aws

Management Guide

Amazon EMR

Amazon EMR: Management Guide

Table of Contents

Was ist Amazon EMR?

Amazon EMR (früher Amazon Elastic genannt MapReduce) ist eine verwaltete Cluster-Plattform, die die Ausführung von Big-Data-Frameworks wie Apache Hadoop und Apache Spark vereinfacht, AWS um riesige Datenmengen zu verarbeiten und zu analysieren. Die Verwendung dieser Frameworks und verwandter Open-Source-Projekte, können Sie Daten zu Analysezwecken und Business-Intelligence-Workloads verarbeiten. Amazon EMR zum Transformieren und Verschieben lässt auch große Datenmengen in und aus anderen AWS -Datenspeichern und Datenbanken verwenden, wie z. B. Amazon Simple Storage Service (Amazon S3) und Amazon DynamoDB.

Wenn Sie Amazon EMR erstmalig verwenden, empfehlen wir, zusätzlich zu diesem Abschnitt die folgenden Abschnitte zu lesen:

- Amazon EMR – Auf dieser Service-Seite finden Sie die Highlights, Produktdetails und Preisinformationen.
- Tutorial: Erste Schritte mit Amazon EMR – Mit diesem Tutorial können Sie schnell mit Amazon EMR beginnen.

In diesem Abschnitt

Übersicht über Amazon EMR

Dieses Thema bietet eine Übersicht über die Amazon-EMR-Cluster, einschließlich der Übermittlung von Aufträgen an einen Cluster, der Verarbeitung von Daten und der verschiedenen Status, die der Cluster während der Verarbeitung durchläuft.

In diesem Thema

Verstehen von Clustern und Knoten

Die zentrale Komponente des Amazon EMR ist der Cluster. Ein Cluster ist eine Sammlung von Amazon Elastic Compute Cloud (Amazon EC2)-Instances. Jede Instance in einem Cluster wird als Knoten bezeichnet. Jeder Knoten verfügt über eine Rolle im Cluster – Knotentyp genannt. Amazon EMR installiert auch verschiedene Softwarekomponenten auf den einzelnen Knotentypen und überträgt so jedem Knoten eine Rolle in einer verteilten Anwendung wie Apache Hadoop.

Amazon EMR verfügt über die folgenden Knotentypen:

- Primärknoten: Knoten, der den Cluster durch die Ausführung von Softwarekomponenten verwaltet, die die Verteilung von Daten und Aufgaben auf andere Knoten zur Verarbeitung koordinieren. Der Primärknoten überwacht den Status der Aufgaben und überwacht den Zustand des Clusters. Jeder Cluster verfügt über einen Primärknoten und es ist möglich, einen Einzelknoten-Cluster nur mit dem Primärknoten zu erstellen.
- Core-Knoten: Knoten mit Software-Komponenten, die Aufgaben ausführen und Daten im HDFS (Hadoop Distributed File System) auf dem Cluster speichern. Multiknoten-Cluster enthalten mindestens einen Core-Knoten.
- Aufgabenknoten: Knoten mit Software-Komponenten, die nur Aufgaben ausführen und keine Daten in HDFS speichern. Aufgabenknoten sind optional.

Übermitteln von Aufträgen an einen Cluster

Bei Ausführung eines Clusters in Amazon EMR haben Sie mehrere Möglichkeiten, die auszuführende Arbeit anzugeben.

- Stellen Sie die gesamte Definition der auszuführenden Arbeit in Funktionen bereit, die Sie als Schritte angeben, wenn Sie einen Cluster erstellen. Dies wird in der Regel für Cluster durchgeführt, die eine bestimmte Datenmenge verarbeiten und nach Abschluss der Verarbeitung beendet werden.
- Erstellen Sie einen Cluster mit langer Laufzeit und verwenden Sie die Amazon EMR-Konsole, die Amazon EMR-API oder die AWS CLI zum Senden von Schritten, die einen oder mehrere Jobs enthalten können. Weitere Informationen finden Sie unter Übermitteln von Arbeit an einen Cluster.
- Erstellen Sie einen Cluster, stellen Sie nach Bedarf eine Verbindung zum Primärknoten und zu anderen Knoten mit SSH her und verwenden Sie die von den installierten Anwendungen bereitgestellten Schnittstellen, um Aufgaben auszuführen und Abfragen zu senden entweder in Skripts oder interaktiv. Weitere Informationen finden Sie im Handbuch zu Amazon-EMR-Versionen.

Verarbeiten von Daten

Wenn Sie einen Cluster starten, bestimmen Sie die zu installierenden Frameworks und Anwendungen, damit Ihren Anforderungen an die Datenverarbeitung entsprochen wird. Um Daten in Ihrem Amazon-EMR-Cluster zu verarbeiten, können Sie Aufträge oder Abfragen direkt an installierte Anwendungen senden oder alternativ die Schritte im Cluster ausführen.

Übermitteln von Aufträgen direkt an die Anwendungen

Sie können Aufträge direkt an die Software übermitteln, die auf Ihrem Amazon-EMR-Cluster installiert ist, und anschließend damit interagieren. Dazu stellen Sie in der Regel eine sichere Verbindung mit dem Primärknoten her und greifen auf die Schnittstellen und Tools zu, die für die Software, die direkt auf Ihrem Cluster ausgeführt wird, verfügbar sind. Weitere Informationen finden Sie unter Verbinden mit einem Cluster.

Ausführen von Schritten zur Verarbeitung von Daten

Sie können einem Amazon-EMR-Cluster einen oder mehrere angeordnete Schritte übermitteln. Jeder Schritt ist eine Arbeitseinheit mit Anweisungen zur Verarbeitung von Daten durch auf dem Cluster installierte Software.

Es folgt ein Beispiel für einen Prozess mit vier Schritten:

1. Übermitteln Sie die Eingabedatenmenge für die Verarbeitung.
2. Verarbeiten Sie die Ausgabe des ersten Schritts mithilfe eines Pig-Programms.
3. Verarbeiten Sie eine zweites Eingabedatenmenge mithilfe eines Hive-Programms.
4. Schreiben Sie einen Ausgabedatensatz.

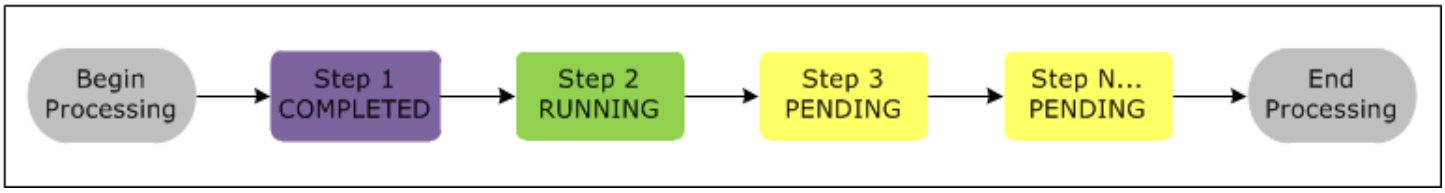
Wenn Sie Daten in Amazon EMR verarbeiten, wird die Eingabe als Daten in Dateien gespeichert, die sich im zugrunde liegenden Dateisystem, wie z. B. Amazon S3 oder HDFS, befinden. Diese Daten werden während des Verarbeitungsablaufs von einem Schritt zum nächsten weitergeleitet. Im letzten Schritt werden die Ausgabedaten in einen bestimmten Speicherort geschrieben, wie zum Beispiel in einen Amazon-S3-Bucket.

Die Schritte werden in der folgenden Reihenfolge ausgeführt:

1. Eine Anfrage wird übermittelt, um mit den Verarbeitungsschritten zu beginnen.
2. Der Status aller Schritte wird auf PENDING (AUSSTEHEND) festgelegt.

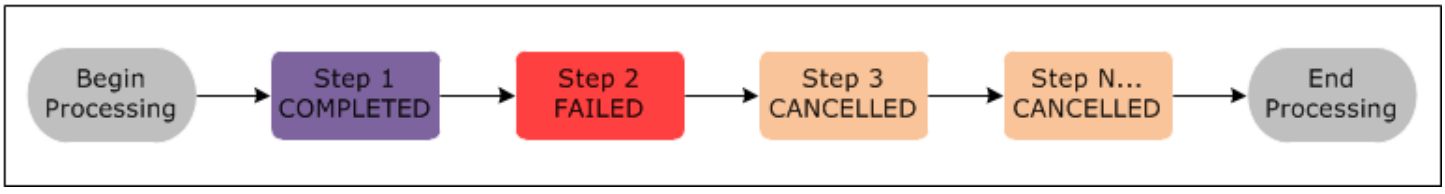
3. Wenn der erste Schritt der Sequenz gestartet wird, wird dessen Status in RUNNING (WIRD AUSGEFÜHRT) geändert. Die anderen Schritte bleiben im Status PENDING (AUSSTEHEND).
4. Nachdem der erste Schritt abgeschlossen ist, wird dessen Status in COMPLETED (ABGESCHLOSSEN) geändert.
5. Der nächste Schritt der Sequenz wird gestartet und dessen Status wird in RUNNING (WIRD AUSGEFÜHRT) geändert. Nachdem er abgeschlossen ist, wird dessen dessen Status in COMPLETED (ABGESCHLOSSEN) geändert.
6. Dieses Muster wiederholt sich für jeden Schritt, bis alle Schritte abgeschlossen sind und die Verarbeitung beendet wird.

Das folgende Diagramm stellt die Schrittsequenz sowie die Statusänderung für die einzelnen Schritte während der Verarbeitung dar.



Wenn ein Schritt während der Verarbeitung fehlschlägt, wechselt der Status zu FEHLGESCHLAGEN. Sie können für jeden Schritt festlegen, was als Nächstes geschieht. Standardmäßig werden alle verbleibenden Schritte in der Sequenz auf ABGEBROCHEN festgelegt und wenn ein vorangehender Schritt fehlschlägt. Außerdem können Sie das Ignorieren des Fehlers aktivieren, damit die verbleibenden Schritte ausgeführt werden oder der Cluster sofort beendet wird.

Das folgende Diagramm stellt die Schrittsequenz sowie die standardmäßige Statusänderung dar, wenn ein Schritt während der Verarbeitung fehlschlägt.



Verstehen des Cluster-Lebenszyklus

Ein erfolgreicher Amazon-EMR-Cluster befolgt diesen Prozess:

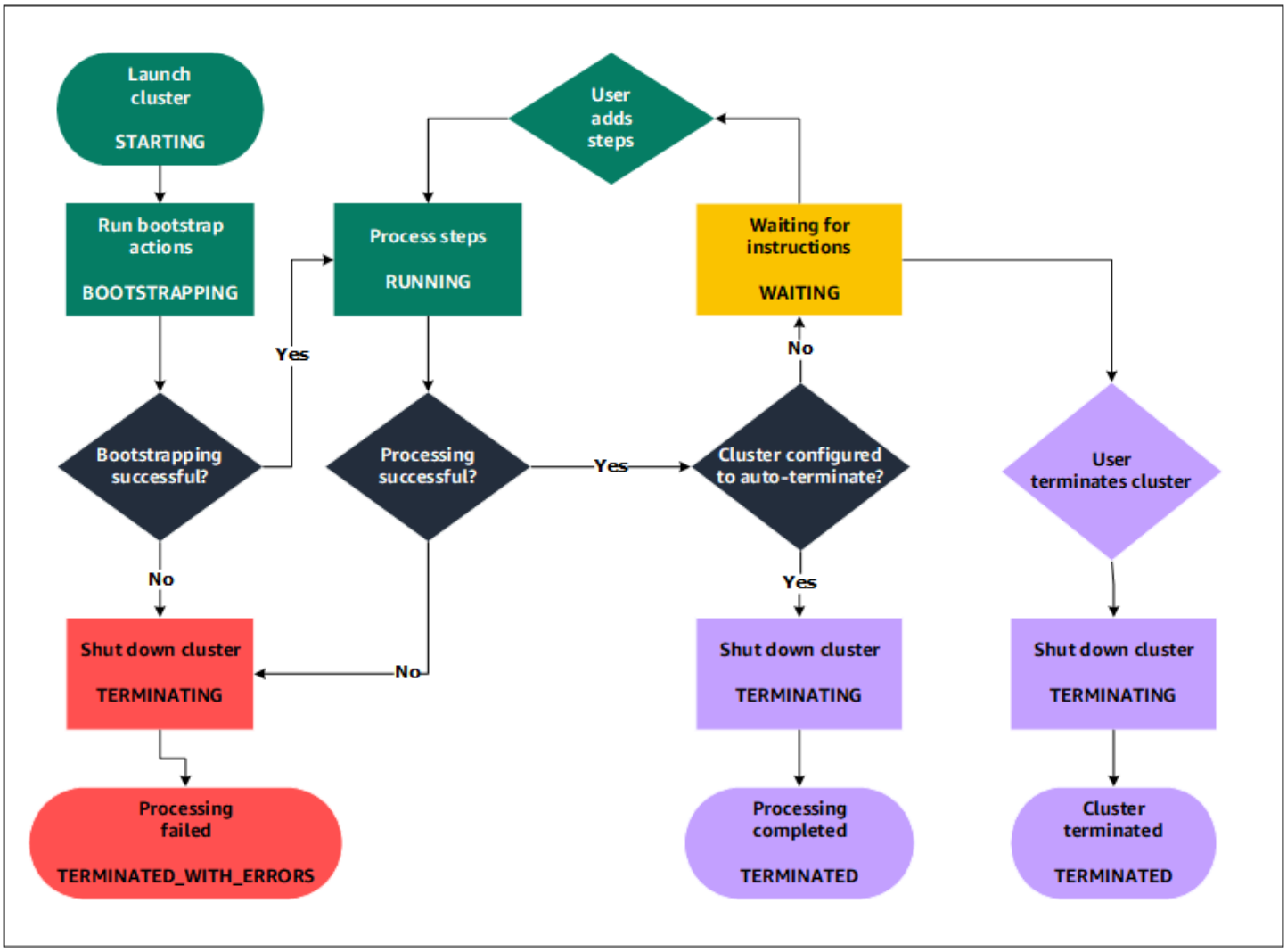
1. Amazon EMR stellt zunächst EC2-Instances im Cluster für jede Instance nach Maßgabe Ihrer Spezifikationen bereit. Weitere Informationen finden Sie unter Cluster-Hardware und Netzwerken

konfigurieren. Amazon EMR verwendet für alle Instances das Standard-AMI für Amazon EMR oder ein von Ihnen angegebenes benutzerdefiniertes Amazon-Linux-AMI. Weitere Informationen finden Sie unter Verwenden Sie ein benutzerdefiniertes AMI. Während dieser Phase ist der Cluster-Status auf `STARTING` gesetzt.

2. Amazon EMR; führt Bootstrap-Aktionen aus, die Sie für jede Instance angeben. Sie können Bootstrap-Aktionen verwenden, um benutzerdefinierte Anwendungen zu installieren und erforderliche Anpassungen vorzunehmen. Weitere Informationen finden Sie unter Erstellen von Bootstrap-Aktionen zur Installation zusätzlicher Software. Während dieser Phase ist der Cluster-Status auf `BOOTSTRAPPING` gesetzt.
3. Amazon EMR installiert die nativen Anwendungen, die Sie angeben, wenn Sie den Cluster erstellen, z. B. Hive, Hadoop, Spark usw.
4. Nachdem Bootstrap-Aktionen erfolgreich abgeschlossen und native Anwendungen installiert wurden, lautet der Cluster-Status `RUNNING`. An diesem Punkt können Sie die Verbindung zu Cluster-Instances herstellen. Der Cluster führt sequenziell die Schritte aus, die Sie beim Erstellen des Clusters angegeben haben. Sie können zusätzliche Schritte senden, die dann nach Abschluss der vorherigen Schritte ausgeführt werden. Weitere Informationen finden Sie unter Übermitteln von Arbeit an einen Cluster.
5. Nachdem die Schritte erfolgreich ausgeführt wurden, erhält der Cluster den Status `WAITING`. Wenn ein Cluster für die automatische Beendigung nach Abschluss des letzten Schritts konfiguriert ist, wechselt der Cluster den Status `TERMINATING`-Zustand und dann in den `TERMINATED`-Zustand. Wenn der Cluster so konfiguriert ist, dass er wartet, müssen Sie ihn manuell herunterfahren, wenn Sie ihn nicht mehr benötigen. Nachdem Sie den Cluster manuell beenden haben, wird dieser in den Status `TERMINATING` versetzt und danach in den Status `TERMINATED`.

Ein Fehler im Cluster-Lebenszyklus veranlasst, Amazon EMR den Cluster und dessen Instances zu beenden, sofern Sie nicht den Beendigungsschutz aktivieren. Wenn ein Cluster wegen eines Fehlers beendet wird, werden alle auf dem Cluster befindlichen Daten gelöscht und dem Cluster-Status wird der Status `TERMINATED_WITH_ERRORS` zugewiesen. Wenn Sie den Beendigungsschutz aktiviert haben, können Sie Daten vom Cluster abrufen und anschließend den Beendigungsschutz entfernen und den Cluster beenden. Weitere Informationen finden Sie unter Verwenden des Beendigungsschutzes.

Das folgende Diagramm stellt den Lebenszyklus eines Clusters dar und wie die einzelnen Lebenszyklusphasen einem bestimmten Cluster-Status zugeordnet sind.



Vorteile der Verwendung von Amazon EMR

Es gibt zahlreiche Vorteile für die Verwendung von Amazon EMR. Dieser Abschnitt bietet eine Übersicht über die Vorteile und stellt Ihnen Links zu weiteren Informationen zur Verfügung.

Themen

- Kosteneinsparungen
- AWS Integration
- Bereitstellung
- Skalierbarkeit und Flexibilität
- Zuverlässigkeit
- Sicherheit

- Überwachen
- Verwaltungsschnittstellen

Kosteneinsparungen

Amazon EMR Preisgestaltung richtet sich nach dem Instance-Typ und der Anzahl der Amazon-EC2-Instances, die Sie bereitstellen, sowie der Region, in der Sie den Cluster starten. On-Demand-Preise bieten einen niedrigen Stundensatz, allerdings können Sie die Kosten weiter senken, indem Sie Reserved Instances erwerben oder auf Spot-Instances bieten. Spot Instances können bedeutende Kostenersparnisse bieten – in einigen Fällen betragen sie nur ein Zehntel der On-Demand-Preise.

> **Note**
>
> Wenn Sie Amazon S3, Amazon Kinesis oder DynamoDB mit Ihrem EMR-Cluster verwenden, fallen für diese Services zusätzliche Gebühren an, die getrennt von Ihrer Amazon-EMR-Nutzung berechnet werden.

> **Note**
>
> Wenn Sie einen Amazon-EMR-Cluster in einem privaten Subnetz einrichten, empfehlen wir, dass Sie auch VPC-Endpunkte für Amazon S3 einrichten. Wenn sich Ihr EMR-Cluster in einem privaten Subnetz ohne VPC-Endpunkte für Amazon S3 befindet, fallen zusätzliche NAT-Gateway-Gebühren an, die mit S3-Verkehr verbunden sind, da der Verkehr zwischen Ihrem EMR-Cluster und S3 nicht innerhalb Ihrer VPC verbleibt.

Weitere Informationen zu Preisoptionen und Details finden Sie unter Amazon-EMR-Preise.

AWS Integration

Amazon EMR lässt sich in andere AWS Services integrieren, um Funktionen und Funktionen in Bezug auf Netzwerk, Speicher, Sicherheit usw. für Ihren Cluster bereitzustellen. In der folgenden Liste finden Sie einige Beispiele für diese Integration:

- Amazon EC2 für die Instances, die als Knoten im Cluster vorhanden sind
- Amazon Virtual Private Cloud (Amazon VPC) zur Konfiguration des virtuellen Netzwerks, in dem Sie Ihre Instances starten

- Amazon S3 zum Speichern von Ein- und Ausgabedaten
- Amazon überwacht CloudWatch die Cluster-Leistung und konfiguriert Alarme
- AWS Identity and Access Management (IAM) zur Konfiguration von Berechtigungen
- AWS CloudTrail um Anfragen an den Service zu prüfen
- AWS Data Pipeline um Ihre Cluster zu planen und zu starten
- AWS Lake Formation um Daten in einem Amazon S3 S3-Data Lake zu entdecken, zu katalogisieren und zu sichern

Bereitstellung

Ihr EMR-Cluster besteht aus EC2-Instances, die die Aufgaben ausführen, die Sie Ihrem Cluster übermitteln. Wenn Sie einen Cluster starten, konfiguriert Amazon EMR die Instances mit den von Ihnen ausgewählten Anwendungen, wie beispielsweise Apache Hadoop oder Spark. Wählen Sie die Größe und den Typ der Instance aus, die am ehesten den Verarbeitungsanforderungen Ihres Clusters entsprechen: Stapelverarbeitung, schnelle Abfragen, Streaming-Daten oder große Datenspeicher. Weitere Informationen zu den für Amazon EMR verfügbaren Instance-Typen finden Sie unter Cluster-Hardware und Netzwerken konfigurieren.

Amazon EMR bietet verschiedene Möglichkeiten zum Konfigurieren von Software auf Ihrem Cluster. Sie können beispielsweise eine Amazon-EMR-Version installieren, die eine Reihe ausgewählter Anwendungen umfasst, einschließlich vielseitiger Frameworks wie Hadoop und Anwendungen, wie beispielsweise Hive, Pig oder Spark. Darüber hinaus können Sie auch eine der zahlreichen MapR-Verteilungen installieren. Amazon EMR verwendet Amazon Linux so können Sie auch Software unter Verwendung des Paket-Managers yum oder direkt von der Quelle manuell auf Ihrem Cluster installieren. Weitere Informationen finden Sie unter Konfigurieren der Cluster-Software.

Skalierbarkeit und Flexibilität

Amazon EMR bietet Flexibilität, sodass Sie Ihren Cluster nach oben oder unten skalieren können, wenn sich Ihre Anforderungen an die Datenverarbeitung ändern. Sie können die Größe des Clusters ändern, um während Spitzenlastzeiten Instances hinzuzufügen, und um Instances zu entfernen, wenn die Spitzenlastzeiten nachlassen. So verfügen Sie über mehr Kontrolle über Ihre Kosten. Weitere Informationen finden Sie unter Manuelle Größenanpassung eines aktiven Clusters.

Amazon EMR bietet außerdem die Option, mehrere Instance-Gruppen auszuführen. So können Sie sie in einer Gruppe On-Demand-Instances verwenden, um die Verarbeitungsleistung sicherzustellen,

während Sie in einer anderen Gruppe Spot Instances verwenden, um Ihre Aufträge schneller abzuschließen und Kosten zu senken. Sie können auch verschiedene Instance-Typen mischen, um die Preisvorteile von bestimmten Spot-Instance-Typen zu nutzen. Weitere Informationen finden Sie unter Wann sollten Sie Spot Instances verwenden?.

Darüber hinaus bietet Amazon EMR die Flexibilität, verschiedene Dateisysteme für Ihre Eingabe-, Ausgabe- und Zwischendaten zu verwenden. Für die Verarbeitung von Daten, die Sie nicht länger als den Lebenszyklus Ihres Clusters speichern müssen, können Sie beispielsweise das Hadoop Distributed File System (HDFS) auswählen, das auf den Primär- und Core-Knoten Ihres Clusters ausgeführt wird. Sie können möglicherweise auch das EMR File System (EMRFS) für die Verwendung mit Amazon S3 auswählen. Es kann als Daten-Layer für Anwendungen auf Ihrem Cluster dienen, sodass Sie die Datenverarbeitung und den Speicher trennen und Daten außerhalb des Lebenszyklus Ihres Clusters erhalten können. EMRFS bietet Ihnen die Möglichkeit, Ihre Anforderungen an die Datenverarbeitung und an den Speicher nach oben oder nach unten zu skalieren. Sie können Ihre Anforderungen an die Datenverarbeitung skalieren, indem Sie die Größe Ihres Clusters verändern, und Ihre Speicheranforderungen skalieren, indem Sie Amazon S3 verwenden. Weitere Informationen finden Sie unter Mit Storage- und Dateisystemen arbeiten.

Zuverlässigkeit

Amazon EMR; überwacht die Knoten in Ihrem Cluster und beendet und ersetzt eine Instance automatisch, wenn ein Fehler auftritt.

Amazon EMR bietet Konfigurationsoptionen, anhand denen Sie steuern, ob der Cluster beendet werden soll automatisch oder manuell. Wenn Sie Ihren Cluster so konfigurieren, dass er automatisch beendet wird, erfolgt das, nachdem alle Schritte abgeschlossen sind. Dies wird auch als vorübergehender Cluster bezeichnet. Sie können den Cluster jedoch auch so konfigurieren, dass er nach Abschluss der Verarbeitung weiter ausgeführt wird. Auf diese Weise können Sie ihn manuell beenden, wenn Sie ihn nicht länger benötigen. Alternativ können Sie einen Cluster erstellen, mit den installierten Anwendungen direkt interagieren und den Cluster, wenn Sie ihn nicht mehr benötigen, manuell beenden. Die Cluster in diesen Beispielen werden als langlebige Cluster bezeichnet.

Zusätzlich können Sie den Beendigungsschutz konfigurieren, um zu verhindern, dass Instances im Cluster aufgrund von Fehlern oder Problemen während der Verarbeitung beendet werden. Wenn der Beendigungsschutz aktiviert ist, können Sie die Daten vor der Beendigung von den Instances wiederherstellen. Die Standardeinstellungen für diese Optionen unterscheiden sich, je nachdem, ob Sie einen Cluster über die Konsole, die CLI oder die API starten. Weitere Informationen finden Sie unter Verwenden des Beendigungsschutzes.

Sicherheit

Amazon EMR nutzt andere AWS Services wie IAM und Amazon VPC sowie Funktionen wie Amazon EC2 EC2-Schlüsselpaare, um Sie bei der Sicherung Ihrer Cluster und Daten zu unterstützen.

IAM

Amazon EMR kann mit IAM integriert werden, um Berechtigungen zu verwalten. Sie definieren Berechtigungen mit IAM-Richtlinien, die Sie Benutzern oder IAM-Gruppen anfügen. Die Berechtigungen, die Sie in den Richtlinie definieren, legen fest, welche Aktionen diese Benutzer oder Gruppenmitglieder ausführen können, und auf welche Ressourcen sie zugreifen können. Weitere Informationen finden Sie unter So EMR arbeitet Amazon mit IAM.

Darüber hinaus verwendet Amazon EMR, IAM-Rollen für den Amazon EMR selbst und das EC2-Instance-Profil für die Instances. Diese Rollen gewähren dem Service und den Instances die Erlaubnis, in Ihrem Namen auf andere AWS Services zuzugreifen. Es gibt sowohl für den Amazon-EMR-Service als auch für das EC2-Instance-Profil eine standardmäßige Rolle. Die Standardrollen verwenden AWS verwaltete Richtlinien, die automatisch für Sie erstellt werden, wenn Sie zum ersten Mal einen EMR-Cluster von der Konsole aus starten und Standardberechtigungen auswählen. Sie können die IAM-Standardrollen auch über die AWS CLI erstellen. Wenn Sie stattdessen die Berechtigungen verwalten möchten AWS, können Sie benutzerdefinierte Rollen für das Service- und Instanzprofil auswählen. Weitere Informationen finden Sie unter IAMServicerollen für EMR Amazon-Berechtigungen für AWS Dienste und Ressourcen konfigurieren.

Sicherheitsgruppen

Amazon EMR verwendet Sicherheitsgruppen, um den ein- und ausgehenden Datenverkehr zu Ihren EC2-Instances zu steuern. Wenn Sie einen Cluster starten, verwendet Amazon EMR eine Sicherheitsgruppe für die primäre-Instance und eine Sicherheitsgruppe, die von den Core-/Aufgaben-Instances gemeinsam genutzt wird. Amazon EMR konfiguriert die Sicherheitsgruppenregeln, um die Kommunikation zwischen den Instances im Cluster sicherzustellen. Optional können Sie, falls Sie erweiterte Regeln benötigen, zusätzliche Sicherheitsgruppen konfigurieren und sie den primäre und Core-/Aufgaben-Instances zuweisen. Weitere Informationen finden Sie unter Steuerung des Netzwerkverkehrs mit Sicherheitsgruppen.

Verschlüsselung

Amazon EMR unterstützt die optionale Amazon S3 serverseitige und clientseitige Verschlüsselung mit EMRFS, um die von Ihnen in Amazon S3 gespeicherten Daten zu schützen. Bei der

serverseitigen Verschlüsselung werden Ihre Daten von Amazon S3 nach dem Hochladen verschlüsselt.

Bei der clientseitigen Verschlüsselung erfolgt der Ver- und Entschlüsselungsvorgang im EMRFS-Client auf Ihrem EMR-Cluster. Sie verwalten den Stammschlüssel für die clientseitige Verschlüsselung entweder mit dem AWS Key Management Service (AWS KMS) oder Ihrem eigenen Schlüsselverwaltungssystem.

Weitere Informationen finden Sie unter Amazon-S3-Verschlüsselung mithilfe von EMRFS-Eigenschaften angeben.

Amazon VPC

Amazon EMR unterstützt das Starten von Clustern in einer Virtual Private Cloud (VPC) in Amazon VPC. Eine VPC ist ein isoliertes, virtuelles Netzwerk, AWS das die Möglichkeit bietet, erweiterte Aspekte der Netzwerkkonfiguration und des Netzwerkzugriffs zu steuern. Weitere Informationen finden Sie unter Netzwerk konfigurieren.

AWS CloudTrail

Amazon EMR lässt sich integrieren CloudTrail , um Informationen über Anfragen zu protokollieren, die von oder im Namen Ihres AWS Kontos gestellt wurden. Anhand dieser Informationen können Sie verfolgen, wer wann auf Ihr Cluster zugreift sowie die IP-Adresse, von der die Anforderung gestellt wird. Weitere Informationen finden Sie unter EMRAPIAmazon-Anrufe protokollieren AWS CloudTrail.

Amazon-EC2-Schlüsselpaare

Indem Sie eine sichere Verbindung zwischen Ihrem Remotecomputer und dem Primärknoten herstellen, können Sie Ihren Cluster überwachen und damit interagieren. Sie verwenden das Netzwerkprotokoll Secure Shell (SSH) für diese Verbindung oder Kerberos für die Authentifizierung. Wenn Sie SSH verwenden, ist ein Amazon-EC2-Schlüsselpaar erforderlich. Weitere Informationen finden Sie unter Verwenden Sie ein EC2 key pair für SSH Anmeldeinformationen.

Überwachen

Sie können die Amazon-EMR-Management-Schnittstellen und Protokolldateien verwenden, um Probleme mit dem Cluster zu beheben, z. B. bei Ausfällen oder Fehlern. Amazon EMR bietet die Möglichkeit, Protokolldateien in Amazon S3 zu archivieren, sodass Sie Protokolle speichern und Probleme beheben können, auch nachdem der Cluster beendet wurde. Amazon EMR bietet in der Amazon-EMR-Konsole auch ein optionales Debugging-Tool, mit dem Sie die Protokolldateien im

Hinblick auf Schritte, Aufträge und Aufgaben durchsuchen können. Weitere Informationen finden Sie unter Konfigurieren der Cluster-Protokollierung und des Debuggings.

Amazon EMR lässt sich integrieren CloudWatch , um Leistungskennzahlen für den Cluster und Jobs innerhalb des Clusters nachzuverfolgen. Sie können Alarme im Hinblick auf eine Vielzahl von Metriken konfigurieren, z. B. ob der Cluster inaktiv ist oder wie viel Prozent des Speicherplatzes verbraucht wurden. Weitere Informationen finden Sie unter Überwachung von EMR Amazon-Metriken mit CloudWatch.

Verwaltungsschnittstellen

Es gibt mehrere Möglichkeiten, mit Amazon EMR zu interagieren:

- Konsole – eine grafische Benutzerschnittstelle, die Sie verwenden können, um Clusters zu starten oder zu verwalten. Hier füllen Sie Webformulare aus, um Detaildaten zum Starten von Clusters anzugeben, Detaildaten von vorhandenen Clusters anzuzeigen und Clusters zu debuggen bzw. zu beenden. Die Konsole bietet die einfachste Möglichkeit für die ersten Schritte mit Amazon EMR keine Programmierkenntnisse erforderlich. Die Konsole ist online unter https://console.aws.amazon.com/elasticmapreduce/home verfügbar.
- AWS Command Line Interface (AWS CLI) — Eine Client-Anwendung, die Sie auf Ihrem lokalen Computer ausführen, um eine Verbindung zu Amazon EMR herzustellen und Cluster zu erstellen und zu verwalten. Das AWS CLI enthält eine Reihe von Befehlen mit vielen Funktionen, die speziell für Amazon EMR gelten. Damit schreiben Sie Skripts, die das Starten und Verwalten der Clusters automatisieren. Wenn Sie lieber von einer Befehlszeile aus arbeiten, AWS CLI ist die Verwendung von die beste Option. Weitere Informationen und Beispiele finden Sie unter Amazon EMR in der AWS CLI -Befehlsreferenz.
- Software Development Kit (SDK) – SDKs stellt Funktionen bereit, die Amazon EMR aufrufen, um Clusters zu erstellen und zu verwalten. Mit ihnen können Sie Anwendungen schreiben, die das Erstellen und Verwalten von Clusters automatisieren. Die Verwendung des SDK ist die beste Option, wenn Sie die Funktionen von Amazon EMR erweitern oder anpassen möchten. Amazon EMR ist derzeit in den folgenden SDKs verfügbar: Go, Java, .NET (C# und VB.NET), Node.js, PHP, Python und Ruby. Weitere Informationen über diese SDKs, finden Sie unter Tools für AWS und Amazon-EMR-Beispielcode und -Bibliotheken.
- Web Service API – eine Low-Level-Schnittstelle, die Sie benutzen können, um den Webservice direkt mithilfe von JSON aufzurufen. Die Verwendung der API ist die beste Option, wenn Sie ein eigenes SDK erstellen wollen, das Amazon EMR aufruft. Weitere Informationen finden Sie in der Amazon-EMR-API-Referenz.

Überblick über die Amazon-EMR-Architektur

Die Service-Architektur von Amazon EMR besteht aus mehreren Ebenen, die dem Cluster jeweils bestimmte Möglichkeiten und Funktionen bereitstellen. Dieser Abschnitt bietet eine Übersicht über die jeweiligen Ebenen und Komponenten.

In diesem Thema

Speicher

Die Speicherschicht umfasst die verschiedenen Dateisysteme, die Sie in Ihrem Cluster verwendet werden. Es gibt mehrere verschiedene Speicheroptionen wie nachfolgend beschrieben.

Hadoop Distributed File System (HDFS)

Hadoop Distributed File System (HDFS) ist ein verteiltes, skalierbares Dateisystem für Hadoop. HDFS verteilt die auf verschiedenen Instances im Cluster gespeicherten Daten, wobei mehrere Kopien von Daten auf unterschiedlichen Instances gespeichert werden, um sicherzustellen, dass bei Ausfall einer einzelnen Instance keine Daten verloren gehen. HDFS ist flüchtiger Speicher, der zurückgefordert wird, wenn Sie einen Cluster beenden. HDFS ist nützlich für das Zwischenspeichern von Zwischenergebnissen während der MapReduce Verarbeitung oder für Workloads mit erheblichen zufälligen I/O-Vorgängen.

Weitere Informationen finden Sie unter Instance-Speicher im HDFS-Benutzerhandbuch auf der Website von Apache Hadoop.

EMR File System (EMRFS)

Amazon EMR erweitert mittels des EMR File System (EMRFS) Hadoop durch die Hinzufügung des direkten Zugriffs auf in Amazon S3 gespeicherte Daten, als ob es sich um ein Dateisystem wie HDFS handeln würde. Sie können entweder HDFS oder Amazon S3 als das Dateisystem Ihres Clusters verwenden. In der Regel wird Amazon S3 zum Speichern der Ein- und Ausgabedaten verwendet, Zwischenergebnisse werden in HDFS gespeichert.

Lokales Dateisystem

Das lokale Dateisystem bezieht sich auf einen lokal verbundenen Datenträger. Wenn Sie einen Hadoop-Cluster erstellen, werden die einzelnen Knoten aus einer Amazon-EC2-Instance erstellt, die einen vorkonfigurierten Block mit bereits zugeordnetem Festplattenspeicher, einen sogenannten Instance-Speicher, aufweist. Die Daten auf den Instance-Speicher-Volumes bleiben nur während des Lebenszyklus der Amazon-EC2-Instance erhalten.

Cluster-Ressourcenverwaltung

Der Ressourcenverwaltungs-Layer ist verantwortlich für die Verwaltung der Cluster-Ressourcen und die Planung der Aufträge für die Datenverarbeitung.

Amazon EMR verwendet standardmäßig YARN (Yet Another Resource Negotiator). Dabei handelt es sich um eine Komponente, die in Apache Hadoop 2.0 eingeführt wurde und mit der die Cluster-Ressourcen für mehrere Datenverarbeitungs-Frameworks zentral verwaltet werden können. Es gibt jedoch auch andere Frameworks und Anwendungen, die in Amazon EMR bereitgestellt werden und nicht YARN als Ressourcenmanager verwenden. Amazon EMR verfügt außerdem auf jedem Knoten, der YARN-Komponenten verwaltet, über einen Agenten, der den Cluster stabil erhält und mit dem Amazon-EMR-Service kommuniziert.

Da Spot Instances häufig zum Ausführen von Aufgabenknoten verwendet werden, verfügt Amazon EMR über Standardfunktionen für die Planung von YARN-Aufträge, sodass laufende Aufträge nicht fehlschlagen, wenn Aufgabenknoten, die auf Spot Instances ausgeführt werden, beendet werden. Amazon EMR ermöglicht dies, indem Anwendungsmasterprozesse nur auf Core-Knoten ausgeführt werden können. Der Anwendungsmasterprozess steuert die Ausführung von Aufträgen und muss während der gesamten Laufzeit des Auftrags aktiv bleiben.

Amazon-EMR-Version 5.19.0 und höher verwendet zu diesem Zweck das integrierte YARN-Knotenbeschriftungsfeature. (Frühere Versionen verwendeten einen Code-Patch). Die Eigenschaften in den Klassifizierungen `yarn-site` und in der `capacity-scheduler`-Konfiguration sind standardmäßig so konfiguriert, dass der YARN-Kapazitätsplaner und der Fair-Scheduler die Vorteile von Knotenbezeichnungen nutzen. Amazon EMR kennzeichnet Core-Knoten automatisch mit dem `CORE`-Label und legt Eigenschaften fest, sodass Anwendungsmaster nur für Knoten mit dem CORE-Label geplant werden. Durch manuelles Ändern verwandter Eigenschaften in den Konfigurationsklassifizierungen von Yarn-Site und Kapazitätsplaner oder direkt in den zugehörigen XML-Dateien könnte diese Feature beeinträchtigt oder verändert werden.

Datenverarbeitungs-Frameworks

Der Datenverarbeitungs-Framework-Layer ist die Engine, die zur Verarbeitung und Analyse der Daten verwendet wird. Es stehen viele Frameworks zur Verfügung, die auf YARN ausgeführt werden oder über ihre eigene Ressourcenverwaltung verfügen. Es gibt unterschiedliche Frameworks für die verschiedenen Verarbeitungsanforderungen, beispielsweise Stapel, Interaktiv, In-Memory, Streaming und so weiter. Das Framework, das Sie auswählen sollten, hängt von Ihrem Anwendungsfall ab. Dies wirkt sich auf die Sprachen und Schnittstellen der Anwendungsebene aus, d. h. der Ebene, über die mit den zu verarbeitenden Daten interagiert wird. Die wichtigsten für Amazon EMR verfügbaren Verarbeitungs-Frameworks sind Hadoop MapReduce und Spark.

Hadoop MapReduce

Hadoop MapReduce ist ein Open-Source-Programmiermodell für verteiltes Rechnen. Es vereinfacht den Prozess der Entwicklung paralleler verteilter Anwendungen, indem die gesamte Logik gehandhabt wird, während Sie die Funktionen "Map" und "Reduce" bereitstellen. Die Funktion "Map" führt eine Zuordnung von Daten und Sätzen von Schlüssel/Wert-Paaren durch, die als Zwischenergebnisse bezeichnet werden. Die Funktion "Reduce" kombiniert die Zwischenergebnisse, wendet weitere Algorithmen an und generiert das Endergebnis. Es stehen mehrere Frameworks zur Verfügung MapReduce, z. B. Hive, das automatisch Map- und Reduce-Programme generiert.

Weitere Informationen finden Sie unter Wie Karten- und Reduziervorgänge tatsächlich ausgeführt werden auf der Wiki-Website von Apache Hadoop.

Apache Spark

Spark ist ein Cluster-Framework und Programmiermodell für die Verarbeitung von Big-Data-Workloads. Wie Hadoop MapReduce ist Spark ein verteiltes Open-Source-Verarbeitungssystem, verwendet jedoch gerichtete azyklische Graphen für Ausführungspläne und In-Memory-Caching für Datensätze. Wenn Sie Spark auf Amazon EMR ausführen, können Sie über EMRFS direkt auf Ihre Daten in Amazon S3 zugreifen. Spark unterstützt mehrere interaktive Abfragen Module wie beispielsweise SparkSQL.

Weitere Informationen finden Sie unter Apache Spark in Amazon-EMR-Clusters in den Amazon-EMR-Versionshinweise.

Anwendungen und Programme

Amazon EMR unterstützt zahlreiche Anwendungen, wie Hive, Pig, und die Spark Streaming-Bibliothek, um beispielsweise mithilfe komplexerer Programmiersprachen Verarbeitungs-Workloads

zu erstellen, Machine-Learning-Algorithmen zu nutzen, Anwendungen für die Stream-Verarbeitung zu erstellen und Data Warehouses zu entwickeln. Darüber hinaus unterstützt Amazon EMR auch Open-Source-Projekte, die ihre eigene Cluster-Management-Funktionalität mitbringen und nicht YARN verwenden.

Sie können verschiedene Bibliotheken und Sprachen verwenden, um mit den Anwendungen, die Sie in Amazon EMR ausführen, zu interagieren. Sie können beispielsweise Java, Hive oder Pig mit MapReduce oder Spark Streaming, Spark SQL, MLLib und GraphX mit Spark verwenden.

Weitere Informationen finden Sie im Handbuch zu Amazon-EMR-Versionen.

Einrichten von Amazon EMR

Führen Sie die Aufgaben in diesem Abschnitt aus, bevor Sie einen Amazon-EMR-Cluster zum ersten Mal starten:

Bevor Sie Amazon EMR zum ersten Mal verwenden, führen Sie die folgenden Schritte aus:

Melde dich an für ein AWS-Konto

Wenn Sie noch keine haben AWS-Konto, führen Sie die folgenden Schritte aus, um eine zu erstellen.

Um sich für eine anzumelden AWS-Konto

1. Öffnen Sie https://portal.aws.amazon.com/billing/signup.
2. Folgen Sie den Online-Anweisungen.

   Bei der Anmeldung müssen Sie auch einen Telefonanruf entgegennehmen und einen Verifizierungscode über die Telefontasten eingeben.

   Wenn Sie sich für eine anmelden AWS-Konto, Root-Benutzer des AWS-Kontoswird eine erstellt. Der Root-Benutzer hat Zugriff auf alle AWS -Services und Ressourcen des Kontos. Aus Sicherheitsgründen sollten Sie einem Benutzer Administratorzugriff zuweisen und nur den Root-Benutzer verwenden, um Aufgaben auszuführen, für die Root-Benutzerzugriff erforderlich ist.

AWS sendet Ihnen nach Abschluss des Anmeldevorgangs eine Bestätigungs-E-Mail. Sie können jederzeit Ihre aktuelle Kontoaktivität anzeigen und Ihr Konto verwalten. Rufen Sie dazu https://aws.amazon.com/ auf und klicken Sie auf Mein Konto.

Erstellen Sie einen Benutzer mit Administratorzugriff

Nachdem Sie sich für einen angemeldet haben AWS-Konto, sichern Sie Ihren Root-Benutzer des AWS-Kontos AWS IAM Identity Center, aktivieren und erstellen Sie einen Administratorbenutzer, sodass Sie den Root-Benutzer nicht für alltägliche Aufgaben verwenden.

Sichern Sie Ihre Root-Benutzer des AWS-Kontos

1. Melden Sie sich AWS Management Consoleals Kontoinhaber an, indem Sie Root-Benutzer auswählen und Ihre AWS-Konto E-Mail-Adresse eingeben. Geben Sie auf der nächsten Seite Ihr Passwort ein.

   Hilfe bei der Anmeldung mit dem Root-Benutzer finden Sie unter Anmelden als Root-Benutzer im AWS-Anmeldung Benutzerhandbuch zu.
2. Aktivieren Sie die Multi-Faktor-Authentifizierung (MFA) für den Root-Benutzer.

   Anweisungen finden Sie unter Aktivieren eines virtuellen MFA-Geräts für Ihren AWS-Konto Root-Benutzer (Konsole) im IAM-Benutzerhandbuch.

Erstellen Sie einen Benutzer mit Administratorzugriff

1. Aktivieren Sie das IAM Identity Center.

   Anweisungen finden Sie unter Aktivieren AWS IAM Identity Center im AWS IAM Identity Center Benutzerhandbuch.
2. Gewähren Sie einem Benutzer in IAM Identity Center Administratorzugriff.

   Ein Tutorial zur Verwendung von IAM-Identity-Center-Verzeichnis als Identitätsquelle finden Sie unter Benutzerzugriff mit der Standardeinstellung konfigurieren IAM-Identity-Center-Verzeichnis im AWS IAM Identity Center Benutzerhandbuch.

Melden Sie sich als Benutzer mit Administratorzugriff an

- Um sich mit Ihrem IAM-Identity-Center-Benutzer anzumelden, verwenden Sie die Anmelde-URL, die an Ihre E-Mail-Adresse gesendet wurde, als Sie den IAM-Identity-Center-Benutzer erstellt haben.

  Hilfe bei der Anmeldung mit einem IAM Identity Center-Benutzer finden Sie im AWS-Anmeldung Benutzerhandbuch unter Anmeldung beim AWS Zugriffsportal.

Weisen Sie weiteren Benutzern Zugriff zu

1. Erstellen Sie in IAM Identity Center einen Berechtigungssatz, der der bewährten Methode zur Anwendung von Berechtigungen mit den geringsten Rechten folgt.

Anweisungen finden Sie im Benutzerhandbuch unter Einen Berechtigungssatz erstellen.AWS IAM Identity Center

2. Weisen Sie Benutzer einer Gruppe zu und weisen Sie der Gruppe dann Single Sign-On-Zugriff zu.

   Anweisungen finden Sie im AWS IAM Identity Center Benutzerhandbuch unter Gruppen hinzufügen.

Erstellen eines Amazon-EC2-Schlüsselpaares für SSH

> **Note**
>
> Mit Amazon-EMR-Version 5.10.0 oder höher können Sie Kerberos zur Authentifizierung von Benutzern und SSH-Verbindungen zu einem Cluster konfigurieren. Weitere Informationen finden Sie unter Verwenden Sie Kerberos für die Authentifizierung bei Amazon EMR.

Um die Knoten in einem Cluster über einen sicheren Kanal mithilfe des Secure Shell (SSH)-Protokolls zu authentifizieren und eine Verbindung zu ihnen herzustellen, erstellen Sie ein Amazon Elastic Compute Cloud (Amazon EC2)-Schlüsselpaar, bevor Sie den Cluster starten. Außerdem können Sie auch einen Cluster ohne ein Schlüsselpaar erstellen. Dies geschieht normalerweise mit vorübergehenden Clusters, die starten, gewisse Schritte ausführen und dann automatisch beendet werden.

Wenn …	Dann …
Sie haben bereits ein Amazon-EC2-Schlüsselpaar, das Sie verwenden möchten, oder Sie müssen sich nicht bei Ihrem Cluster authentifizieren.	Überspringen Sie diesen Schritt.
Sie müssen ein Schlüsselpaar erstellen.	Sehen Sie unter Erstellen Ihres Schlüsselpaares mithilfe von Amazon EC2.

Nächste Schritte

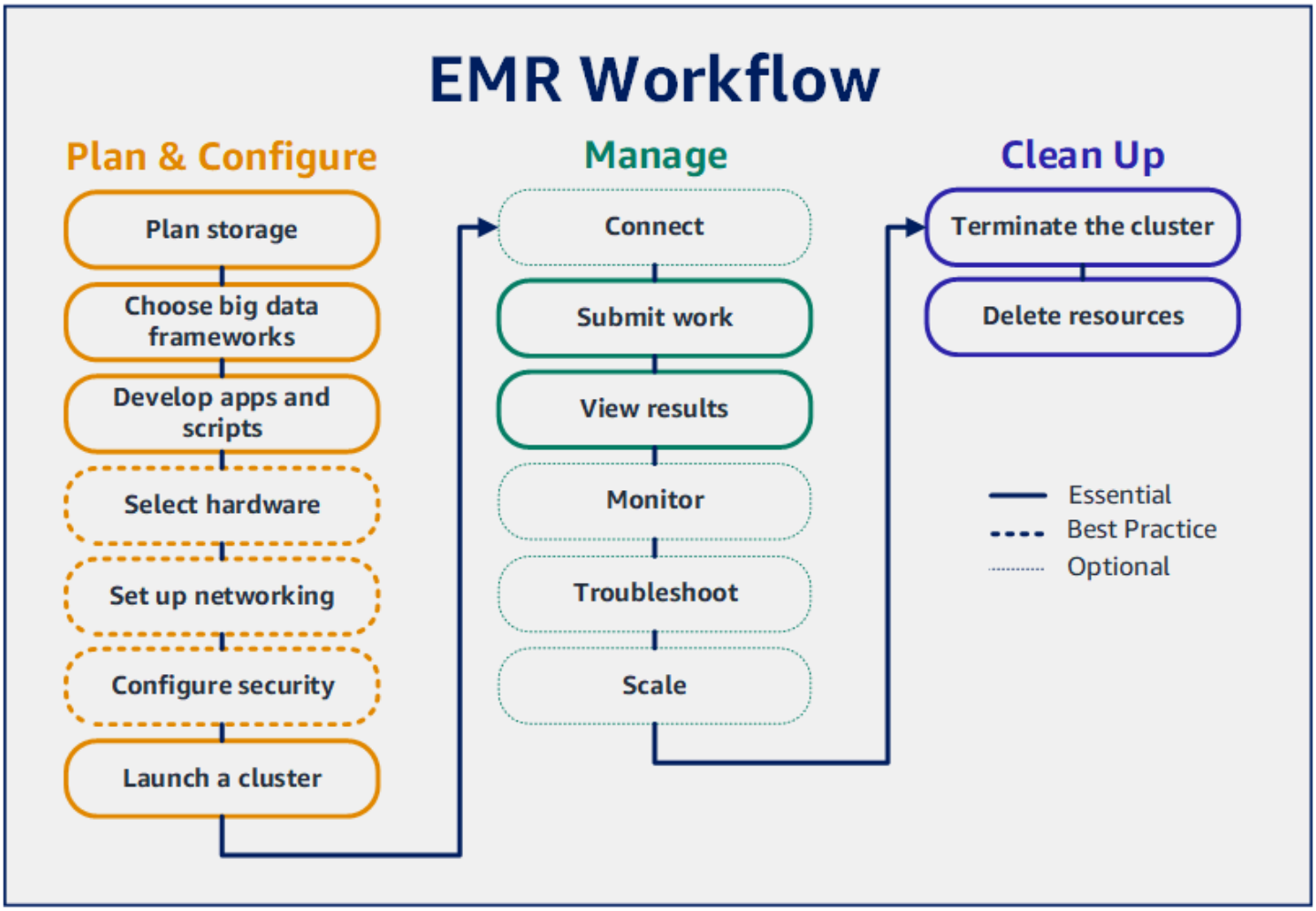
- Hinweise zur Erstellung eines Beispielclusters finden Sie unter Tutorial: Erste Schritte mit Amazon EMR.
- Weitere Informationen zur Konfiguration eines benutzerdefinierten Clusters und zur Steuerung des Zugriffs darauf finden Sie unter Cluster planen und konfigurieren und Sicherheit bei Amazon EMR.

Tutorial: Erste Schritte mit Amazon EMR

Übersicht

Mit Amazon können EMR Sie in wenigen Minuten einen Cluster einrichten, um Daten mit Big-Data-Frameworks zu verarbeiten und zu analysieren. Dieses Tutorial zeigt Ihnen, wie Sie einen Beispielcluster mit Spark starten und wie Sie ein einfaches PySpark Skript ausführen, das in einem Amazon S3 S3-Bucket gespeichert ist. Es behandelt wichtige EMR Amazon-Aufgaben in drei Hauptkategorien von Workflows: Planen und Konfigurieren, Verwalten und Aufräumen.

Während Sie das Tutorial durcharbeiten, finden Sie Links zu detaillierteren Themen und im Nächste Schritte Abschnitt Ideen für weitere Schritte. Wenn Sie Fragen haben oder nicht weiterkommen, wenden Sie sich in unserem Diskussionsforum an das EMR Amazon-Team.



Voraussetzungen

- Bevor Sie einen EMR Amazon-Cluster starten, stellen Sie sicher, dass Sie die Aufgaben in abgeschlossen habenEinrichten von Amazon EMR.

Kosten

- Der erstellte Beispiel-Cluster wird in einer Live-Umgebung ausgeführt. Für den Cluster fallen nur minimale Gebühren an. Stellen Sie sicher, dass Sie die Bereinigungsaufgaben im letzten Schritt dieses Tutorials ausführen, um zusätzliche Kosten zu vermeiden. Die Gebühren fallen pro Sekunde gemäß den EMR Amazon-Preisen an. Die Gebühren variieren auch je nach Region. Weitere Informationen finden Sie unter EMRAmazon-Preise.
- Für kleine Dateien, die Sie in Amazon S3 speichern, können geringe Gebühren anfallen. Einige oder alle Gebühren für Amazon S3 können erlassen werden, wenn Sie sich innerhalb der Nutzungsgrenzen des AWS kostenlosen Kontingents befinden. Weitere Informationen finden Sie unter Amazon-S3-Preise und AWS kostenloses Kontingent.

Schritt 1: Planung und Konfiguration eines EMR Amazon-Clusters

Speicher für Amazon vorbereiten EMR

Wenn Sie Amazon verwendenEMR, können Sie aus einer Vielzahl von Dateisystemen wählen, um Eingabedaten, Ausgabedaten und Protokolldateien zu speichern. In diesem Tutorial verwenden EMRFS Sie das Speichern von Daten in einem S3-Bucket. EMRFSist eine Implementierung des Hadoop-Dateisystems, mit der Sie reguläre Dateien in Amazon S3 lesen und schreiben können. Weitere Informationen finden Sie unter Mit Storage- und Dateisystemen arbeiten.

Um einen Bucket zu erstellen, befolgen Sie die Anweisungen unter Wie wird ein S3 Bucket erstellt? im Konsolen-Benutzerhandbuch zu Amazon Simple Storage Service. Erstellen Sie den Bucket in derselben AWS Region, in der Sie Ihren EMR Amazon-Cluster starten möchten. Zum Beispiel USA West (Oregon) us-west-2.

Für Buckets und Ordner, die Sie mit Amazon verwenden, gelten EMR die folgenden Einschränkungen:

- Namen können Kleinbuchstaben, Zahlen, Bindestriche (-) und Punkte (.) enthalten.
- Namen dürfen nicht mit Zahlen enden.

- Bucket-Namen müssen in allen AWS -Konten eindeutig sein.
- Ein Ausgabeordner muss leer sein.

Bereiten Sie eine Anwendung mit Eingabedaten für Amazon vor EMR

Die gängigste Methode, eine Bewerbung für Amazon vorzubereiten, EMR besteht darin, die Bewerbung und ihre Eingabedaten auf Amazon S3 hochzuladen. Wenn Sie dann Arbeit an Ihren Cluster senden, geben Sie die Amazon-S3-Speicherorte für Ihr Skript und Ihre Daten an.

In diesem Schritt laden Sie ein PySpark Beispielskript in Ihren Amazon S3 S3-Bucket hoch. Wir haben ein PySpark Skript bereitgestellt, das Sie verwenden können. Das Skript verarbeitet Inspektionsdaten von Lebensmittelbetrieben und gibt eine Ergebnisdatei in Ihrem S3-Bucket zurück. In der Ergebnisdatei sind die zehn Einrichtungen mit den meisten Verstößen vom Typ „Rot“ aufgeführt.

Sie laden auch Beispieleingabedaten in Amazon S3 hoch, damit das PySpark Skript sie verarbeiten kann. Bei den Eingabedaten handelt es sich um eine modifizierte Version der Inspektionsergebnisse des Gesundheitsministeriums in King County, Washington, von 2006 bis 2020. Weitere Informationen finden Sie unter King County Open Data: Daten zur Inspektion von Lebensmittelbetrieben. Nachfolgend sehen Sie einige Beispielzeilen aus dem Datensatz.

```
name, inspection_result, inspection_closed_business, violation_type, violation_points
100 LB CLAM, Unsatisfactory, FALSE, BLUE, 5
100 PERCENT NUTRICION, Unsatisfactory, FALSE, BLUE, 5
7-ELEVEN #2361-39423A, Complete, FALSE, , 0
```

Um das PySpark Beispielskript vorzubereiten für EMR

1. Kopieren Sie den Beispielcode unten mit einem Editor Ihrer Wahl in eine neue Datei.

```
import argparse

from pyspark.sql import SparkSession

def calculate_red_violations(data_source, output_uri):
    """
    Processes sample food establishment inspection data and queries the data to
 find the top 10 establishments
    with the most Red violations from 2006 to 2020.
```

```
    :param data_source: The URI of your food establishment data CSV, such as 's3://
DOC-EXAMPLE-BUCKET/food-establishment-data.csv'.
    :param output_uri: The URI where output is written, such as 's3://DOC-EXAMPLE-
BUCKET/restaurant_violation_results'.
    """
    with SparkSession.builder.appName("Calculate Red Health
 Violations").getOrCreate() as spark:
        # Load the restaurant violation CSV data
        if data_source is not None:
            restaurants_df = spark.read.option("header", "true").csv(data_source)

        # Create an in-memory DataFrame to query
        restaurants_df.createOrReplaceTempView("restaurant_violations")

        # Create a DataFrame of the top 10 restaurants with the most Red violations
        top_red_violation_restaurants = spark.sql("""SELECT name, count(*) AS
 total_red_violations
          FROM restaurant_violations
          WHERE violation_type = 'RED'
          GROUP BY name
          ORDER BY total_red_violations DESC LIMIT 10""")

        # Write the results to the specified output URI
        top_red_violation_restaurants.write.option("header",
 "true").mode("overwrite").csv(output_uri)

if __name__ == "__main__":
    parser = argparse.ArgumentParser()
    parser.add_argument(
        '--data_source', help="The URI for you CSV restaurant data, like an S3
 bucket location.")
    parser.add_argument(
        '--output_uri', help="The URI where output is saved, like an S3 bucket
 location.")
    args = parser.parse_args()

    calculate_red_violations(args.data_source, args.output_uri)
```

2. Speichern Sie die Datei als `health_violations.py`.
3. Laden Sie Ihre `health_violations.py` in Amazon S3 in den Bucket hoch, den Sie als Voraussetzung für dieses Tutorial erstellt haben. Anweisungen finden Sie unter Hochladen eines Objekts in Ihren Bucket im Handbuch „Erste Schritte“ für Amazon Simple Storage Service.

Um die Beispiel-Eingabedaten vorzubereiten für EMR

1. Laden Sie die ZIP-Datei food_establishment_data.zip herunter.
2. Entpacken und speichern Sie `food_establishment_data.zip` als `food_establishment_data.csv` auf Ihrem Computer.
3. Laden Sie die CSV Datei in den S3-Bucket hoch, den Sie für dieses Tutorial erstellt haben. Anweisungen finden Sie unter Hochladen eines Objekts in Ihren Bucket im Handbuch „Erste Schritte“ für Amazon Simple Storage Service.

Weitere Informationen zum Einrichten von Daten für EMR finden Sie unterEingabedaten vorbereiten.

Starten Sie einen EMR Amazon-Cluster

Nachdem Sie einen Speicherort und Ihre Anwendung vorbereitet haben, können Sie einen EMR Amazon-Beispielcluster starten. In diesem Schritt starten Sie einen Apache Spark-Cluster mit der neuesten EMRAmazon-Release-Version.

Console

Um einen Cluster zu starten, auf dem Spark zusammen mit der Konsole installiert ist

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und anschließend Create cluster aus.
3. Notieren Sie sich auf der Seite Cluster erstellen die Standardwerte für Version, Instance-Typ, Anzahl der Instances und Berechtigungen. Diese Felder werden automatisch mit Werten aufgefüllt, die für Allzweck-Cluster geeignet sind.
4. Geben Sie im Feld Clustername einen eindeutigen Clusternamen ein, um Ihren Cluster leichter identifizieren zu können, z. B. *`My first cluster`*. Ihr Clustername darf die Zeichen <, >, $, | oder `nicht enthalten (Backtick).
5. Wählen Sie unter Anwendungen die Spark-Option, um Spark auf Ihrem Cluster zu installieren.

> **Note**
>
> Wählen Sie die Anwendungen aus, die Sie in Ihrem EMR Amazon-Cluster haben möchten, bevor Sie den Cluster starten. Sie können nach dem Start keine Anwendungen zu einem Cluster hinzufügen oder daraus entfernen.

6. Aktivieren Sie unter Cluster-Protokolle das Kontrollkästchen Cluster-spezifische Protokolle in Amazon S3 veröffentlichen. Ersetzen Sie den Amazon-S3-Standortwert durch den Amazon-S3-Bucket, den Sie erstellt haben, gefolgt von **`/logs`**. Beispiel, **`s3://DOC-EXAMPLE-BUCKET/logs`**. Durch das Hinzufügen wird ein neuer Ordner namens „logs“ in Ihrem Bucket **`/logs`** erstellt, in den Amazon die Protokolldateien Ihres Clusters kopieren EMR kann.
7. Wählen Sie unter Sicherheitskonfiguration und Berechtigungen Ihr EC2key pair aus. Wählen Sie im selben Abschnitt das EMR Dropdownmenü Servicerolle für Amazon aus und wählen Sie EMR_ DefaultRole aus. Wählen Sie dann das Drop-down-Menü IAMRolle für Instanzprofil und wählen Sie EMR_ _ EC2. DefaultRole
8. Wählen Sie Cluster erstellen aus, um den Cluster zu starten und die Cluster-Detailseite zu öffnen.
9. Suchen Sie den Cluster-Status neben dem Clusternamen. Der Status ändert sich von Starting zu Running zu Waiting, wenn Amazon EMR den Cluster bereitstellt. Möglicherweise müssen Sie das Aktualisierungs-Symbol auf der rechten Seite betätigen oder Ihren Browser aktualisieren, um Updates zu sehen.

Ihr Clusterstatus ändert sich in Wartend, wenn der Cluster betriebsbereit ist, läuft und bereit ist, Arbeit anzunehmen. Weitere Informationen zum Lesen der Cluster-Zusammenfassung finden Sie unter Cluster-Status und -Details anzeigen. Weitere Informationen zu Cluster-Status finden Sie unter Verstehen des Cluster-Lebenszyklus.

CLI

Um einen Cluster zu starten, auf dem Spark mit dem installiert ist AWS CLI

1. Erstellen Sie IAM Standardrollen, die Sie dann verwenden können, um Ihren Cluster zu erstellen, indem Sie den folgenden Befehl verwenden.

```
aws emr create-default-roles
```

Weitere Informationen zu `create-default-roles` finden Sie in der AWS CLI - Befehlsreferenz.

2. Erstellen Sie einen Spark-Cluster mit dem folgenden Befehl. Geben Sie mit der `--name` Option einen Namen für Ihren Cluster ein und geben Sie mit der `--ec2-attributes` Option den Namen Ihres EC2 key pair an.

```
aws emr create-cluster \
--name "<My First EMR Cluster>" \
--release-label <emr-5.36.2> \
--applications Name=Spark \
--ec2-attributes KeyName=<myEMRKeyPairName> \
--instance-type m5.xlarge \
--instance-count 3 \
--use-default-roles
```

Notieren Sie sich die anderen erforderlichen Werte für `--instance-type`, `--instance-count` und `--use-default-roles`. Diese Werte wurden für Allzweck-Cluster ausgewählt. Weitere Informationen zu `create-cluster` finden Sie in der AWS CLI -Befehlsreferenz.

> **Note**
>
> Linux-Zeilenfortsetzungszeichen (\) sind aus Gründen der Lesbarkeit enthalten. Sie können entfernt oder in Linux-Befehlen verwendet werden. Entfernen Sie sie unter Windows oder ersetzen Sie sie durch ein Caret-Zeichen (^).

Die Ausgabe sollte ungefähr wie die folgende aussehen. Die Ausgabe zeigt `ClusterId` und `ClusterArn` Ihres neuen Clusters. Notieren Sie sich Ihre `ClusterId`. Sie verwenden `ClusterId`, um den Clusterstatus zu überprüfen und Arbeiten einzureichen.

```
{
    "ClusterId": "myClusterId",
    "ClusterArn": "myClusterArn"
}
```

3. Überprüfen Sie Ihren Clusterstatus mit dem folgenden Befehl.

```
aws emr describe-cluster --cluster-id <myClusterId>
```

Mit dem `Status`-Objekt für Ihren neuen Cluster sollten Sie eine Ausgabe wie die folgende sehen.

```
{
    "Cluster": {
        "Id": "myClusterId",
        "Name": "My First EMR Cluster",
        "Status": {
            "State": "STARTING",
            "StateChangeReason": {
                "Message": "Configuring cluster software"
            }
        }
     }
}
```

Der `State` Wert ändert sich von `STARTING` bis `RUNNING` zu`WAITING`, wenn Amazon EMR den Cluster bereitstellt.

Der Cluster-Status ändert sich zu **`WAITING`**, in dem ein Cluster betriebsbereit und bereit ist, Arbeit anzunehmen. Weitere Informationen zu Cluster-Status finden Sie unter Verstehen des Cluster-Lebenszyklus.

Schritt 2: Verwalten Sie Ihren EMR Amazon-Cluster

Arbeit bei Amazon einreichen EMR

Nachdem Sie einen Cluster gestartet haben, können Sie Arbeiten an den laufenden Cluster senden, um Daten zu verarbeiten und zu analysieren. In einem Schritt reichen Sie Arbeiten an einen EMR Amazon-Cluster ein. Ein Schritt ist eine Arbeitseinheit, die aus einer oder mehreren Aktionen besteht. Sie könnten beispielsweise einen Schritt zur Berechnung von Werten oder zur Übertragung und Verarbeitung von Daten einreichen. Sie können Schritte beim Erstellen eines Clusters oder an einen laufenden Cluster senden. In diesem Teil des Tutorials übermitteln Sie `health_violations.py` als Schritt an Ihren laufenden Cluster. Weitere Informationen zu Schritten finden Sie unter Übermitteln von Arbeit an einen Cluster.

Console

Um eine Spark-Anwendung als Schritt mit der Konsole einzureichen

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Cluster und wählen Sie dann den Cluster aus, für den Sie Arbeit einreichen möchten. Der Clusterstatus muss Wartend lauten.
3. Wählen Sie Schritte und dann Schritt hinzufügen.
4. Konfigurieren Sie den Schritt anhand der folgenden Richtlinien:

   - Wählen Sie für Typ die Option Spark-Anwendung aus. Sie sollten zusätzliche Felder für den Bereitstellungsmodus, den Speicherort der Anwendung und die Optionen Spark-Submit sehen.
   - Geben Sie unter Name einen neuen Namen ein. Wenn Sie viele Schritte in einem Cluster haben, hilft Ihnen die Benennung der einzelnen Schritte dabei, den Überblick zu behalten.
   - Behalten Sie für den Bereitstellungsmodus den Standardwert Clustermodus bei. Weitere Informationen zu Spark-Bereitstellungsmodi finden Sie unter Übersicht über den Clustermodus in der Apache-Spark-Dokumentation.
   - Geben Sie unter Anwendungsort den Speicherort Ihres `health_violations.py` Skripts in Amazon S3 ein, z. B. *`s3://DOC-EXAMPLE-BUCKET/health_violations.py`*.
   - Lassen Sie das Feld mit den Spark-Submit-Optionen leer. Weitere Informationen zu den `spark-submit`-Optionen finden Sie unter Starten von Anwendungen mit spark-submit.
   - Geben Sie im Feld Argumente die folgenden Argumente und Werte ein:

     ```
     --data_source s3://DOC-EXAMPLE-BUCKET/food_establishment_data.csv
     --output_uri s3://DOC-EXAMPLE-BUCKET/myOutputFolder
     ```

     Ersetzen *`s3://DOC-EXAMPLE-BUCKET/food_establishment_data.csv`* mit dem S3-Bucket URI der Eingabedaten, in dem Sie die Daten vorbereitet habenBereiten Sie eine Anwendung mit Eingabedaten für Amazon vor EMR.

     Ersetzen *`DOC-EXAMPLE-BUCKET`* mit dem Namen des Buckets, den Sie für dieses Tutorial erstellt haben, und ersetzen Sie *`myOutputFolder`* mit einem Namen für Ihren Cluster-Ausgabeordner.

- Übernehmen Sie unter Aktion bei Fehler des Schrittes die Standardeinstellung Fortfahren. Auf diese Weise wird der Cluster weiter ausgeführt, wenn der Schritt fehlschlägt.

5. Wählen Sie Hinzufügen, um den Schritt zu senden. Der Schritt wird in der Konsole mit dem Status Ausstehend angezeigt.
6. Überwachen Sie den Status des Schritts. Der Wert sollte sich von Ausstehend zu Wird ausgeführt zu Abgeschlossen ändern. Um den Status in der Konsole zu aktualisieren, wählen Sie das Aktualisierungssymbol rechts neben dem Filter aus. Die Ausführung des Skripts dauert etwa eine Minute. Wenn sich der Status in Abgeschlossen, ändert, wurde der Schritt erfolgreich abgeschlossen.

CLI

Um eine Spark-Anwendung als Schritt einzureichen, verwenden Sie AWS CLI

1. Stellen Sie sicher, dass Sie `ClusterId` des Clusters haben, den Sie in Starten Sie einen EMR Amazon-Cluster gestartet haben. Sie können Ihre Cluster-ID auch mit dem folgenden Befehl abrufen.

```
aws emr list-clusters --cluster-states WAITING
```

2. Senden Sie `health_violations.py` als Schritt mit dem `add-steps`-Befehl und Ihrem `ClusterId`.

   - Sie können einen Namen für Ihren Schritt angeben, indem Sie ihn ersetzen *"My Spark Application"*. Ersetzen Sie im `Args` Array *s3://DOC-EXAMPLE-BUCKET/health_violations.py* mit dem Standort Ihrer `health_violations.py` Anwendung.
   - Ersetzen *s3://DOC-EXAMPLE-BUCKET/food_establishment_data.csv* mit dem S3-Standort Ihres `food_establishment_data.csv` Datensatzes.
   - Ersetzen *s3://DOC-EXAMPLE-BUCKET/MyOutputFolder* mit dem S3-Pfad Ihres angegebenen Buckets und einem Namen für Ihren Cluster-Ausgabeordner.
   - `ActionOnFailure=CONTINUE` bedeutet, dass der Cluster weiter ausgeführt wird, wenn der Schritt fehlschlägt.

```
aws emr add-steps \
--cluster-id <myClusterId> \
```

```
--steps Type=Spark,Name="<My Spark
 Application>",ActionOnFailure=CONTINUE,Args=[<s3://DOC-EXAMPLE-
BUCKET/health_violations.py>,--data_source,<s3://DOC-EXAMPLE-BUCKET/
food_establishment_data.csv>,--output_uri,<s3://DOC-EXAMPLE-BUCKET/
MyOutputFolder>]
```

Weitere Informationen zum Senden von Schritten mithilfe von finden Sie in der AWS CLI Befehlsreferenz. CLI

Nachdem Sie den Schritt eingereicht haben, sollten Sie eine Ausgabe wie die folgende mit einer Liste von `StepIds` sehen. Da Sie einen Schritt eingereicht haben, wird in der Liste nur eine ID angezeigt. Kopieren Sie Ihre Schritt-ID. Sie verwenden Ihre Schritt-ID, um den Status des Schritts zu überprüfen.

```
{
    "StepIds": [
        "s-1XXXXXXXXXXA"
    ]
}
```

3. Fragen Sie den Status Ihres Schritts mit dem `describe-step`-Befehl ab.

```
aws emr describe-step --cluster-id <myClusterId> --step-id <s-1XXXXXXXXXXA>
```

Die Ausgabe sollte ungefähr wie die folgende aussehen, mit Informationen zu Ihrem Schritt.

```
{
    "Step": {
        "Id": "s-1XXXXXXXXXXA",
        "Name": "My Spark Application",
        "Config": {
            "Jar": "command-runner.jar",
            "Properties": {},
            "Args": [
                "spark-submit",
                "s3://DOC-EXAMPLE-BUCKET/health_violations.py",
                "--data_source",
                "s3://DOC-EXAMPLE-BUCKET/food_establishment_data.csv",
                "--output_uri",
                "s3://DOC-EXAMPLE-BUCKET/myOutputFolder"
```

```
            ]
        },
        "ActionOnFailure": "CONTINUE",
        "Status": {
            "State": "COMPLETED"
        }
    }
}
```

Der `State`-Wert des Schritts ändert sich mit der Ausführung des Schritts von `PENDING` zu `RUNNING` zu `COMPLETED`. Die Ausführung des Schritts dauert etwa eine Minute, sodass Sie den Status möglicherweise einige Male überprüfen müssen.

Sie wissen, dass der Schritt erfolgreich war, wenn sich `State` in **`COMPLETED`** ändert.

Weitere Informationen zum Schrittlebenszyklus finden Sie unter Ausführen von Schritten zur Verarbeitung von Daten.

Ergebnisse anzeigen

Nachdem ein Schritt erfolgreich ausgeführt wurde, können Sie seine Ausgabeergebnisse in Ihrem Amazon-S3-Ausgabeordner anzeigen.

So sehen Sie die Ergebnisse von **`health_violations.py`**

1. Öffnen Sie die Amazon S3 S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie den Bucket-Namen und dann den Ausgabeordner aus, den Sie beim Absenden des Schritts angegeben haben. Zum Beispiel *`DOC-EXAMPLE-BUCKET`* und dann *`myOutputFolder`*.
3. Stellen Sie sicher, dass die folgenden Elemente in Ihrem Ausgabeordner angezeigt werden:
   - Ein kleines Objekt namens `_SUCCESS`.
   - Eine CSV Datei, die mit dem Präfix beginnt`part-`, das Ihre Ergebnisse enthält.
4. Wählen Sie das Objekt mit Ihren Ergebnissen aus und klicken Sie dann auf Herunterladen, um die Ergebnisse in Ihrem lokalen Dateisystem zu speichern.
5. Öffnen Sie die Ergebnisse in Ihrem Editor Ihrer Wahl. In der Ausgabedatei sind die zehn Lebensmittelbetriebe mit den meisten roten Verstößen aufgeführt. Die Ausgabedatei zeigt auch die Gesamtzahl der roten Verstöße für jeden Betrieb.

Es folgt ein Beispiel für ein `health_violations.py`-Ergebnis.

```
name, total_red_violations
SUBWAY, 322
T-MOBILE PARK, 315
WHOLE FOODS MARKET, 299
PCC COMMUNITY MARKETS, 251
TACO TIME, 240
MCDONALD'S, 177
THAI GINGER, 153
SAFEWAY INC #1508, 143
TAQUERIA EL RINCONSITO, 134
HIMITSU TERIYAKI, 128
```

Weitere Informationen zur EMR Amazon-Cluster-Ausgabe finden Sie unterEinen Ausgabespeicherort konfigurieren.

(Optional) Connect zu Ihrem laufenden EMR Amazon-Cluster her

Wenn Sie Amazon verwendenEMR, möchten Sie möglicherweise eine Verbindung zu einem laufenden Cluster herstellen, um Protokolldateien zu lesen, den Cluster zu debuggen oder CLI Tools wie die Spark-Shell zu verwenden. EMRMit Amazon können Sie mithilfe des Secure Shell (SSH) - Protokolls eine Verbindung zu einem Cluster herstellen. In diesem Abschnitt erfahren Sie, wie Sie Ihren Cluster konfigurierenSSH, eine Verbindung zu Ihrem Cluster herstellen und Protokolldateien für Spark anzeigen. Weitere Informationen zum Herstellen einer Verbindung mit einem Cluster finden Sie unter Authentifizieren Sie sich Amazon EMR Amazon-Cluster-Knoten.

Autorisieren Sie SSH Verbindungen zu Ihrem Cluster

Bevor Sie eine Verbindung zu Ihrem Cluster herstellen, müssen Sie Ihre Cluster-Sicherheitsgruppen ändern, um eingehende SSH Verbindungen zu autorisieren. EC2Amazon-Sicherheitsgruppen dienen als virtuelle Firewalls zur Steuerung des ein- und ausgehenden Datenverkehrs zu Ihrem Cluster. Als Sie Ihren Cluster für dieses Tutorial EMR erstellt haben, hat Amazon in Ihrem Namen die folgenden Sicherheitsgruppen erstellt:

ElasticMapReduce-Master

Die standardmäßige von Amazon EMR verwaltete Sicherheitsgruppe, die dem primären Knoten zugeordnet ist. In einem EMR Amazon-Cluster ist der primäre Knoten eine EC2 Amazon-Instance, die den Cluster verwaltet.

ElasticMapReduce-Slave

Die Standardsicherheitsgruppe, die Core- und Aufgabenknoten zugeordnet ist.

Console

Um der primären Sicherheitsgruppe über die Konsole SSH Zugriff auf vertrauenswürdige Quellen zu gewähren

Um Ihre Sicherheitsgruppen bearbeiten zu können, benötigen Sie die Berechtigung, Sicherheitsgruppen für die Gruppe zu verwaltenVPC, in der sich der Cluster befindet. Weitere Informationen finden Sie unter Ändern der Berechtigungen für einen Benutzer und unter der Beispielrichtlinie, die die Verwaltung von EC2 Sicherheitsgruppen ermöglicht, im IAMBenutzerhandbuch.

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und dann den Cluster aus, den Sie aktualisieren möchten. Dadurch wird die Cluster-Detailseite geöffnet. Die Registerkarte Eigenschaften auf dieser Seite sollte vorausgewählt sein.
3. Wählen Sie auf der Registerkarte Eigenschaften unter Netzwerk den Pfeil neben EC2Sicherheitsgruppen (Firewall) aus, um diesen Abschnitt zu erweitern. Wählen Sie unter Primärknoten den Link zur Sicherheitsgruppe aus. Wenn Sie die folgenden Schritte abgeschlossen haben, können Sie optional zu diesem Schritt zurückkehren, Core- und Task-Knoten auswählen und die folgenden Schritte wiederholen, um dem SSH Client Zugriff auf Core- und Task-Knoten zu gewähren.
4. Daraufhin wird die EC2-Konsole geöffnet. Wählen Sie die Registerkarte Eingehende Regeln und anschließend Eingehende Regeln bearbeiten aus.
5. Suchen Sie mit den folgenden Einstellungen nach einer Regel für eingehenden Datenverkehr, die öffentlichen Zugriff ermöglicht. Falls sie existiert, wählen Sie Löschen, um sie zu entfernen.

   - Typ

SSH

- Port

  22

- Quelle

  Benutzerdefiniert 0.0.0.0/0

> **Warning**
>
> Vor Dezember 2020 verfügte die Sicherheitsgruppe ElasticMapReduce -master über eine vorkonfigurierte Regel, die eingehenden Datenverkehr auf Port 22 aus allen Quellen zuließ. Diese Regel wurde erstellt, um die ersten SSH Verbindungen zum Master-Knoten zu vereinfachen. Wir empfehlen Ihnen dringend, diese Eingangsregel zu entfernen und den Datenverkehr auf vertrauenswürdige Quellen zu beschränken.

6. Scrollen Sie zum Ende der Regelliste und wählen Sie Regel hinzufügen.
7. Wählen Sie als Typ die Option aus SSH. Bei Auswahl SSH werden automatisch Protokoll und 22 für Portbereich eingegeben TCP.
8. Wählen Sie als Quelle Meine IP aus, um Ihre IP-Adresse automatisch als Quelladresse hinzuzufügen. Sie können auch einen Bereich benutzerdefinierter vertrauenswürdiger Client-IP-Adressen hinzufügen oder zusätzliche Regeln für andere Clients erstellen. In vielen Netzwerkumgebungen werden IP-Adressen dynamisch zugewiesen, sodass Sie in Zukunft möglicherweise Ihre IP-Adressen für vertrauenswürdige Clients aktualisieren müssen.
9. Wählen Sie Save (Speichern) aus.
10. Wählen Sie optional Core- und Task-Knoten aus der Liste aus und wiederholen Sie die obigen Schritte, um dem SSH Client Zugriff auf Core- und Task-Knoten zu gewähren.

Connect zu Ihrem Cluster her, indem Sie AWS CLI

Unabhängig von Ihrem Betriebssystem können Sie mit dem eine SSH Verbindung zu Ihrem Cluster herstellen AWS CLI.

Um eine Verbindung zu Ihrem Cluster herzustellen und Protokolldateien anzuzeigen, verwenden Sie den AWS CLI

1. Verwenden Sie den folgenden Befehl, um eine SSH Verbindung zu Ihrem Cluster herzustellen. Ersetzen *<mykeypair.key>* mit dem vollständigen Pfad und Dateinamen Ihrer Schlüsselpaardatei. Beispiel, `C:\Users\<username>\.ssh\mykeypair.pem`.

   ```
   aws emr ssh --cluster-id <j-2AL4XXXXXX5T9> --key-pair-file <~/mykeypair.key>
   ```

2. Navigieren Sie zu `/mnt/var/log/spark`, um auf die Spark-Protokolle auf dem Hauptknoten Ihres Clusters zuzugreifen. Sehen Sie sich dann die Dateien an diesem Speicherort an. Eine Liste zusätzlicher Protokolldateien auf dem Hauptknoten finden Sie unter Protokolldateien auf dem Primärknoten anzeigen.

   ```
   cd /mnt/var/log/spark
   ls
   ```

Schritt 3: Bereinigen Sie Ihre EMR Amazon-Ressourcen

So beenden Sie Ihren Cluster

Nachdem Sie Arbeiten an Ihren Cluster übermittelt und die Ergebnisse Ihrer PySpark Bewerbung eingesehen haben, können Sie den Cluster beenden. Durch das Beenden eines Clusters werden alle mit dem Cluster verbundenen EMR Amazon-Gebühren und EC2 Amazon-Instances gestoppt.

Wenn Sie einen Cluster kündigen, EMR bewahrt Amazon Metadaten über den Cluster zwei Monate lang kostenlos auf. Archivierte Metadaten helfen Ihnen dabei, den Cluster für einen neuen Auftrag zu klonen oder die Cluster-Konfiguration zu Referenzzwecken wiederaufzugreifen. Zu den Metadaten gehören keine Daten, die der Cluster in S3 schreibt, oder Daten, die HDFS im Cluster gespeichert sind.

> **Note**
>
> Mit der EMR Amazon-Konsole können Sie nach dem Beenden des Clusters keinen Cluster aus der Listenansicht löschen. Ein beendeter Cluster verschwindet von der Konsole, wenn Amazon seine Metadaten EMR löscht.

Console

Um den Cluster mit der Konsole zu beenden

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie Clusters und dann den Cluster aus, den Sie beenden möchten.
3. Wählen Sie im Dropdownmenü Aktionen die Option Cluster beenden aus.
4. Wählen Sie im Dialogfenster Beenden. Je nach Clusterkonfiguration kann die Kündigung 5 bis 10 Minuten dauern. Weitere Informationen zur Verwendung von EMR Amazon-Clustern finden Sie unterEinen Cluster beenden.

CLI

Um den Cluster mit dem zu beenden AWS CLI

1. Initiieren Sie den Vorgang zur Clusterbeendigung mit dem folgenden Befehl. Ersetzen *<myClusterId>* mit der ID Ihres Probenclusters. Der Befehl gibt keine Ausgabe zurück.

```
aws emr terminate-clusters --cluster-ids <myClusterId>
```

2. Um zu überprüfen, ob der Clusterbeendigungsprozess im Gange ist, überprüfen Sie den Clusterstatus mit dem folgenden Befehl.

```
aws emr describe-cluster --cluster-id <myClusterId>
```

Im Folgenden finden Sie eine Beispielausgabe im JSON Format. Der Cluster `Status` sollte sich von **`TERMINATING`** zu **`TERMINATED`** ändern. Die Beendigung kann je nach Clusterkonfiguration 5 bis 10 Minuten dauern. Weitere Informationen zum Beenden eines EMR Amazon-Clusters finden Sie unterEinen Cluster beenden.

```
{
    "Cluster": {
        "Id": "j-xxxxxxxxxxxx",
        "Name": "My Cluster Name",
        "Status": {
            "State": "TERMINATED",
            "StateChangeReason": {
                "Code": "USER_REQUEST",
```

```
                "Message": "Terminated by user request"
            }
        }
    }
}
```

Löschen von S3-Ressourcen

Um zusätzliche Gebühren zu vermeiden, sollten Sie Ihren Amazon-S3-Bucket löschen. Durch das Löschen des Buckets werden alle Amazon-S3-Ressourcen für dieses Tutorial entfernt. Ihr Bucket sollte Folgendes enthalten:

- Das Skript PySpark
- Der Eingabedatensatz
- Ihr Ordner mit den Ausgabeergebnissen
- Ihr Ordner für Protokolldateien

Möglicherweise müssen Sie zusätzliche Schritte unternehmen, um gespeicherte Dateien zu löschen, wenn Sie PySpark das Skript oder die Ausgabe an einem anderen Ort gespeichert haben.

> **Note**
>
> Ihr Cluster muss beendet werden, bevor Sie Ihren Bucket löschen können. Andernfalls dürfen Sie den Bucket möglicherweise nicht leeren.

Um Ihren Bucket zu löschen, befolgen Sie die Anweisungen unter Wie lösche ich einen S3-Bucket? im Benutzerhandbuch zu Amazon Simple Storage Service.

Nächste Schritte

Sie haben jetzt Ihren ersten EMR Amazon-Cluster von Anfang bis Ende gestartet. Sie haben auch wichtige EMR Aufgaben wie das Vorbereiten und Einreichen von Big-Data-Anwendungen, das Anzeigen von Ergebnissen und das Beenden eines Clusters erledigt.

In den folgenden Themen erfahren Sie mehr darüber, wie Sie Ihren EMR Amazon-Workflow anpassen können.

Erkunden Sie Big-Data-Anwendungen für Amazon EMR

Entdecken und vergleichen Sie die Big-Data-Anwendungen, die Sie auf einem Cluster installieren können, im Amazon EMR Release Guide. Der Versionshandbuch beschreibt jede EMR Release-Version und enthält Tipps zur Verwendung von Frameworks wie Spark und Hadoop auf AmazonEMR.

Planen Sie Cluster-Hardware, Netzwerke und Sicherheit

In diesem Tutorial haben Sie einen einfachen EMR Cluster erstellt, ohne erweiterte Optionen zu konfigurieren. Mit den erweiterten Optionen können Sie EC2 Amazon-Instance-Typen, Cluster-Netzwerke und Cluster-Sicherheit angeben. Weitere Informationen zur Planung und Einführung eines Clusters, der Ihren Anforderungen entspricht, finden Sie unter Cluster planen und konfigurieren und Sicherheit bei Amazon EMR.

Verwalten von Clustern

Erfahren Sie mehr über die Arbeit mit laufenden Clustern unter Verwalten von Clustern. Um einen Cluster zu verwalten, können Sie eine Verbindung zum Cluster herstellen, Schritte debuggen und die Clusteraktivitäten und den Zustand verfolgen. Mit EMRverwalteter Skalierung können Sie die Cluster-Ressourcen auch an die Workload-Anforderungen anpassen.

Verwenden Sie eine andere Schnittstelle

Zusätzlich zur EMR Amazon-Konsole können Sie Amazon EMR über den AWS Command Line Interface, den Webservice API oder einen der vielen unterstützten verwalten AWS SDKs. Weitere Informationen finden Sie unter Verwaltungsschnittstellen.

Sie können auch auf vielfältige Weise mit Anwendungen interagieren, die auf EMR Amazon-Clustern installiert sind. Einige Anwendungen wie Apache Hadoop veröffentlichen Weboberflächen, die Sie sich ansehen können. Weitere Informationen finden Sie unter Auf EMR Amazon-Clustern gehostete Weboberflächen anzeigen.

Stöbern Sie im EMR technischen Blog

Beispielhafte Komplettlösungen und ausführliche technische Diskussionen zu neuen EMR Amazon-Funktionen finden Sie im AWS Big-Data-Blog.

Amazon EMR-Konsole

Die Konsole bietet eine aktualisierte Oberfläche, die Ihnen eine intuitive Möglichkeit bietet, Ihre Amazon EMR-Umgebung zu verwalten, und bietet Ihnen bequemen Zugriff auf Dokumentation, Produktinformationen und andere Ressourcen.

Funktionen der Konsole

Die Amazon EMR-Konsole ist unter der folgenden URL verfügbar:

- Konsolen-URL — https://console.aws.amazon.com/emr

In der folgenden Tabelle ist der Status der wichtigsten Komponenten der Amazon EMR-Konsole aufgeführt.

Komponente für Amazon-EMR-Konsole	Konsole	
EMR Studio	✓	
Cluster erstellen und verwalten	✓	
Blockieren des öffentlichen Zugriffs	✓	
Überwachen Sie CloudWatch Amazon-Ereignisse	✓	
Sicherheitskonfigurationen	✓	
Virtuelle Cluster (Amazon EMR in EKS)	✓	
Ihre Amazon Virtual Private Cloud-Subnetze anzeigen und verwalten 1	✓	
Notizbücher 2	✓	

[1] In der Konsole können Sie Ihre Amazon VPC-Subnetze im Bereich Netzwerk anzeigen und verwalten, wenn Sie einen Cluster erstellen.

[2] EMR Notebooks sind als EMR Studio-Workspaces in der Konsole verfügbar. Mit der Schaltfläche „Arbeitsbereich erstellen“ in der Konsole können Sie neue Notizbücher erstellen. Um auf Workspaces zuzugreifen oder diese zu erstellen, benötigen EMR-Notebook-Benutzer zusätzliche IAM-Rollenberechtigungen. Weitere Informationen finden Sie unter Amazon EMR Notebooks sind Amazon EMR Studio Workspaces in der Konsole und Amazon EMR-Konsole.

Zusammenfassung der Unterschiede

In diesem Abschnitt werden die Funktionen der Amazon EMR-Konsole beschrieben. Diese Funktionen lassen sich in die folgenden Kategorien einteilen:

- Cluster-Kompatibilität in der Konsole
- Cluster erstellen
- Clusterdetails anzeigen oder bearbeiten
- Cluster anzeigen und nach ihnen suchen
- Unterschiede bei der Arbeit mit Sicherheitskonfigurationen

Cluster-Kompatibilität in der Konsole

In einigen Fällen ist ein von Ihnen erstellter Cluster möglicherweise nicht mit der Konsole kompatibel. In der folgenden Liste werden die Kompatibilitätsanforderungen für die Amazon EMR-Konsole beschrieben.

- Die Konsole unterstützt Cluster, die in den Amazon EMR-Versionen 5.20.1 und höher erstellt wurden.
- Sie können Cluster mit automatischer Skalierung in der Konsole klonen, aber Sie können nur neue Cluster erstellen, wenn Sie sie manuell skalieren oder verwaltete Skalierung verwenden möchten.

Um Cluster der Version 5.20.1 und früher zu erstellen und mit ihnen zu arbeiten, können Sie das AWS Command Line Interface (AWS CLI) oder das AWS SDK verwenden.

Cluster erstellen

Funktion	Konsole	
Terminologie: Amazon-EMR-Cluster-Knotentypen	Primär, Core, Aufgabe	
Von Amazon EMR unterstützte Versionen 1	Amazon-EMR-Version 5.20.1 und höher	
Schnelles Starten eines Clusters	Verwenden Sie die Schaltfläche „Cluster erstellen“ im Bereich „Zusammenfassung“. Ihr Clustername darf die Zeichen <, >, $, | oder `(Backtick) nicht enthalten.	
Konfiguration eines Timeouts für die Spot-Bereitstellung	Definieren Sie einen Timeout-Zeitraum für die Bereitstellung von Instances für jede Flotte in Ihrem Cluster.	
Servicerollen und Amazon-EC2-Instance-Profilrolle	Die Konsole erstellt keine Standardrollen. Sie müssen Rollen mit der IAM-Konsole erstellen oder eine bereits erstellte IAM-Rolle auswählen	
Transparenz der Cluster	Von der Amazon-EMR-Konsole aus können Sie einen Cluster nicht für alle Benutzer sichtbar machen. Ihre IAM-Richtlinie bestimmt den Clusterzugriff	

Funktion	Konsole	
Netzwerk – konfigurieren Sie private Subnetze	Sie müssen Amazon-S3 -Endpunkte und NAT-Gateways von ihren jeweiligen Amazon-S3- und Amazon-VPC-Konsolen aus konfigurieren.	
Konsistente Ansicht des EMR-Dateisystems (EMRFS CV)	Mit der Veröffentlichung von Amazon S3 Strong read-after-write Consistency am 1. Dezember 2020 müssen Sie EMRFS CV nicht mit Ihren EMR-Clustern verwenden	
Debugging	Sie können Aufträge mithilfe der Benutzeroberfläche der Anwendung auf der Cluster-Detailseite debuggen	

[1] Sie können in der Konsole keine Cluster mit Versionen vor Amazon EMR 5.20.1 erstellen oder bearbeiten, aber alle vorhandenen Cluster, die mit Versionen vor 5.20.1 erstellt wurden, funktionieren weiterhin. Verwenden Sie die API oder CLI, um Cluster mit Amazon EMR-Versionen vor 5.20.1 zu erstellen und zu bearbeiten. Sie können alle Cluster mit der Konsole anzeigen, aber Konsolen, die vor 5.20.1 erstellt wurden, sind möglicherweise nicht mit neueren Funktionen kompatibel.

Cluster anzeigen und nach ihnen suchen

In der folgenden Tabelle wird erläutert, wie Sie die Amazon EMR-Konsole verwenden können, um Cluster anzuzeigen und nach ihnen zu suchen.

> **Note**
>
> Wenn Sie einen Datenfilter auf die Cluster-Liste anwenden, wird die gesamte Datenbank abgefragt. Wenn Sie jedoch eine Textzeichenfolge in das Suchfeld eingeben, gilt die Suche nur für die Ergebnisse, die die Liste clientseitig geladen hat.

Funktion	Konsole	
Anzeigen von Cluster-Details	Sie können die Cluster-ID auswählen, um umfassende Cluster-Details wie Konfigurationsoptionen, persistente Anwendungs-Benutzeroberflächen und Protokolle anzuzeigen.	
Suchen nach Clustern	Verwenden Sie ein einziges Suchfeld, um Textsuchabfragen einzugeben und Datenfilter wie „Status = Beliebiger aktiver Status“ zu erstellen und anzuwenden.	
Nach ausgefallenen Clustern suchen	Um nach ausgefallenen Clustern zu suchen, wenden Sie den Filter Status = Mit Fehlern beendet an.	

Clusterdetails anzeigen oder bearbeiten

Funktion	Konsole	

Funktion	Konsole	
Anzeige der Instances in Ihren Instance-Gruppen und Instance-Flotten sowie der Optionen für Skalierung, Bereitstellung, Größenänderung und Kündigung	Instance-Optionen und -Details finden Sie auf der Registerkarte Instances. Sehen Sie sich die Kündigungsoptionen auf der Registerkarte „Eigenschaften“ an.	
Benutzeroberflächen, Protokolle und Konfigurationen von Apps anzeigen (Apache-Spark-Benutzeroberfläche, Spark-Verlaufsservice, Apache-Tez-Benutzeroberfläche, YARN-Zeitleistenserver)	Sehen Sie sich die Cluster-Konfigurationen auf der Registerkarte Konfigurationen an. Starten Sie eine persistente Live-Anwendungsoberfläche, um die Protokolle für eine Anwendung auf der Registerkarte „Anwendungen“ anzuzeigen.	
Exportieren eines Clusters nach CLI	Die Option ist in den Aktionsmenüs „Cluster-Detail“ und „Listenansicht“ als „Befehl zum Klonen eines Clusters anzeigen“ verfügbar	

Unterschiede bei der Arbeit mit Sicherheitskonfigurationen

Funktion	Konsole	
Klonen von Sicherheitskonfigurationen	✓	
Föderierte Verwaltung mit Trino und Apache Ranger	✓	

Funktion	Konsole	
Eine Runtime-Rolle verwenden, um Arbeit an einen Cluster zu übermitteln [1]	✓	
Zugriff auf EMR-Dateisystem (EMRFS)	Amazon S3 Access Points	
AWS Lake Formation Zugriffskontrollen	Laufzeit-Rollen	

[1] Um eine Rolle während der Schrittübermittlung zu übergeben, muss Ihr Cluster eine Sicherheitskonfiguration mit einer angehängten IAM-Berechtigungsrichtlinie verwenden, sodass ein Benutzer nur die genehmigten Rollen weitergeben kann und Ihre Aufträge auf Amazon-EMR-Ressourcen zugreifen können. Weitere Informationen finden Sie unter EMRSchritte zu Runtime-Rollen für Amazon.

Amazon EMR Studio

Amazon EMR Studio ist eine webbasierte integrierte Entwicklungsumgebung (IDE) für vollständig verwaltete Jupyter Notebooks, die auf Amazon-EMR-Clustern ausgeführt werden. Sie können ein EMR Studio für Ihr Team einrichten, um in R, Python, Scala und geschriebene Anwendungen zu entwickeln, zu visualisieren und zu debuggen PySpark. EMR Studio ist in AWS Identity and Access Management (IAM) und IAM Identity Center integriert, sodass sich Benutzer mit ihren Unternehmensanmeldedaten anmelden können.

Sie können ein EMR Studio kostenlos erstellen. Wenn Sie EMR Studio verwenden, fallen Gebühren für Amazon-S3-Speicher und Amazon-EMR-Cluster an. Highlights, weitere Produktdetails und Preise finden Sie auf der Serviceseite für Amazon EMR Studio.

Hauptfeatures von EMR Studio

Amazon EMR Studio bietet die folgenden Features:

- Authentifizieren Sie Benutzer mit AWS Identity and Access Management (IAM) oder AWS IAM Identity Center mit oder ohne Verbreitung vertrauenswürdiger Identitäten und Ihrem Unternehmensidentitätsanbieter.
- Greifen Sie bei Bedarf auf Amazon-EMR-Cluster zu und starten Sie sie, um Jupyter-Notebook-Aufträge auszuführen.
- Stellen Sie auf EKS-Clustern eine Verbindung zu Amazon EMR her, um Arbeit einzureichen, während der Auftrag ausgeführt wird.
- Erkunden und speichern Sie Beispiel-Notebooks. Weitere Informationen zu Beispiel-Notebooks finden Sie im EMR Studio Notebook-Beispiel- GitHub Repository .
- Analysieren Sie Daten mit Python, PySpark, Spark Scala, Spark R oder SparkSQL und installieren Sie benutzerdefinierte Kernel und Bibliotheken.
- Arbeiten Sie in Echtzeit mit anderen Benutzern in demselben Workspace zusammen. Weitere Informationen finden Sie unter Konfigurieren Sie die Zusammenarbeit im Workspace.
- Verwenden Sie den EMR Studio SQL Explorer, um Ihren Datenkatalog zu durchsuchen, SQL-Abfragen auszuführen und Ergebnisse herunterzuladen, bevor Sie mit den Daten in einem Notebook arbeiten.
- Führen Sie parametrisierte Notebooks als Teil von geplanten Workflows mit einem Orchestrierungstool wie Apache Airflow oder Amazon Managed Workflows für Apache Airflow aus.

Weitere Informationen finden Sie unter Orchestrieren von Analyseaufträgen auf EMR Notebooks mithilfe von MWAA im AWS-Big-Data-Blog.

- Verknüpfen Sie Code-Repositorys wie GitHub und BitBucket.
- Verfolgen und debuggen Sie Jobs mit dem Spark History Server, der Tez-Benutzeroberfläche oder dem YARN-Timeline-Server.

EMR Studio ist auch HIPAA-fähig und nach HITRUST CSF und SOC 2 zertifiziert. Weitere Informationen über HIPAA-Compliance für AWS-Services finden Sie unter https://aws.amazon.com/compliance/hipaa-compliance/. Weitere Informationen zur HITRUST CSF-Konformität für AWS-Services finden Sie unter https://aws.amazon.com/compliance/hitrust/. Weitere Informationen zu anderen Compliance-Programmen für AWS-Services finden Sie unter AWS-Services im Leistungsumfang nach Compliance-Programmen.

Verlauf der Features von Amazon EMR Studio

In dieser Tabelle sind Aktualisierungen zur Funktion Amazon EMR Managed Scaling aufgeführt.

Datum der Veröffentlichung	Funktion
5. Januar 2024	Unterstützung für EMR Studio wurde in AWS GovCloud (USA-Ost) und AWS GovCloud (USA-West) hinzugefügt.
26. November 2023	Unterstützung für die Verbeitung vertrauenswürdiger Identitäten für EMR Studio mit IAM-Identity-Center-Authentifizierung hinzugefügt.
26. Oktober 2023	Es wurde die Möglichkeit hinzugefügt, eine Serverless-EMR-Anwendung mit interaktiven Funktionen zu erstellen.
28. Februar 2023	Kundenverwaltete AWS KMS-Schlüsselunterstützung für die Speicherung von Anwendungsprotokollen für Serverless-EMR-Anwendungen hinzugefügt.
23. Februar 2023	Es wurde die Erstellung von IAM-Rollen mit einem Klick für die Serverless-EMR-Auftragsübermittlung hinzugefügt. ECR-Suche hinzugefügt, wenn Sie ein benutzerdefiniertes Image für EMR-Serverless-Anwendungen auswählen.

Datum der Veröffentlichung	Funktion
27. Januar 2023	Notebooks mit Headless-Ausführung können den Fortschritt jeder einzelnen Zellenausführung `%execute_notebook` auf magische Weise verfolgen.
23. Januar 2023	Persistente Anwendungen wurden für schnellere Startzeiten optimiert.

Wie Amazon EMR Studio funktioniert

Ein Amazon EMR Studio ist eine Amazon-EMR-Ressource, die Sie für ein Team von Benutzern erstellen. EMR Studio ist eine webbasierte, integrierte Entwicklungsumgebung für vollständig verwaltete Jupyter-Notebooks, die auf Amazon-EMR-Clustern ausgeführt werden. Benutzer melden sich mit Unternehmensanmeldeinformationen bei einem Studio an.

Jedes EMR Studio, das Sie erstellen, verwendet die folgenden AWS-Ressourcen:

- Eine Amazon Virtual Private Cloud (VPC) mit Subnetzen – Benutzer führen Studio-Kernel und -Anwendungen auf Amazon EMR und Amazon EMR auf EKS-Clustern in der angegebenen VPC aus. Ein EMR Studio kann eine Verbindung zu jedem Cluster in den Subnetzen herstellen, die Sie beim Erstellen des Studios angeben.
- IAM-Rollen und Berechtigungsrichtlinien – Um Benutzerberechtigungen zu verwalten, erstellen Sie IAM-Berechtigungsrichtlinien, die Sie der IAM-Identität eines Benutzers oder einer Benutzerrolle zuordnen. EMR Studio verwendet auch eine IAM-Servicerolle und Sicherheitsgruppen, um mit anderen AWS-Services zusammenzuarbeiten. Weitere Informationen erhalten Sie unter Zugriffskontrolle und Definieren Sie Sicherheitsgruppen zur Steuerung des EMR Studio-Netzwerkverkehrs.
- Sicherheitsgruppen – EMR Studio verwendet Sicherheitsgruppen, um einen sicheren Netzwerkkanal zwischen dem Studio und einem EMR-Cluster einzurichten.
- Ein Amazon-S3-Backup-Speicherort – EMR Studio speichert Notebookarbeiten an einem Amazon-S3-Speicherort.

In den folgenden Schritten wird beschrieben, wie Sie ein EMR Studio erstellen und verwalten:

1. Erstellen Sie ein Studio in Ihrem AWS-Konto mit entweder IAM- oder IAM-Identity-Center-Authentifizierung. Detaillierte Anweisungen finden Sie unter Richten Sie ein Amazon EMR Studio ein.
2. Weisen Sie Ihrem Studio Benutzer und Gruppen zu. Verwenden Sie Berechtigungsrichtlinien, um detaillierte Berechtigungen für jeden Benutzer festzulegen. Weitere Informationen finden Sie im Thema EMRStudio-Benutzer zuweisen und verwalten .
3. Beginnen Sie mit der Überwachung von EMR-Studio-Aktionen mit AWS CloudTrail-Ereignissen. Weitere Informationen finden Sie unter Amazon EMR Studio-Aktionen überwachen.
4. Bieten Sie Studio-Benutzern mehr Cluster-Optionen mit Cluster-Vorlagen und Amazon EMR in EKS-verwalteten Endpunkten.

Authentifizierung und Benutzeranmeldung

Amazon EMR Studio unterstützt zwei Authentifizierungsmodi: den IAM-Authentifizierungsmodus und den IAM-Identity-Center-Authentifizierungsmodus. Der IAM-Modus verwendet AWS Identity and Access Management (IAM), während der IAM-Identity-Center-Modus AWS IAM Identity Center verwendet. Wenn Sie ein EMR Studio erstellen, wählen Sie den Authentifizierungsmodus für alle Benutzer dieses Studios.

IAM-Authentifizierungsmodus

Im IAM-Authentifizierungsmodus können Sie entweder die IAM-Authentifizierung oder den IAM-Verbund verwenden.

Mit der IAM-Authentifizierung können Sie IAM-Identitäten wie Benutzer, Gruppen und Rollen in IAM verwalten. Sie gewähren Benutzern Zugriff auf ein Studio mit IAM-Berechtigungsrichtlinien und attributbasierter Zugriffskontrolle (ABAC).

Mit dem IAM-Verbund können Sie Vertrauen zwischen einem externen Identitätsanbieter (IdP) aufbauen und AWS-Benutzeridentitäten über Ihren IdP verwalten.

Authentifizierungsmodus von IAM Identity Center

Mit dem IAM-Identity-Center-Authentifizierungsmodus können Sie Benutzern Verbundzugriff auf ein EMR Studio gewähren. Sie können IAM Identity Center verwenden, um Benutzer und Gruppen aus Ihrem IAM-Identity-Center-Verzeichnis, Ihrem vorhandenen Unternehmensverzeichnis oder einem externen IdP wie Azure Active Directory (AD) zu authentifizieren. Sie verwalten dann Benutzer mit Ihrem Identitätsanbieter (IdP).

EMR Studio unterstützt die Verwendung der folgenden Identitätsanbieter für IAM Identity Center:

- AWS Managed Microsoft AD und selbstverwaltetes Active Directory – Weitere Informationen finden Sie unter Verbindung mit Ihrem Microsoft-AD-Verzeichnis herstellen.
- SAML-basierte Anbieter – Eine vollständige Liste finden Sie unter Unterstützte Identitätsanbieter.
- Das Verzeichnis von IAM Identity Center – Weitere Informationen finden Sie unter Identitäten in IAM Identity Center verwalten und Verbreitung vertrauenswürdiger Identitäten zwischen Anwendungen im AWS IAM Identity Center-Benutzerhandbuch.

Wie sich die Authentifizierung auf die Anmeldung und die Benutzerzuweisung auswirkt

Der Authentifizierungsmodus, den Sie für EMR Studio wählen, wirkt sich darauf aus, wie sich Benutzer bei einem Studio anmelden, wie Sie einen Benutzer einem Studio zuweisen und wie Sie Benutzer autorisieren (ihnen Berechtigungen erteilen), Aktionen wie das Erstellen neuer Amazon-EMR-Cluster auszuführen.

In der folgenden Tabelle sind die Anmeldemethoden für EMR Studio nach Authentifizierungsmodus zusammengefasst.

EMR-Studio-Anmeldeoptionen nach Authentifizierungsmodus

Authentifizierungsmodus	Anmelde-Methode	Beschreibung
• IAM (Authentifizierung und Verbund) • IAM Identity Center	EMR-Studio-URL	Benutzer melden sich mit der Studio-Zugriffs-URL bei einem Studio an. Beispiel: `https://xxxxxxxxxxxxxxxxxxxxxx.emrstudio-prod.us-east-1.amazonaws.com` . Benutzer geben IAM-Anmeldeinformationen ein, wenn Sie die IAM-Authentifizierung verwenden. Wenn Sie IAM-Verbund oder IAM Identity Center verwenden, leitet EMR Studio Benutzer zur Eingabe der Anmeldeinformationen zur Anmelde-URL Ihres Identitätsanbieters weiter.

Authentifizierungsmodus	Anmelde-Methode	Beschreibung
		Im Zusammenhang mit dem Identitätsverbund wird diese Anmeldeoption als vom Servicean bieter (SP) initiierte Anmeldung bezeichnet.
• IAM (Verbund) • IAM Identity Center	Identitätsanbieter-(IdP)-Portal	Benutzer melden sich beim Portal Ihres Identität sanbieters an, z. B. beim Azure-Portal, und starten die Amazon-EMR-Konsole. Nach dem Start der Amazon-EMR-Konsole wählen Benutzer ein Studio aus der Studio-Liste aus und öffnen es. Sie können EMR Studio auch als SAML-Anwe ndung konfigurieren, sodass sich Benutzer über das Portal Ihres Identitätsanbieters bei einem bestimmten Studio anmelden können. Anweisung en finden Sie unter So konfigurieren Sie ein EMR Studio als SAML-Anwendung in Ihrem IdP-Portal. Im Zusammenhang mit dem Identitätsverbund wird diese Anmeldeoption als vom Identität sanbieter (IdP) initiierte Anmeldung bezeichnet.
• IAM (Authenti fizierung)	AWS Management Console	Benutzer melden sich bei AWS Management Console mit IAM-Anmeldeinformationen an und öffnen ein Studio aus der Studios-Liste in der Amazon-EMR-Konsole.

In der folgenden Tabelle werden die Benutzerzuweisung und Autorisierung für EMR Studio nach Authentifizierungsmodus beschrieben.

EMR Studio Benutzerzuweisung und Autorisierung im Authentifizierungsmodus

Authentifizierungsmodus	Benutzerzuweisung	Benutzer-Autorisierung
IAM (Authentifizierung und Verbund)	Zulassen der `CreateStudioPresignedUrl` Aktion in einer IAM-Berechtigungsrichtlinie, die an eine IAM-Identität (Benutzer, Gruppe oder Rolle) angefügt ist. Lassen Sie für Verbundbenutzer die `CreateStudioPresignedUrl` -Aktion in einem IAM in der Berechtigungsrichtlinie zu, die Sie für die IAM-Rolle konfigurieren, die Sie für den Verbund verwenden. Verwenden Sie die attributbasierte Zugriffskontrolle (ABAC), um das Studio oder die Studios anzugeben, auf die der Benutzer zugreifen kann. Detaillierte Anweisungen finden Sie unter Weisen Sie einem EMR Studio einen Benutzer oder eine Gruppe zu.	Definieren Sie IAM-Berechtigungsrichtlinien, die bestimmte EMR-Studio-Aktionen zulassen. Hängen Sie für native Benutzer die IAM-Berechtigungsrichtlinie an eine IAM-Identität (Benutzer, Gruppe oder Rolle) an. Lassen Sie für Verbundbenutzer Studio-Aktionen in der Berechtigungsrichtlinie zu, die Sie für die IAM-Rolle konfigurieren, die Sie für den Verbund verwenden. Weitere Informationen finden Sie unter EMRStudio-Benutzerberechtigungen für Amazon EC2 oder Amazon konfigurieren EKS.
IAM Identity Center	Ordnen Sie bei Studios, bei denen `IdCUserAssignment` auf `REQUIRED` eingestellt wurde, Benutzer dem Studio mit einer bestimmten Sitzungsrichtlinie zu. Weitere Informationen finden Sie unter Weisen Sie einem EMR	Optional: Definieren Sie IAM-Sitzungsrichtlinien, die bestimmte EMR-Studio-Aktionen zulassen. Ordnen Sie einem Benutzer eine Sitzungsrichtlinie zu, wenn Sie ihn einem Studio zuweisen.

Authentifizierungsmodus	Benutzerzuweisung	Benutzer-Autorisierung
	Studio einen Benutzer oder eine Gruppe zu. Bei Studios, bei denen `IdCUserAssignment` auf `OPTIONAL` eingestellt wurde, kann jeder Identity-Center-Benutzer oder jede Identity-Center-Gruppe auf das Studio zugreifen.	Weitere Informationen finden Sie unter Benutzerberechtigungen für den IAM-Identity-Center-Authentifizierungsmodus.

Zugriffskontrolle

In Amazon EMR Studio konfigurieren Sie die Benutzerautorisierung (Berechtigungen) mit identitätsbasierten AWS Identity and Access Management (IAM)-Richtlinien. In diesen Richtlinien geben Sie zulässige Aktionen und Ressourcen sowie die Bedingungen an, unter denen die Aktionen zulässig sind.

Benutzerberechtigungen für den IAM-Authentifizierungsmodus

Um Benutzerberechtigungen festzulegen, wenn Sie die IAM-Authentifizierung für EMR Studio verwenden, lassen Sie Aktionen zu, z. B. `elasticmapreduce:RunJobFlow` in einer IAM-Berechtigungsrichtlinie. Sie können eine oder mehrere zu verwendende Berechtigungsrichtlinien erstellen. Sie könnten beispielsweise eine grundlegende Richtlinie erstellen, die es einem Benutzer nicht erlaubt, neue Amazon-EMR-Cluster zu erstellen, und eine weitere Richtlinie, die die Clustererstellung zulässt. Eine Liste aller Studio-Aktionen finden Sie unter AWS Identity and Access Management Berechtigungen für EMR Studio-Benutzer.

Benutzerberechtigungen für den IAM-Identity-Center-Authentifizierungsmodus

Wenn Sie die IAM-Identity-Center-Authentifizierung verwenden, erstellen Sie eine einzelne EMR-Studio-Benutzerrolle. Die Benutzerrolle ist eine dedizierte IAM-Rolle, die ein Studio annimmt, wenn sich ein Benutzer anmeldet.

Sie fügen der EMR-Studio-Benutzerrolle IAM-Sitzungsrichtlinien hinzu. Eine Sitzungsrichtlinie ist eine spezielle Art von IAM-Berechtigungsrichtlinie, die einschränkt, was ein Verbundbenutzer während einer Studio-Anmeldesitzung tun kann. Mit Sitzungsrichtlinien können Sie spezifische Berechtigungen

für einen Benutzer oder eine Gruppe festlegen, ohne mehrere Benutzerrollen für EMR Studio erstellen zu müssen.

Wenn Sie einem Studio Benutzer und Gruppen zuweisen, ordnen Sie diesem Benutzer oder dieser Gruppe eine Sitzungsrichtlinie zu, um detaillierte Berechtigungen anzuwenden. Sie können die Sitzungsrichtlinie eines Benutzers oder einer Gruppe auch jederzeit aktualisieren. Amazon EMR speichert jede Sitzungsrichtlinienzuweisung, die Sie erstellen.

Weitere allgemeine Informationen zu Sitzungs-Richtlinien finden Sie unter Berechtigungen und Richtlinien im AWS Identity and Access Management-Benutzerhandbuch.

Workspaces

Workspaces sind die wichtigsten Bausteine von Amazon EMR Studio. Um Notebooks zu organisieren, erstellen Benutzer einen oder mehrere Workspaces in einem Studio. Weitere Informationen finden Sie unter Informationen über Workspace-Grundlagen.

Ähnlich wie Workspaces in JupyterLab behält ein Workspace den Status der Notebook-Arbeit bei. Die Workspace-Benutzeroberfläche erweitert jedoch die Open-Source-Benutzeroberfläche von JupyterLab um zusätzliche Tools, mit denen Sie EMR-Cluster erstellen und anhängen, Aufträge ausführen, Beispiel-Notebooks durchsuchen und Git-Repositorys verknüpfen können.

Die folgende Liste enthält die wichtigsten Features von EMR Studio Workspaces:

- Die Sichtbarkeit von Workspace basiert auf Studio. Workspaces, die Sie in einem Studio erstellen, sind in anderen Studios nicht sichtbar.
- Standardmäßig wird ein Workspace geteilt und kann von allen Studio-Benutzern gesehen werden. Es kann jedoch jeweils nur ein Benutzer einen Workspace öffnen und darin arbeiten. Um gleichzeitig mit anderen Benutzern zu arbeiten, können Sie Konfigurieren Sie die Zusammenarbeit im Workspace
- Sie können gleichzeitig mit anderen Benutzern in einem Workspace zusammenarbeiten, wenn Sie die Workspace-Zusammenarbeit aktivieren. Weitere Informationen finden Sie unter Konfigurieren Sie die Zusammenarbeit im Workspace.
- Notebooks in einem Workspace verwenden denselben EMR-Cluster, um Befehle auszuführen. Sie können einen Workspace an einen Amazon-EMR-Cluster anhängen, der auf Amazon EC2 ausgeführt wird, oder an einen virtuellen Amazon EMR in EKS-Cluster und verwalteten Endpunkt.
- Workspaces können zu einer anderen Availability Zone wechseln, die Sie den Subnetzen eines Studios zuordnen. Sie können einen Workspace beenden und neu starten, um den Failover-

Prozess einzuleiten. Wenn Sie einen Workspace neu starten, startet EMR Studio den Workspace in einer anderen Availability Zone in der VPC des Studios, wenn das Studio mit Zugriff auf mehrere Availability Zones konfiguriert ist. Wenn das Studio nur über eine Availability Zone verfügt, versucht EMR Studio, den Workspace in einem anderen Subnetz zu starten. Weitere Informationen finden Sie unter Beheben von Workspace-Verbindungsproblemen.

- Ein Workspace kann eine Verbindung zu Clustern in allen Subnetzen herstellen, die einem Studio zugeordnet sind.

Weitere Informationen zum Erstellen und Konfigurieren von EMR Studio Workspaces finden Sie unter Informationen über Workspace-Grundlagen.

Notebook-Speicher in Amazon EMR Studio

Wenn Sie einen Workspace verwenden, speichert EMR Studio die Zellen in Notebookdateien automatisch in regelmäßigen Abständen an dem Amazon-S3-Speicherort, der mit Ihrem Studio verknüpft ist. Bei diesem Backup-Prozess bleibt die Arbeit zwischen den Sitzungen erhalten, sodass Sie später darauf zurückgreifen können, ohne Änderungen an ein Git-Repository zu übertragen. Weitere Informationen finden Sie unter Workspace-Inhalt speichern.

Wenn Sie eine Notebook-Datei aus einem Workspace löschen, löscht EMR Studio die Backup-Version für Sie aus Amazon S3. Wenn Sie jedoch einen Workspace löschen, ohne zuerst die zugehörigen Notebookdateien zu löschen, verbleiben die Notebookdateien in Amazon S3 und es fallen weiterhin Speichergebühren an. Weitere Informationen hierzu finden Sie unter Löschen Sie einen Workspace und Notebookdateien.

Überlegungen zu EMR Studio

Überlegungen

Beachten Sie Folgendes, wenn Sie mit EMR Studio arbeiten:

- EMR Studio ist in den folgenden AWS-Regionen Versionen verfügbar:
  - USA Ost (Ohio): (us-east-2)
  - USA Ost (Nord-Virginia): (us-east-1)
  - USA West (Nordkalifornien) (us-west-1)
  - USA West (Oregon): (us-west-2)
  - Afrika (Kapstadt) (af-south-1)

- Asien-Pazifik (Hongkong) (ap-east-1)
- Asien-Pazifik (Jakarta) (ap-southeast-3) *
- Asien-Pazifik (Melbourne) (ap-southeast-4) *
- Asien-Pazifik (Mumbai): (ap-south-1)
- Asien-Pazifik (Osaka) (ap-northeast-3) *
- Asien-Pazifik (Seoul): (ap-northeast-2)
- Asien-Pazifik (Singapur): (ap-southeast-1)
- Asien-Pazifik (Sydney): (ap-southeast-2)
- Asien-Pazifik (Tokyo) (ap-northeast-1)
- Kanada (Zentral): (ca-central-1)
- Europa (Frankfurt) (eu-central-1)
- Europa (Irland) (eu-west-1)
- Europa (London) (eu-west-2)
- Europa (Mailand) (eu-south-1)
- Europa (Paris) (eu-west-3)
- Europa (Spanien) (eu-south-2)
- Europa (Stockholm) (eu-north-1)
- Europa (Zürich) (eu-central-2) *
- Israel (Tel Aviv) il-central-1) *
- Naher Osten (VAE) (me-central-1) *
- Südamerika (São Paulo) (sa-east-1)
- AWS GovCloud (US-Ost) (-1) gov-us-east
- AWS GovCloud (US-West) (gov-us-west-1)

  * Die Live-Spark-Benutzeroberfläche wird in diesen Regionen nicht unterstützt.

- Damit Benutzer neue EMR-Cluster, die auf Amazon EC2 laufen, für einen Workspace bereitstellen können, können Sie ein EMR Studio mit einer Reihe von Cluster-Vorlagen verknüpfen. Administratoren können Clustervorlagen mit Service Catalog definieren und wählen, ob ein Benutzer oder eine Gruppe innerhalb eines Studios auf die Clustervorlagen zugreifen kann oder keine Clustervorlagen.
- Verwenden Sie die Amazon EMR-Servicerolle, wenn Sie Zugriffsberechtigungen für Notizbuchdateien definieren AWS Secrets Manager, die in Amazon S3 gespeichert sind oder aus

denen Geheimnisse gelesen werden. Sitzungsrichtlinien werden mit diesen Berechtigungen nicht unterstützt.

- Sie können mehrere EMR-Studios erstellen, um den Zugriff auf EMR-Cluster in verschiedenen VPCs zu steuern.
- Verwenden Sie die AWS CLI , um Amazon EMR auf EKS-Clustern einzurichten. Anschließend können Sie die Studio-Oberfläche verwenden, um Cluster an Workspaces mit einem verwalteten Endpunkt anzuhängen, um Notebook-Jobs auszuführen.
- Wenn Sie Trusted Identity Propagation mit Amazon EMR verwenden, gibt es weitere Überlegungen, die auch für EMR Studio gelten. Weitere Informationen finden Sie unter Überlegungen und Einschränkungen für Amazon EMR bei der Identity Center-Integration.
- EMR Studio unterstützt die folgenden magischen Python-Befehle nicht:
  - `%alias`
  - `%alias_magic`
  - `%automagic`
  - `%macro`
  - `%%js`
  - `%%javascript`
  - Ändern von `proxy_user` mit `%configure`
  - Ändern von `KERNEL_USERNAME` mit `%env` oder `%set_env`
- Amazon EMR auf EKS-Clustern unterstützt keine SparkMagic Befehle für EMR Studio.
- Um mehrzeilige Scala-Anweisungen in Notebookzellen zu schreiben, stellen Sie sicher, dass alle Zeilen bis auf die letzte mit einem Punkt enden. Im folgenden Beispiel wird die richtige Syntax für mehrzeilige Scala-Anweisungen verwendet.

```
val df = spark.sql("SELECT * from table_name).
        filter("col1=='value'").
        limit(50)
```

- Um die Sicherheit der Anwendungen außerhalb der Konsole zu erhöhen, die Sie möglicherweise mit Amazon EMR verwenden, sind die Anwendungs-Hosting-Domains in der Public Suffix List (PSL) registriert. Zu diesen Hosting-Domains gehören beispielsweise die folgenden: `emrstudio-prod.us-east-1.amazonaws.com`, `emrnotebooks-prod.us-east-1.amazonaws.com`, `emrappui-prod.us-east-1.amazonaws.com`. Aus Sicherheitsgründen empfehlen wir Ihnen, Cookies mit einem `__Host--`Präfix zu verwenden, falls Sie jemals sensible Cookies im Standard-

Domainnamen einrichten müssen. Diese Vorgehensweise hilft Ihnen dabei, Ihre Domain vor CSRF (Cross-Site Request Forgery Attempts, Anforderungsfälschung zwischen Websites)-Versuchen zu schützen. Weitere Informationen finden Sie auf der Set-Cookie-Seite im Mozilla Developer Network.

Bekannte Probleme

- Ein EMR Studio, das IAM Identity Center mit aktivierter Weitergabe vertrauenswürdiger Identitäten verwendet, kann nur EMR-Clustern zugeordnet werden, die auch vertrauenswürdige Identitätsverteilung verwenden.
- Stellen Sie sicher, dass Sie Proxy-Management-Tools wie FoxyProxy oder SwitchyOmega im Browser deaktivieren, bevor Sie ein Studio erstellen. Aktive Proxys können Fehler verursachen, wenn Sie Studio erstellen wählen, und zu einer Netzwerkfehler-Fehlermeldung führen.
- Kernel, die auf Amazon EMR in EKS-Clustern ausgeführt werden, können aufgrund von Timeout-Problemen nicht gestartet werden. Wenn beim Starten des Kernels ein Fehler oder ein Problem auftritt, schließen Sie die Notebook-Datei, fahren Sie den Kernel herunter und öffnen Sie die Notebook-Datei erneut.
- Der Kernel-Neustartvorgang funktioniert nicht wie erwartet, wenn Sie einen Cluster von Amazon EMR in EKS verwenden. Nachdem Sie Kernel neu starten ausgewählt haben, aktualisieren Sie den Workspace, damit der Neustart wirksam wird.
- Wenn ein Workspace nicht an einen Cluster angehängt ist, wird eine Fehlermeldung angezeigt, wenn ein Studio-Benutzer eine Notebook-Datei öffnet und versucht, einen Kernel auszuwählen. Sie können diese Fehlermeldung ignorieren, indem Sie OK wählen, aber Sie müssen den Workspace an einen Cluster anhängen und einen Kernel auswählen, bevor Sie Notebook-Code ausführen können.
- Wenn Sie Amazon EMR 6.2.0 mit einer Sicherheitskonfiguration verwenden, um die Clustersicherheit einzurichten, erscheint die Workspace-Oberfläche leer und funktioniert nicht wie erwartet. Wir empfehlen Ihnen, eine andere unterstützte Version von Amazon EMR zu verwenden, wenn Sie Datenverschlüsselung oder Amazon-S3-Autorisierung für EMRFS für einen Cluster konfigurieren möchten. EMR Studio funktioniert mit den Amazon-EMR-Versionen 5.32.0 (Amazon-EMR-5.x-Serie) und 6.2.0 (Amazon-EMR-6.x-Serie) und höher.
- Wenn Sie Debuggen Sie Amazon, EMR das auf EC2 Amazon-Jobs ausgeführt wird, funktionieren die Links zur Spark-Benutzeroberfläche auf dem Cluster möglicherweise nicht oder werden nicht angezeigt. Um die Links zu regenerieren, erstellen Sie eine neue Notebook-Zelle und führen Sie den `%%info`-Befehl aus.

- Jupyter Enterprise Gateway bereinigt in den folgenden Amazon-EMR-Release-Versionen keine inaktiven Kernel auf dem Primärknoten eines Clusters: 5.32.0, 5.33.0, 6.2.0 und 6.3.0. Kernel im Leerlauf verbrauchen Rechenressourcen und können dazu führen, dass Cluster mit langer Laufzeit ausfallen. Mit dem folgenden Beispielskript können Sie die Kernelbereinigung im Leerlauf für Jupyter Enterprise Gateway konfigurieren. Sie können Connect zum Primärknoten her mit SSH oder das Skript als Schritt einreichen. Weitere Informationen finden Sie unter Befehle und Skripts auf einem Amazon-EMR-Cluster ausführen.

  ```
  #!/bin/bash
  sudo tee -a /emr/notebook-env/conf/jupyter_enterprise_gateway_config.py << EOF
  c.MappingKernelManager.cull_connected = True
  c.MappingKernelManager.cull_idle_timeout = 10800
  c.MappingKernelManager.cull_interval = 300
  EOF
  sudo systemctl daemon-reload
  sudo systemctl restart jupyter_enterprise_gateway
  ```

- Wenn Sie eine automatische Terminierungsrichtlinie mit den Amazon-EMR-Versionen 5.32.0, 5.33.0, 6.2.0 oder 6.3.0 verwenden, markiert Amazon EMR einen Cluster als inaktiv und beendet den Cluster möglicherweise automatisch, auch wenn Sie einen aktiven Python3-Kernel haben. Das liegt daran, dass bei der Ausführung eines Python3-Kernels kein Spark-Job auf dem Cluster gesendet wird. Um die automatische Terminierung mit einem Python3-Kernel zu verwenden, empfehlen wir die Verwendung von Amazon-EMR-Version 6.4.0 oder höher. Weitere Informationen zum Auto-Beenden finden Sie unter Verwenden einer Richtlinie zur automatischen Beendigung.
- Wenn Sie `%%display` einen Spark DataFrame in einer Tabelle anzeigen, können sehr breite Tabellen gekürzt werden. Sie können mit der rechten Maustaste auf die Ausgabe klicken und Neue Ansicht für Ausgabe erstellen auswählen, um eine scrollbare Ansicht der Ausgabe zu erhalten.
- Wenn Sie einen Spark-basierten Kernel wie PySpark Spark oder SparkR starten, wird eine Spark-Sitzung gestartet, und wenn Sie eine Zelle in einem Notizbuch ausführen, werden Spark-Jobs in dieser Sitzung in die Warteschlange gestellt. Wenn Sie eine laufende Zelle unterbrechen, wird der Spark-Auftrag weiter ausgeführt. Um den Spark-Auftrag zu beenden, sollten Sie die Cluster-interne Spark-Benutzeroberfläche verwenden. Weitere Informationen zur Verbindung mit einer Spark-Benutzeroberfläche finden Sie unter Debuggen Sie Anwendungen und Jobs mit Studio EMR.
- Die Verwendung von Amazon EMR Studio Workspaces als Root-Benutzer in einem AWS-Konto verursacht einen `403: Forbidden` Fehler. Dies liegt daran, dass die Jupyter Enterprise Gateway-Konfiguration in Amazon EMR dem Root-Benutzer keinen Zugriff gewährt. Wir empfehlen, den Root-Benutzer nicht für Ihre täglichen Aufgaben zu verwenden. Weitere Authentifizierungsoptionen finden Sie unter AWS Identity and Access Management Amazon EMR.

Feature-Einschränkungen

Amazon EMR Studio unterstützt die folgenden Amazon-EMR-Feature nicht:

- Anhängen und Ausführen von Aufträgen auf EMR-Clustern mit einer Sicherheitskonfiguration, die die Kerberos-Authentifizierung spezifiziert
- Cluster mit mehreren Primärknoten
- Cluster, die Amazon EC2 EC2-Instances verwenden, die auf AWS Graviton2 für Amazon EMR 6.x-Versionen unter 6.9.0 und 5.x-Versionen unter 5.36.1 basieren

Die folgenden Features werden von einem Studio, das die Verbreitung vertrauenswürdiger Identitäten verwendet, nicht unterstützt:

- Erstellen von EMR-Clustern ohne Vorlage.
- Verwenden von EMR-Serverless-Anwendungen.
- Starten von Amazon EMR in EKS-Clustern.
- Verwenden einer Laufzeitrolle.
- Aktivieren der Zusammenarbeit mit SQL Explorer oder Workspace.

Service-Limits für EMR Studio

In der folgenden Tabelle werden die Service-Limits für EMR Studio aufgeführt.

Item	Limit
EMR Studios	AWS Maximal 100 pro Konto
Subnetze	Maximal fünf für jedes EMR-Studio
IAM-Identity-Center-Gruppen	Maximal fünf für jedes EMR-Studio
Benutzer von IAM Identity Center	Maximal 100 für jedes EMR-Studio

Bewährte Methoden für VPC und Subnetze

Verwenden Sie die folgenden bewährten Methoden, um eine Amazon Virtual Private Cloud (Amazon VPC) mit Subnetzen für EMR Studio einzurichten:

- Sie können in Ihrer VPC maximal fünf Subnetze angeben, die Sie dem Studio zuordnen möchten. Wir empfehlen, dass Sie mehrere Subnetze in verschiedenen Availability Zones bereitstellen, um die Workspace-Verfügbarkeit zu unterstützen und Studio-Benutzern Zugriff auf Cluster in verschiedenen Availability Zones zu gewähren. Weitere Informationen zur Arbeit mit VPCs, Subnetzen und Availability Zones finden Sie unter VPCs und Subnetze im Benutzerhandbuch für Amazon Virtual Private Cloud .
- Die von Ihnen angegebenen Subnetze sollten miteinander kommunizieren können.
- Damit Benutzer einen Workspace mit öffentlich gehosteten Git-Repositorys verknüpfen können, sollten Sie nur private Subnetze angeben, die über Network Address Translation (NAT) Zugriff auf das Internet haben. Weitere Informationen zum Einrichten eines privaten Subnetzes für Amazon EMR finden Sie unter Private Subnetze.
- Wenn Sie Amazon EMR auf EKS mit EMR Studio verwenden, muss mindestens ein gemeinsames Subnetz zwischen Ihrem Studio und dem Amazon-EKS-Cluster vorhanden sein, den Sie zum Registrieren eines virtuellen Clusters verwenden. Andernfalls wird Ihr verwalteter Endpunkt nicht als Option in Studio Workspaces angezeigt. Sie können einen Amazon-EKS-Cluster erstellen und ihn einem Subnetz zuordnen, das zum Studio gehört, oder Sie können ein Studio erstellen und die Subnetze Ihres EKS-Clusters angeben.
- Wenn Sie Amazon EMR in EKS mit EMR Studio verwenden möchten, wählen Sie die VPC für Ihre Amazon-EKS-Cluster-Worker-Knoten aus.

Cluster-Anforderungen für Amazon EMR Studio

Amazon-EMR-Cluster, die auf Amazon EC2 ausgeführt werden

Alle Amazon-EMR-Cluster, die auf Amazon EC2 ausgeführt werden und die Sie für einen EMR Studio Workspace erstellen, müssen die folgenden Anforderungen erfüllen. Cluster, die Sie mit der EMR-Studio-Oberfläche erstellen, erfüllen diese Anforderungen automatisch.

- Der Cluster muss Amazon-EMR-Versionen 5.32.0 (Amazon EMR 5.x-Serie) oder 6.2.0 (Amazon EMR 6.x-Serie) oder höher verwenden. Sie können mit der Amazon EMR-Konsole oder dem SDK einen Cluster erstellen und ihn dann an einen EMR Studio Workspace anhängen. AWS Command

Line Interface Studio-Benutzer können auch Cluster bereitstellen und anhängen, wenn sie einen Amazon-EMR-Workspace erstellen oder darin arbeiten. Weitere Informationen finden Sie unter Hängen Sie einen Computer an einen EMR Studio-Workspace an.

- Dieser Cluster muss sich innerhalb einer Amazon Virtual Private Cloud befinden. Die EC2-Classic-Plattform wird nicht unterstützt.
- Auf dem Cluster müssen Spark, Livy und Jupyter Enterprise Gateway installiert sein. Wenn Sie den Cluster für SQL Explorer verwenden möchten, sollten Sie sowohl Presto als auch Spark installieren.
- Um SQL Explorer verwenden zu können, muss der Cluster Amazon-EMR-Version 5.34.0 oder höher oder Version 6.4.0 oder höher verwenden und Presto installiert sein. Wenn Sie den AWS Glue-Datenkatalog als Hive-Metastore für Presto angeben möchten, müssen Sie ihn auf dem Cluster konfigurieren. Weitere Informationen finden Sie unter Verwendung von Presto mit dem AWS Glue Data Catalog.
- Der Cluster muss sich in einem privaten Subnetz mit Network Address Translation (NAT) befinden, um öffentlich gehostete Git-Repositorys mit EMR Studio verwenden zu können.

Wir empfehlen die folgenden Clusterkonfigurationen, wenn Sie mit EMR Studio arbeiten.

- Stellen Sie den Bereitstellungsmodus für Spark-Sitzungen auf den Clustermodus ein. Im Clustermodus werden die Anwendungsmasterprozesse auf den Core-Knoten und nicht auf dem Primärknoten eines Clusters platziert. Dadurch wird der Primärknoten von potenziellem Speicherdruck entlastet. Weitere Informationen finden Sie unter Cluster Mode Overview in der Apache Spark-Dokumentation.
- Ändern Sie das Livy-Timeout wie in der folgenden Beispielkonfiguration von der Standardeinstellung von einer Stunde auf sechs Stunden.

```
{
    "classification":"livy-conf",
        "Properties":{
            "livy.server.session.timeout":"6h",
            "livy.spark.deploy-mode":"cluster"
        }
}
```

- Erstellen Sie verschiedene Instance-Flotten mit bis zu 30 Instances und wählen Sie mehrere Instance-Typen in Ihrer Spot Instance-Flotte aus. Sie könnten beispielsweise die folgenden arbeitsspeicheroptimierten Instance-Typen für Spark-Workloads angeben: r5.2x, r5.4x, r5.8x,

r5.12x, r5.16x, r4.2x, r4.4x, r4.8x, r4.12, usw. Weitere Informationen finden Sie unter Instance-Flotten konfigurieren.

- Verwenden Sie die kapazitätsoptimierte Zuweisungsstrategie für Spot Instances, um Amazon EMR dabei zu unterstützen, eine effektive Instance-Auswahl auf der Grundlage von Echtzeit-Kapazitätsinformationen von Amazon EC2 zu treffen. Weitere Informationen finden Sie unter Zuweisungsstrategie für Flotten.
- Aktivieren Sie die verwaltete Skalierung in Ihrem Cluster. Legen Sie den Parameter für die maximale Anzahl an Core-Knoten auf die minimale persistente Kapazität fest, die Sie verwenden möchten, und konfigurieren Sie die Skalierung für eine gut diversifizierte Task-Flotte, die auf Spot Instances ausgeführt wird, um Kosten zu sparen. Weitere Informationen finden Sie unter Verwenden von verwalteter Skalierung in Amazon EMR.

Wir bitten Sie außerdem dringend, Amazon EMR Block Public Access aktiviert zu lassen und den eingehenden SSH-Verkehr auf vertrauenswürdige Quellen zu beschränken. Durch den eingehenden Zugriff auf einen Cluster können Benutzer Notebooks auf dem Cluster ausführen. Weitere Informationen finden Sie unter Verwenden Sie Amazon, um EMR den öffentlichen Zugriff zu blockieren und Steuerung des Netzwerkverkehrs mit Sicherheitsgruppen.

Cluster von Amazon EMR in EKS

Zusätzlich zu EMR-Clustern, die auf Amazon EC2 ausgeführt werden, können Sie Amazon EMR auf EKS-Clustern für EMR Studio mithilfe von AWS CLI einrichten und verwalten. Richten Sie Amazon EMR auf EKS-Clustern gemäß den folgenden Richtlinien ein:

- Erstellen Sie einen verwalteten HTTPS-Endpunkt für den Cluster von Amazon EMR in EKS. Benutzer hängen einen Workspace an einen verwalteten Endpunkt an. Der Amazon Elastic Kubernetes Service (EKS)-Cluster, den Sie zur Registrierung eines virtuellen Clusters verwenden, muss über ein privates Subnetz verfügen, um verwaltete Endgeräte zu unterstützen.
- Verwenden Sie einen Amazon-EKS-Cluster mit mindestens einem privaten Subnetz und Network Address Translation (NAT), wenn Sie öffentlich gehostete Git-Repositorys verwenden möchten.
- Vermeiden Sie die Verwendung von Amazon-EKS-optimierten Arm-Amazon-Linux-AMIs, die für Endpunkte, die von Amazon EMR auf EKS verwaltet werden, nicht unterstützt werden.
- Vermeiden Sie die AWS Fargate ausschließliche Verwendung von Amazon EKS-Clustern, die nicht unterstützt werden.

Amazon EMR Studio konfigurieren

Dieser Abschnitt richtet sich an EMR Studio-Administratoren. Er behandelt, wie Sie ein EMR Studio für Ihr Team einrichten, und enthält Anweisungen für Aufgaben wie das Zuweisen von Benutzern und Gruppen, das Einrichten von Cluster-Vorlagen und das Optimieren von Apache Spark für EMR Studio.

Themen

Administratorrechte zum Erstellen und Verwalten eines Studios EMR

Die auf dieser Seite beschriebenen IAM Berechtigungen ermöglichen es Ihnen, ein EMR Studio zu erstellen und zu verwalten. Weitere Informationen zu erforderlichen Berechtigungen finden Sie unter Für die Verwaltung eines EMR Studios sind Berechtigungen erforderlich.

Für die Verwaltung eines EMR Studios sind Berechtigungen erforderlich

In der folgenden Tabelle sind die Vorgänge im Zusammenhang mit der Erstellung und Verwaltung eines EMR Studios aufgeführt. In der Tabelle werden auch die für jeden Vorgang erforderlichen Berechtigungen angezeigt.

> **Note**
>
> Sie benötigen IAM Identity Center- und `SessionMapping` Studio-Aktionen nur, wenn Sie den IAM Identity Center-Authentifizierungsmodus verwenden.

Berechtigungen zum Erstellen und Verwalten eines EMR Studios

Operation	Berechtigungen
Ein Studio erstellen	`"elasticmapreduce:CreateStudio", "sso:CreateApplication", "sso:PutApplicationAuthentic ationMethod", "sso:PutApplicationGrant", "sso:PutApplicationAccessScope", "sso:PutApplicationAssignmentConfi guration", "iam:PassRole"`
Ein Studio beschreiben	`"elasticmapreduce:DescribeStudio", "sso:GetManagedApplicationInstance"`
Studios auflisten	`"elasticmapreduce:ListStudios"`
Ein Studio löschen	`"elasticmapreduce:DeleteStudio", "sso:DeleteApplication", "sso:DeleteApplicationAuthentica tionMethod", "sso:DeleteApplicationAccessScope", "sso:DeleteApplicationGrant"`
Additional permissions required when you use IAM Identity Center mode	
Einem Studio Benutzer oder Gruppen zuweisen	`"elasticmapreduce:CreateStudioSessio nMapping", "sso:GetProfile", "sso:ListDirectoryAssociations", "sso:ListProfiles", "sso:AssociateProfile", "sso-directory:SearchUsers", "sso-directory:SearchGroups", "sso-directory:DescribeUser", "sso-directory:DescribeGroup", "sso:ListInstances", "sso:CreateApplicationAssignment",`

Operation	Berechtigungen
	"sso:DescribeInstance", "organizations:DescribeOrga nization", "organizations:ListDelegatedAdmini strators", "sso:CreateInstance", "sso:DescribeRegisteredRegions", "sso:GetSharedSsoConfiguration", "iam:ListPolicies"
Rufen Sie die Studio-Zuweisungsdetails für einen bestimmten Benutzer oder eine bestimmte Gruppe ab	"sso-directory:SearchUsers", "sso-directory:SearchGroups", "sso-directory:DescribeUser", "sso-directory:DescribeGroup", "sso:DescribeApplication", "elasticmapreduce:GetStudioSessio nMapping"
Alle Benutzer und Gruppen auflisten, die einem Studio zugewiesen sind	"elasticmapreduce:ListStudioSessionM appings"
Aktualisieren Sie die Sitzungsrichtlinie, die einem Benutzer oder einer Gruppe zugewiesen ist, die einem Studio zugewiesen ist	"sso-directory:SearchUsers", "sso-directory:SearchGroups", "sso-directory:DescribeUser", "sso-directory:DescribeGroup", "sso:DescribeApplication", "sso:DescribeInstance", "elasticmapreduce:UpdateStu dioSessionMapping"

Operation	Berechtigungen
Einen Benutzer oder eine Gruppe aus einem Studio entfernen	`"elasticmapreduce:DeleteStudioSessionMapping", "sso-directory:SearchUsers", "sso-directory:SearchGroups", "sso-directory:DescribeUser", "sso-directory:DescribeGroup", "sso:ListDirectoryAssociations", "sso:GetProfile", "sso:DescribeApplication", "sso:DescribeInstance", "sso:ListProfiles", "sso:DisassociateProfile", "sso:DeleteApplicationAssignment", "sso:ListApplicationAssignments"`

Um eine Richtlinie mit Administratorberechtigungen für EMR Studio zu erstellen

1. Folgen Sie den Anweisungen unter IAMRichtlinien erstellen, um eine Richtlinie anhand eines der folgenden Beispiele zu erstellen. Welche Berechtigungen Sie benötigen, hängt von Ihrem Authentifizierungsmodus für EMR Studio ab.

   Fügen Sie Ihre eigenen Werte für diese Elemente ein:

   - Ersetzen *<deine-ressource- >ARN* um den Amazon-Ressourcennamen (ARN) des Objekts oder der Objekte anzugeben, die die Anweisung für Ihre Anwendungsfälle abdeckt.
   - Ersetzen *<region>* mit dem Code des Ortes AWS-Region , an dem Sie das Studio erstellen möchten.
   - Ersetzen *<aws-account_id>* mit der ID des AWS Kontos für das Studio.
   - Ersetzen *<EMRStudio-Service-Role>* and *<EMRStudio-User-Role>* mit den Namen Ihrer EMRStudio-Dienstrolle und EMRStudio-Benutzerrolle.

   Example Beispielrichtlinie: Administratorberechtigungen, wenn Sie den IAM Authentifizierungsmodus verwenden

   ```
   {
       "Version": "2012-10-17",
   ```

```
    "Statement": [
        {
            "Effect": "Allow",
            "Resource": "arn:aws:elasticmapreduce:<region>:<aws-account-id>:studio/
*",
            "Action": [
                "elasticmapreduce:CreateStudio",
                "elasticmapreduce:DescribeStudio",
                "elasticmapreduce:DeleteStudio"
            ]
        },
        {
            "Effect": "Allow",
            "Resource": "<your-resource-ARN>",
            "Action": [
                "elasticmapreduce:ListStudios"
            ]
        },
        {
            "Effect": "Allow",
            "Resource": [
                "arn:aws:iam::<aws-account-id>:role/<EMRStudio-Service-Role>"
            ],
            "Action": "iam:PassRole"
        }
    ]
}
```

Example Beispielrichtlinie: Administratorberechtigungen, wenn Sie den IAM Identity Center-Authentifizierungsmodus verwenden

> **Note**
>
> Identity Center und das Identity Center-Verzeichnis unterstützen die Angabe eines ARN im Ressourcenelement einer IAM Richtlinienerklärung APIs nicht. Um den Zugriff auf IAM Identity Center und das IAM Identity Center-Verzeichnis zu ermöglichen, spezifizieren die folgenden Berechtigungen alle Ressourcen, „Ressource“ :"*“, für IAM Identity Center-Aktionen. Weitere Informationen finden Sie unter Aktionen, Ressourcen und Bedingungsschlüssel für IAM Identity Center Directory.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Resource": "arn:aws:elasticmapreduce:<region>:<aws-account-id>:studio/
*",
            "Action": [
                "elasticmapreduce:CreateStudio",
                "elasticmapreduce:DescribeStudio",
                "elasticmapreduce:DeleteStudio",
                "elasticmapreduce:CreateStudioSessionMapping",
                "elasticmapreduce:GetStudioSessionMapping",
                "elasticmapreduce:UpdateStudioSessionMapping",
                "elasticmapreduce:DeleteStudioSessionMapping"
            ]
        },
        {
            "Effect": "Allow",
            "Resource": "<your-resource-ARN>",
            "Action": [
                "elasticmapreduce:ListStudios",
                "elasticmapreduce:ListStudioSessionMappings"
            ]
        },
        {
            "Effect": "Allow",
            "Resource": [
                "arn:aws:iam::<aws-account-id>:role/<EMRStudio-Service-Role>",
                "arn:aws:iam::<aws-account-id>:role/<EMRStudio-User-Role>"
            ],
            "Action": "iam:PassRole"
        },
        {
            "Effect": "Allow",
            "Resource": "*",
            "Action": [
                "sso:CreateApplication",
                "sso:PutApplicationAuthenticationMethod",
                "sso:PutApplicationGrant",
                "sso:PutApplicationAccessScope",
                "sso:PutApplicationAssignmentConfiguration",
```

```
                "sso:DescribeApplication",
                "sso:DeleteApplication",
                "sso:DeleteApplicationAuthenticationMethod",
                "sso:DeleteApplicationAccessScope",
                "sso:DeleteApplicationGrant",
                "sso:ListInstances",
                "sso:CreateApplicationAssignment",
                "sso:DeleteApplicationAssignment",
                "sso:ListApplicationAssignments",
                "sso:DescribeInstance",
                "sso:AssociateProfile",
                "sso:DisassociateProfile",
                "sso:GetProfile",
                "sso:ListDirectoryAssociations",
                "sso:ListProfiles",
                "sso-directory:SearchUsers",
                "sso-directory:SearchGroups",
                "sso-directory:DescribeUser",
                "sso-directory:DescribeGroup",
                "organizations:DescribeOrganization",
                "organizations:ListDelegatedAdministrators",
                "sso:CreateInstance",
                "sso:DescribeRegisteredRegions",
                "sso:GetSharedSsoConfiguration",
                "iam:ListPolicies"
            ]
        }
    ]
}
```

2. Ordnen Sie die Richtlinie Ihrer IAM Identität (Benutzer, Rolle oder Gruppe) zu. Anweisungen finden Sie unter Hinzufügen und Entfernen von IAM Identitätsberechtigungen.

Richten Sie ein Amazon EMR Studio ein

Gehen Sie wie folgt vor, um ein Amazon EMR Studio einzurichten.

Bevor Sie beginnen

> **Note**
>
> Wenn Sie EMR Studio mit Amazon EMR on verwenden möchtenEKS, empfehlen wir Ihnen, zuerst Amazon EMR on EKS für EMR Studio einzurichten, bevor Sie ein Studio einrichten.

Bevor Sie ein EMR Studio einrichten, stellen Sie sicher, dass Sie über die folgenden Elemente verfügen:

- Ein AWS-Konto. Detaillierte Anweisungen finden Sie unter Einrichten von Amazon EMR.
- Berechtigungen zum Erstellen und Verwalten eines EMR Studios. Weitere Informationen finden Sie unter the section called “Administratorberechtigungen zum Erstellen eines EMR Studios”.
- Ein Amazon S3 S3-Bucket, in dem EMR Studio die Workspaces und Notizbuchdateien in Ihrem Studio sichern kann. Anweisungen finden Sie unter Erstellen eines Buckets im Amazon Simple Storage Service (S3)-Benutzerhandbuch.
- Wenn Sie eine Verbindung zu einem Amazon EMR on EC2 - oder Amazon EMR EKS on-Cluster herstellen oder Git-Repositorys verwenden möchten, benötigen Sie eine Amazon Virtual Private Cloud (VPC) für das Studio und maximal fünf Subnetze. Sie benötigen kein, um EMR Studio mit VPC EMR Serverless zu verwenden. Tipps zur Netzwerkkonfiguration finden Sie unter Bewährte Methoden für VPC und Subnetze.

Um ein EMR Studio einzurichten

1. Wählen Sie einen Authentifizierungsmodus für Amazon EMR Studio
2. Erstellen Sie die folgenden Studio-Ressourcen.
   - Erstellen Sie eine EMR Studio-Dienstrolle
   - EMRStudio-Benutzerberechtigungen für Amazon EC2 oder Amazon konfigurieren EKS
   - (Optional) Definieren Sie Sicherheitsgruppen zur Steuerung des EMR Studio-Netzwerkverkehrs.
3. Erstellen Sie ein Studio EMR
4. Weisen Sie einem EMR Studio einen Benutzer oder eine Gruppe zu

Nachdem Sie diese Einrichtungs-Schritte abgeschlossen haben, fahren Sie mit Verwenden Sie ein Amazon EMR Studio fort.

Wählen Sie einen Authentifizierungsmodus für Amazon EMR Studio

EMRStudio unterstützt zwei Authentifizierungsmodi: den IAM Authentifizierungsmodus und den IAM Identity Center-Authentifizierungsmodus. IAMDer Modus verwendet AWS Identity and Access Management (IAM), während der IAM Identity Center-Modus verwendet AWS IAM Identity Center. Wenn Sie ein EMR Studio erstellen, wählen Sie den Authentifizierungsmodus für alle Benutzer dieses Studios. Weitere Informationen zu diesen verschiedenen Authentifizierungsmodi finden Sie unter Authentifizierung und Benutzeranmeldung.

Verwenden Sie die folgende Tabelle, um einen Authentifizierungsmodus für EMR Studio auszuwählen.

Wenn Sie …	Wir empfehlen...
Sie kennen sich bereits mit IAM Authentif izierung oder Verbund aus oder haben diese eingerichtet	IAMAuthentifizierungsmodus, welches die folgenden Vorteile bietet: • Bietet eine schnelle Einrichtung für EMR Studio, wenn Sie bereits Identitäten wie Benutzer und Gruppen in IAM verwalten. • Funktioniert mit Identitätsanbietern, die mit OpenID Connect (OIDC) oder Security Assertion Markup Language 2.0 (SAML2.0) kompatibel sind. • Unterstützt die Verwendung mehrerer Identitätsanbieter mit demselben AWS-Konto . • Verfügbar in einer Vielzahl von. AWS-Regio nen • Konform mit SOC 2.
Neu bei AWS Amazon EMR	IAMIdentity Center-Authentifizierungsmodus, welches die folgenden Features bietet: • Unterstützt die einfache Zuweisung von Benutzern und Gruppen zu AWS Ressource n.

Wenn Sie …	Wir empfehlen...
	• Funktioniert mit Microsoft Active Directory und SAML 2.0-Identitätsanbietern. • Erleichtert die Einrichtung eines Verbunds für mehrere Konten, sodass Sie den Verbund nicht für jedes Konto AWS-Konto in Ihrer Organisation separat konfigurieren müssen.

IAMAuthentifizierungsmodus für Amazon EMR Studio einrichten

Im IAM Authentifizierungsmodus können Sie entweder die IAM Authentifizierung oder den IAM Verbund verwenden. IAMMit der Authentifizierung können Sie IAM Identitäten wie Benutzer, Gruppen und Rollen in IAM verwalten. Sie gewähren Benutzern Zugriff auf ein Studio mit IAM Berechtigungsrichtlinien und attributbasierter Zugriffskontrolle (). ABAC IAMMit einem Verbund können Sie Vertrauen zwischen einem externen Identitätsanbieter (IdP) aufbauen AWS und Benutzeridentitäten über Ihren IdP verwalten.

> **Note**
>
> Wenn Sie IAM den Zugriff auf AWS Ressourcen bereits steuern oder Ihren Identitätsanbieter (IdP) bereits für konfiguriert haben, finden Sie weitere Informationen unter Benutzerberechtigungen für den IAM-Authentifizierungsmodus So legen Sie Benutzerberechtigungen festIAM, wenn Sie den IAM Authentifizierungsmodus für EMR Studio verwenden.

Verwenden Sie den IAM Verbund für Amazon EMR Studio

Um den IAM Verbund für EMR Studio zu verwenden, erstellen Sie eine Vertrauensbeziehung zwischen Ihnen AWS-Konto und Ihrem Identitätsanbieter (IdP) und ermöglichen Verbundbenutzern den Zugriff auf. AWS Management Console Die Schritte, die Sie zum Aufbau dieser Vertrauensbeziehung ergreifen, hängen vom Verbundstandard Ihres IdP ab.

Im Allgemeinen führen Sie die folgenden Aufgaben aus, um den Verbund mit einem externen IdP zu konfigurieren. Vollständige Anweisungen finden Sie unter Aktivieren des Zugriffs von SAML 2.0-Verbundbenutzern auf AWS Management Console und Aktivieren des benutzerdefinierten Identity

Broker-Zugriffs auf das AWS Management ConsoleAWS Identity and Access Management im Benutzerhandbuch.

1. Sammeln Sie Informationen von Ihrem IdP. Dies bedeutet in der Regel die Generierung eines Metadatendokuments, um SAML Authentifizierungsanfragen von Ihrem IdP zu validieren.
2. Erstellen Sie eine IAM Identitätsanbieter-Entität, um Informationen über Ihren IdP zu speichern. Anweisungen finden Sie unter IAMIdentitätsanbieter erstellen.
3. Erstellen Sie eine oder mehrere IAM Rollen für Ihren IdP. EMRStudio weist einem Verbundbenutzer eine Rolle zu, wenn sich der Benutzer anmeldet. Die Rolle ermöglicht es dem Identitätsanbieter, temporäre Sicherheitsanmeldeinformationen für den Zugriff auf AWS anzufordern. Anweisungen finden Sie unter Erstellen einer Rolle für einen Drittanbieter-Identitätsanbieter (Verbund). Die Berechtigungsrichtlinien, die Sie der Rolle zuweisen, bestimmen, was Verbundbenutzer in AWS und in einem Studio tun können. EMR Weitere Informationen finden Sie unter Benutzerberechtigungen für den IAM-Authentifizierungsmodus.
4. (Für SAML Anbieter) Vervollständigen Sie die SAML Vertrauensstellung, indem Sie Ihren IdP mit Informationen zu AWS und den Rollen konfigurieren, die Verbundbenutzer annehmen sollen. Dieser Konfigurationsprozess schafft Vertrauen zwischen Ihrem IdP und AWS. Weitere Informationen finden Sie unter Konfiguration Ihres SAML 2.0-IdP mit dem Vertrauen der vertrauenden Partei und Hinzufügen von Ansprüchen.

Um ein EMR Studio als SAML Anwendung in Ihrem IdP-Portal zu konfigurieren

Sie können ein bestimmtes EMR Studio über einen Deep-Link zum Studio als SAML Anwendung konfigurieren. Auf diese Weise können sich Benutzer bei Ihrem IdP-Portal anmelden und ein bestimmtes Studio starten, anstatt durch die EMR Amazon-Konsole zu navigieren.

- Verwenden Sie das folgende Format, um einen Deep-Link zu Ihrem EMR Studio als Landing URL nach der SAML Bestätigung der Bestätigung zu konfigurieren.

```
https://console.aws.amazon.com/emr/home?region=<aws-region>#studio/<your-studio-id>/start
```

IAMIdentity Center-Authentifizierungsmodus für Amazon EMR Studio einrichten

Um sich auf EMR Studio AWS IAM Identity Center vorzubereiten, müssen Sie Ihre Identitätsquelle konfigurieren und Benutzer und Gruppen bereitstellen. Bei der Bereitstellung werden Benutzer- und

Gruppeninformationen für Identity Center und Anwendungen, die IAM Identity Center verwendenIAM, zur Verfügung gestellt. Weitere Informationen finden Sie unter Benutzer- und Gruppenbereitstellung.

EMRStudio unterstützt die Verwendung der folgenden Identitätsanbieter für IAM Identity Center:

- AWS Managed Microsoft AD und selbstverwaltetes Active Directory — Weitere Informationen finden Sie unter Connect Ihrem Microsoft AD-Verzeichnis herstellen.
- SAMLbasierte Anbieter — Eine vollständige Liste finden Sie unter Unterstützte Identitätsanbieter.
- Das IAM Identity Center-Verzeichnis — Weitere Informationen finden Sie unter Identitäten in IAM Identity Center verwalten.

So richten Sie IAM Identity Center für Studio ein EMR

1. Um IAM Identity Center for EMR Studio einzurichten, benötigen Sie Folgendes:

   - Ein Verwaltungskonto in Ihrer AWS Organisation, wenn Sie mehrere Konten in Ihrer Organisation verwenden.

     > **Note**
     >
     > Sie sollten Ihr Verwaltungskonto nur verwenden, um IAM Identity Center zu aktivieren und Benutzer und Gruppen bereitzustellen. Nachdem Sie IAM Identity Center eingerichtet haben, verwenden Sie ein Mitgliedskonto, um ein EMR Studio zu erstellen und Benutzer und Gruppen zuzuweisen. Weitere Informationen zur AWS Terminologie finden Sie unter AWS Organizations Terminologie und Konzepte.

   - Wenn Sie IAM Identity Center vor dem 25. November 2019 aktiviert haben, müssen Sie möglicherweise Anwendungen aktivieren, die IAM Identity Center für die Konten in Ihrer AWS Organisation verwenden. Weitere Informationen finden Sie unter Aktivieren von IAM Identity Center-integrierten Anwendungen in AWS Konten.
   - Vergewissern Sie sich, dass die Voraussetzungen auf der Seite mit den Voraussetzungen für IAMIdentity Center aufgeführt sind.
2. Folgen Sie den Anweisungen unter IAMIdentity Center aktivieren, um IAM Identity Center dort zu aktivieren, AWS-Region wo Sie das EMR Studio erstellen möchten.
3. Connect IAM Identity Center mit Ihrem Identitätsanbieter und stellen Sie die Benutzer und Gruppen bereit, die Sie dem Studio zuweisen möchten.

Wenn Sie …	Vorgehensweise
Ein Microsoft-AD-Verzeichnis verwenden	1. Folgen Sie den Anweisungen unter Connect zu Ihrem Microsoft AD-Verzeichnis herstellen, um Ihr selbstverwaltetes Active Directory oder AWS Managed Microsoft AD Verzeichnis mithilfe von AWS Directory Service zu verbinden. 2. Um Benutzer und Gruppen für IAM Identity Center bereitzustellen, können Sie Identitätsdaten aus Ihrem Quell-AD mit IAM Identity Center synchronisieren. Sie können Identitäten aus Ihrem Quell-AD auf viele Arten synchronisieren. Eine Möglichkeit besteht darin, AD-Benutzer oder -Gruppen einem AWS -Konto in Ihrer Organisation zuzuweisen. Anweisungen finden Sie unter Single Sign-On. Die Synchronisation kann bis zu zwei Stunden dauern. Nach diesem Schritt werden synchronisierte Benutzer und Gruppen in Ihrem Identitätsspeicher angezeigt. **Note** Benutzer und Gruppen werden erst in Ihrem Identity Store angezeigt, wenn Sie Benutzer- und Gruppeninformationen synchronisieren oder die Benutzerverwaltung just-in-time (JIT) verwenden. Weitere Informationen finden Sie unter Bereitste

Wenn Sie …	Vorgehensweise
	llung, wenn Benutzer von Active Directory kommen. 3. (Optional) Nachdem Sie AD-Benutzer und -Gruppen synchronisiert haben, können Sie ihnen den Zugriff auf Ihr AWS Konto, das Sie im vorherigen Schritt konfiguriert haben, entziehen. Anweisungen finden Sie unter Benutzerzugriff entfernen.
Ein externer Identitätsanbieter	Folgen Sie den Anweisungen unter Verbindung zu Ihrem externen Identität sanbieter herstellen.
Das IAM Identity Center-Verzeichnis	Wenn Sie Benutzer und Gruppen in IAM Identity Center erstellen, erfolgt die Bereitste llung automatisch. Weitere Informationen finden Sie unter Identitäten in IAM Identity Center verwalten.

Sie können jetzt Benutzer und Gruppen aus Ihrem Identity Store einem EMR Studio zuweisen. Detaillierte Anweisungen finden Sie unter Weisen Sie einem EMR Studio einen Benutzer oder eine Gruppe zu.

Erstellen Sie eine EMR Studio-Dienstrolle

Über die EMR Studio-Servicerolle

Jedes EMR Studio verwendet eine IAM Rolle mit Berechtigungen, die es dem Studio ermöglichen, mit anderen AWS Diensten zu interagieren. Diese Servicerolle muss Berechtigungen beinhalten, die es EMR Studio ermöglichen, einen sicheren Netzwerkkanal zwischen Workspaces und Clustern einzurichten, Notebook-Dateien darin zu speichern und darauf zuzugreifen Amazon S3 Control, AWS Secrets Manager während ein Workspace mit einem Git-Repository verknüpft wird.

Verwenden Sie die Studio-Servicerolle (anstelle von Sitzungsrichtlinien), um alle Amazon-S3-Zugriffsberechtigungen für das Speichern von Notebookdateien und AWS Secrets Manager Zugriffsberechtigungen zu definieren.

So erstellen Sie eine Servicerolle für EMR Studio bei Amazon EC2 oder Amazon EKS

1. Folgen Sie den Anweisungen unter Erstellen einer Rolle zum Delegieren von Berechtigungen für einen AWS Dienst, um die Dienstrolle mit der folgenden Vertrauensrichtlinie zu erstellen.

   > **Important**
   >
   > Die folgende Vertrauensrichtlinie umfasst die `aws:SourceArn`und `aws:SourceAccount`globale Bedingungsschlüssel, mit denen Sie die Berechtigungen einschränken können, die Sie EMR Studio auf bestimmte Ressourcen in Ihrem Konto gewähren. Auf diese Weise können Sie sich vor dem Problem des verwirrten Stellvertreters schützen.

   ```
   {
     "Version": "2012-10-17",
     "Statement": [
       {
         "Effect": "Allow",
         "Principal": {
           "Service": "elasticmapreduce.amazonaws.com"
         },
         "Action": "sts:AssumeRole",
         "Condition": {
           "StringEquals": {
             "aws:SourceAccount": "<account-id>"
           },
           "ArnLike": {
             "aws:SourceArn": "arn:aws:elasticmapreduce:<region>:<account-id>:*"
           }
         }
       }
     ]
   }
   ```

2. Entfernen Sie die standardmäßigen Rollenberechtigungen. Fügen Sie dann die Berechtigungen aus der folgenden IAM Beispielberechtigungsrichtlinie hinzu. Alternativ können Sie eine

benutzerdefinierte Richtlinie erstellen, die das EMRBerechtigungen für die Studio-Dienstrolle verwendet.

> **Important**
>
> - Damit die EC2 Tag-basierte Zugriffskontrolle von Amazon mit EMR Studio funktioniert, müssen Sie den Zugriff `ModifyNetworkInterfaceAttribute` API wie in der folgenden Richtlinie gezeigt einrichten.
> - Damit EMR Studio mit der Servicerolle funktioniert, dürfen Sie die folgenden Anweisungen nicht ändern: `AllowAddingEMRTagsDuringDefaultSecurityGroupCreation` und`AllowAddingTagsDuringEC2ENICreation`.
> - Um die Beispielrichtlinie verwenden zu können, müssen Sie die folgenden Ressourcen mit dem Schlüssel **`"for-use-with-amazon-emr-managed-policies"`** und dem Wert **`"true"`** kennzeichnen.
>   - Ihre Amazon Virtual Private Cloud (VPC) für EMR Studio.
>   - Jedes Subnetz, das Sie mit dem Studio verwenden möchten.
>   - Alle benutzerdefinierten EMR Studio-Sicherheitsgruppen. Sie müssen alle Sicherheitsgruppen, die Sie während der EMR Studio-Vorschauphase erstellt haben, taggen, wenn Sie sie weiterhin verwenden möchten.
>   - Geheimnisse AWS Secrets Manager , die in Studio-Benutzern verwaltet werden, um Git-Repositorys mit einem Workspace zu verknüpfen.
>
>   Sie können Tags auf Ressourcen anwenden, indem Sie die Registerkarte Tags auf dem entsprechenden Ressourcenbildschirm in verwenden. AWS Management Console

Falls zutreffend, ändern Sie die Angabe `* "Resource":"*"` in der folgenden Richtlinie, um den Amazon-Ressourcennamen (ARN) der Ressourcen anzugeben, auf die sich die Erklärung für Ihren Anwendungsfall bezieht.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
```

```
      "Sid": "AllowEMRReadOnlyActions",
      "Effect": "Allow",
      "Action": [
        "elasticmapreduce:ListInstances",
        "elasticmapreduce:DescribeCluster",
        "elasticmapreduce:ListSteps"
      ],
      "Resource": "*"
    },
    {
      "Sid": "AllowEC2ENIActionsWithEMRTags",
      "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterfacePermission",
        "ec2:DeleteNetworkInterface"
      ],
      "Resource": [
        "arn:aws:ec2:*:*:network-interface/*"
      ],
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/for-use-with-amazon-emr-managed-policies": "true"
        }
      }
    },
    {
      "Sid": "AllowEC2ENIAttributeAction",
      "Effect": "Allow",
      "Action": [
        "ec2:ModifyNetworkInterfaceAttribute"
      ],
      "Resource": [
        "arn:aws:ec2:*:*:instance/*",
        "arn:aws:ec2:*:*:network-interface/*",
        "arn:aws:ec2:*:*:security-group/*"
      ]
    },
    {
      "Sid": "AllowEC2SecurityGroupActionsWithEMRTags",
      "Effect": "Allow",
      "Action": [
        "ec2:AuthorizeSecurityGroupEgress",
        "ec2:AuthorizeSecurityGroupIngress",
        "ec2:RevokeSecurityGroupEgress",
```

```
        "ec2:RevokeSecurityGroupIngress",
        "ec2:DeleteNetworkInterfacePermission"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/for-use-with-amazon-emr-managed-policies": "true"
        }
      }
    },
    {
      "Sid": "AllowDefaultEC2SecurityGroupsCreationWithEMRTags",
      "Effect": "Allow",
      "Action": [
        "ec2:CreateSecurityGroup"
      ],
      "Resource": [
        "arn:aws:ec2:*:*:security-group/*"
      ],
      "Condition": {
        "StringEquals": {
          "aws:RequestTag/for-use-with-amazon-emr-managed-policies": "true"
        }
      }
    },
    {
      "Sid": "AllowDefaultEC2SecurityGroupsCreationInVPCWithEMRTags",
      "Effect": "Allow",
      "Action": [
        "ec2:CreateSecurityGroup"
      ],
      "Resource": [
        "arn:aws:ec2:*:*:vpc/*"
      ],
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/for-use-with-amazon-emr-managed-policies": "true"
        }
      }
    },
    {
      "Sid": "AllowAddingEMRTagsDuringDefaultSecurityGroupCreation",
      "Effect": "Allow",
      "Action": [
```

```
        "ec2:CreateTags"
      ],
      "Resource": "arn:aws:ec2:*:*:security-group/*",
      "Condition": {
        "StringEquals": {
          "aws:RequestTag/for-use-with-amazon-emr-managed-policies": "true",
          "ec2:CreateAction": "CreateSecurityGroup"
        }
      }
    },
    {
      "Sid": "AllowEC2ENICreationWithEMRTags",
      "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterface"
      ],
      "Resource": [
        "arn:aws:ec2:*:*:network-interface/*"
      ],
      "Condition": {
        "StringEquals": {
          "aws:RequestTag/for-use-with-amazon-emr-managed-policies": "true"
        }
      }
    },
    {
      "Sid": "AllowEC2ENICreationInSubnetAndSecurityGroupWithEMRTags",
      "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterface"
      ],
      "Resource": [
        "arn:aws:ec2:*:*:subnet/*",
        "arn:aws:ec2:*:*:security-group/*"
      ],
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/for-use-with-amazon-emr-managed-policies": "true"
        }
      }
    },
    {
      "Sid": "AllowAddingTagsDuringEC2ENICreation",
      "Effect": "Allow",
```

```
      "Action": [
        "ec2:CreateTags"
      ],
      "Resource": "arn:aws:ec2:*:*:network-interface/*",
      "Condition": {
        "StringEquals": {
          "ec2:CreateAction": "CreateNetworkInterface"
        }
      }
    },
    {
      "Sid": "AllowEC2ReadOnlyActions",
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeSecurityGroups",
        "ec2:DescribeNetworkInterfaces",
        "ec2:DescribeTags",
        "ec2:DescribeInstances",
        "ec2:DescribeSubnets",
        "ec2:DescribeVpcs"
      ],
      "Resource": "*"
    },
    {
      "Sid": "AllowSecretsManagerReadOnlyActionsWithEMRTags",
      "Effect": "Allow",
      "Action": [
        "secretsmanager:GetSecretValue"
      ],
      "Resource": "arn:aws:secretsmanager:*:*:secret:*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/for-use-with-amazon-emr-managed-policies": "true"
        }
      }
    },
    {
      "Sid": "AllowWorkspaceCollaboration",
      "Effect": "Allow",
      "Action": [
        "iam:GetUser",
        "iam:GetRole",
        "iam:ListUsers",
        "iam:ListRoles",
```

```
        "sso:GetManagedApplicationInstance",
        "sso-directory:SearchUsers"
      ],
      "Resource": "*"
    }
  ]
}
```

3. Erteilen Sie Ihrer Servicerolle Lese- und Schreibzugriff auf Ihren Amazon S3 S3-Standort für EMR Studio. Verwenden Sie die folgenden Mindestberechtigungen. Für weitere Informationen finden Sie unter Amazon S3: Gewährt Lese- und Schreibzugriff auf Objekte in einem S3-Bucket programmgesteuert und in der Konsole.

```
"s3:PutObject",
"s3:GetObject",
"s3:GetEncryptionConfiguration",
"s3:ListBucket",
"s3:DeleteObject"
```

   Wenn Sie Ihren Amazon-S3-Bucket verschlüsseln, fügen Sie die folgenden Berechtigungen für AWS Key Management Service hinzu.

```
"kms:Decrypt",
"kms:GenerateDataKey",
"kms:ReEncryptFrom",
"kms:ReEncryptTo",
"kms:DescribeKey"
```

4. Wenn Sie den Zugriff auf Git-Secrets auf Benutzerebene kontrollieren möchten, fügen Sie der EMR Studio-Benutzerrollenrichtlinie tagbasierte Berechtigungen hinzu und entfernen Sie die Richtlinienberechtigungen aus der EMR Studio-Dienstrollenrichtlinie. `secretsmanager:GetSecretValue` `secretsmanager:GetSecretValue` Weitere Informationen zum Festlegen von differenzierten Benutzerberechtigungen finden Sie unter Erstellen Sie Berechtigungsrichtlinien für EMR Studio-Benutzer.

Minimale Servicerolle für Serverless EMR

Wenn Sie interaktive Workloads mit EMR Serverless über EMR Studio-Notebooks ausführen möchten, verwenden Sie dieselbe Vertrauensrichtlinie, die Sie für die Einrichtung von EMR Studio im

vorherigen Abschnitt verwendet haben. So erstellen Sie eine Servicerolle für EMR Studio bei Amazon EC2 oder Amazon EKS

Für Ihre IAM Richtlinie verfügt die Mindestrichtlinie über die folgenden Berechtigungen. Geben Sie *bucket-name* bei der Konfiguration von EMR Studio und Workspace den Namen des Buckets an, den Sie verwenden möchten. EMRStudio verwendet den Bucket, um die Workspaces und Notizbuchdateien in Ihrem Studio zu sichern.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ObjectActions",
            "Effect": "Allow",
            "Action": [
               "s3:PutObject",
               "s3:GetObject",
               "s3:DeleteObject"
            ],
            "Resource": ["arn:aws:s3:::bucket-name/*"]
        },
        {
            "Sid": "BucketActions",
            "Effect": "Allow",
            "Action": [
               "s3:ListBucket",
               "s3:GetEncryptionConfiguration"
            ],
            "Resource": ["arn:aws:s3:::bucket-name"]
        }
    ]
}
```

Wenn Sie beabsichtigen, einen verschlüsselten Amazon-S3-Bucket zu verwenden, fügen Sie Ihrer Richtlinie die folgenden Berechtigungen hinzu:

```
"kms:Decrypt",
"kms:GenerateDataKey",
"kms:ReEncryptFrom",
"kms:ReEncryptTo",
"kms:DescribeKey"
```

EMRBerechtigungen für die Studio-Dienstrolle

In der folgenden Tabelle sind die Operationen aufgeführt, die EMR Studio mithilfe der Servicerolle ausführt, sowie die für jeden Vorgang erforderlichen IAM Aktionen.

Operation	Aktionen
Richten Sie einen sicheren Netzwerkkanal zwischen einem Workspace und einem EMR Cluster ein und führen Sie die erforderlichen Bereinigungsaktionen durch.	`"ec2:CreateNetworkInterface", "ec2:CreateNetworkInterfacePermission", "ec2:DeleteNetworkInterface", "ec2:DeleteNetworkInterfacePermission", "ec2:DescribeNetworkInterfaces", "ec2:ModifyNetworkInterfaceAttribute", "ec2:AuthorizeSecurityGroupEgress", "ec2:AuthorizeSecurityGroupIngress", "ec2:CreateSecurityGroup", "ec2:DescribeSecurityGroups", "ec2:RevokeSecurityGroupEgress", "ec2:DescribeTags", "ec2:DescribeInstances", "ec2:DescribeSubnets", "ec2:DescribeVpcs", "elasticmapreduce:ListInstances", "elasticmapreduce:DescribeCluster", "elasticmapreduce:ListSteps"`
Verwenden Sie die in gespeicherten Git-Anmeldeinformationen AWS Secrets Manager , um Git-Repositorys mit einem Workspace zu verknüpfen.	`"secretsmanager:GetSecretValue"`
Wenden Sie AWS Tags auf die Netzwerkschnittstelle und die Standardsicherheitsgruppen an, die EMR Studio bei der Einrichtung des sicheren Netzwerkkanals erstellt. Weitere Informationen finden Sie	`"ec2:CreateTags"`

Operation	Aktionen
unter Markieren von AWS -Ressourcen.	

Greifen Sie auf Notebook-Dateien und Metadaten zu oder laden Sie sie in Amazon S3 hoch.	`"s3:PutObject",`
	`"s3:GetObject",`
	`"s3:GetEncryptionConfiguration",`
	`"s3:ListBucket",`
	`"s3:DeleteObject"`
	
	Wenn Sie einen verschlüsselten Amazon-S3-Bucket verwenden, schließen Sie die folgenden Berechtigungen ein.
	
	`"kms:Decrypt",`
	`"kms:GenerateDataKey",`
	`"kms:ReEncryptFrom",`
	`"kms:ReEncryptTo",`
	`"kms:DescribeKey"`

Aktivieren und konfigurieren Sie die Workspace-Zusammenarbeit.	`"iam:GetUser",`
	`"iam:GetRole",`
	`"iam:ListUsers",`
	`"iam:ListRoles",`
	`"sso:GetManagedApplicationInstance",`
	`"sso-directory:SearchUsers"`

Verschlüsseln Sie EMR Studio Workspace-Notizbücher und -Dateien mithilfe von vom Kunden verwalteten Schlüsseln (CMK) mit AWS Key Management Service	`"kms:Decrypt",`
	`"kms:GenerateDataKey",`
	`"kms:ReEncryptFrom",`
	`"kms:ReEncryptTo",`
	`"kms:DescribeKey"`

EMRStudio-Benutzerberechtigungen für Amazon EC2 oder Amazon konfigurieren EKS

Sie müssen Benutzerberechtigungsrichtlinien für Amazon EMR Studio konfigurieren, damit Sie detaillierte Benutzer- und Gruppenberechtigungen festlegen können. Informationen zur

Funktionsweise von Benutzerberechtigungen in EMR Studio finden Sie unterZugriffskontrolle. Wie Amazon EMR Studio funktioniert

> **Note**
>
> Die in diesem Abschnitt behandelten Berechtigungen erzwingen keine Datenzugriffskontrolle. Um den Zugriff auf Eingabe-Datensätze zu verwalten, sollten Sie Berechtigungen für die Cluster konfigurieren, die Ihr Studio verwendet. Weitere Informationen finden Sie unter Sicherheit bei Amazon EMR.

Erstellen Sie eine EMR Studio-Benutzerrolle für den IAM Identity Center-Authentifizierungsmodus

Sie müssen eine EMR Studio-Benutzerrolle erstellen, wenn Sie den IAM Identity Center-Authentifizierungsmodus verwenden.

Um eine Benutzerrolle für EMR Studio zu erstellen

1. Folgen Sie den Anweisungen unter Erstellen einer Rolle zum Delegieren von Berechtigungen für einen AWS Dienst im Benutzerhandbuch, um eine AWS Identity and Access Management Benutzerrolle zu erstellen.

   Verwenden Sie beim Erstellen der Rolle die folgende Vertrauensbeziehungsrichtlinie.

   ```
   {
     "Version": "2008-10-17",
     "Statement": [
       {
         "Effect": "Allow",
         "Principal": {
           "Service": "elasticmapreduce.amazonaws.com"
         },
         "Action": [
                 "sts:AssumeRole",
                 "sts:SetContext"
                 ]
       }
     ]
   }
   ```

2. Entfernen Sie die standardmäßigen Rollenberechtigungen und -richtlinien.

3. Bevor Sie einem Studio Benutzer und Gruppen zuweisen, fügen Sie der Benutzerrolle Ihre EMR Studio-Sitzungsrichtlinien hinzu. Anweisungen zum Erstellen von Sitzungsrichtlinien finden Sie unter Erstellen Sie Berechtigungsrichtlinien für EMR Studio-Benutzer.

Erstellen Sie Berechtigungsrichtlinien für EMR Studio-Benutzer

In den folgenden Abschnitten finden Sie Informationen zum Erstellen von Berechtigungsrichtlinien für EMR Studio.

Themen

- Die Berechtigungsrichtlinien erstellen
- Legen Sie die Eigentümerschaft für die Workspace-Zusammenarbeit fest
- Git-Secrets-Richtlinie auf Benutzerebene erstellen
- Ordnen Sie die Berechtigungsrichtlinie Ihrer IAM Identität zu

> **Note**
>
> Verwenden Sie die EMR Studio-Servicerolle, um Amazon S3 S3-Zugriffsberechtigungen für das Speichern von Notizbuchdateien und AWS Secrets Manager Zugriffsberechtigungen für das Lesen von Geheimnissen festzulegen, wenn Sie Workspaces mit Git-Repositorys verknüpfen.

Die Berechtigungsrichtlinien erstellen

Erstellen Sie eine oder mehrere IAM Berechtigungsrichtlinien, die festlegen, welche Aktionen ein Benutzer in Ihrem Studio ausführen kann. Mit den Beispielrichtlinien auf dieser Seite können Sie beispielsweise drei separate Richtlinien für einfache, fortgeschrittene und erfahrene Studio-Benutzer erstellen.

Eine Aufschlüsselung der einzelnen Studio-Operationen, die ein Benutzer ausführen kann, sowie der IAM Mindestaktionen, die für die einzelnen Operationen erforderlich sind, finden Sie unterAWS Identity and Access Management Berechtigungen für EMR Studio-Benutzer. Die Schritte zum Erstellen der Richtlinien finden Sie unter IAMRichtlinien erstellen im IAMBenutzerhandbuch.

Ihre Berechtigungsrichtlinie muss die folgenden Aussagen enthalten.

```
{
```

```
            "Sid": "AllowAddingTagsOnSecretsWithEMRStudioPrefix",
            "Effect": "Allow",
            "Action": "secretsmanager:TagResource",
            "Resource": "arn:aws:secretsmanager:*:*:secret:emr-studio-*"
},
{
            "Sid": "AllowPassingServiceRoleForWorkspaceCreation",
            "Action": "iam:PassRole",
            "Resource": [
                "arn:aws:iam::*:role/your-emr-studio-service-role"
            ],
            "Effect": "Allow"
}
```

Legen Sie die Eigentümerschaft für die Workspace-Zusammenarbeit fest

Mithilfe von Workspace Collaboration können mehrere Benutzer gleichzeitig im selben Workspace arbeiten. Sie kann über das Collaboration-Bedienfeld in der Workspace-Benutzeroberfläche konfiguriert werden. Um das Collaboration Panel sehen und verwenden zu können, muss ein Benutzer über die folgenden Berechtigungen verfügen. Jeder Benutzer mit diesen Berechtigungen kann das Panel Zusammenarbeit sehen und verwenden.

```
"elasticmapreduce:UpdateEditor",
"elasticmapreduce:PutWorkspaceAccess",
"elasticmapreduce:DeleteWorkspaceAccess",
"elasticmapreduce:ListWorkspaceAccessIdentities"
```

Um den Zugriff auf das Panel Zusammenarbeit einzuschränken, können Sie die tagbasierte Zugriffskontrolle verwenden. Wenn ein Benutzer einen Workspace erstellt, wendet EMR Studio ein Standard-Tag mit einem Schlüssel an`creatorUserId`, dessen Wert der ID des Benutzers entspricht, der den Workspace erstellt.

> **Note**
>
> EMRStudio fügt das `creatorUserId` Tag zu Arbeitsbereichen hinzu, die nach dem 16. November 2021 erstellt wurden. Um einzuschränken, wer die Zusammenarbeit für vor dem Datum erstellte Workspaces konfigurieren kann, empfehlen wir, das `creatorUserId`-Tag manuell zu Ihrem Workspace hinzuzufügen und dann die tagbasierte Zugriffskontrolle in Ihren Benutzerberechtigungsrichtlinien zu verwenden.

Die folgende Beispielanweisung ermöglicht es einem Benutzer, die Zusammenarbeit für jeden Workspace mit dem Tag-Schlüssel `creatorUserId` zu konfigurieren, dessen Wert der Benutzer-ID entspricht (angegeben durch die Richtlinienvariable `aws:userId`). Mit anderen Worten, die Anweisung ermöglicht es einem Benutzer, die Zusammenarbeit für die von ihm erstellten Workspaces zu konfigurieren. Weitere Informationen zu Richtlinienvariablen finden Sie im IAMBenutzerhandbuch unter IAMRichtlinienelemente: Variablen und Tags.

```
    {
        "Sid": "UserRolePermissionsForCollaboration",
        "Action": [
            "elasticmapreduce:UpdateEditor",
            "elasticmapreduce:PutWorkspaceAccess",
            "elasticmapreduce:DeleteWorkspaceAccess",
            "elasticmapreduce:ListWorkspaceAccessIdentities"
        ],
        "Resource": "*",
        "Effect": "Allow",
        "Condition": {
            "StringEquals": {
                "elasticmapreduce:ResourceTag/creatorUserId": "${aws:userid}"
            }
        }
    }
```

Git-Secrets-Richtlinie auf Benutzerebene erstellen

Themen

- So verwenden Sie Berechtigungen auf Benutzerebene
- So gehen Sie von Berechtigungen auf Serviceebene zu Berechtigungen auf Benutzerebene über
- Berechtigungen auf Serviceebene

So verwenden Sie Berechtigungen auf Benutzerebene

EMRStudio fügt das `for-use-with-amazon-emr-managed-user-policies` Tag automatisch hinzu, wenn es Git-Secrets erstellt. Wenn Sie den Zugriff auf Git-Secrets auf Benutzerebene kontrollieren möchten, fügen Sie der EMR Studio-Benutzerrollenrichtlinie tagbasierte Berechtigungen hinzu, `secretsmanager:GetSecretValue` wie im folgenden So gehen Sie von Berechtigungen auf Serviceebene zu Berechtigungen auf Benutzerebene über Abschnitt gezeigt.

Wenn Sie `secretsmanager:GetSecretValue` in der EMR Studio-Dienstrollenrichtlinie bereits über Berechtigungen verfügen, sollten Sie diese Berechtigungen entfernen.

So gehen Sie von Berechtigungen auf Serviceebene zu Berechtigungen auf Benutzerebene über

> **Note**
>
> Das `for-use-with-amazon-emr-managed-user-policies`-Tag stellt sicher, dass die Berechtigungen aus Schritt 1 unten dem Ersteller des Workspace Zugriff auf das Git-Secret gewähren. Wenn Sie Git-Repositorys jedoch vor dem 1. September 2023 verlinkt haben, wird den entsprechenden Git-Secrets der Zugriff verweigert, da das `for-use-with-amazon-emr-managed-user-policies`-Tag nicht auf sie angewendet wurde. Um Berechtigungen auf Benutzerebene anzuwenden, müssen Sie die alten Geheimnisse aus den entsprechenden Git-Repositorys neu erstellen JupyterLab und die entsprechenden Git-Repositorys erneut verknüpfen.
> Weitere Informationen zu Richtlinienvariablen finden Sie unter IAMRichtlinienelemente: Variablen und Tags im IAM Benutzerhandbuch.

1. Fügen Sie der EMRStudio-Benutzerrollenrichtlinie die folgenden Berechtigungen hinzu. Sie verwendet den `for-use-with-amazon-emr-managed-user-policies`-Schlüssel mit dem Wert `"${aws:userid}"`.

```
{
   "Sid": "AllowSecretsManagerReadOnlyActionsWithEMRTags",
    "Effect": "Allow",
    "Action": "secretsmanager:GetSecretValue",
    "Resource": "arn:aws:secretsmanager:*:*:secret:*",
    "Condition": {
        "StringEquals": {
            "secretsmanager:ResourceTag/for-use-with-amazon-emr-managed-user-
policies": "${aws:userid}"
        }
    }
}
```

2. Falls vorhanden, entfernen Sie die folgende Berechtigung aus der EMRStudio-Dienstrollenrichtlinie. Da die Servicerollenrichtlinie für alle von jedem Benutzer definierten Secrets gilt, müssen Sie dies nur einmal tun.

```
{
    "Sid": "AllowSecretsManagerReadOnlyActionsWithEMRTags",
    "Effect": "Allow",
    "Action": [
        "secretsmanager:GetSecretValue"
     ],
    "Resource": "arn:aws:secretsmanager:*:*:secret:*",
    "Condition": {
        "StringEquals": {
            "aws:ResourceTag/for-use-with-amazon-emr-managed-policies": "true"
        }
    }
}
```

Berechtigungen auf Serviceebene

Ab dem 1. September 2023 fügt EMR Studio automatisch das `for-use-with-amazon-emr-managed-user-policies` Tag für die Zugriffskontrolle auf Benutzerebene hinzu. Da es sich um eine zusätzliche Funktion handelt, können Sie weiterhin den Zugriff auf Dienstebene verwenden, der über die `GetSecretValue` Berechtigung in der EMRStudio-Servicerolle verfügbar ist.

Für Geheimnisse, die vor dem 1. September 2023 erstellt wurden, hat EMR Studio das `for-use-with-amazon-emr-managed-user-policies` Tag nicht hinzugefügt. Um weiterhin Berechtigungen auf Dienstebene zu verwenden, behalten Sie einfach Ihre bestehenden EMRStudio-Dienstrollen- und Benutzerrollenberechtigungen bei. Um jedoch einzuschränken, wer auf ein einzelnes Secret zugreifen kann, empfehlen wir, dass Sie die Schritte unter So verwenden Sie Berechtigungen auf Benutzerebene zum Hinzufügen des `for-use-with-amazon-emr-managed-user-policies`-Tags zu Ihren Secrets befolgen und dann die tagbasierte Zugriffskontrolle in Ihren Benutzerberechtigungsrichtlinien verwenden.

Weitere Informationen zu Richtlinienvariablen finden Sie im IAMBenutzerhandbuch unter IAMRichtlinienelemente: Variablen und Tags.

Ordnen Sie die Berechtigungsrichtlinie Ihrer IAM Identität zu

In der folgenden Tabelle wird zusammengefasst, an welche IAM Identität Sie je nach EMR Studio-Authentifizierungsmodus eine Berechtigungsrichtlinie anhängen. Anweisungen zum Anhängen einer Richtlinie finden Sie unter Hinzufügen und Entfernen von IAM Identitätsberechtigungen.

Wenn Sie …	Fügen Sie die Richtlinie an …
IAMAuthentifizierung	Ihre IAM Identitäten (Benutzer, Benutzergruppen oder Rollen). Sie können einem Benutzer in Ihrem AWS-Konto eine Berechtigungsrichtlinie zuweisen.
IAMVerbund mit einem externen Identitätsanbieter (IdP)	Die IAM Rolle oder Rollen, die Sie für Ihren externen IdP erstellen. Zum Beispiel ein IAM For SAML 2.0-Föderation. EMRStudio verwendet die Berechtigungen, die Sie Ihren IAM Rollen zuordnen, für Benutzer mit Verbundzugriff auf ein Studio.
IAMIdentity Center	Ihre Amazon EMR Studio-Benutzerrolle.

Beispielbenutzerrichtlinien

Die folgende grundlegende Benutzerrichtlinie erlaubt die meisten EMR Studio-Aktionen, erlaubt es einem Benutzer jedoch nicht, neue EMR Amazon-Cluster zu erstellen.

Grundlegende Richtlinien

> **Important**
>
> Die Beispielrichtlinie beinhaltet nicht die `CreateStudioPresignedUrl` Erlaubnis, die Sie einem Benutzer gewähren müssen, wenn Sie den IAM Authentifizierungsmodus verwenden. Weitere Informationen finden Sie unter Weisen Sie einem EMR Studio einen Benutzer oder eine Gruppe zu.

Die Beispielrichtlinie enthält `Condition` Elemente zur Durchsetzung einer tagbasierten Zugriffskontrolle (TBAC), sodass Sie die Richtlinie zusammen mit der Beispieldienstrolle für EMR Studio verwenden können. Weitere Informationen finden Sie unter Erstellen Sie eine EMR Studio-Dienstrolle.

```
{
```

```
    "Version":"2012-10-17",
    "Statement":[
       {
          "Sid":"AllowDefaultEC2SecurityGroupsCreationInVPCWithEMRTags",
          "Effect":"Allow",
          "Action":[
             "ec2:CreateSecurityGroup"
          ],
          "Resource":[
             "arn:aws:ec2:*:*:vpc/*"
          ],
          "Condition":{
             "StringEquals":{
                "aws:ResourceTag/for-use-with-amazon-emr-managed-policies":"true"
             }
          }
       },
       {
          "Sid":"AllowAddingEMRTagsDuringDefaultSecurityGroupCreation",
          "Effect":"Allow",
          "Action":[
             "ec2:CreateTags"
          ],
          "Resource":"arn:aws:ec2:*:*:security-group/*",
          "Condition":{
             "StringEquals":{
                "aws:RequestTag/for-use-with-amazon-emr-managed-policies":"true",
                "ec2:CreateAction":"CreateSecurityGroup"
             }
          }
       },
       {
          "Sid":"AllowSecretManagerListSecrets",
          "Action":[
             "secretsmanager:ListSecrets"
          ],
          "Resource":"*",
          "Effect":"Allow"
       },
       {
          "Sid":"AllowSecretCreationWithEMRTagsAndEMRStudioPrefix",
          "Effect":"Allow",
          "Action":"secretsmanager:CreateSecret",
          "Resource":"arn:aws:secretsmanager:*:*:secret:emr-studio-*",
```

```
            "Condition":{
               "StringEquals":{
                  "aws:RequestTag/for-use-with-amazon-emr-managed-policies":"true"
               }
            }
         },
         {
            "Sid":"AllowAddingTagsOnSecretsWithEMRStudioPrefix",
            "Effect":"Allow",
            "Action":"secretsmanager:TagResource",
            "Resource":"arn:aws:secretsmanager:*:*:secret:emr-studio-*"
         },
         {
            "Sid":"AllowPassingServiceRoleForWorkspaceCreation",
            "Action":"iam:PassRole",
            "Resource":[
               "arn:aws:iam::*:role/<your-emr-studio-service-role>"
            ],
            "Effect":"Allow"
         },
         {
            "Sid":"AllowS3ListAndLocationPermissions",
            "Action":[
               "s3:ListAllMyBuckets",
               "s3:ListBucket",
               "s3:GetBucketLocation"
            ],
            "Resource":"arn:aws:s3:::*",
            "Effect":"Allow"
         },
         {
            "Sid":"AllowS3ReadOnlyAccessToLogs",
            "Action":[
               "s3:GetObject"
            ],
            "Resource":[
               "arn:aws:s3:::aws-logs-<aws-account-id>-<region>/elasticmapreduce/*"
            ],
            "Effect":"Allow"
         },
         {
            "Sid":"AllowConfigurationForWorkspaceCollaboration",
            "Action":[
               "elasticmapreduce:UpdateEditor",
```

```
            "elasticmapreduce:PutWorkspaceAccess",
            "elasticmapreduce:DeleteWorkspaceAccess",
            "elasticmapreduce:ListWorkspaceAccessIdentities"
         ],
         "Resource":"*",
         "Effect":"Allow",
         "Condition":{
            "StringEquals":{
               "elasticmapreduce:ResourceTag/creatorUserId":"${aws:userId}"
            }
         }
      },
      {
         "Sid":"DescribeNetwork",
         "Effect":"Allow",
         "Action":[
            "ec2:DescribeVpcs",
            "ec2:DescribeSubnets",
            "ec2:DescribeSecurityGroups"
         ],
         "Resource":"*"
      },
      {
         "Sid":"ListIAMRoles",
         "Effect":"Allow",
         "Action":[
            "iam:ListRoles"
         ],
         "Resource":"*"
      }
   ]
}
```

Die folgende Richtlinie für Zwischenbenutzer erlaubt die meisten EMR Studio-Aktionen und ermöglicht es einem Benutzer, mithilfe einer Cluster-Vorlage neue EMR Amazon-Cluster zu erstellen.

Zwischenrichtlinie

> **Important**
>
> Die Beispielrichtlinie beinhaltet nicht die `CreateStudioPresignedUrl` Berechtigung, die Sie einem Benutzer gewähren müssen, wenn Sie den IAM Authentifizierungsmodus

verwenden. Weitere Informationen finden Sie unter Weisen Sie einem EMR Studio einen Benutzer oder eine Gruppe zu.

Die Beispielrichtlinie enthält `Condition` Elemente zur Durchsetzung einer tagbasierten Zugriffskontrolle (TBAC), sodass Sie die Richtlinie zusammen mit der Beispieldienstrolle für EMR Studio verwenden können. Weitere Informationen finden Sie unter Erstellen Sie eine EMR Studio-Dienstrolle.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"AllowEMRBasicActions",
         "Action":[
            "elasticmapreduce:CreateEditor",
            "elasticmapreduce:DescribeEditor",
            "elasticmapreduce:ListEditors",
            "elasticmapreduce:StartEditor",
            "elasticmapreduce:StopEditor",
            "elasticmapreduce:DeleteEditor",
            "elasticmapreduce:OpenEditorInConsole",
            "elasticmapreduce:AttachEditor",
            "elasticmapreduce:DetachEditor",
            "elasticmapreduce:CreateRepository",
            "elasticmapreduce:DescribeRepository",
            "elasticmapreduce:DeleteRepository",
            "elasticmapreduce:ListRepositories",
            "elasticmapreduce:LinkRepository",
            "elasticmapreduce:UnlinkRepository",
            "elasticmapreduce:DescribeCluster",
            "elasticmapreduce:ListInstanceGroups",
            "elasticmapreduce:ListBootstrapActions",
            "elasticmapreduce:ListClusters",
            "elasticmapreduce:ListSteps",
            "elasticmapreduce:CreatePersistentAppUI",
            "elasticmapreduce:DescribePersistentAppUI",
            "elasticmapreduce:GetPersistentAppUIPresignedURL",
            "elasticmapreduce:GetOnClusterAppUIPresignedURL"
         ],
         "Resource":"*",
         "Effect":"Allow"
```

```
      },
      {
         "Sid":"AllowEMRContainersBasicActions",
         "Action":[
            "emr-containers:DescribeVirtualCluster",
            "emr-containers:ListVirtualClusters",
            "emr-containers:DescribeManagedEndpoint",
            "emr-containers:ListManagedEndpoints",
            "emr-containers:DescribeJobRun",
            "emr-containers:ListJobRuns"
         ],
         "Resource":"*",
         "Effect":"Allow"
      },
      {
          "Sid": "AllowRetrievingManagedEndpointCredentials",
          "Effect": "Allow",
          "Action": [
              "emr-containers:GetManagedEndpointSessionCredentials"
          ],
          "Resource": [
              "arn:aws:emr-containers:<region>:<account-id>:/virtualclusters/<virtual-
cluster-id>/endpoints/<managed-endpoint-id>"
          ],
          "Condition": {
              "StringEquals": {
                  "emr-containers:ExecutionRoleArn": [
                      "arn:aws:iam::<account-id>:role/<emr-on-eks-execution-role>"
                  ]
              }
          }
      },
      {
         "Sid":"AllowSecretManagerListSecrets",
         "Action":[
            "secretsmanager:ListSecrets"
         ],
         "Resource":"*",
         "Effect":"Allow"
      },
      {
         "Sid":"AllowSecretCreationWithEMRTagsAndEMRStudioPrefix",
         "Effect":"Allow",
         "Action":"secretsmanager:CreateSecret",
```

```
            "Resource":"arn:aws:secretsmanager:*:*:secret:emr-studio-*",
            "Condition":{
               "StringEquals":{
                  "aws:RequestTag/for-use-with-amazon-emr-managed-policies":"true"
               }
            }
         },
         {
            "Sid":"AllowAddingTagsOnSecretsWithEMRStudioPrefix",
            "Effect":"Allow",
            "Action":"secretsmanager:TagResource",
            "Resource":"arn:aws:secretsmanager:*:*:secret:emr-studio-*"
         },
         {
            "Sid":"AllowClusterTemplateRelatedIntermediateActions",
            "Action":[
               "servicecatalog:DescribeProduct",
               "servicecatalog:DescribeProductView",
               "servicecatalog:DescribeProvisioningParameters",
               "servicecatalog:ProvisionProduct",
               "servicecatalog:SearchProducts",
               "servicecatalog:UpdateProvisionedProduct",
               "servicecatalog:ListProvisioningArtifacts",
               "servicecatalog:ListLaunchPaths",
               "servicecatalog:DescribeRecord",
               "cloudformation:DescribeStackResources"
            ],
            "Resource":"*",
            "Effect":"Allow"
         },
         {
            "Sid":"AllowPassingServiceRoleForWorkspaceCreation",
            "Action":"iam:PassRole",
            "Resource":[
               "arn:aws:iam::*:role/<your-emr-studio-service-role>"
            ],
            "Effect":"Allow"
         },
         {
            "Sid":"AllowS3ListAndLocationPermissions",
            "Action":[
               "s3:ListAllMyBuckets",
               "s3:ListBucket",
               "s3:GetBucketLocation"
```

```
        ],
        "Resource":"arn:aws:s3:::*",
        "Effect":"Allow"
    },
    {
        "Sid":"AllowS3ReadOnlyAccessToLogs",
        "Action":[
            "s3:GetObject"
        ],
        "Resource":[
            "arn:aws:s3:::aws-logs-<aws-account-id>-<region>/elasticmapreduce/*"
        ],
        "Effect":"Allow"
    },
    {
        "Sid":"AllowConfigurationForWorkspaceCollaboration",
        "Action":[
            "elasticmapreduce:UpdateEditor",
            "elasticmapreduce:PutWorkspaceAccess",
            "elasticmapreduce:DeleteWorkspaceAccess",
            "elasticmapreduce:ListWorkspaceAccessIdentities"
        ],
        "Resource":"*",
        "Effect":"Allow",
        "Condition":{
            "StringEquals":{
                "elasticmapreduce:ResourceTag/creatorUserId":"${aws:userId}"
            }
        }
    },
    {
        "Sid":"DescribeNetwork",
        "Effect":"Allow",
        "Action":[
            "ec2:DescribeVpcs",
            "ec2:DescribeSubnets",
            "ec2:DescribeSecurityGroups"
        ],
        "Resource":"*"
    },
    {
        "Sid":"ListIAMRoles",
        "Effect":"Allow",
        "Action":[
```

```
            "iam:ListRoles"
         ],
         "Resource":"*"
      },
      {
         "Sid": "AllowServerlessActions",
         "Action": [
            "emr-serverless:CreateApplication",
            "emr-serverless:UpdateApplication",
            "emr-serverless:DeleteApplication",
            "emr-serverless:ListApplications",
            "emr-serverless:GetApplication",
            "emr-serverless:StartApplication",
            "emr-serverless:StopApplication",
            "emr-serverless:StartJobRun",
            "emr-serverless:CancelJobRun",
            "emr-serverless:ListJobRuns",
            "emr-serverless:GetJobRun",
            "emr-serverless:GetDashboardForJobRun",
            "emr-serverless:AccessInteractiveEndpoints"
         ],
         "Resource": "*",
         "Effect": "Allow"
        },
        {
           "Sid": "AllowPassingRuntimeRoleForRunningServerlessJob",
           "Action": "iam:PassRole",
           "Resource": "arn:aws:iam::*:role/serverless-runtime-role",
           "Effect": "Allow"
        }
   ]
}
```

Die folgende erweiterte Benutzerrichtlinie erlaubt alle EMR Studio-Aktionen und ermöglicht es einem Benutzer, mithilfe einer EMR Cluster-Vorlage oder durch Bereitstellung einer Cluster-Konfiguration neue Amazon-Cluster zu erstellen.

Erweiterte Richtlinien

> **Important**
>
> Die Beispielrichtlinie beinhaltet nicht die `CreateStudioPresignedUrl` Erlaubnis, die Sie einem Benutzer gewähren müssen, wenn Sie den IAM Authentifizierungsmodus verwenden.

Weitere Informationen finden Sie unter Weisen Sie einem EMR Studio einen Benutzer oder eine Gruppe zu.

Die Beispielrichtlinie enthält `Condition` Elemente zur Durchsetzung einer tagbasierten Zugriffskontrolle (TBAC), sodass Sie die Richtlinie zusammen mit der Beispieldienstrolle für EMR Studio verwenden können. Weitere Informationen finden Sie unter Erstellen Sie eine EMR Studio-Dienstrolle.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"AllowEMRBasicActions",
         "Action":[
            "elasticmapreduce:CreateEditor",
            "elasticmapreduce:DescribeEditor",
            "elasticmapreduce:ListEditors",
            "elasticmapreduce:StartEditor",
            "elasticmapreduce:StopEditor",
            "elasticmapreduce:DeleteEditor",
            "elasticmapreduce:OpenEditorInConsole",
            "elasticmapreduce:AttachEditor",
            "elasticmapreduce:DetachEditor",
            "elasticmapreduce:CreateRepository",
            "elasticmapreduce:DescribeRepository",
            "elasticmapreduce:DeleteRepository",
            "elasticmapreduce:ListRepositories",
            "elasticmapreduce:LinkRepository",
            "elasticmapreduce:UnlinkRepository",
            "elasticmapreduce:DescribeCluster",
            "elasticmapreduce:ListInstanceGroups",
            "elasticmapreduce:ListBootstrapActions",
            "elasticmapreduce:ListClusters",
            "elasticmapreduce:ListSteps",
            "elasticmapreduce:CreatePersistentAppUI",
            "elasticmapreduce:DescribePersistentAppUI",
            "elasticmapreduce:GetPersistentAppUIPresignedURL",
            "elasticmapreduce:GetOnClusterAppUIPresignedURL"
         ],
         "Resource":"*",
         "Effect":"Allow"
```

```
      },
      {
         "Sid":"AllowEMRContainersBasicActions",
         "Action":[
            "emr-containers:DescribeVirtualCluster",
            "emr-containers:ListVirtualClusters",
            "emr-containers:DescribeManagedEndpoint",
            "emr-containers:ListManagedEndpoints",
            "emr-containers:DescribeJobRun",
            "emr-containers:ListJobRuns"
         ],
         "Resource":"*",
         "Effect":"Allow"
      },
      {
          "Sid": "AllowRetrievingManagedEndpointCredentials",
          "Effect": "Allow",
          "Action": [
              "emr-containers:GetManagedEndpointSessionCredentials"
          ],
          "Resource": [
              "arn:aws:emr-containers:<region>:<account-id>:/virtualclusters/<virtual-
cluster-id>/endpoints/<managed-endpoint-id>"
          ],
          "Condition": {
              "StringEquals": {
                  "emr-containers:ExecutionRoleArn": [
                      "arn:aws:iam::<account-id>:role/<emr-on-eks-execution-role>"
                  ]
              }
          }
      },
      {
         "Sid":"AllowSecretManagerListSecrets",
         "Action":[
            "secretsmanager:ListSecrets"
         ],
         "Resource":"*",
         "Effect":"Allow"
      },
      {
         "Sid":"AllowSecretCreationWithEMRTagsAndEMRStudioPrefix",
         "Effect":"Allow",
         "Action":"secretsmanager:CreateSecret",
```

```
            "Resource":"arn:aws:secretsmanager:*:*:secret:emr-studio-*",
            "Condition":{
               "StringEquals":{
                  "aws:RequestTag/for-use-with-amazon-emr-managed-policies":"true"
               }
            }
         },
         {
            "Sid":"AllowAddingTagsOnSecretsWithEMRStudioPrefix",
            "Effect":"Allow",
            "Action":"secretsmanager:TagResource",
            "Resource":"arn:aws:secretsmanager:*:*:secret:emr-studio-*"
         },
         {
            "Sid":"AllowClusterTemplateRelatedIntermediateActions",
            "Action":[
               "servicecatalog:DescribeProduct",
               "servicecatalog:DescribeProductView",
               "servicecatalog:DescribeProvisioningParameters",
               "servicecatalog:ProvisionProduct",
               "servicecatalog:SearchProducts",
               "servicecatalog:UpdateProvisionedProduct",
               "servicecatalog:ListProvisioningArtifacts",
               "servicecatalog:ListLaunchPaths",
               "servicecatalog:DescribeRecord",
               "cloudformation:DescribeStackResources"
            ],
            "Resource":"*",
            "Effect":"Allow"
         },
         {
            "Sid":"AllowEMRCreateClusterAdvancedActions",
            "Action":[
               "elasticmapreduce:RunJobFlow"
            ],
            "Resource":"*",
            "Effect":"Allow"
         },
         {
            "Sid":"AllowPassingServiceRoleForWorkspaceCreation",
            "Action":"iam:PassRole",
            "Resource":[
               "arn:aws:iam::*:role/<your-emr-studio-service-role>",
               "arn:aws:iam::*:role/EMR_DefaultRole_V2",
```

```
            "arn:aws:iam::*:role/EMR_EC2_DefaultRole"
         ],
         "Effect":"Allow"
      },
      {
         "Sid":"AllowS3ListAndLocationPermissions",
         "Action":[
            "s3:ListAllMyBuckets",
            "s3:ListBucket",
            "s3:GetBucketLocation"
         ],
         "Resource":"arn:aws:s3:::*",
         "Effect":"Allow"
      },
      {
         "Sid":"AllowS3ReadOnlyAccessToLogs",
         "Action":[
            "s3:GetObject"
         ],
         "Resource":[
            "arn:aws:s3:::aws-logs-<aws-account-id>-<region>/elasticmapreduce/*"
         ],
         "Effect":"Allow"
      },
      {
         "Sid":"AllowConfigurationForWorkspaceCollaboration",
         "Action":[
            "elasticmapreduce:UpdateEditor",
            "elasticmapreduce:PutWorkspaceAccess",
            "elasticmapreduce:DeleteWorkspaceAccess",
            "elasticmapreduce:ListWorkspaceAccessIdentities"
         ],
         "Resource":"*",
         "Effect":"Allow",
         "Condition":{
            "StringEquals":{
               "elasticmapreduce:ResourceTag/creatorUserId":"${aws:userId}"
            }
         }
      },
      {
         "Sid" : "SageMakerDataWranglerForEMRStudio",
         "Effect" : "Allow",
         "Action" : [
```

```
            "sagemaker:CreatePresignedDomainUrl",
            "sagemaker:DescribeDomain",
            "sagemaker:ListDomains",
            "sagemaker:ListUserProfiles"
         ],
         "Resource":"*"
      },
      {
         "Sid":"DescribeNetwork",
         "Effect":"Allow",
         "Action":[
            "ec2:DescribeVpcs",
            "ec2:DescribeSubnets",
            "ec2:DescribeSecurityGroups"
         ],
         "Resource":"*"
      },
      {
         "Sid":"ListIAMRoles",
         "Effect":"Allow",
         "Action":[
            "iam:ListRoles"
         ],
         "Resource":"*"
      },
      {
         "Sid": "AllowServerlessActions",
         "Action": [
            "emr-serverless:CreateApplication",
            "emr-serverless:UpdateApplication",
            "emr-serverless:DeleteApplication",
            "emr-serverless:ListApplications",
            "emr-serverless:GetApplication",
            "emr-serverless:StartApplication",
            "emr-serverless:StopApplication",
            "emr-serverless:StartJobRun",
            "emr-serverless:CancelJobRun",
            "emr-serverless:ListJobRuns",
            "emr-serverless:GetJobRun",
            "emr-serverless:GetDashboardForJobRun",
            "emr-serverless:AccessInteractiveEndpoints"
         ],
         "Resource": "*",
         "Effect": "Allow"
```

```
        },
        {
            "Sid": "AllowPassingRuntimeRoleForRunningServerlessJob",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::*:role/serverless-runtime-role",
            "Effect": "Allow"
        },
        {
            "Sid": "AllowCodeWhisperer",
            "Effect": "Allow",
            "Action": [ "codewhisperer:GenerateRecommendations" ],
            "Resource": "*"
        },
        {
            "Sid": "AllowAthenaSQL",
            "Action": [
                "athena:StartQueryExecution",
                "athena:StopQueryExecution",
                "athena:GetQueryExecution",
                "athena:GetQueryRuntimeStatistics",
                "athena:GetQueryResults",
                "athena:ListQueryExecutions",
                "athena:BatchGetQueryExecution",
                "athena:GetNamedQuery",
                "athena:ListNamedQueries",
                "athena:BatchGetNamedQuery",
                "athena:UpdateNamedQuery",
                "athena:DeleteNamedQuery",
                "athena:ListDataCatalogs",
                "athena:GetDataCatalog",
                "athena:ListDatabases",
                "athena:GetDatabase",
                "athena:ListTableMetadata",
                "athena:GetTableMetadata",
                "athena:ListWorkGroups",
                "athena:GetWorkGroup",
                "athena:CreateNamedQuery",
                "athena:GetPreparedStatement",
                "glue:CreateDatabase",
                "glue:DeleteDatabase",
                "glue:GetDatabase",
                "glue:GetDatabases",
                "glue:UpdateDatabase",
                "glue:CreateTable",
```

```
            "glue:DeleteTable",
            "glue:BatchDeleteTable",
            "glue:UpdateTable",
            "glue:GetTable",
            "glue:GetTables",
            "glue:BatchCreatePartition",
            "glue:CreatePartition",
            "glue:DeletePartition",
            "glue:BatchDeletePartition",
            "glue:UpdatePartition",
            "glue:GetPartition",
            "glue:GetPartitions",
            "glue:BatchGetPartition",
            "kms:ListAliases",
            "kms:ListKeys",
            "kms:DescribeKey",
            "lakeformation:GetDataAccess",
            "s3:GetBucketLocation",
            "s3:GetBucketLocation",
            "s3:GetObject",
            "s3:ListBucket",
            "s3:ListBucketMultipartUploads",
            "s3:ListMultipartUploadParts",
            "s3:AbortMultipartUpload",
            "s3:PutObject",
            "s3:PutBucketPublicAccessBlock",
            "s3:ListAllMyBuckets"
        ],
        "Resource": "*",
        "Effect": "Allow"
    }
  ]
}
```

Die folgende Benutzerrichtlinie enthält die Mindestbenutzerberechtigungen, die für die Verwendung einer EMR serverlosen interaktiven Anwendung mit EMR Studio Workspaces erforderlich sind.

EMRInteraktive Richtlinie für serverlose Server

Ersetzen Sie in dieser Beispielrichtlinie, die Benutzerberechtigungen für EMR serverlose interaktive Anwendungen mit EMR Studio enthält, die Platzhalter für *`serverless-runtime-role`* and *`emr-studio-service-role`* mit Ihrer richtigen EMRStudio-Dienstrolle und EMRServerless-Runtime-Rolle.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowServerlessActions",
            "Action": [
                "emr-serverless:CreateApplication",
                "emr-serverless:UpdateApplication",
                "emr-serverless:DeleteApplication",
                "emr-serverless:ListApplications",
                "emr-serverless:GetApplication",
                "emr-serverless:StartApplication",
                "emr-serverless:StopApplication",
                "emr-serverless:StartJobRun",
                "emr-serverless:CancelJobRun",
                "emr-serverless:ListJobRuns",
                "emr-serverless:GetJobRun",
                "emr-serverless:GetDashboardForJobRun",
                "emr-serverless:AccessInteractiveEndpoints"
            ],
            "Resource": "*",
            "Effect": "Allow"
        },
        {
            "Sid": "AllowEMRBasicActions",
            "Action": [
                "elasticmapreduce:CreateEditor",
                "elasticmapreduce:DescribeEditor",
                "elasticmapreduce:ListEditors",
                "elasticmapreduce:UpdateStudio",
                "elasticmapreduce:StartEditor",
                "elasticmapreduce:StopEditor",
                "elasticmapreduce:DeleteEditor",
                "elasticmapreduce:OpenEditorInConsole",
                "elasticmapreduce:AttachEditor",
                "elasticmapreduce:DetachEditor",
                "elasticmapreduce:CreateStudio",
                "elasticmapreduce:DescribeStudio",
                "elasticmapreduce:DeleteStudio",
                "elasticmapreduce:ListStudios",
                "elasticmapreduce:CreateStudioPresignedUrl"
            ],
            "Resource": "*",
```

```
            "Effect": "Allow"
        },
        {
            "Sid": "AllowPassingRuntimeRoleForRunningEMRServerlessJob",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::*:role/serverless-runtime-role",
            "Effect": "Allow"
        },
        {
            "Sid": "AllowPassingServiceRoleForWorkspaceCreation",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::*:role/emr-studio-service-role",
            "Effect": "Allow"
        },
        {
            "Sid": "AllowS3ListAndGetPermissions",
            "Action": [
                "s3:ListAllMyBuckets",
                "s3:ListBucket",
                "s3:GetBucketLocation",
                "s3:GetObject"
            ],
            "Resource": "arn:aws:s3:::*",
            "Effect": "Allow"
        },
        {
            "Sid":"DescribeNetwork",
            "Effect":"Allow",
            "Action":[
                "ec2:DescribeVpcs",
                "ec2:DescribeSubnets",
                "ec2:DescribeSecurityGroups"
            ],
            "Resource":"*"
        },
        {
            "Sid": "ListIAMRoles",
            "Effect": "Allow",
            "Action": [
                "iam:ListRoles"
            ],
            "Resource": "*"
        }
    ]
```

```
}
```

AWS Identity and Access Management Berechtigungen für EMR Studio-Benutzer

Die folgende Tabelle enthält jeden Amazon EMR Studio-Vorgang, den ein Benutzer ausführen könnte, und listet die IAM Mindestaktionen auf, die zur Durchführung dieses Vorgangs erforderlich sind. Sie erlauben diese Aktionen in Ihren IAM Berechtigungsrichtlinien (wenn Sie die IAM Authentifizierung verwenden) oder in Ihren Sitzungsrichtlinien für Benutzerrollen (wenn Sie die IAM Identity Center-Authentifizierung verwenden) für EMR Studio.

In der Tabelle werden auch die Operationen angezeigt, die in den einzelnen Beispielberechtigungsrichtlinien für EMR Studio zulässig sind. Weitere Informationen zu Beispiel-Berechtigungsrichtlinien finden Sie unter Erstellen Sie Berechtigungsrichtlinien für EMR Studio-Benutzer.

Aktion	Basic	Intermediär	Advanced	Zugeordnete Aktionen
Arbeitsbereiche erstellen und löschen	Ja	Ja	Ja	`"elasticmapreduce:CreateEditor", "elasticmapreduce:DescribeEditor", "elasticmapreduce:ListEditors", "elasticmapreduce:DeleteEditor"`
Rufen Sie das Panel Zusammenarbeit auf, aktivieren Sie die Workspace-Zusammenarbeit und fügen Sie Mitarbeiter hinzu. Weitere Informationen finden Sie unter Legen Sie die Eigentümerschaft für die Workspace-Zusammenarbeit fest.	Ja	Ja	Ja	`"elasticmapreduce:UpdateEditor", "elasticmapreduce:PutWorkspaceAccess", "elasticmapreduce:DeleteWorkspaceAccess", "elasticmapreduce:ListWorkspaceAccessIdentities"`

Aktion	Basic	Intermediär	Advanced	Zugeordnete Aktionen
Sehen Sie sich beim Erstellen eines neuen EMR Clusters eine Liste der Amazon S3 Control Speicher-Buckets in demselben Konto wie Studio an und greifen Sie auf Container-Protokolle zu, wenn Sie eine Weboberfläche zum Debuggen von Anwendungen verwenden	Ja	Ja	Ja	`"s3:ListAllMyBuckets", "s3:ListBucket", "s3:GetBucketLocation", "s3:GetObject"`
Auf Workspaces zugreifen	Ja	Ja	Ja	`"elasticmapreduce: DescribeEditor", "elasticmapreduce:ListEditors", "elasticmapreduce:StartEditor", "elasticmapreduce:StopEditor", "elasticmapreduce:OpenEditorInConsole"`
Vorhandene EMR Amazon-Cluster, die mit dem Workspace verknüpft sind, anhängen oder trennen	Ja	Ja	Ja	`"elasticmapreduce: AttachEditor", "elasticmapreduce:DetachEditor", "elasticmapreduce:ListClusters", "elasticmapreduce: DescribeCluster", "elasticmapreduce: ListInstanceGroups", "elasticmapreduce:ListBootstrapActions"`

Aktion	Basic	Intermediär	Advanced	Zugeordnete Aktionen
Amazon EMR an Clustern anhängen oder trennen EKS	Ja	Ja	Ja	`"elasticmapreduce:AttachEditor", "elasticmapreduce:DetachEditor", "emr-containers:ListVirtualClusters", "emr-containers:DescribeVirtualCluster", "emr-containers:ListManagedEndpoints", "emr-containers:DescribeManagedEndpoint", "emr-containers:GetManagedEndpointSessionCredentials"`

Aktion	Basic	Intermediär	Advanced	Zugeordnete Aktionen
EMRServerlose Anwendungen, die dem Workspace zugeordnet sind, anhängen oder trennen	Nein	Ja	Ja	`"elasticmapreduce:AttachEditor", "elasticmapreduce:DetachEditor", "emr-serverless:GetApplication", "emr-serverless:StartApplication", "emr-serverless:ListApplications", "emr-serverless:GetDashboardForJobRun", "emr-serverless:AccessInteractiveEndpoints", "iam:PassRole"` Die `PassRole` Berechtigung ist erforderlich, um die Runtime-Rolle für EMR serverlose Jobs zu bestehen. Weitere Informationen finden Sie unter Job Runtime Roles im Amazon EMR Serverless User Guide.

Aktion	Basic	Intermediär	Advanced	Zugeordnete Aktionen
Debuggen Sie Amazon EMR bei EC2 Jobs mit persistenten Anwendungsbenutzeroberflächen	Ja	Ja	Ja	`"elasticmapreduce:CreatePersistentAppUI", "elasticmapreduce:DescribePersistentAppUI", "elasticmapreduce:GetPersistentAppUIPresignedURL", "elasticmapreduce:ListClusters", "elasticmapreduce:ListSteps", "elasticmapreduce:DescribeCluster", "s3:ListBucket", "s3:GetObject"`
Debuggen Sie Amazon EMR bei EC2 Jobs mit Benutzeroberflächen für Cluster-Anwendungen	Ja	Ja	Ja	`"elasticmapreduce:GetOnClusterAppUIPresignedURL"`

Aktion	Basic	Intermediär	Advanced	Zugeordnete Aktionen
Debuggen Sie Amazon EMR bei EKS Jobläufen mit dem Spark History Server	Ja	Ja	Ja	`"elasticmapreduce:CreatePersistentAppUI", "elasticmapreduce:DescribePersistentAppUI", "elasticmapreduce:GetPersistentAppUIPresignedURL", "emr-containers:ListVirtualClusters", "emr-containers:DescribeVirtualCluster", "emr-containers:ListJobRuns", "emr-containers:DescribeJobRun", "s3:ListBucket", "s3:GetObject"`
Git-Repositorys erstellen und löschen	Ja	Ja	Ja	`"elasticmapreduce:CreateRepository", "elasticmapreduce:DeleteRepository", "elasticmapreduce:ListRepositories", "elasticmapreduce:DescribeRepository", "secretsmanager:CreateSecret", "secretsmanager:ListSecrets", "secretsmanager:TagResource"`

Aktion	Basic	Intermediär	Advanced	Zugeordnete Aktionen
Git-Repositorys verknüpfen und trennen	Ja	Ja	Ja	`"elasticmapreduce:LinkRepository", "elasticmapreduce:UnlinkRepository", "elasticmapreduce:ListRepositories", "elasticmapreduce:DescribeRepository"`
Neue Cluster aus vordefinierten Cluster-Vorlagen erstellen	Nein	Ja	Ja	`"servicecatalog:SearchProducts", "servicecatalog:DescribeProduct", "servicecatalog:DescribeProductView", "servicecatalog:DescribeProvisioningParameters", "servicecatalog:ProvisionProduct", "servicecatalog:UpdateProvisionedProduct", "servicecatalog:ListProvisioningArtifacts", "servicecatalog:DescribeRecord", "servicecatalog:ListLaunchPaths", "cloudformation:DescribeStackResources", "elasticmapreduce:ListClusters", "elasticmapreduce:DescribeCluster"`

Aktion	Basic	Intermediär	Advanced	Zugeordnete Aktionen
Stellen Sie eine Clusterkonfiguration bereit, um neue Cluster zu erstellen.	Nein	Nein	Ja	`"elasticmapreduce:RunJobFlow", "iam:PassRole", "elasticmapreduce:ListClusters", "elasticmapreduce:DescribeCluster"`
Weisen Sie einem Studio einen Benutzer zu, wenn Sie den IAM Authentifizierungsmodus verwenden.	Nein	Nein	Nein	`"elasticmapreduce:CreateStudioPresignedUrl"`
Beschreiben Sie Netzwerkobjekte.	Ja	Ja	Ja	`{ "Version": "2012-10-17", "Statement": [ { "Sid": "DescribeNetwork", "Effect": "Allow", "Action": [ "ec2:DescribeVpcs", "ec2:DescribeSubnets", "ec2:DescribeSecurityGroups" ], "Resource": "*" } ] }`

Aktion	Basic	Intermediär	Advanced	Zugeordnete Aktionen

IAMRollen auflisten.	Ja	Ja	Ja	<pre>{
				    "Version": "2012-10-17",
				    "Statement": [
				        {
				            "Sid": "ListIAMRoles",
				            "Effect": "Allow",
				            "Action": [
				                "iam:ListRoles"
				            ],
				            "Resource": "*"
				        }
				    ]
				}</pre>

Stellen Sie von Amazon EMR Studio aus eine Connect zu SageMaker Studio her und verwenden Sie die visuelle Oberfläche von Data Wrangler.	Nein	Nein	Ja	<pre>"sagemaker:CreatePresignedDomainUrl",
				"sagemaker:DescribeDomain",
				"sagemaker:ListDomains",
				"sagemaker:ListUserProfiles"</pre>

Verwenden Sie Amazon CodeWhisperer in Ihrem EMR Studio.	Nein	Nein	Ja	<pre>"codewhisperer:GenerateRecommendations"</pre>

Aktion	Basic	Intermediär	Advanced	Zugeordnete Aktionen
Greifen Sie von Ihrem EMR Studio aus auf den Amazon Athena SQL Athena-Editor zu. Diese Liste enthält möglicher weise nicht alle Berechtig ungen, die Sie für die Nutzung aller Athena-Fe atures benötigen. Die up-to-date Liste der meisten finden Sie in der Athena-Vollzugriffsrichtlinie.	Nein	Nein	Ja	"athena:StartQueryExecution", "athena:StopQueryExecution", "athena:GetQueryExecution", "athena:GetQueryRuntimeStatistics", "athena:GetQueryResults", "athena:ListQueryExecutions", "athena:BatchGetQueryExecution", "athena:GetNamedQuery", "athena:ListNamedQueries", "athena:BatchGetNamedQuery", "athena:UpdateNamedQuery", "athena:DeleteNamedQuery", "athena:ListDataCatalogs", "athena:GetDataCatalog", "athena:ListDatabases", "athena:GetDatabase", "athena:ListTableMetadata", "athena:GetTableMetadata", "athena:ListWorkGroups", "athena:GetWorkGroup", "athena:CreateNamedQuery", "athena:GetPreparedStatement", "glue:CreateDatabase", "glue:DeleteDatabase", "glue:GetDatabase",

Aktion	Basic	Intermediär	Advanced	Zugeordnete Aktionen
				"glue:GetDatabases", "glue:UpdateDatabase", "glue:CreateTable", "glue:DeleteTable", "glue:BatchDeleteTable", "glue:UpdateTable", "glue:GetTable", "glue:GetTables", "glue:BatchCreatePartition", "glue:CreatePartition", "glue:DeletePartition", "glue:BatchDeletePartition", "glue:UpdatePartition", "glue:GetPartition", "glue:GetPartitions", "glue:BatchGetPartition", "kms:ListAliases", "kms:ListKeys", "kms:DescribeKey", "lakeformation:GetDataAccess", "s3:GetBucketLocation", "s3:GetBucketLocation", "s3:GetObject", "s3:ListBucket", "s3:ListBucketMultipartUploads", "s3:ListMultipartUploadParts", "s3:AbortMultipartUpload", "s3:PutObject", "s3:PutBucketPublicAccessBlock", "s3:ListAllMyBuckets"

Erstellen Sie ein Studio EMR

Sie können ein EMR Studio für Ihr Team mit der EMR Amazon-Konsole oder dem erstellen AWS CLI. Das Erstellen einer Studio-Instanz ist Teil der Einrichtung von Amazon EMR Studio.

Voraussetzungen

Bevor Sie ein Studio erstellen, stellen Sie sicher, dass Sie die vorherigen Aufgaben in Richten Sie ein Amazon EMR Studio ein abgeschlossen haben.

Um ein Studio mit dem zu erstellen AWS CLI, sollten Sie die neueste Version installiert haben. Weitere Informationen finden Sie unter Installieren oder Aktualisierung auf die neueste Version von AWS CLI.

> **Important**
>
> Deaktivieren Sie Proxy-Management-Tools wie FoxyProxy oder SwitchyOmega im Browser, bevor Sie ein Studio erstellen. Aktive Proxys können zu einer Netzwerkfehler-Fehlermeldung führen, wenn Sie Studio erstellen wählen.

Amazon EMR bietet Ihnen eine einfache Konsolenerfahrung zum Erstellen eines Studios, sodass Sie schnell mit den Standardeinstellungen beginnen können, um interaktive Workloads oder Batch-Jobs mit den Standardeinstellungen auszuführen. Durch das Erstellen eines EMR Studios wird auch eine EMR serverlose Anwendung erstellt, die für Ihre interaktiven Jobs bereit ist.

Wenn Sie die volle Kontrolle über die Einstellungen Ihres Studios haben möchten, können Sie Benutzerdefiniert wählen, wodurch Sie alle zusätzlichen Einstellungen konfigurieren können.

Interactive workloads

Um ein EMR Studio für interaktive Workloads zu erstellen

1. Öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie in der linken Navigationsleiste unter EMRStudio die Option Erste Schritte aus. Sie können auf der Studio-Seite auch ein neues Studio erstellen.
3. Amazon EMR bietet Standardeinstellungen für Sie, wenn Sie ein EMR Studio für interaktive Workloads erstellen, aber Sie können diese Einstellungen bearbeiten. Zu den konfigurierbaren Einstellungen gehören der Name des EMR Studios, der S3-Standort für

Ihren Workspace, die zu verwendende Servicerolle, die Workspace (s), die Sie verwenden möchten, der Name der EMR serverlosen Anwendung und die zugehörige Runtime-Rolle.

4. Wählen Sie Create Studio und starten Sie Workspace, um den Vorgang abzuschließen und zur Studios-Seite zu navigieren. Ihr neues Studio wird in der Liste mit Details wie Studio-Name, Erstellungsdatum und Studio-Zugriff angezeigtURL. Ihr Workspace wird in einem neuen Tab in Ihrem Browser geöffnet.

Batch jobs

Um ein EMR Studio für interaktive Workloads zu erstellen

1. Öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie in der linken Navigationsleiste unter EMRStudio die Option Erste Schritte aus. Sie können auf der Studio-Seite auch ein neues Studio erstellen.
3. Amazon EMR bietet Standardeinstellungen für Sie, wenn Sie ein EMR Studio für Batch-Jobs erstellen, aber Sie können diese Einstellungen bearbeiten. Zu den konfigurierbaren Einstellungen gehören der Name des EMR Studios, der Name der EMR serverlosen Anwendung und die zugehörige Runtime-Rolle.
4. Wählen Sie Create Studio und starten Sie Workspace, um den Vorgang abzuschließen und zur Studios-Seite zu navigieren. Ihr neues Studio wird in der Liste mit Details wie Studio-Name, Erstellungsdatum und Studio-Zugriff angezeigtURL. Ihr EMR Studio wird in einem neuen Tab in Ihrem Browser geöffnet.

Custom settings

Um ein EMR Studio mit benutzerdefinierten Einstellungen zu erstellen

1. Öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie in der linken Navigationsleiste unter EMRStudio die Option Erste Schritte aus. Sie können auf der Studio-Seite auch ein neues Studio erstellen.
3. Wählen Sie Studio erstellen, um die Seite Studio erstellen zu öffnen.
4. Geben Sie einen Studio-Namen ein.
5. Wählen Sie, ob Sie einen neuen S3-Bucket erstellen oder einen vorhandenen Speicherort verwenden möchten.

6. Wählen Sie den Workspace aus, der dem Studio hinzugefügt werden soll. Sie können bis zu 3 Arbeitsbereiche hinzufügen.
7. Wählen Sie unter Authentifizierung einen Authentifizierungsmodus für das Studio und geben Sie die Informationen gemäß der folgenden Tabelle ein. Weitere Informationen zur Authentifizierung für EMR Studio finden Sie unterWählen Sie einen Authentifizierungsmodus für Amazon EMR Studio.

Wenn Sie …	Vorgehensweise
IAMAuthentifizierung oder Verbund	Die Standardauthentifizierungsmethode ist AWS Identity and Access Management (IAM). Am unteren Bildschirmrand können Sie auch Tags hinzufügen, um bestimmte n Benutzern Zugriff auf das Studio zu gewähren, wie unter Weisen Sie einem EMR Studio einen Benutzer oder eine Gruppe zu beschrieben. Wenn Sie möchten, dass sich Verbundbe nutzer mit Studio URL und Anmeldein formationen für Ihren Identity Provider (IdP) anmelden, wählen Sie Ihren IdP aus der Dropdownliste aus und geben Sie Ihren Identity Provider-Anmeldenamen (IdP) und den Parameternamen ein. URL RelayState Eine Liste der IdP-Authentifizierung URLs und der RelayState Namen finden Sie unter RelayState Parameter und Authentif izierung des Identitätsanbieters URLs.
IAMIdentity Center-Authentifizierung	Wählen Sie Ihre EMR Studio-Dienstrolle und Ihre Benutzerrolle aus. Weitere Informationen erhalten Sie unter Erstellen Sie eine EMR Studio-Dienstrolle und Erstellen Sie eine EMR Studio-Benutzerrol

Wenn Sie …	Vorgehensweise
	le für den IAM Identity Center-Authentifiz ierungsmodus. Wenn Sie die IAMIdentity Center-Au thentifizierung (früher AWS Single Sign On) für das Studio verwenden, können Sie die Anmeldung für Benutzer mit der Option Weitergabe vertrauenswürdiger Identität en aktivieren optimieren. Mit Trusted Identity Propagation können sich Benutzer mit ihren Identity Center-Anmeldeinfo rmationen anmelden und ihre Identitäten an nachgelagerte AWS Dienste weitergeben lassen, wenn sie das Studio verwenden. Im Abschnitt Application access (Anwendun gszugriff) können Sie auch angeben, ob alle Benutzer und Gruppen in Ihrem Identity Center Zugriff auf das Studio haben sollen oder ob nur zugewiesene Benutzer und Gruppen, die Sie auswählen, auf das Studio zugreifen können. Weitere Informationen finden Sie unter Integrieren Sie Amazon EMR mit AWS IAM Identity Center und auch Weitergab e vertrauenswürdiger Identitäten über Anwendungen hinweg im AWS IAMIdentity Center-Benutzerhandbuch.

8. Wählen Sie für VPC eine Amazon Virtual Private Cloud (VPC) für das Studio aus der Drop-down-Liste aus.
9. Wählen Sie unter Subnetze maximal fünf Subnetze aus, die Sie dem Studio VPC zuordnen möchten. Sie haben die Möglichkeit, weitere Subnetze hinzuzufügen, nachdem Sie das Studio erstellt haben.

10. Wählen Sie für Sicherheitsgruppen entweder die Standardsicherheitsgruppen oder benutzerdefinierte Sicherheitsgruppen aus. Weitere Informationen finden Sie unter Definieren Sie Sicherheitsgruppen zur Steuerung des EMR Studio-Netzwerkverkehrs.

Wenn Sie folgendes auswählen …	Vorgehensweise
Die Standard-Sicherheitsgruppen von EMR Studio	Um die Git-basierte Repository-Verknüp fung für das Studio zu aktivieren, wählen Sie Cluster/Endpunkte und Git-Repository aktivieren. Wählen Sie andernfalls Cluster/ Endpunkte aktivieren.
Benutzerdefinierte Sicherheitsgruppen für Ihr Studio	• Wählen Sie unter Cluster-/Endpunkts icherheitsgruppe die Engine-Sicherheits gruppe aus, die Sie aus der Dropdownl iste konfiguriert haben. Ihr Studio verwendet diese Sicherheitsgruppe, um eingehenden Zugriff von verbundenen Workspaces aus zu ermöglichen. • Wählen Sie unter Workspace-Sicherhe itsgruppe die Workspace-Sicherhe itsgruppe aus, die Sie aus der Dropdownl iste konfiguriert haben. Ihr Studio verwendet diese Sicherheitsgruppe mit Workspaces, um ausgehenden Zugriff auf verbundene EMR Amazon-Cluster und öffentlich gehostete Git-Repositorys zu ermöglichen.

11. Fügen Sie Ihrem Studio und anderen Ressourcen Tags hinzu. Weitere Informationen zu Tags finden Sie unter Tag-Cluster.
12. Wählen Sie Create Studio und starten Sie Workspace, um den Vorgang abzuschließen und zur Studios-Seite zu navigieren. Ihr neues Studio wird in der Liste mit Details wie Studio-Name, Erstellungsdatum und Studio-Zugriff angezeigtURL.

Nachdem Sie ein Studio erstellt haben, folgen Sie den Anweisungen unter Weisen Sie einem EMR Studio einen Benutzer oder eine Gruppe zu.

CLI

> **Note**
>
> Linux-Zeilenfortsetzungszeichen (\) sind aus Gründen der Lesbarkeit enthalten. Sie können entfernt oder in Linux-Befehlen verwendet werden. Entfernen Sie sie unter Windows oder ersetzen Sie sie durch ein Caret-Zeichen (^).

Example — Erstellen Sie ein EMR Studio, das IAM zur Authentifizierung verwendet wird

Der folgende AWS CLI Beispielbefehl erstellt ein EMR Studio mit IAM Authentifizierungsmodus. Wenn Sie die IAM Authentifizierung oder den Verbund für das Studio verwenden, geben Sie keinen an`--user-role`.

Damit sich Verbundbenutzer mit Studio URL und den Anmeldeinformationen für Ihren Identitätsanbieter (IdP) anmelden können, geben Sie Ihr `--idp-auth-url` und an. `--idp-relay-state-parameter-name` Eine Liste der IdP-Authentifizierung URLs und der RelayState Namen finden Sie unter RelayState Parameter und Authentifizierung des Identitätsanbieters URLs.

```
aws emr create-studio \
--name <example-studio-name> \
--auth-mode IAM \
--vpc-id <example-vpc-id> \
--subnet-ids <subnet-id-1> <subnet-id-2>... <subnet-id-5>  \
--service-role <example-studio-service-role-name> \
--user-role studio-user-role-name \
--workspace-security-group-id <example-workspace-sg-id> \
--engine-security-group-id <example-engine-sg-id> \
--default-s3-location <example-s3-location> \
--idp-auth-url <https://EXAMPLE/login/> \
--idp-relay-state-parameter-name <example-RelayState>
```

Example — Erstellen Sie ein EMR Studio, das Identity Center für die Authentifizierung verwendet

Mit dem folgenden AWS CLI Beispielbefehl wird ein EMR Studio erstellt, das den IAM Identity Center-Authentifizierungsmodus verwendet. Wenn Sie die IAM Identity Center-Authentifizierung verwenden, müssen Sie einen angeben`--user-role`.

Weitere Informationen zum IAM Identity Center-Authentifizierungsmodus finden Sie unterIAMIdentity Center-Authentifizierungsmodus für Amazon EMR Studio einrichten.

```
aws emr create-studio \
--name <example-studio-name> \
--auth-mode SSO \
--vpc-id <example-vpc-id> \
--subnet-ids <subnet-id-1> <subnet-id-2>... <subnet-id-5>  \
--service-role <example-studio-service-role-name> \
--user-role <example-studio-user-role-name> \
--workspace-security-group-id <example-workspace-sg-id> \
--engine-security-group-id <example-engine-sg-id> \
--default-s3-location <example-s3-location>
--trusted-identity-propagation-enabled \
--idc-user-assignment OPTIONAL \
--idc-instance-arn <iam-identity-center-instance-arn>
```

Example — CLI Ausgabe für **`aws emr create-studio`**

Es folgt ein Beispiel für die Ausgabe, die nach dem Erstellen eines Studios erscheint.

```
{
    StudioId: "es-123XXXXXXXXX",
    Url: "https://es-123XXXXXXXXX.emrstudio-prod.us-east-1.amazonaws.com"
}
```

Weitere Informationen über den Befehl `create-studio` finden Sie unter AWS CLI - Befehlsreferenz.

RelayState Parameter und Authentifizierung des Identitätsanbieters URLs

Wenn Sie den IAM Verbund verwenden und möchten, dass sich Benutzer mit Ihrem Studio URL und den Anmeldeinformationen für Ihren Identity Provider (IdP) anmelden, können Sie den Anmeldenamen URL und den RelayStateParameternamen Ihres Identity Providers (IdP) angeben, wenn Sie Erstellen Sie ein Studio EMR

Die folgende Tabelle zeigt die Standardauthentifizierung URL und den RelayState Parameternamen für einige beliebte Identitätsanbieter.

Identitätsanbieter	Parameter	Authentifizierung URL
Auth0	`RelayState`	`https://<sub_domain> .auth0.com/samlp/<app_id>`
Google-Konten	`RelayState`	`https://accounts.google.com/o/saml2/initsso?idpid= <idp_id>&spid=<sp_id>&forceauthn=false`
Microsoft Azure	`RelayState`	`https://myapps.microsoft.com/signin/ <app_name> /<app_id>?tenantId= <tenant_id>`
Okta	`RelayState`	`https://<sub_domain> .okta.com/app/<app_name> /<app_id>/sso/saml`
PingFederate	`TargetResource`	`https://<host>/idp/<idp_id>/startSSO.ping?PartnerSpId= <sp_id>`
PingOne	`TargetResource`	`https://sso.connect.pingidentity.com/sso/sp/initsso?saasid= <app_id>&idpid=<idp_id>`

EMRStudio-Benutzer zuweisen und verwalten

Nachdem Sie ein EMR Studio erstellt haben, können Sie ihm Benutzer und Gruppen zuweisen. Die Methode, mit der Sie Benutzer zuweisen, aktualisieren und entfernen, hängt vom Studio-Authentifizierungsmodus ab.

- Wenn Sie den IAM Authentifizierungsmodus verwenden, konfigurieren Sie die EMR Studio-Benutzerzuweisung und -berechtigungen in IAM oder mit IAM Ihrem Identitätsanbieter.
- Im IAM Identity Center-Authentifizierungsmodus verwenden Sie die Amazon EMR Management Console oder die AWS CLI , um Benutzer zu verwalten.

Weitere Informationen zur Authentifizierung für Amazon EMR Studio finden Sie unterWählen Sie einen Authentifizierungsmodus für Amazon EMR Studio.

Weisen Sie einem EMR Studio einen Benutzer oder eine Gruppe zu

IAM

Wenn Sie es verwendenIAMAuthentifizierungsmodus für Amazon EMR Studio einrichten, müssen Sie die `CreateStudioPresignedUrl` Aktion in der IAM Berechtigungsrichtlinie eines Benutzers zulassen und den Benutzer auf ein bestimmtes Studio beschränken. Sie können `CreateStudioPresignedUrl` in Ihre eigene Benutzerberechtigungen für den IAM-Authentifizierungsmodus aufnehmen oder eine separate Richtlinie verwenden.

Um einen Benutzer auf ein Studio (oder eine Gruppe von Studios) zu beschränken, können Sie die attributebasierte Zugriffskontrolle (ABAC) verwenden oder den Amazon-Ressourcennamen (ARN) eines Studios im `Resource` Element der Berechtigungsrichtlinie angeben.

Example Weisen Sie mithilfe eines Studios einen Benutzer einem Studio zu ARN

Die folgende Beispielrichtlinie gewährt einem Benutzer Zugriff auf ein bestimmtes EMR Studio, indem sie die `CreateStudioPresignedUrl` Aktion zulässt und den Amazon-Ressourcennamen (ARN) des Studios im `Resource` Element angibt.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowCreateStudioPresignedUrl",
            "Effect": "Allow",
            "Action": [
                "elasticmapreduce:CreateStudioPresignedUrl"
            ],
            "Resource": "arn:aws:elasticmapreduce:<region>:<account-
id>:studio/<studio-id>"
        }
    ]
}
```

Example Weisen Sie einen Benutzer ABAC zur IAM Authentifizierung einem Studio mit zu

Es gibt mehrere Möglichkeiten, die attributbasierte Zugriffskontrolle (ABAC) für ein Studio zu konfigurieren. Sie können beispielsweise einem EMR Studio ein oder mehrere Tags zuordnen

und dann eine IAM Richtlinie erstellen, die die `CreateStudioPresignedUrl` Aktion auf ein bestimmtes Studio oder eine Gruppe von Studios mit diesen Tags beschränkt.

Sie können Tags während oder nach der Erstellung von Studio hinzufügen. Verwenden Sie den Befehl AWS CLI`emr add-tags`, um Tags zu einem bestehenden Studio hinzuzufügen. Im folgenden Beispiel wird einem Studio ein Tag mit dem Schlüssel-Wert-Paar `Team = Data Analytics` hinzugefügt. EMR

```
aws emr add-tags --resource-id <example-studio-id> --tags Team="Data Analytics"
```

Das folgende Beispiel für eine Berechtigungsrichtlinie ermöglicht die `CreateStudioPresignedUrl` Aktion für EMR Studios mit dem Schlüssel-Wert-Paar des Tags. `Team = DataAnalytics` Weitere Informationen zur Verwendung von Tags zur Zugriffssteuerung finden Sie unter Zugriffskontrolle für Benutzer und Rollen mithilfe von Tags oder Zugriffskontrolle auf AWS -Ressourcen mithilfe von Tags.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowCreateStudioPresignedUrl",
            "Effect": "Allow",
            "Action": [
                "elasticmapreduce:CreateStudioPresignedUrl"
            ],
            "Resource": "arn:aws:elasticmapreduce:<region>:<account-id>:studio/*",
            "Condition": {
                "StringEquals": {
                    "elasticmapreduce:ResourceTag/Team": "Data Analytics"
                }
            }
        }
    ]
}
```

Example Weisen Sie einem Studio mithilfe des SourceIdentity globalen Bedingungsschlüssels aws: einen Benutzer zu

Wenn Sie den IAM Verbund verwenden, können Sie den globalen Bedingungsschlüssel `aws:SourceIdentity` in einer Berechtigungsrichtlinie verwenden, um Benutzern Zugriff auf Studio zu gewähren, wenn sie Ihre IAM Rolle für den Verbund übernehmen.

Sie müssen zunächst Ihren Identitätsanbieter (IdP) so konfigurieren, dass er eine identifizierende Zeichenfolge zurückgibt, z. B. eine E-Mail-Adresse oder einen Benutzernamen, wenn sich ein Benutzer authentifiziert und Ihre IAM Rolle für den Verbund übernimmt. IAMsetzt den globalen Bedingungsschlüssel `aws:SourceIdentity` auf die identifizierende Zeichenfolge, die von Ihrem IdP zurückgegeben wurde.

Weitere Informationen finden Sie im Blogbeitrag Wie man IAM Rollenaktivitäten mit Corporate Identity in Beziehung setzt im AWS Security Blog und im SourceIdentity Eintrag aws: in der Referenz zu globalen Bedingungsschlüsseln.

Die folgende Beispielrichtlinie ermöglicht die `CreateStudioPresignedUrl` Aktion und gibt Benutzern eine`aws:SourceIdentity`, die der *<example-source-identity>* Zugriff auf das EMR Studio, angegeben durch *<example-studio-arn>*.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "elasticmapreduce:CreateStudioPresignedUrl",
      "Resource": "<example-studio-arn>",
      "Condition": {
        "StringLike": {
          "aws:SourceIdentity": "<example-source-identity>"
        }
      }
    }
  ]
}
```

IAM Identity Center

Wenn Sie einem EMR Studio einen Benutzer oder eine Gruppe zuweisen, geben Sie eine Sitzungsrichtlinie an, die detaillierte Berechtigungen für diesen Benutzer oder diese Gruppe definiert, z. B. die Möglichkeit, einen neuen EMR Cluster zu erstellen. Amazon EMR speichert diese Zuordnungen von Sitzungsrichtlinien. Sie können die Sitzungsrichtlinie eines Benutzers oder einer Gruppe nach der Zuweisung aktualisieren.

> **Note**
>
> Die endgültigen Berechtigungen für einen Benutzer oder eine Gruppe stellen eine Schnittmenge zwischen den in Ihrer EMR Studio-Benutzerrolle definierten Berechtigungen und den in der Sitzungsrichtlinie für diesen Benutzer oder diese Gruppe definierten Berechtigungen dar. Wenn ein Benutzer zu mehr als einer Gruppe gehört, die dem Studio zugewiesen ist, verwendet EMR Studio eine Kombination von Berechtigungen für diesen Benutzer.

So weisen Sie einem EMR Studio mithilfe der EMR Amazon-Konsole Benutzer oder Gruppen zu

1. Navigieren Sie zur neuen EMR Amazon-Konsole und wählen Sie in der Seitennavigation die Option Zur alten Konsole wechseln aus. Weitere Informationen darüber, was Sie erwartet, wenn Sie zur alten Konsole wechseln, finden Sie unter Verwenden der alten Konsole.
2. Wählen Sie in der linken Navigationsleiste EMRStudio aus.
3. Wählen Sie Ihren Studio-Namen aus der Studio-Liste oder wählen Sie das Studio aus und klicken Sie auf Details anzeigen, um die Studio-Detailseite zu öffnen.
4. Wählen Sie die Benutzer hinzufügen um die Suchtabelle Benutzer und Gruppen anzuzeigen.
5. Wählen Sie die Registerkarte Benutzer oder die Registerkarte Gruppen und geben Sie einen Suchbegriff in die Suchleiste ein, um einen Benutzer oder eine Gruppe zu finden.
6. Wählen Sie einen oder mehrere Benutzer oder Gruppen aus der Suchergebnisliste aus. Sie können zwischen der Registerkarte Benutzer und der Registerkarte Gruppen hin- und herwechseln.
7. Nachdem Sie Benutzer und Gruppen ausgewählt haben, die Sie dem Studio hinzufügen möchten, wählen Sie Hinzufügen. Sie sollten sehen, dass die Benutzer und Gruppen in der Studio-Benutzerliste angezeigt werden. Es kann einige Sekunden dauern, bis die Liste aktualisiert wird.
8. Folgen Sie den Anweisungen unter Aktualisieren Sie die Berechtigungen für einen Benutzer oder eine Gruppe, die einem Studio zugewiesen ist, um die Studio-Berechtigungen für einen Benutzer oder eine Gruppe zu verfeinern.

Um einem EMR Studio einen Benutzer oder eine Gruppe zuzuweisen, verwenden Sie AWS CLI

Fügen Sie Ihre eigenen Werte für die folgenden `create-studio-session-mapping`-Argumente ein. Weitere Informationen über den Befehl `create-studio-session-mapping` finden Sie unter AWS CLI -Befehlsreferenz.

- **`--studio-id`** – Die ID des Studios, dem Sie den Benutzer oder die Gruppe zuweisen möchten. Anleitungen zum Abrufen einer Studio-ID finden Sie unter Studio-Details anzeigen.
- **`--identity-name`** – Der Name des Benutzers oder der Gruppe aus dem Identitätsspeicher. Weitere Informationen finden Sie unter UserNameFür Benutzer und DisplayNamefür Gruppen in der Identity API Store-Referenz.
- **`--identity-type`** – Verwenden Sie entweder `USER` oder `GROUP`, um den Identitätstyp anzugeben.
- **`--session-policy-arn`**— Der Amazon-Ressourcenname (ARN) für die Sitzungsrichtlinie, die Sie dem Benutzer oder der Gruppe zuordnen möchten. Beispiel, **`arn:aws:iam::`*`<aws-account-id>`*`:policy/`*`EMRStudio_Advanced_User_Policy`**. Weitere Informationen finden Sie unter Erstellen Sie Berechtigungsrichtlinien für EMR Studio-Benutzer.

```
aws emr create-studio-session-mapping \
 --studio-id <example-studio-id> \
 --identity-name <example-identity-name> \
 --identity-type <USER-or-GROUP> \
 --session-policy-arn <example-session-policy-arn>
```

> **Note**
>
> Linux-Zeilenfortsetzungszeichen (\) sind aus Gründen der Lesbarkeit enthalten. Sie können entfernt oder in Linux-Befehlen verwendet werden. Entfernen Sie sie unter Windows oder ersetzen Sie sie durch ein Caret-Zeichen (^).

Verwenden Sie den `get-studio-session-mapping`-Befehl, um die neue Zuweisung zu überprüfen. Ersetzen *`<example-identity-name>`* mit dem IAM Identity Center-Namen des Benutzers oder der Gruppe, den Sie aktualisiert haben.

```
aws emr get-studio-session-mapping \
 --studio-id <example-studio-id> \
 --identity-type <USER-or-GROUP> \
 --identity-name <user-or-group-name> \
```

Aktualisieren Sie die Berechtigungen für einen Benutzer oder eine Gruppe, die einem Studio zugewiesen ist

IAM

Um Benutzer- oder Gruppenberechtigungen zu aktualisieren, wenn Sie den IAM AuthentifizierungsmodusverwendenIAM, können Sie die IAM Berechtigungsrichtlinien ändern, die Ihren IAM Identitäten (Benutzern, Gruppen oder Rollen) zugeordnet sind.

Weitere Informationen finden Sie unter Benutzerberechtigungen für den IAM-Authentifizierungsmodus.

IAM Identity Center

So aktualisieren Sie die EMR Studio-Berechtigungen für einen Benutzer oder eine Gruppe mithilfe der Konsole

1. Navigieren Sie zur neuen EMR Amazon-Konsole und wählen Sie in der Seitennavigation die Option Zur alten Konsole wechseln aus. Weitere Informationen darüber, was Sie erwartet, wenn Sie zur alten Konsole wechseln, finden Sie unter Verwenden der alten Konsole.
2. Wählen Sie in der linken Navigationsleiste EMRStudio aus.
3. Wählen Sie Ihren Studio-Namen aus der Studio-Liste oder wählen Sie das Studio aus und klicken Sie auf Details anzeigen, um die Studio-Detailseite zu öffnen.
4. Suchen Sie in der Studio-Benutzerliste auf der Studio-Detailseite nach dem Benutzer oder der Gruppe, die Sie aktualisieren möchten. Sie können nach Namen oder Identitätstyp suchen.
5. Wählen Sie den Benutzer oder die Gruppe aus, den bzw. die Sie aktualisieren möchten, und wählen Sie Richtlinie zuweisen, um das Dialogfeld Sitzungsrichtlinie zu öffnen.
6. Wählen Sie eine Richtlinie aus, die auf den Benutzer oder die Gruppe angewendet werden soll, den Sie in Schritt 5 ausgewählt haben, und klicken Sie dann auf Richtlinie anwenden. In der Liste Studio-Benutzer sollte der Richtlinienname in der Spalte Sitzungsrichtlinie für den Benutzer oder die Gruppe angezeigt werden, den Sie aktualisiert haben.

Um die EMR Studio-Berechtigungen für einen Benutzer oder eine Gruppe mit dem zu aktualisieren AWS CLI

Fügen Sie Ihre eigenen Werte für die folgenden `update-studio-session-mappings`-Argumente ein. Weitere Informationen über den Befehl `update-studio-session-mappings` finden Sie unter AWS CLI -Befehlsreferenz.

```
aws emr update-studio-session-mapping \
 --studio-id <example-studio-id> \
 --identity-name <name-of-user-or-group-to-update> \
 --session-policy-arn <new-session-policy-arn-to-apply> \
 --identity-type <USER-or-GROUP> \
```

Verwenden Sie den `get-studio-session-mapping`-Befehl, um die neue Zuweisung der Sitzungsrichtlinie zu überprüfen. Ersetzen *<example-identity-name>* mit dem IAM Identity Center-Namen des Benutzers oder der Gruppe, den Sie aktualisiert haben.

```
aws emr get-studio-session-mapping \
 --studio-id <example-studio-id> \
 --identity-type <USER-or-GROUP> \
 --identity-name <user-or-group-name> \
```

Einen Benutzer oder eine Gruppe aus einem Studio entfernen

IAM

Um einen Benutzer oder eine Gruppe aus einem EMR Studio zu entfernen, wenn Sie den IAM Authentifizierungsmodus verwenden, müssen Sie dem Benutzer den Zugriff auf das Studio entziehen, indem Sie die IAM Berechtigungsrichtlinie des Benutzers neu konfigurieren.

Gehen Sie in der folgenden Beispielrichtlinie davon aus, dass Sie über ein EMR Studio mit dem Schlüssel-Wert-Paar des Tags verfügen. `Team = Quality Assurance` Gemäß der Richtlinie kann der Benutzer auf Studios zugreifen, die mit dem `Team`-Schlüssel gekennzeichnet sind, dessen Wert entweder `Data Analytics` oder `Quality Assurance` entspricht. Um den Benutzer aus dem Studio zu entfernen, das mit `Team = Quality Assurance` markiert ist, entfernen Sie `Quality Assurance` aus der Liste der Tag-Werte.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowCreateStudioPresignedUrl",
            "Effect": "Allow",
```

```
            "Action": [
                "elasticmapreduce:CreateStudioPresignedUrl"
            ],
            "Resource": "arn:aws:elasticmapreduce:<region>:<account-id>:studio/*",
            "Condition": {
                "StringEquals": {
                    "emr:ResourceTag/Team": [
                        "Data Analytics",
                        "Quality Assurance"
                    ]
                }
            }
        }
    ]
}
```

IAM Identity Center

Um einen Benutzer oder eine Gruppe mithilfe der Konsole aus einem EMR Studio zu entfernen

1. Navigieren Sie zur neuen EMR Amazon-Konsole und wählen Sie in der Seitennavigation die Option Zur alten Konsole wechseln aus. Weitere Informationen darüber, was Sie erwartet, wenn Sie zur alten Konsole wechseln, finden Sie unter Verwenden der alten Konsole.
2. Wählen Sie in der linken Navigationsleiste EMRStudio aus.
3. Wählen Sie Ihren Studio-Namen aus der Studio-Liste oder wählen Sie das Studio aus und klicken Sie auf Details anzeigen, um die Studio-Detailseite zu öffnen.
4. Suchen Sie in der Liste Studio-Benutzer auf der Studio-Detailseite nach dem Benutzer oder der Gruppe, die Sie aus dem Studio entfernen möchten. Sie können nach Namen oder Identitätstyp suchen.
5. Wählen Sie den Benutzer oder die Gruppe aus, die Sie löschen möchten, wählen Sie Löschen und bestätigen Sie das Löschen. Der Benutzer oder die Gruppe, den Sie gelöscht haben, verschwindet aus der Liste Studio-Benutzer.

Um einen Benutzer oder eine Gruppe aus einem EMR Studio zu entfernen, verwenden Sie AWS CLI

Fügen Sie Ihre eigenen Werte für die folgenden `delete-studio-session-mapping`-Argumente ein. Weitere Informationen über den Befehl `delete-studio-session-mapping` finden Sie unter AWS CLI -Befehlsreferenz.

```
aws emr delete-studio-session-mapping \
 --studio-id <example-studio-id> \
 --identity-type <USER-or-GROUP> \
 --identity-name <name-of-user-or-group-to-delete> \
```

Ein Amazon EMR Studio verwalten

Dieser Abschnitt enthält Anweisungen, die Ihnen helfen, eine EMR Studio-Ressource zu überwachen, zu aktualisieren oder zu löschen. Informationen zum Zuweisen von Benutzern oder zum Aktualisieren von Benutzerberechtigungen finden Sie unter EMRStudio-Benutzer zuweisen und verwalten.

Studio-Details anzeigen

Console

Um Details zu einem EMR Studio mit der neuen Konsole anzuzeigen

1. Öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie in der linken Navigationsleiste unter EMRStudio die Option Studios aus.
3. Wählen Sie das Studio aus der Studio-Liste aus, um die Studio-Detailseite zu öffnen. Die Studio-Detailseite enthält Informationen zu den Studio-Einstellungen, z. B. die Studio-Beschreibung und Subnetze. VPC

CLI

Um Details für ein EMR Studio anhand der Studio-ID abzurufen, verwenden Sie AWS CLI

Verwenden Sie den folgenden `describe-studio` AWS CLI Befehl, um detaillierte Informationen zu einem bestimmten EMR Studio abzurufen. Weitere Informationen finden Sie in der AWS CLI - Befehlsreferenz.

```
aws emr describe-studio \
 --studio-id <id-of-studio-to-describe> \
```

Um eine Liste von EMR Studios abzurufen, verwenden Sie AWS CLI

Verwenden Sie den folgenden `list-studios` AWS CLI -Befehl. Weitere Informationen finden Sie in der AWS CLI -Befehlsreferenz.

```
aws emr list-studios
```

Im Folgenden finden Sie ein Beispiel für einen Rückgabewert für den `list-studios` Befehl im JSON Format.

```
{
    "Studios": [
        {
            "AuthMode": "IAM",
            "VpcId": "vpc-b21XXXXX",
            "Name": "example-studio-name",
            "Url": "https://es-7HWP74SNGDXXXXXXXXXXXXXX.emrstudio-prod.us-
east-1.amazonaws.com",
            "CreationTime": 1605672582.781,
            "StudioId": "es-7HWP74SNGDXXXXXXXXXXXXXX",
            "Description": "example studio description"
        }
    ]
}
```

Amazon EMR Studio-Aktionen überwachen

EMRStudio und API Aktivitäten anzeigen

EMRStudio ist in einen Dienst integriert AWS CloudTrail, der eine Aufzeichnung der Aktionen bereitstellt, die von einem Benutzer, einer IAM Rolle oder einem anderen AWS Dienst in EMR Studio ausgeführt wurden. CloudTrail erfasst API Aufrufe für EMR Studio als Ereignisse. Sie können Ereignisse mithilfe der CloudTrail Konsole unter anzeigen https://console.aws.amazon.com/cloudtrail/.

EMRStudio-Ereignisse enthalten Informationen darüber, welches Studio oder welcher IAM Benutzer eine Anfrage stellt und um welche Art von Anfrage es sich handelt.

> **Note**
>
> Clusterinterne Aktionen wie das Ausführen von Notebook-Aufträgen werden AWS CloudTrail nicht ausgegeben.

Sie können auch einen Trail für die kontinuierliche Bereitstellung von EMR CloudTrail Studio-Ereignissen an einen Amazon S3 S3-Bucket erstellen. Weitere Informationen finden Sie im AWS CloudTrail -Benutzerhandbuch.

CloudTrail Beispielereignis: Ein Benutzer ruft den DescribeStudio API

Das Folgende ist ein AWS CloudTrail Beispielereignis, das erzeugt wird, wenn ein Benutzer,`admin`, den aufruft DescribeStudioAPI. CloudTrail zeichnet den Benutzernamen auf als`admin`.

> **Note**
>
> Um die Studio-Details zu schützen, DescribeStudio schließt das EMR API Studio-Ereignis für einen Wert für `responseElements` aus.


```
{
   "eventVersion":"1.08",
   "userIdentity":{
      "type":"IAMUser",
      "principalId":"AIDXXXXXXXXXXXXXXXXX",
      "arn":"arn:aws:iam::653XXXXXXXXX:user/admin",
      "accountId":"653XXXXXXXXX",
      "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
      "userName":"admin"
   },
   "eventTime":"2021-01-07T19:13:58Z",
   "eventSource":"elasticmapreduce.amazonaws.com",
   "eventName":"DescribeStudio",
   "awsRegion":"us-east-1",
   "sourceIPAddress":"72.XX.XXX.XX",
   "userAgent":"aws-cli/1.18.188 Python/3.8.5 Darwin/18.7.0 botocore/1.19.28",
   "requestParameters":{
      "studioId":"es-9O5XXXXXXXXXXXXXXXXXXXX"
   },
   "responseElements":null,
   "requestID":"0fxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
   "eventID":"b0xxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
   "readOnly":true,
   "eventType":"AwsApiCall",
   "managementEvent":true,
   "eventCategory":"Management",
   "recipientAccountId":"653XXXXXXXXX"
```

```
}
```

Spark-Benutzer- und Jobaktivitäten anzeigen

Um die Spark-Jobaktivitäten von Amazon EMR Studio-Benutzern anzuzeigen, können Sie den Benutzerwechsel in einem Cluster konfigurieren. Beim Identitätswechsel wird jeder Spark-Job, der von einem Workspace aus eingereicht wird, dem Studio-Benutzer zugeordnet, der den Code ausgeführt hat.

Wenn der Benutzerwechsel aktiviert ist, EMR erstellt Amazon ein HDFS Benutzerverzeichnis auf dem primären Knoten des Clusters für jeden Benutzer, der Code im Workspace ausführt. Wenn beispielsweise ein Benutzer `studio-user-1@example.com`-Code ausführt, können Sie eine Verbindung zum Primärknoten herstellen und sehen, dass `hadoop fs -ls /user` ein Verzeichnis für `studio-user-1@example.com` hat.

Um den Spark-Benutzerwechsel einzurichten, legen Sie die folgenden Eigenschaften in den folgenden Konfigurationsklassifizierungen fest:

- `core-site`
- `livy-conf`

```
[
    {
        "Classification": "core-site",
        "Properties": {
          "hadoop.proxyuser.livy.groups": "*",
          "hadoop.proxyuser.livy.hosts": "*"
        }
    },
    {
        "Classification": "livy-conf",
        "Properties": {
          "livy.impersonation.enabled": "true"
        }
    }
]
```

Informationen zum Anzeigen von Verlaufsserverseiten finden Sie unter Debuggen Sie Anwendungen und Jobs mit Studio EMR. Sie können auch eine Verbindung zum primären Knoten des Clusters

herstellen, SSH um die Webschnittstellen der Anwendung anzuzeigen. Weitere Informationen finden Sie unter Auf EMR Amazon-Clustern gehostete Weboberflächen anzeigen.

Ein Amazon EMR Studio aktualisieren

Nachdem Sie ein EMR Studio erstellt haben, können Sie die folgenden Attribute mit dem aktualisieren AWS CLI:

- Name
- Beschreibung
- Standard-S3-Speicherort
- Subnetze

Um ein EMR Studio zu aktualisieren, verwenden Sie AWS CLI

Verwenden Sie den `update-studio` AWS CLI Befehl, um ein EMR Studio zu aktualisieren. Weitere Informationen finden Sie in der AWS CLI -Befehlsreferenz.

> **Note**
>
> Sie können ein Studio mit maximal 5 Subnetzen verknüpfen. Diese Subnetze müssen zu demselben gehören VPC wie das Studio. Die Liste der SubnetzeIDs, die Sie an den `update-studio` Befehl senden, kann ein neues Subnetz enthaltenIDs, muss aber auch das gesamte Subnetz enthalten, IDs das Sie dem Studio bereits zugeordnet haben. Sie können keine Subnetze aus einem Studio entfernen.

```
aws emr update-studio \
 --studio-id <example-studio-id-to-update> \
 --name <example-new-studio-name> \
 --subnet-ids <old-subnet-id-1 old-subnet-id-2 old-subnet-id-3 new-subnet-id> \
```

Verwenden Sie den `describe-studio` AWS CLI Befehl und geben Sie Ihre Studio-ID an, um die Änderungen zu überprüfen. Weitere Informationen finden Sie in der AWS CLI -Befehlsreferenz.

```
aws emr describe-studio \
 --studio-id <id-of-updated-studio> \
```

Löschen Sie ein Amazon EMR Studio und Workspaces

Wenn Sie ein Studio löschen, löscht EMR Studio alle IAM Identity Center-Benutzer- und Gruppenzuweisungen, die dem Studio zugeordnet sind.

> **Note**
>
> Wenn Sie ein Studio löschen, EMR löscht Amazon die mit diesem Studio verknüpften Workspaces nicht. Sie müssen die Workspaces in Ihrem Studio separat löschen.

WorkSpaces löschen

Console

Da es sich bei jedem EMR Studio Workspace um eine EMR Notebook-Instance handelt, können Sie die EMR Amazon-Managementkonsole verwenden, um Workspaces zu löschen. Sie können Workspaces mit der EMR Amazon-Konsole löschen, bevor oder nachdem Sie Ihr Studio gelöscht haben.

Um einen Workspace mit der EMR Amazon-Konsole zu löschen

1. Navigieren Sie zur neuen EMR Amazon-Konsole und wählen Sie in der Seitennavigation die Option Zur alten Konsole wechseln aus. Weitere Informationen darüber, was Sie erwartet, wenn Sie zur alten Konsole wechseln, finden Sie unter Verwenden der alten Konsole.
2. Wählen Sie Notebooks.
3. Wählen Sie die Arbeitsbereiche aus, die Sie löschen möchten.
4. Wählen Sie Löschen und nochmals Löschen aus, um den Vorgang zu bestätigen.
5. Folgen Sie den Anweisungen zum Löschen von Objekten im Konsolen-Benutzerhandbuch für Amazon Simple Storage Service, um die mit dem gelöschten Workspace verknüpften Notebookdateien aus Amazon S3 zu entfernen.

EMR Studio UI

From the Workspace UI

Löschen Sie einen Workspace und die zugehörigen Backup-Dateien aus EMR Studio

1. Melden Sie sich mit Ihrem EMR Studio-Zugriff bei Ihrem Studio an URL und wählen Sie in der linken Navigationsleiste Workspaces aus.
2. Suchen Sie in der Liste nach Ihrem Workspace und aktivieren Sie das Kontrollkästchen neben dessen Namen. Sie können mehrere Arbeitsbereiche zum gleichzeitigen Löschen auswählen.
3. Wählen Sie oben rechts in der Liste der Arbeitsbereiche die Option Löschen aus und bestätigen Sie, dass Sie die ausgewählten Arbeitsbereiche löschen möchten. Wählen Sie zur Bestätigung Delete.
4. Wenn Sie die Notebookdateien, die mit dem gelöschten Workspace verknüpft waren, aus Amazon S3 entfernen möchten, folgen Sie den Anweisungen zum Löschen von Objekten im Konsole-Benutzerhandbuch von Amazon Simple Storage Service. Wenn Sie das Studio nicht erstellt haben, wenden Sie sich an Ihren Studio-Administrator, um den Amazon-S3-Backup-Speicherort für den gelöschten Workspace zu ermitteln.

From the Workspaces list

Löschen Sie einen Workspace und die zugehörigen Backup-Dateien aus der Workspaces-Liste

1. Navigieren Sie in der Konsole zur Workspace-Liste.
2. Wählen Sie den Workspace, den Sie löschen möchten, aus der Liste aus und klicken Sie dann auf Aktionen.
3. Wählen Sie Löschen.
4. Wenn Sie die Notebookdateien, die mit dem gelöschten Workspace verknüpft waren, aus Amazon S3 entfernen möchten, folgen Sie den Anweisungen zum Löschen von Objekten im Konsole-Benutzerhandbuch von Amazon Simple Storage Service. Wenn Sie das Studio nicht erstellt haben, wenden Sie sich an Ihren Studio-Administrator, um den Amazon-S3-Backup-Speicherort für den gelöschten Workspace zu ermitteln.

Lösche ein Studio EMR

Console

Um ein EMR Studio mit der neuen Konsole zu löschen

1. Öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie in der linken Navigationsleiste unter EMRStudio die Option Studios aus.
3. Wählen Sie das Studio aus der Studio-Liste mit dem Schalter links neben dem Studio-Namen aus. Wählen Sie Löschen.

Old console

Um ein EMR Studio mit der alten Konsole zu löschen

1. Öffnen Sie die EMR Amazon-Konsole zu https://console.aws.amazon.com/elasticmapreduce/Hause.
2. Wählen Sie in der linken Navigationsleiste EMRStudio aus.
3. Wählen Sie das Studio aus der Studio-Liste aus und klicken Sie auf Löschen.

CLI

Um ein EMR Studio mit dem zu löschen AWS CLI

Verwenden Sie den `delete-studio` AWS CLI Befehl, um ein EMR Studio zu löschen. Weitere Informationen finden Sie in der AWS CLI -Befehlsreferenz.

```
aws emr delete-studio --studio-id <id-of-studio-to-delete>
```

Verschlüsselung von EMR Studio-Workspace-Notizbüchern und -Dateien

In EMR Studio können Sie verschiedene Arbeitsbereiche erstellen und konfigurieren, um Notizbücher zu organisieren und auszuführen. In diesen Arbeitsbereichen werden Notizbücher und zugehörige Dateien in Ihrem angegebenen Amazon S3 S3-Bucket gespeichert. Standardmäßig werden diese Dateien mit von Amazon SSE S3 verwalteten Schlüsseln (-S3) mit serverseitiger Verschlüsselung als Basisverschlüsselungsebene verschlüsselt. Sie können sich auch dafür entscheiden, vom Kunden verwaltete KMS Schlüssel (SSE-KMS) zum Verschlüsseln Ihrer Dateien zu verwenden. Sie können dies über die EMR Amazon-Managementkonsole oder über AWS CLI und AWS SDK beim Erstellen eines EMR Studios tun.

EMRDie Studio-Workspace-Speicherverschlüsselung ist in allen Regionen verfügbar, in denen EMR Studio verfügbar ist.

Voraussetzungen

Bevor Sie das EMR Studio-Workspace-Notizbuch und die Dateien verschlüsseln können, müssen Sie einen symmetrischen Kundenmanager-Schlüssel (CMK) in derselben AWS-Konto Region wie Ihr EMR Studio erstellen. AWS Key Management Service

Ihre Ressourcenrichtlinie AWS KMS muss über die erforderlichen Zugriffsberechtigungen für die Servicerolle Ihres EMR Studios verfügen. Im Folgenden finden Sie eine IAM Beispielrichtlinie, die Mindestzugriffsberechtigungen für die EMR Studio Workspace-Speicherverschlüsselung gewährt:

```
{
    "Sid": "AllowEMRStudioServiceRoleAccess",
    "Effect": "Allow",
    "Principal": {
        "AWS": "arn:aws:iam::<ACCOUNT_ID>:role/<ROLE_NAME>"
    },
    "Action": [
        "kms:Decrypt",
        "kms:GenerateDataKey",
        "kms:ReEncryptFrom",
        "kms:ReEncryptTo",
        "kms:DescribeKey"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "kms:CallerAccount": "<ACCOUNT_ID>",
            "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::<S3_BUCKET_NAME>",
            "kms:ViaService": "s3.<AWS_REGION>.amazonaws.com"
        }
    }
}
```

Ihre EMR Studio-Dienstrolle muss auch über die Zugriffsberechtigungen verfügen, um Ihren AWS KMS Schlüssel verwenden zu können. Im Folgenden finden Sie eine IAM Beispielrichtlinie, die die Mindestzugriffsberechtigungen für die EMR Studio Workspace-Speicherverschlüsselung gewährt:

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowEMRStudioWorkspaceStorageEncryptionAccess",
            "Effect": "Allow",
            "Action": [
               "kms:Decrypt",
               "kms:GenerateDataKey",
               "kms:ReEncryptFrom",
               "kms:ReEncryptTo",
               "kms:DescribeKey"
            ],
            "Resource": ["arn:aws:kms:<REGION>:<ACCOUNT_ID>:key/<KEY_IDENTIFIER>"]
        }
    ]
}
```

Aufstellen

Gehen Sie wie folgt vor, um ein neues EMR Studio zu erstellen, das die Workspace-Speicherverschlüsselung verwendet.

1. Öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/elasticmapreduce/.
2. Wählen Sie Studios und dann Create Studio.
3. Geben Sie für S3-Speicherort einen Amazon S3 S3-Pfad ein oder wählen Sie ihn aus. Dies ist der Amazon S3 S3-Standort, an dem Amazon Workspace-Notizbücher und Dateien EMR speichert.
4. Geben Sie unter Servicerolle eine Rolle ein, oder wählen Sie sie aus. IAM Dies ist die IAM Rolle, die Amazon EMR einnimmt.
5. Wählen Sie Workspace-Dateien mit Ihrem eigenen AWS KMS Schlüssel verschlüsseln.
6. Geben Sie einen AWS KMS Schlüssel ein, der zum Verschlüsseln von Workspace-Notizbüchern und -Dateien in Amazon S3 verwendet werden soll, oder wählen Sie einen aus.
7. Wählen Sie Create Studio oder Create Studio and Launch Workspaces.
8. Wählen Sie Workspace-Dateien mit Ihrem eigenen AWS KMS Schlüssel verschlüsseln.
9. Geben Sie eine ein AWS KMS , die zum Verschlüsseln von Workspace-Notizbüchern und -Dateien in Amazon S3 verwendet werden soll, oder wählen Sie eine aus.
10. Wählen Sie Save Changes.

Die folgenden Schritte zeigen, wie Sie ein EMR Studio aktualisieren und die Workspace-Speicherverschlüsselung einrichten.

1. Öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/elasticmapreduce/.
2. Wählen Sie ein vorhandenes EMR Studio aus der Liste aus und klicken Sie dann auf Bearbeiten.
3. Wählen Sie Workspace-Dateien mit Ihrem eigenen AWS KMS Schlüssel verschlüsseln.
4. Geben Sie eine ein AWS KMS , die zum Verschlüsseln von Workspace-Notizbüchern und -Dateien in Amazon S3 verwendet werden soll, oder wählen Sie eine aus.
5. Wählen Sie Save Changes.

Definieren Sie Sicherheitsgruppen zur Steuerung des EMR Studio-Netzwerkverkehrs

Über die EMR Studio-Sicherheitsgruppen

Amazon EMR Studio verwendet zwei Sicherheitsgruppen, um den Netzwerkverkehr zwischen Workspaces im Studio und einem angeschlossenen EMR Amazon-Cluster, der auf Amazon EC2 läuft, zu kontrollieren:

- Eine Engine-Sicherheitsgruppe, die Port 18888 verwendet, um mit einem verbundenen EMR Amazon-Cluster zu kommunizieren, der auf Amazon EC2 läuft.
- Eine Workspace-Sicherheitsgruppe, die den Workspaces in einem Studio zugeordnet ist. Diese Sicherheitsgruppe enthält eine ausgehende HTTPS Regel, die es dem Workspace ermöglicht, Datenverkehr ins Internet weiterzuleiten, und muss ausgehenden Datenverkehr ins Internet an Port 443 zulassen, um die Verknüpfung von Git-Repositorys mit einem Workspace zu ermöglichen.

EMRStudio verwendet diese Sicherheitsgruppen zusätzlich zu allen Sicherheitsgruppen, die einem EMR Cluster zugeordnet sind, der an einen Workspace angeschlossen ist.

Sie müssen diese Sicherheitsgruppen erstellen, wenn Sie das verwenden AWS CLI , um ein Studio zu erstellen.

> **Note**
>
> Sie können die Sicherheitsgruppen für EMR Studio mit Regeln anpassen, die auf Ihre Umgebung zugeschnitten sind. Sie müssen jedoch die auf dieser Seite aufgeführten Regeln

einbeziehen. Ihre Workspace-Sicherheitsgruppe kann keinen eingehenden Datenverkehr zulassen, und die Engine-Sicherheitsgruppe muss eingehenden Datenverkehr von der Workspace-Sicherheitsgruppe zulassen.

Verwenden Sie die Standard-Sicherheitsgruppen von EMR Studio

Wenn Sie die EMR Amazon-Konsole verwenden, können Sie die folgenden Standardsicherheitsgruppen auswählen. Die Standardsicherheitsgruppen werden von EMR Studio in Ihrem Namen erstellt und enthalten die mindestens erforderlichen Regeln für eingehende und ausgehende Nachrichten für Workspaces in einem Studio. EMR

- `DefaultEngineSecurityGroup`
- `DefaultWorkspaceSecurityGroupGit` oder `DefaultWorkspaceSecurityGroupWithoutGit`

Voraussetzungen

Um die Sicherheitsgruppen für EMR Studio zu erstellen, benötigen Sie eine Amazon Virtual Private Cloud (VPC) für das Studio. Sie wählen diesVPC, wenn Sie die Sicherheitsgruppen erstellen. Dies sollte derselbe seinVPC, den Sie bei der Erstellung des Studios angegeben haben. Wenn Sie Amazon Amazon EMR on EKS mit EMR Studio verwenden möchten, wählen Sie den VPC für Ihre EKS Amazon-Cluster-Worker-Knoten aus.

Anweisungen

Folgen Sie den Anweisungen unter Erstellen einer Sicherheitsgruppe im EC2Amazon-Benutzerhandbuch für Linux-Instances, um eine Engine-Sicherheitsgruppe und eine Workspace-Sicherheitsgruppe in Ihrem zu erstellenVPC. Die Sicherheitsgruppen müssen die in den folgenden Tabellen zusammengefassten Regeln enthalten.

Wenn Sie Sicherheitsgruppen für EMR Studio erstellen, beachten Sie die IDs für beide. Sie geben jede Sicherheitsgruppe anhand ihrer ID an, wenn Sie ein Studio erstellen.

Engine-Sicherheitsgruppe

EMRStudio verwendet Port 18888 für die Kommunikation mit einem angeschlossenen Cluster.

Regeln für eingehenden Datenverkehr

Typ	Protokoll	Port	Bestimmungsort	Beschreibung
TCP	TCP	18888	Ihre EMR Studio Workspace-Sicherheitsgruppe.	Lassen Sie Datenverkehr von allen Ressourcen in der Workspace-Sicherheitsgruppe für EMR Studio zu.

WorkSpaces-Sicherheitsgruppe

Diese Sicherheitsgruppe ist den Workspaces in einem EMR Studio zugeordnet.

Regeln für ausgehenden Datenverkehr

Typ	Protokoll	Port	Bestimmungsort	Beschreibung
TCP	TCP	18888	Ihre EMR Studio-Engine-Sicherheitsgruppe.	Lassen Sie den Datenverkehr zu allen Ressourcen in der Engine-Sicherheitsgruppe für EMR Studio zu.
HTTPS	TCP	443	0.0.0.0/0	Erlaube Datenverkehr im Internet, um öffentlich gehostete Git-Repositorys mit Workspaces zu verknüpfen.

AWS CloudFormation Vorlagen für Amazon EMR Studio erstellen

Über EMR Studio-Cluster-Vorlagen

Sie können AWS CloudFormation Vorlagen erstellen, um EMR Studio-Benutzern zu helfen, neue EMR Amazon-Cluster in einem Workspace zu starten. CloudFormation Vorlagen sind formatierte Textdateien in JSON oderYAML. In einer Vorlage beschreiben Sie einen Stapel von AWS Ressourcen und erklären, CloudFormation wie Sie diese Ressourcen für Sie bereitstellen können. Für EMR Studio können Sie eine oder mehrere Vorlagen erstellen, die einen EMR Amazon-Cluster beschreiben.

Sie organisieren Ihre Vorlagen in AWS Service Catalog. AWS Service Catalog ermöglicht es Ihnen, häufig bereitgestellte IT-Services, sogenannte Produkte, zu erstellen und zu verwalten AWS. Sie sammeln Ihre Vorlagen als Produkte in einem Portfolio, das Sie mit Ihren EMR Studio-Benutzern teilen. Nachdem Sie Cluster-Vorlagen erstellt haben, können Studio-Benutzer mit einer Ihrer Vorlagen einen neuen Cluster für einen Workspace starten. Benutzer müssen über die Berechtigung zum Erstellen neuer Cluster aus Vorlagen verfügen. Sie können Benutzerberechtigungen in Ihren EMRStudio-Berechtigungsrichtlinien festlegen.

Weitere Informationen zu CloudFormation Vorlagen finden Sie im AWS CloudFormation Benutzerhandbuch unter Vorlagen. Weitere Informationen zu AWS Service Catalog finden Sie unter Was ist AWS Service Catalog.

Das folgende Video zeigt, wie Sie Cluster-Vorlagen in AWS Service Catalog EMR Studio einrichten. Weitere Informationen finden Sie auch im Blogbeitrag Aufbau einer Self-Service-Umgebung für jeden Geschäftsbereich mithilfe von Amazon EMR und Service Catalog.

Optionale Vorlageparameter

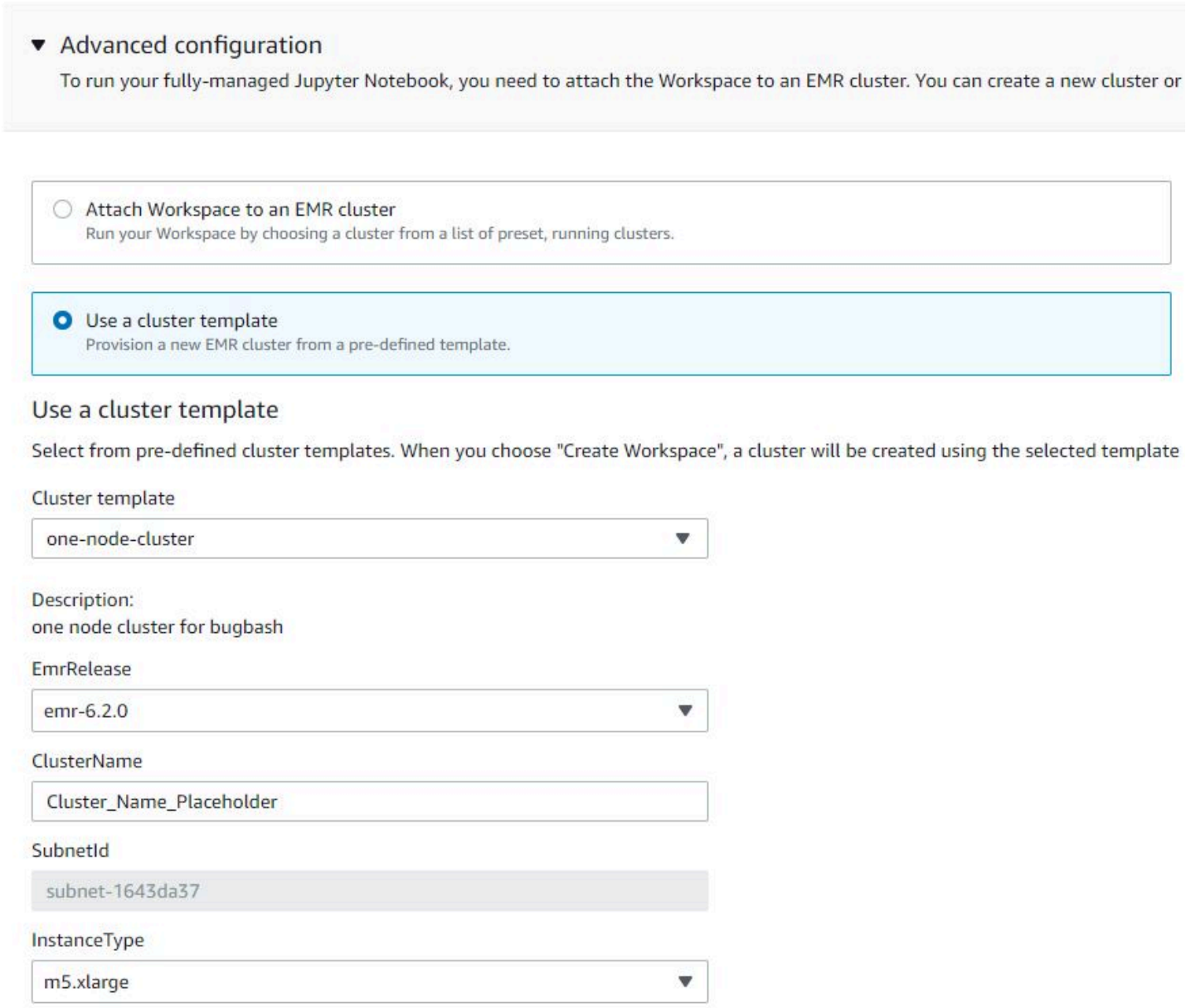
Sie können zusätzliche Optionen in den `Parameters` Abschnitt Ihrer Vorlage aufnehmen. Mit Parametern können Studio-Benutzer benutzerdefinierte Werte für einen Cluster eingeben oder auswählen. Sie könnten beispielsweise einen Parameter hinzufügen, mit dem Benutzer eine bestimmte EMR Amazon-Version auswählen können. Weitere Informationen finden Sie unter Parameter im AWS CloudFormation -Benutzerhandbuch.

Der folgende `Parameters`-Beispielabschnitt definiert zusätzliche Eingabeparameter wie `ClusterName`, `EmrRelease`-Version und `ClusterInstanceType`.

```
Parameters:
```

```
  ClusterName:
    Type: "String"
    Default: "Cluster_Name_Placeholder"
  EmrRelease:
    Type: "String"
    Default: "emr-6.2.0"
    AllowedValues:
    - "emr-6.2.0"
    - "emr-5.32.0"
  ClusterInstanceType:
    Type: "String"
    Default: "m5.xlarge"
    AllowedValues:
    - "m5.xlarge"
    - "m5.2xlarge"
```

Wenn Sie Parameter hinzufügen, werden Studio-Benutzern nach der Auswahl einer Clustervorlage zusätzliche Formularoptionen angezeigt. Die folgende Abbildung zeigt zusätzliche Formularoptionen für EmrReleaseVersion ClusterName, und InstanceType.



Voraussetzungen

Bevor Sie eine Clustervorlage erstellen, stellen Sie sicher, dass Sie über die IAM Berechtigungen für den Zugriff auf die Ansicht der Administratorkonsole von Service Catalog verfügen. Sie benötigen außerdem die erforderlichen IAM Berechtigungen, um die administrativen Aufgaben von Service Catalog auszuführen. Weitere Informationen finden Sie unter Service-Catalog-Administratoren Berechtigungen erteilen.

Anweisungen

So erstellen Sie EMR Clustervorlagen mit Service Catalog

1. Erstellen Sie eine oder mehrere CloudFormation Vorlagen. Wo Sie Ihre Vorlagen speichern, liegt bei Ihnen. Da es sich bei Vorlagen um formatierte Textdateien handelt, können Sie sie auf Amazon S3 hochladen oder in Ihrem lokalen Dateisystem speichern. Weitere Informationen zu CloudFormation Vorlagen finden Sie unter Vorlagen im AWS CloudFormation Benutzerhandbuch.

   Verwenden Sie die folgenden Regeln, um Ihre Vorlagen zu benennen, oder vergleichen Sie Ihre Namen mit dem Muster `[a-zA-Z0-9][a-zA-Z0-9._-]*`.

   - Vorlagennamen müssen mit einer Ziffer oder einem Buchstaben beginnen.
   - Vorlagennamen dürfen nur aus Buchstaben, Ziffern, Punkten (.), Unterstrichen (_) und Bindestrichen (-) bestehen.

   Jede Cluster-Vorlage, die Sie erstellen, muss die folgenden Optionen enthalten:

   Eingabeparameter

   - ClusterName — Ein Name für den Cluster, der Benutzern hilft, ihn nach der Bereitstellung zu identifizieren.

   Ausgabe

   - `ClusterId`— Die ID des neu bereitgestellten ClustersEMR.

   Im Folgenden finden Sie eine AWS CloudFormation Beispielvorlage im YAML Format für einen Cluster mit zwei Knoten. Die Beispielvorlage enthält die erforderlichen Vorlagenoptionen und definiert zusätzliche Eingabeparameter für `EmrRelease` und `ClusterInstanceType`.

```
awsTemplateFormatVersion: 2010-09-09

Parameters:
  ClusterName:
    Type: "String"
    Default: "Example_Two_Node_Cluster"
  EmrRelease:
```

```
    Type: "String"
    Default: "emr-6.2.0"
    AllowedValues:
    - "emr-6.2.0"
    - "emr-5.32.0"
  ClusterInstanceType:
    Type: "String"
    Default: "m5.xlarge"
    AllowedValues:
    - "m5.xlarge"
    - "m5.2xlarge"

Resources:
  EmrCluster:
    Type: AWS::EMR::Cluster
    Properties:
      Applications:
      - Name: Spark
      - Name: Livy
      - Name: JupyterEnterpriseGateway
      - Name: Hive
      EbsRootVolumeSize: '10'
      Name: !Ref ClusterName
      JobFlowRole: EMR_EC2_DefaultRole
      ServiceRole: EMR_DefaultRole_V2
      ReleaseLabel: !Ref EmrRelease
      VisibleToAllUsers: true
      LogUri:
        Fn::Sub: 's3://aws-logs-${AWS::AccountId}-${AWS::Region}/elasticmapreduce/'
      Instances:
        TerminationProtected: false
        Ec2SubnetId: 'subnet-ab12345c'
        MasterInstanceGroup:
          InstanceCount: 1
          InstanceType: !Ref ClusterInstanceType
        CoreInstanceGroup:
          InstanceCount: 1
          InstanceType: !Ref ClusterInstanceType
          Market: ON_DEMAND
          Name: Core

Outputs:
  ClusterId:
    Value:
```

```
      Ref: EmrCluster
    Description: The ID of the  EMR cluster
```

2. Erstellen Sie ein Portfolio für Ihre Cluster-Vorlagen in demselben AWS Konto wie Ihr Studio.

   a. Öffnen Sie die AWS Service Catalog Konsole unter https://console.aws.amazon.com/servicecatalog/.
   b. Wählen Sie im linken Navigationsmenü Portfolios.
   c. Geben Sie auf der Seite Portfolio erstellen die erforderlichen Informationen ein.
   d. Wählen Sie Erstellen. AWS Service Catalog erstellt das Portfolio und zeigt die Portfoliodetails an.

3. Führen Sie die folgenden Schritte aus, um Ihre Cluster-Vorlagen als AWS Service Catalog -Produkte hinzuzufügen.

   a. Navigieren Sie in der AWS Service Catalog -Managementkonsole unter Administration zur Seite Produkte.
   b. Wählen Sie Neues Produkt hochladen.
   c. Geben Sie einen Produktnamen und einen Eigentümer ein.
   d. Geben Sie Ihre Vorlagendatei unter Versionsdetails an.
   e. Wähle Überprüfen, um deine Produkteinstellungen zu überprüfen, und wähle dann Produkt erstellen.

4. Führen Sie die folgenden Schritte aus, um Ihre Produkte zu Ihrem Portfolio hinzuzufügen.

   a. Navigieren Sie in der AWS Service Catalog -Managementkonsole unter Administration zur Seite Produkte.
   b. Wählen Sie Ihr Produkt aus, klicken Sie auf Aktionen und anschließend auf Produkt zum Portfolio hinzufügen.
   c. Wählen Sie Ihr Portfolio aus und klicken Sie dann auf Produkt zum Portfolio hinzufügen.

5. Legen Sie eine Beschränkung für die Markteinführung deiner Produkte fest. Eine Startbeschränkung ist eine IAM Rolle, die Benutzerberechtigungen für die Markteinführung eines Produkts festlegt. Sie können Ihre Startbeschränkungen anpassen, müssen jedoch Berechtigungen zur Nutzung CloudFormation von Amazon EMR und gewähren AWS Service Catalog. Weitere Informationen und Anweisungen finden Sie unter Startbeschränkungen für den Service Catalog.

6. Wenden Sie Ihre Markteinführungsbeschränkung auf jedes Produkt in Ihrem Portfolio an. Sie müssen die Markteinführungsbeschränkung auf jedes Produkt einzeln anwenden.

a. Wählen Sie Ihr Portfolio auf der Portfolio-Seite in der AWS Service Catalog - Managementkonsole aus.

b. Wählen Sie die Registerkarte Constraints (Einschränkungen) und dann Create constraint (Einschränkung erstellen).

c. Wählen Sie Ihr Produkt aus und wählen Sie unter Einschränkungstyp die Option Starten aus. Klicken Sie auf Weiter.

d. Wählen Sie Ihre Startbeschränkungsrolle im Abschnitt Startbeschränkung aus, und wählen Sie dann Erstellen.

7. Gewähren Sie Zugriff auf Ihr Portfolio.

a. Wählen Sie Ihr Portfolio auf der Portfolio-Seite in der AWS Service Catalog - Managementkonsole aus.

b. Erweitern Sie den Tab Gruppen, Rollen und Benutzer und wählen Sie Gruppen, Rollen, Benutzer hinzufügen aus.

c. Suchen Sie auf der Registerkarte Rollen nach Ihrer EMR IAM Studio-Rolle, wählen Sie Ihre Rolle aus und klicken Sie auf Zugriff hinzufügen.

Wenn Sie …	Zugriff gewähren auf …
IAMAuthentifizierung	Ihre nativen Benutzer
IAMFöderation	Deine IAM Rolle für die Föderation
IAMIdentity Center-Verbund	Ihre EMRStudio-Benutzerrolle

Zugriff und Berechtigungen für Git-basierte Repositorys einrichten

EMRStudio unterstützt die folgenden Git-basierten Dienste:

- AWS CodeCommit
- GitHub
- Bitbucket
- GitLab

Damit EMR Studio-Benutzer ein Git-Repository mit einem Workspace verknüpfen können, richten Sie die folgenden Zugriffs- und Berechtigungsanforderungen ein. Sie können auch Git-basierte Repositorys konfigurieren, die Sie in einem privaten Netzwerk hosten, indem Sie den Anweisungen unter Ein privat gehostetes Git-Repository für EMR Studio konfigurieren folgen.

Cluster-Internetzugang

Sowohl EMR Amazon-Cluster, die auf Amazon EC2 ausgeführt werden, als auch Amazon EMR auf EKS Clustern, die an Studio Workspaces angeschlossen sind, müssen sich in einem privaten Subnetz befinden, das ein Network Address Translation (NAT) -Gateway verwendet, oder sie müssen über ein virtuelles privates Gateway auf das Internet zugreifen können. Weitere Informationen finden Sie unter VPCAmazon-Optionen.

Die Sicherheitsgruppen, die Sie mit EMR Studio verwenden, müssen auch eine Regel für ausgehenden Datenverkehr enthalten, die es Workspaces ermöglicht, den Datenverkehr von einem angeschlossenen Cluster ins Internet weiterzuleiten. EMR Weitere Informationen finden Sie unter Definieren Sie Sicherheitsgruppen zur Steuerung des EMR Studio-Netzwerkverkehrs.

> **Important**
>
> Wenn sich die Netzwerkschnittstelle in einem öffentlichen Subnetz befindet, kann sie nicht über ein Internet-Gateway () mit dem Internet kommunizieren. IGW

Berechtigungen für AWS Secrets Manager

Um EMR Studio-Benutzern den Zugriff auf Git-Repositorys mit gespeicherten Geheimnissen zu ermöglichen AWS Secrets Manager, fügen Sie der Servicerolle für EMR Studio eine Berechtigungsrichtlinie hinzu, die den `secretsmanager:GetSecretValue` Vorgang ermöglicht.

Informationen zum Verknüpfen von Git-basierten Repositorys mit Workspaces finden Sie unter Git-basierte Repositorys mit einem EMR Studio-Workspace verknüpfen.

Ein privat gehostetes Git-Repository für EMR Studio konfigurieren

Verwenden Sie die folgenden Anweisungen, um privat gehostete Repositorys für Amazon EMR Studio zu konfigurieren. Stellen Sie eine Konfigurationsdatei mit Informationen zu Ihren DNS und Git-Servern bereit. EMRStudio verwendet diese Informationen, um Workspaces zu konfigurieren, die den Datenverkehr an Ihre selbstverwalteten Repositorys weiterleiten können.

> **Note**
>
> Wenn Sie konfigurieren`DnsServerIpV4`, verwendet EMR Studio Ihren DNS Server, um `GitServerDnsName` sowohl Ihren als auch Ihren EMR Amazon-Endpunkt aufzulösen, z. `elasticmapreduce.us-east-1.amazonaws.com` Um einen Endpunkt für Amazon einzurichtenEMR, stellen Sie über den, den Sie mit Ihrem Studio verwendenVPC, eine Verbindung zu Ihrem Endpunkt her. Dadurch wird sichergestellt, dass der EMR Amazon-Endpunkt in eine private IP aufgelöst wird. Weitere Informationen finden Sie unter Stellen Sie EMR über einen VPC Schnittstellenendpunkt eine Connect zu Amazon her.

Voraussetzungen

Bevor Sie ein privat gehostetes Git-Repository für EMR Studio konfigurieren, benötigen Sie einen Amazon S3 S3-Speicherort, an dem EMR Studio die Workspaces und Notizbuchdateien im Studio sichern kann. Verwenden Sie denselben S3-Bucket, den Sie beim Erstellen eines Studios angegeben haben.

Um ein oder mehrere privat gehostete Git-Repositorys für EMR Studio zu konfigurieren

1. Erstellen Sie eine Konfigurationsdatei mithilfe der folgenden Vorlage. Geben Sie für jeden Git-Server, den Sie in Ihrer Konfiguration angeben möchten, die folgenden Werte an:

   - **`DnsServerIpV4`**- Die IPv4 Adresse Ihres DNS Servers. Wenn Sie Werte für sowohl als auch `DnsServerIpV4` angeben`GitServerIpV4List`, hat der Wert für `DnsServerIpV4` Vorrang und EMR Studio verwendet, um Ihr `GitServerDnsName` Problem `DnsServerIpV4` zu lösen.

     > **Note**
     >
     > Um privat gehostete Git-Repositorys verwenden zu können, muss Ihr DNS Server eingehenden Zugriff von Studio aus EMR zulassen. Wir bitten Sie dringend, Ihren DNS Server vor anderen, unbefugten Zugriffen zu schützen.

   - **`GitServerDnsName`**- Der DNS Name Ihres Git-Servers. Zum Beispiel `"git.example.com"`.
   - **`GitServerIpV4List`**- Eine Liste von IPv4 Adressen, die zu deinen Git-Servern gehören.

```
[
```

```
    {
        "Type": "PrivatelyHostedGitConfig",
        "Value": [
            {
                "DnsServerIpV4": "<10.24.34.xxx>",
                "GitServerDnsName": "<enterprise.git.com>",
                "GitServerIpV4List": [
                    "<xxx.xxx.xxx.xxx>",
                    "<xxx.xxx.xxx.xxx>"
                ]
            },
            {
                "DnsServerIpV4": "<10.24.34.xxx>",
                "GitServerDnsName": "<git.example.com>",
                "GitServerIpV4List": [
                    "<xxx.xxx.xxx.xxx>",
                    "<xxx.xxx.xxx.xxx>"
                ]
            }
        ]
    }
]
```

2. Speichern Sie Ihre Konfigurationsdatei unter `configuration.json`.
3. Laden Sie die Konfigurationsdatei in Ihren Amazon-S3-Speicherort in einem Ordner mit dem `life-cycle-configuration`-Namen hoch. Wenn Ihr Standard-S3-Speicherort beispielsweise `s3://`*`DOC-EXAMPLE-BUCKET`*`/studios` lautet, befindet sich Ihre Konfigurationsdatei in `s3://`*`DOC-EXAMPLE-BUCKET`*`/studios/`**`life-cycle-configuration/configuration.json`**.

> **Important**
>
> Wir bitten Sie dringend, den Zugriff auf Ihren `life-cycle-configuration` Ordner auf Studio-Administratoren und Ihre EMR Studio-Servicerolle zu beschränken und sich `configuration.json` vor unbefugtem Zugriff zu schützen. Anweisungen finden Sie unter Steuern des Zugriffs auf einen Bucket mit Benutzerrichtlinien oder Bewährte Sicherheitsmethoden für Amazon S3.

Anweisungen zum Hochladen finden Sie unter Erstellen eines Ordners und Hochladen von Objekten im Benutzerhandbuch für Amazon Simple Storage Service. Um Ihre Konfiguration auf

einen vorhandenen Workspace anzuwenden, schließen Sie den Workspace und starten Sie ihn neu, nachdem Sie Ihre Konfigurationsdatei auf Amazon S3 hochgeladen haben.

Optimieren Sie Spark-Jobs in EMR Studio

Wenn Sie einen Spark-Job mit EMR Studio ausführen, können Sie einige Schritte unternehmen, um sicherzustellen, dass Sie Ihre EMR Amazon-Cluster-Ressourcen optimieren.

Ihre Livy-Sitzung verlängern

Wenn Sie Apache Livy zusammen mit Spark auf Ihrem EMR Amazon-Cluster verwenden, empfehlen wir Ihnen, Ihr Livy-Sitzungs-Timeout zu erhöhen, indem Sie einen der folgenden Schritte ausführen:

- Wenn Sie einen EMR Amazon-Cluster erstellen, legen Sie diese Konfigurationsklassifizierung im Feld Konfiguration eingeben fest.

```
[
    {
        "Classification": "livy-conf",
        "Properties": {
          "livy.server.session.timeout": "8h"
        }
    }
]
```

- Stellen Sie für einen bereits laufenden EMR Cluster eine Verbindung zu Ihrem Cluster her `ssh` und legen Sie die `livy-conf` Konfigurationsklassifizierung unter fest. `/etc/livy/conf/livy.conf`

```
[
    {
        "Classification": "livy-conf",
        "Properties": {
          "livy.server.session.timeout": "8h"
        }
    }
]
```

  Möglicherweise müssen Sie Livy neu starten, nachdem Sie die Konfiguration geändert haben.

- Wenn Sie nicht möchten, dass es bei Ihrer Livy-Sitzung zu einem Timeout kommt, setzen Sie die Eigenschaft `livy.server.session.timeout-check` auf `false` in `/etc/livy/conf/livy.conf`.

Spark im Cluster-Modus ausführen

Im Clustermodus wird der Spark-Treiber auf einem Core-Knoten statt auf dem Primärknoten ausgeführt, wodurch die Ressourcennutzung auf dem Primärknoten verbessert wird.

Um Ihre Spark-Anwendung im Cluster-Modus statt im Standard-Client-Modus auszuführen, wählen Sie Cluster-Modus, wenn Sie bei der Konfiguration Ihres Spark-Schritts in Ihrem neuen EMR Amazon-Cluster den Bereitstellungsmodus festlegen. Weitere Informationen finden Sie unter Übersicht über den Clustermodus in der Apache-Spark-Dokumentation.

Den Spark-Treiberspeicher erhöhen

Um den Speicher des Spark-Treibers zu vergrößern, konfigurieren Sie Ihre Spark-Sitzung mit dem `%%configure` magischen Befehl in Ihrem EMR Notizbuch, wie im folgenden Beispiel.

```
%%configure -f
{"driverMemory": "6000M"}
```

Verwenden Sie ein Amazon EMR Studio

Dieser Abschnitt enthält Themen, die Ihnen bei der Konfiguration und Interaktion mit einem Amazon EMR Studio helfen.

Das folgende Video enthält praktische Informationen wie das Erstellen eines neuen Workspace und das Starten eines neuen EMR Amazon-Clusters mit einer Cluster-Vorlage. Das Video zeigt auch ein Beispiel-Notebook.

Dieser Abschnitt enthält die folgenden Themen, die Ihnen bei der Arbeit in einem EMR Studio helfen sollen:

- Informationen über Workspace-Grundlagen
- Konfigurieren Sie die Zusammenarbeit im Workspace
- Führen Sie einen EMR Studio-Workspace mit einer Runtime-Rolle aus
- Führen Sie Workspace-Notebooks programmgesteuert aus

- Durchsuchen Sie Daten mit dem Explorer SQL
- Hängen Sie einen Computer an einen EMR Studio-Workspace an
- Git-basierte Repositorys mit einem EMR Studio-Workspace verknüpfen
- Verwenden Sie den Amazon Athena SQL Athena-Editor in Studio EMR
- CodeWhisperer Amazon-Integration mit EMR Studio Workspaces
- Debuggen Sie Anwendungen und Jobs mit Studio EMR
- Installieren Sie Kernel und Bibliotheken in einem EMR Studio-Arbeitsbereich
- Verbessern Sie die Kernel mit Befehlen magic
- Verwenden Sie mehrsprachige Notebooks mit Spark-Kerneln

Informationen über Workspace-Grundlagen

Wenn Sie ein EMR Studio verwenden, können Sie verschiedene Workspaces erstellen und konfigurieren, um Notebooks zu organisieren und auszuführen. In diesem Abschnitt wird das Erstellen von und das Arbeiten mit Workspaces behandelt. Eine konzeptionelle Übersicht über die Workspaces finden Sie unter Wie Amazon EMR Studio funktioniert.

In diesem Abschnitt werden die folgenden Themen behandelt, die Ihnen bei der Verwendung von EMR Studio-Arbeitsbereichen helfen sollen:

- Erstellen Sie einen EMR Studio-Arbeitsbereich
- Einen WorkSpace starten
- Machen Sie sich mit der Workspace-Benutzeroberfläche vertraut
- Erkunden Sie Notebookbeispiele
- Workspace-Inhalt speichern
- Löschen Sie einen Workspace und Notebookdateien
- Informationen über Workspace-Status
- Beheben von Workspace-Verbindungsproblemen

Erstellen Sie einen EMR Studio-Arbeitsbereich

Sie können EMR Studio-Arbeitsbereiche erstellen, um Notebook-Code über die EMR Studio-Oberfläche auszuführen.

Um einen Workspace in einem EMR Studio zu erstellen

1. Loggen Sie sich in Ihr EMR Studio ein.
2. Wählen Sie Workspace erstellen.
3. Geben Sie Workspace-Name und Beschreibung ein. Wenn Sie einen Workspace benennen, können Sie ihn auf der Seite Workspaces leichter identifizieren.
4. Wenn Sie in Echtzeit mit anderen Studio-Benutzern in diesem Workspace zusammenarbeiten möchten, aktivieren Sie die Workspace-Zusammenarbeit. Sie können Mitarbeiter konfigurieren, nachdem Sie den Workspace gestartet haben.
5. Wenn Sie einen Cluster an einen Workspace anhängen möchten, erweitern Sie den Abschnitt Erweiterte Konfiguration. Wenn Sie möchten, können Sie später einen Cluster anhängen. Weitere Informationen finden Sie unter Hängen Sie einen Computer an einen EMR Studio-Workspace an.

   > **Note**
   >
   > Um einen neuen Cluster bereitzustellen, benötigen Sie Zugriffsberechtigungen von Ihrem Administrator.

   Wählen Sie eine der Clusteroptionen für den Workspace und hängen Sie den Cluster an. Weitere Informationen zum Bereitstellen eines Clusters beim Erstellen eines Workspace finden Sie unter Erstellen Sie einen neuen EMR Cluster und fügen Sie ihn einem Studio-Workspace hinzu EMR.
6. Wählen Sie unten rechts auf der Seite die Option Workspace erstellen aus.

Nachdem Sie einen Workspace erstellt haben, öffnet EMR Studio die Workspaces-Seite. Oben auf der Seite wird ein grünes Erfolgsbanner angezeigt und Sie können den neu erstellten Workspace in der Liste finden.

Standardmäßig wird ein Workspace geteilt und kann von allen Studio-Benutzern gesehen werden. Es kann jedoch jeweils nur ein Benutzer einen Workspace öffnen und darin arbeiten. Um gleichzeitig mit anderen Benutzern zu arbeiten, können Sie Konfigurieren Sie die Zusammenarbeit im Workspace

Einen WorkSpace starten

Um mit der Arbeit mit Notebookdateien zu beginnen, starten Sie einen Workspace, um auf den Notebook-Editor zuzugreifen. Auf der Seite Workspaces in einem Studio werden alle Workspaces

aufgeführt, auf die Sie Zugriff haben, mit Details wie Name, Status, Erstellungszeit und Letzte Änderung.

> **Note**
>
> Wenn Sie in der alten EMR Amazon-Konsole EMR Notizbücher hatten, finden Sie sie in der Konsole als EMR Studio-Arbeitsbereiche. EMRNotebook-Benutzer benötigen zusätzliche IAM Rollenberechtigungen, um auf Workspaces zuzugreifen oder diese zu erstellen. Wenn Sie kürzlich ein Notizbuch in der alten Konsole erstellt haben, müssen Sie möglicherweise die Liste der Arbeitsbereiche aktualisieren, damit es in der Konsole angezeigt wird. Weitere Informationen zum Übergang finden Sie unter Amazon EMR Notebooks sind als Amazon EMR Studio Workspaces in der Konsole verfügbar. und Amazon EMR-Konsole

So starten Sie einen Workspace zum Bearbeiten und Ausführen von Notebooks

1. Suchen Sie auf der Workspaces-Seite Ihres Studios nach dem Workspace. Sie können die Liste nach Schlüsselwort oder Spaltenwert filtern.
2. Wählen Sie den Workspace-Namen, um den Workspace in einer neuen Browser-Registerkarte zu starten. Es kann einige Minuten dauern, bis der Workspace geöffnet wird, wenn er inaktiv ist. Wählen Sie alternativ die Zeile für den Workspace aus und wählen Sie dann Workspace starten aus. Sie können aus den folgenden Startoptionen auswählen:

   - Schnellstart – Starten Sie Ihren Workspace schnell mit den Standardoptionen. Wählen Sie Schnellstart, wenn Sie Cluster an den Workspace in JupyterLab anhängen möchten.
   - Mit Optionen starten – Starten Sie Ihren Workspace mit benutzerdefinierten Optionen. Sie können entweder in Jupyter starten oder JupyterLab Ihren Workspace an einen EMR Cluster anhängen und Ihre Sicherheitsgruppen auswählen.

   > **Note**
   >
   > Es kann jeweils nur ein Benutzer einen Workspace öffnen und darin arbeiten. Wenn Sie einen Workspace auswählen, der bereits verwendet wird, zeigt EMR Studio eine Benachrichtigung an, wenn Sie versuchen, ihn zu öffnen. In der Spalte Benutzer auf der Workspaces-Seite wird der Benutzer angezeigt, der im Workspace arbeitet.

Machen Sie sich mit der Workspace-Benutzeroberfläche vertraut

Die Benutzeroberfläche von EMR Studio Workspace basiert auf der JupyterLabOberfläche mit den mit Symbolen versehenen Registerkarten in der linken Seitenleiste. Wenn Sie den Mauszeiger über einem Symbol halten, wird ein Tooltip mit dem Namen der Registerkarte angezeigt. Wählen Sie in der linken Seitenleiste Registerkarten aus, um auf die folgenden Bereiche zuzugreifen.

- Dateibrowser – Zeigt die Dateien und Verzeichnisse im Workspace sowie die Dateien und Verzeichnisse der verknüpften Git-Repositorys an.
- Ausführen von Kernels und Terminals – Listet alle Kernel und Terminals auf, die im Workspace laufen. Weitere Informationen finden Sie in der offiziellen Dokumentation unter Kernel und Terminals verwalten. JupyterLab
- Git – Stellt eine grafische Benutzeroberfläche für die Ausführung von Befehlen in den Git-Repositorys bereit, die an den Workspace angehängt sind. Dieses Panel ist eine JupyterLab Erweiterung namens jupyterlab-git. Weitere Informationen finden Sie unter jupyterlab-git.
- EMRCluster — Ermöglicht das Anhängen eines Clusters an den Workspace oder das Trennen eines Clusters vom Workspace, um Notebook-Code auszuführen. Das EMR Cluster-Konfigurationsfenster bietet auch erweiterte Konfigurationsoptionen, mit denen Sie einen neuen Cluster erstellen und an den Workspace anhängen können. Weitere Informationen finden Sie unter Erstellen Sie einen neuen EMR Cluster und fügen Sie ihn einem Studio-Workspace hinzu EMR.
- Amazon EMR Git Repository — Hilft Ihnen, den Workspace mit bis zu drei Git-Repositorys zu verknüpfen. Weitere Informationen und Anweisungen finden Sie in Git-basierte Repositorys mit einem EMR Studio-Workspace verknüpfen.
- Notebook-Beispiele – Enthält eine Liste mit Notebookbeispielen, die Sie im Workspace speichern können. Sie können auf die Beispiele auch zugreifen, indem Sie auf der Launcher-Seite des Workspace die Option Notebook-Beispiele auswählen.
- Befehle — Bietet eine tastaturgesteuerte Möglichkeit, nach Befehlen zu suchen und diese auszuführen. JupyterLab Weitere Informationen finden Sie auf der Seite mit der Befehlspalette in der JupyterLab Dokumentation.
- Notebook-Tools – Ermöglicht die Auswahl und Einstellung von Optionen wie Zellfolientyp und Metadaten. Die Option Notebook-Tools wird in der linken Seitenleiste angezeigt, nachdem Sie eine Notebookdatei geöffnet haben.
- Registerkarten öffnen – Listet die geöffneten Dokumente und Aktivitäten im Haupt-Workspace auf, sodass Sie zu einer geöffneten Registerkarte springen können. Weitere Informationen finden Sie in der JupyterLab Dokumentation auf der Seite „Tabs und Einzeldokumentmodus“.

- Zusammenarbeit – Ermöglicht es Ihnen, die Zusammenarbeit im Workspace zu aktivieren oder zu deaktivieren und Mitarbeiter zu verwalten. Sie benötigen die nötigen Berechtigungen, um das Panel Zusammenarbeit anzeigen zu können. Weitere Informationen finden Sie unter Legen Sie die Eigentümerschaft für die Workspace-Zusammenarbeit fest.

Erkunden Sie Notebookbeispiele

Jeder EMR Studio-Arbeitsbereich enthält eine Reihe von Notizbuchbeispielen, anhand derer Sie die EMR Studio-Funktionen erkunden können. Um ein Notebook-Beispiel zu bearbeiten oder auszuführen, können Sie es im Workspace speichern.

Um ein Notebookbeispiel in einem Workspace zu speichern

1. Wählen Sie in der linken Seitenleiste den Tab Notebook-Beispiele, um den Bereich Notebook-Beispiele zu öffnen. Sie können auf die Beispiele auch zugreifen, indem Sie auf der Launcher-Seite des Workspace die Option Notebook-Beispiele auswählen.
2. Wählen Sie ein Notebookbeispiel aus, um es im Hauptarbeitsbereich als Vorschau anzuzeigen. Das Beispiel ist schreibgeschützt.
3. Um das Notebookbeispiel im Workspace zu speichern, wählen Sie In Workspace speichern. EMRStudio speichert das Beispiel in Ihrem Home-Verzeichnis. Nachdem Sie ein Notebook-Beispiel im Workspace gespeichert haben, können Sie es umbenennen, bearbeiten und ausführen.

Weitere Informationen zu den Notebook-Beispielen finden Sie im EMRStudio Notebook Examples GitHub Repository.

Workspace-Inhalt speichern

Wenn Sie im Notebook-Editor eines Workspace arbeiten, speichert EMR Studio den Inhalt der Notebookzellen und gibt ihn für Sie an dem Amazon S3 S3-Speicherort aus, der dem Studio zugeordnet ist. Bei diesem Backup-Vorgang bleibt die Arbeit zwischen den Sitzungen erhalten.

Sie können ein Notizbuch auch speichern, indem Sie auf der Registerkarte „Notizbuch öffnen“ CTRL +S drücken oder eine der Speicheroptionen unter Datei verwenden.

Eine weitere Möglichkeit, die Notebookdateien in einem Workspace zu sichern, besteht darin, den Workspace mit einem Git-basierten Repository zu verknüpfen und Ihre Änderungen mit dem Remote-Repository zu synchronisieren. Auf diese Weise können Sie auch Notizbücher

speichern und mit Teammitgliedern teilen, die einen anderen Workspace oder Studio verwenden. Detaillierte Anweisungen finden Sie unter Git-basierte Repositorys mit einem EMR Studio-Workspace verknüpfen.

Löschen Sie einen Workspace und Notebookdateien

Wenn Sie eine Notizbuchdatei aus einem EMR Studio-Arbeitsbereich löschen, löschen Sie die Datei aus dem Dateibrowser, und EMR Studio entfernt ihre Sicherungskopie in Amazon S3. Sie müssen keine weiteren Schritte unternehmen, um Speichergebühren zu vermeiden, wenn Sie eine Datei aus einem Workspace löschen.

Wenn Sie einen gesamten Workspace löschen, verbleiben die zugehörigen Notebookdateien und -ordner am Amazon-S3-Speicherort. Für die Dateien fallen weiterhin Speichergebühren an. Um Speichergebühren zu vermeiden, entfernen Sie alle gesicherten Dateien und Ordner, die mit Ihrem gelöschten Workspace verknüpft sind, aus Amazon S3.

Um eine Notebook-Datei aus einem EMR Studio-Arbeitsbereich zu löschen

1. Wählen Sie in der linken Seitenleiste des Workspace den Bereich Dateibrowser aus.
2. Wählen Sie die Datei oder den Ordner aus, die Sie löschen möchten. Klicken Sie mit der rechten Maustaste auf die ausgewählten Dateien oder Ordner und wählen Sie Löschen. Die Datei verschwindet aus der Liste. EMRStudio entfernt die Datei oder den Ordner für Sie aus Amazon S3.

From the Workspace UI

Löschen Sie einen Workspace und die zugehörigen Backup-Dateien aus EMR Studio

1. Melden Sie sich mit Ihrem EMR Studio-Zugriff bei Ihrem Studio an URL und wählen Sie in der linken Navigationsleiste Workspaces aus.
2. Suchen Sie in der Liste nach Ihrem Workspace und aktivieren Sie das Kontrollkästchen neben dessen Namen. Sie können mehrere Arbeitsbereiche zum gleichzeitigen Löschen auswählen.
3. Wählen Sie oben rechts in der Liste der Arbeitsbereiche die Option Löschen aus und bestätigen Sie, dass Sie die ausgewählten Arbeitsbereiche löschen möchten. Wählen Sie zur Bestätigung Delete.
4. Wenn Sie die Notebookdateien, die mit dem gelöschten Workspace verknüpft waren, aus Amazon S3 entfernen möchten, folgen Sie den Anweisungen zum Löschen von Objekten im

Konsole-Benutzerhandbuch von Amazon Simple Storage Service. Wenn Sie das Studio nicht erstellt haben, wenden Sie sich an Ihren Studio-Administrator, um den Amazon-S3-Backup-Speicherort für den gelöschten Workspace zu ermitteln.

From the Workspaces list

Löschen Sie einen Workspace und die zugehörigen Backup-Dateien aus der Workspaces-Liste

1. Navigieren Sie in der Konsole zur Workspace-Liste.
2. Wählen Sie den Workspace, den Sie löschen möchten, aus der Liste aus und klicken Sie dann auf Aktionen.
3. Wählen Sie Löschen.
4. Wenn Sie die Notebookdateien, die mit dem gelöschten Workspace verknüpft waren, aus Amazon S3 entfernen möchten, folgen Sie den Anweisungen zum Löschen von Objekten im Konsole-Benutzerhandbuch von Amazon Simple Storage Service. Wenn Sie das Studio nicht erstellt haben, wenden Sie sich an Ihren Studio-Administrator, um den Amazon-S3-Backup-Speicherort für den gelöschten Workspace zu ermitteln.

Informationen über Workspace-Status

Nachdem Sie einen EMR Studio-Arbeitsbereich erstellt haben, wird er in der Workspaces-Liste in Ihrem Studio als Zeile mit seinem Namen, Status, Erstellungszeit und dem Zeitstempel der letzten Änderung angezeigt. Die folgende Tabelle beschreibt den Workspace-Status.

Status	Description
Wird gestartet	Der Workspace wird vorbereitet, ist aber noch nicht einsatzbereit. Sie können einen Workspace nicht öffnen, wenn er den Status „Wird gestartet“ hat.
Bereit	Sie können den Workspace öffnen, um den Notebook-Editor zu verwenden, aber Sie müssen den Workspace an einen EMR Cluster anhängen, bevor Sie Notebook-Code ausführen können.

Status	Description
Anfügen	Der Workspace wird an einen Cluster angehängt.
Attached (Angefügt)	Der Workspace ist mit einem EMR Cluster verbunden und bereit, damit Sie Notebook-Code schreiben und ausführen können. Wenn der Status eines Workspaces nicht Angefügt lautet, müssen Sie ihn an einen Cluster anhängen, bevor Sie Notebook-Code ausführen können.
Inaktiv	Der Workspace wurde gestoppt. Um einen inaktiven Workspace zu reaktivieren, wählen Sie ihn aus der Liste Workspaces aus. Wenn Sie den Workspace auswählen, ändert sich der Status von Inaktiv zu Beginnt zu Bereit.
Wird angehalten	Der Workspace wird heruntergefahren und auf Inaktiv gesetzt. Wenn Sie einen Workspace beenden, beendet er alle entsprechenden Notebook-Kernel. EMRStudio stoppt Notebooks , die lange Zeit inaktiv waren.
Wird gelöscht	Wenn Sie einen Workspace löschen, markiert EMR Studio ihn zum Löschen und startet den Löschvorgang. Nach Abschluss des Löschvorgangs verschwindet der Workspace aus der Liste. Wenn Sie einen Workspace löschen, verbleiben seine Notebookdateien im Amazon-S3-Speicherort.

Beheben von Workspace-Verbindungsproblemen

Um Probleme mit der Workspace-Konnektivität zu lösen, können Sie einen Workspace beenden und neu starten. Wenn Sie einen Workspace neu starten, startet EMR Studio den Workspace in einer anderen Availability Zone oder einem anderen Subnetz, das Ihrem Studio zugeordnet ist.

Um einen EMR Studio-Workspace zu beenden und neu zu starten

1. Schließen Sie den Workspace in Ihrem Browser.
2. Navigieren Sie in der Konsole zur Workspace-Liste.
3. Wählen Sie Ihren Workspace aus der Liste aus und klicken Sie auf Aktionen.
4. Wählen Sie Stopp und warten Sie, bis sich der Workspace-Status von Stopp auf Inaktiv ändert.
5. Wählen Sie erneut Aktionen und dann Start, um den Workspace neu zu starten.
6. Warten Sie, bis sich der Workspace-Status von Beginnt auf Bereit ändert, und wählen Sie dann den Workspace-Namen, um ihn in einer neuen Browser-Registerkarte erneut zu öffnen.

Konfigurieren Sie die Zusammenarbeit im Workspace

Mit Workspace-Zusammenarbeit können Sie Notebook-Code gleichzeitig mit anderen Mitgliedern Ihres Teams schreiben und ausführen. Wenn Sie an derselben Notebookdatei arbeiten, sehen Sie die Änderungen, die Ihre Mitarbeiter vornehmen. Sie können die Zusammenarbeit aktivieren, wenn Sie einen Workspace erstellen, oder die Zusammenarbeit in einem vorhandenen Workspace ein- und ausschalten.

> **Note**
>
> EMRDie Zusammenarbeit in Studio Workspace wird bei EMRserverlosen interaktiven Anwendungen oder wenn die Verbreitung vertrauenswürdiger Identitäten aktiviert ist, nicht unterstützt.

Voraussetzungen

Bevor Sie die Zusammenarbeit für einen Workspace konfigurieren, stellen Sie sicher, dass Sie die folgenden Aufgaben abgeschlossen haben:

- Stellen Sie sicher, dass Ihr EMR Studio-Administrator Ihnen die erforderlichen Berechtigungen erteilt hat. Die folgende Beispielanweisung ermöglicht es einem Benutzer, die Zusammenarbeit

für jeden Workspace mit dem Tag-Schlüssel `creatorUserId` zu konfigurieren, dessen Wert der Benutzer-ID entspricht (angegeben durch die Richtlinienvariable `aws:userId`).

```
{
    "Sid": "UserRolePermissionsForCollaboration",
    "Action": [
        "elasticmapreduce:UpdateEditor",
        "elasticmapreduce:PutWorkspaceAccess",
        "elasticmapreduce:DeleteWorkspaceAccess",
        "elasticmapreduce:ListWorkspaceAccessIdentities"
    ],
    "Resource": "*",
    "Effect": "Allow",
    "Condition": {
        "StringEquals": {
            "elasticmapreduce:ResourceTag/creatorUserId": "${aws:userid}"
        }
    }
}
```

- Stellen Sie sicher, dass die Ihrem EMR Studio zugeordnete Servicerolle über die erforderlichen Berechtigungen verfügt, um die Workspace-Zusammenarbeit zu aktivieren und zu konfigurieren, wie in der folgenden Beispielanweisung dargestellt.

```
{
    "Sid": "AllowWorkspaceCollaboration",
    "Effect": "Allow",
    "Action": [
        "iam:GetUser",
        "iam:GetRole",
        "iam:ListUsers",
        "iam:ListRoles",
        "sso:GetManagedApplicationInstance",
        "sso-directory:SearchUsers"
    ],
    "Resource": "*"
}
```

Weitere Informationen finden Sie unter Erstellen Sie eine EMR Studio-Dienstrolle.

Um die Workspace-Zusammenarbeit zu aktivieren und Mitarbeiter hinzuzufügen

1. Wähle in deinem Workspace im Launcher-Bildschirm oder unten im linken Bereich das Zusammenarbeit-Symbol aus.

   > **Note**
   >
   > Das Panel Zusammenarbeit wird Ihnen nur angezeigt, wenn Ihr Studio-Administrator Ihnen die Erlaubnis erteilt hat, die Zusammenarbeit für den Workspace zu konfigurieren. Weitere Informationen finden Sie unter Legen Sie die Eigentümerschaft für die Workspace-Zusammenarbeit fest.

2. Vergewissern Sie sich, dass der Schalter Workspace-Zusammenarbeit zulassen aktiviert ist. Wenn Sie die Zusammenarbeit aktivieren, können nur Sie und die von Ihnen hinzugefügten Mitarbeiter den Workspace in der Liste auf der Studio-Workspaces-Seite sehen.
3. Geben Sie einen Namen für den Mitarbeiter ein. In Ihrem Workspace können maximal fünf Mitarbeiter enthalten sein, einschließlich Ihnen. Ein Mitarbeiter kann jeder Benutzer sein, der Zugriff auf Ihr EMR Studio hat. Wenn Sie keinen Mitarbeiter angeben, ist der Workspace ein privater Workspace, auf den nur Sie zugreifen können.

   In der folgenden Tabelle sind die zutreffenden Werte für Mitarbeiter angegeben, die je nach Identitätstyp des Inhabers eingegeben werden müssen.

   > **Note**
   >
   > Ein Inhaber kann nur Mitarbeiter mit demselben Identitätstyp einladen. Beispielsweise kann ein Benutzer nur andere Benutzer hinzufügen, und ein IAM Identity Center-Benutzer kann nur andere IAM Identity Center-Benutzer hinzufügen.

Authentifizierungsmodus	Wert, der für den Namen des Mitarbeiters eingegeben werden soll
IAMAuthentifizierung	Ein Benutzername. Dies ist der Name, den ein Benutzer sieht, wenn er bei der AWS Management Console angemeldet ist.

Authentifizierungsmodus	Wert, der für den Namen des Mitarbeiters eingegeben werden soll
IAMFöderation	Der Name einer IAM Rolle und ein optionaler Sitzungsname. Um alle Verbundbenutzer hinzuzufügen, die dieselbe IAM Rolle übernehmen, geben Sie den Namen einer IAM Rolle für den Verbund an. Um einen einzelnen Benutzer als Mitarbeiter hinzuzufügen, geben Sie eine Rolle und einen Sitzungsnamen an. Beispiel, `MyRoleName:MySessionName` .
SSO	Ein IAM Identity Center-Benutzername wie `user@example.com`.

4. Wählen Sie Hinzufügen aus. Der Mitarbeiter kann den Workspace jetzt auf seiner EMR Studio-Workspaces-Seite sehen und den Workspace starten, um ihn in Echtzeit mit Ihnen zu verwenden.

> **Note**
>
> Wenn Sie die Workspace-Zusammenarbeit deaktivieren, kehrt der Workspace in seinen gemeinsamen Status zurück und kann von allen Studio-Benutzern eingesehen werden. Im geteilten Status kann jeweils nur ein Studio-Benutzer den Workspace öffnen und darin arbeiten.

Führen Sie einen EMR Studio-Workspace mit einer Runtime-Rolle aus

> **Note**
>
> Die auf dieser Seite beschriebene Runtime-Rollenfunktionalität gilt nur für Amazon, die auf Amazon EMR ausgeführt wirdEC2, und bezieht sich nicht auf die Runtime-Rollenfunktionalität in EMR serverlosen interaktiven Anwendungen. Weitere Informationen zur Verwendung von

Runtime-Rollen in EMR Serverless finden Sie unter Job Runtime Roles im Amazon EMR Serverless User Guide.

Eine Runtime-Rolle ist eine AWS Identity and Access Management (IAM) -Rolle, die Sie angeben können, wenn Sie einen Job oder eine Anfrage an einen EMR Amazon-Cluster senden. Der Job oder die Abfrage, die Sie an Ihren EMR Cluster senden, verwendet die Runtime-Rolle, um auf AWS Ressourcen wie Objekte in Amazon S3 zuzugreifen.

Wenn Sie einen EMR Studio-Arbeitsbereich an einen EMR Cluster anhängen, der Amazon EMR 6.11 oder höher verwendet, können Sie eine Runtime-Rolle für den Job oder die Abfrage auswählen, die Sie einreichen, um sie beim Zugriff auf Ressourcen zu verwenden. AWS Wenn der EMR Cluster jedoch keine Runtime-Rollen unterstützt, übernimmt der EMR Cluster die Rolle nicht, wenn er auf Ressourcen zugreift AWS .

Bevor Sie eine Runtime-Rolle mit einem Amazon EMR Studio-Workspace verwenden können, muss ein Administrator Benutzerberechtigungen so konfigurieren, dass der Studio-Benutzer die `elasticmapreduce:GetClusterSessionCredentials` API On-the-Runtime-Rolle aufrufen kann. Starten Sie dann einen neuen Cluster mit einer Runtime-Rolle, die Sie mit Ihrem Amazon EMR Studio Workspace verwenden können.

Auf dieser Seite

Konfigurieren Sie Benutzerberechtigungen für die Laufzeit-Rolle

Konfigurieren Sie Benutzerberechtigungen, sodass der Studio-Benutzer die `elasticmapreduce:GetClusterSessionCredentials` API Runtime-Rolle aufrufen kann, die der Benutzer verwenden möchte. Sie müssen auch the section called “Studio-Benutzerberechtigungen (EC2,EKS)” konfigurieren, bevor der Benutzer Studio verwenden kann.

> **Warning**
>
> Um diese Berechtigung zu erteilen, erstellen Sie eine auf dem `elasticmapreduce:ExecutionRoleArn` Kontextschlüssel basierende Bedingung, wenn Sie einem Anrufer Zugriff auf den `GetClusterSessionCredentials` APIs gewähren. Das folgende Beispiel veranschaulicht die Vorgehensweise hierfür.

```
{
      "Sid": "AllowSpecificExecRoleArn",
      "Effect": "Allow",
      "Action": [
          "elasticmapreduce:GetClusterSessionCredentials"
      ],
      "Resource": "*",
      "Condition": {
          "StringEquals": {
              "elasticmapreduce:ExecutionRoleArn": [
                  "arn:aws:iam::111122223333:role/test-emr-demo1",
                  "arn:aws:iam::111122223333:role/test-emr-demo2"
              ]
          }
      }
  }
```

Das folgende Beispiel zeigt, wie einem IAM Prinzipal die Verwendung einer IAM Rolle mit dem Namen `test-emr-demo3` Runtime-Rolle ermöglicht wird. Darüber hinaus kann der Versicherungsnehmer nur mit der Cluster-ID auf EMR Amazon-Cluster zugreifen`j-123456789`.

```
{
    "Sid":"AllowSpecificExecRoleArn",
    "Effect":"Allow",
    "Action":[
        "elasticmapreduce:GetClusterSessionCredentials"
    ],
    "Resource": [
          "arn:aws:elasticmapreduce:<region>:111122223333:cluster/j-123456789"
     ],
    "Condition":{
        "StringEquals":{
            "elasticmapreduce:ExecutionRoleArn":[
```

```
                "arn:aws:iam::111122223333:role/test-emr-demo3"
            ]
        }
    }
}
```

Im folgenden Beispiel kann ein IAM Principal jede IAM Rolle, deren Name mit der Zeichenfolge beginnt, `test-emr-demo4` als Laufzeitrolle verwenden. Darüber hinaus kann der Versicherungsnehmer nur auf EMR Amazon-Cluster zugreifen, die mit dem Schlüssel-Wert-Paar gekennzeichnet sind. `tagKey: tagValue`

```
{
    "Sid":"AllowSpecificExecRoleArn",
    "Effect":"Allow",
    "Action":[
        "elasticmapreduce:GetClusterSessionCredentials"
    ],
    "Resource": "*",
    "Condition":{
        "StringEquals":{
             "elasticmapreduce:ResourceTag/tagKey": "tagValue"
        },
        "StringLike":{
            "elasticmapreduce:ExecutionRoleArn":[
                "arn:aws:iam::111122223333:role/test-emr-demo4*"
            ]
        }
    }
}
```

Starten Sie einen neuen Cluster mit einer Laufzeit-Rolle

Nachdem Sie über die erforderlichen Berechtigungen verfügen, starten Sie einen neuen Cluster mit einer Runtime-Rolle, die Sie mit Ihrem Amazon EMR Studio Workspace verwenden können.

Wenn Sie bereits einen neuen Cluster mit einer Laufzeit-Rolle gestartet haben, können Sie mit dem Abschnitt the section called “Verwenden Sie den Cluster mit Ihrem Workspace” fortfahren.

1. Erfüllen Sie zunächst die Voraussetzungen im Abschnitt EMRSchritte zu Runtime-Rollen für Amazon.

2. Starten Sie dann einen Cluster mit den folgenden Einstellungen, um Runtime-Rollen mit Amazon EMR Studio Workspaces zu verwenden. Anweisungen zum Start Ihres Clusters finden Sie unter Angabe einer Sicherheitskonfiguration für einen Cluster.

   - Wählen Sie für Release die Option emr-6.11.0 oder höher aus.
   - Wählen Sie Spark, Livy und Jupyter Enterprise Gateway als Ihre Cluster-Anwendungen aus.
   - Verwenden Sie die Sicherheitskonfiguration, die Sie im vorherigen Schritt erstellt haben.
   - Optional können Sie Lake Formation für Ihren EMR Cluster aktivieren. Weitere Informationen finden Sie unter Aktivieren Sie Lake Formation mit Amazon EMR.

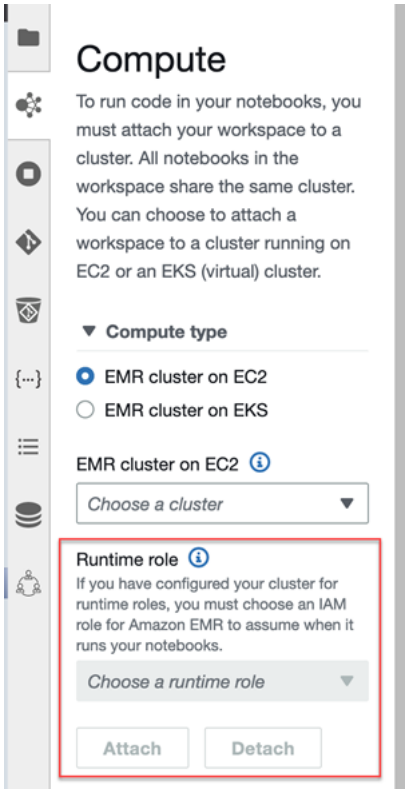
Nachdem Sie Ihren Cluster gestartet haben, können Sie den Runtime-Cluster mit aktivierter Rolle und einem EMR Studio-Arbeitsbereich verwenden.

> **Note**
>
> Der ExecutionRoleArnWert wird derzeit bei der StartNotebookExecutionAPIOperation nicht unterstützt, obwohl der `ExecutionEngineConfig.Type` Wert `EMR`

Verwenden Sie den EMR Cluster mit einer Runtime-Rolle in Workspaces

Sobald Sie Ihren Cluster eingerichtet und gestartet haben, können Sie den rollenfähigen Runtime-Cluster mit Ihrem Studio Workspace verwenden. EMR

1. Erstellen Sie einen neuen Workspace oder starten Sie einen vorhandenen Workspace. Weitere Informationen finden Sie unter Erstellen Sie einen EMR Studio-Arbeitsbereich.
2. Wählen Sie in der linken Seitenleiste Ihres geöffneten Workspace die Registerkarte EMRCluster, erweitern Sie den Abschnitt Compute-Typ und wählen Sie Ihren Cluster aus dem EC2 Cluster-On-Menü und die EMR Runtime-Rolle aus dem Runtime-Rollenmenü aus.



3. Wählen Sie Anhängen, um den Cluster mit der Laufzeit-Rolle an Ihren Workspace anzuhängen.

Überlegungen

Beachten Sie die folgenden Überlegungen, wenn Sie einen rollenfähigen Runtime-Cluster mit Ihrem Amazon EMR Studio Workspace verwenden:

- Sie können eine Runtime-Rolle nur auswählen, wenn Sie einen EMR Studio-Workspace an einen EMR Cluster anhängen, der Amazon EMR Version 6.11 oder höher verwendet.
- Die auf dieser Seite beschriebene Runtime-Rollenfunktionalität wird nur unterstützt, wenn Amazon auf Amazon EMR läuftEC2, und nicht für EMR serverlose interaktive Anwendungen. Weitere Informationen zu Runtime-Rollen für EMR Serverless finden Sie unter Job Runtime Roles im Amazon EMR Serverless User Guide.
- Sie müssen zwar zusätzliche Berechtigungen konfigurieren, bevor Sie eine Runtime-Rolle angeben können, wenn Sie einen Job an einen Cluster senden, aber Sie benötigen keine zusätzlichen Berechtigungen, um auf die von einem EMR Studio-Workspace generierten Dateien zuzugreifen. Die Berechtigungen für solche Dateien sind dieselben wie für Dateien, die aus Clustern ohne Laufzeit-Rollen generiert wurden.

- Sie können SQL Explorer nicht in einem EMR Studio-Arbeitsbereich mit einem Cluster verwenden, der über eine Runtime-Rolle verfügt. Amazon EMR deaktiviert SQL Explorer in der Benutzeroberfläche, wenn ein Workspace an einen Runtime-Cluster mit aktivierter Rolle EMR angehängt ist.
- Sie können den Kollaborationsmodus nicht in einem EMR Studio-Arbeitsbereich mit einem Cluster verwenden, der über eine Runtime-Rolle verfügt. Amazon EMR deaktiviert die Workspace-Kollaborationsfunktionen, wenn ein Workspace an einen rollenfähigen EMR Runtime-Cluster angehängt ist. Der Workspace bleibt nur für den Benutzer zugänglich, der den Workspace angehängt hat.
- Sie können Runtime-Rollen nicht in einem Studio verwenden, in dem IAM Identity Center Trusted Identity Propagation aktiviert ist.
- Möglicherweise wird die Warnung Die Seite ist möglicherweise nicht sicher! angezeigt von der Spark-Benutzeroberfläche für einen Laufzeit-Cluster mit aktivierter Rolle. In diesem Fall umgehen Sie die Warnung, um weiterhin die Spark-Benutzeroberfläche zu sehen.

Führen Sie Workspace-Notebooks programmgesteuert aus

> **Note**
>
> Die programmatische Ausführung von Notebooks wird mit interaktiven Amazon EMR Serverless-Anwendungen nicht unterstützt.

Sie können Ihre Amazon EMR Studio Workspace-Notebooks programmgesteuert mit einem Skript oder auf dem ausführen. AWS CLI Informationen zum programmgesteuerten Ausführen Ihres Notebooks finden Sie unter Beispielbefehle zur programmgesteuerten Ausführung von EMR Notebooks.

Durchsuchen Sie Daten mit dem Explorer SQL

> **Note**
>
> SQLExplorer for EMR Studio wird mit interaktiven Amazon EMR Serverless-Anwendungen oder in einem Studio, in dem IAM Identity Center vertrauenswürdige Identitätsverbreitung aktiviert ist, nicht unterstützt.

Dieses Thema enthält Informationen, die Ihnen bei den ersten Schritten mit SQL Explorer in Amazon EMR Studio helfen sollen. SQLExplorer ist ein einseitiges Tool in Ihrem Workspace, das Ihnen hilft, die Datenquellen im Datenkatalog Ihres EMR Clusters zu verstehen. Sie können den SQL Explorer verwenden, um Ihre Daten zu durchsuchen, SQL Abfragen zum Abrufen von Daten auszuführen und Abfrageergebnisse herunterzuladen.

SQLDer Explorer unterstützt Presto. Bevor Sie SQL Explorer verwenden, stellen Sie sicher, dass Sie über einen Cluster verfügen, der Amazon EMR Version 5.34.0 oder höher oder Version 6.4.0 oder höher verwendet und Presto installiert hat. Der Amazon EMR Studio SQL Explorer unterstützt keine Presto-Cluster, die Sie mit Verschlüsselung bei der Übertragung konfiguriert haben. Das liegt daran, dass Presto auf diesen Clustern im TLS Modus läuft.

Durchsuchen Sie den Datenkatalog Ihres Clusters

SQLExplorer bietet eine Katalogbrowser-Oberfläche, mit der Sie untersuchen und verstehen können, wie Ihre Daten organisiert sind. Sie können beispielsweise den Datenkatalog-Browser verwenden, um Tabellen- und Spaltennamen zu überprüfen, bevor Sie eine SQL Abfrage schreiben.

Wie Sie Ihren Datenkatalog durchsuchen

1. Öffnen Sie den SQL Explorer in Ihrem Workspace.
2. Stellen Sie sicher, dass Ihr Workspace mit einem EMR Cluster verbunden istEC2, auf dem Amazon EMR Version 6.4.0 oder höher mit installiertem Presto ausgeführt wird. Sie können einen vorhandenen Cluster auswählen oder einen neuen erstellen. Weitere Informationen finden Sie unter Hängen Sie einen Computer an einen EMR Studio-Workspace an.
3. Wählen Sie eine Datenbank aus der Drop-down-Liste aus, um sie zu durchsuchen.
4. Erweitern Sie eine Tabelle in Ihrer Datenbank, um die Spaltennamen der Tabelle zu sehen. Sie können in der Suchleiste auch ein Schlüsselwort eingeben, um die Tabellenergebnisse zu filtern.

Führen Sie eine SQL Abfrage aus, um Daten abzurufen

Um Daten mit einer SQL Abfrage abzurufen und die Ergebnisse herunterzuladen

1. Öffnen Sie den SQL Explorer in Ihrem Workspace.
2. Stellen Sie sicher, dass Ihr Workspace EC2 mit einem EMR Cluster verbunden ist, auf dem Presto und Spark installiert sind. Sie können einen vorhandenen Cluster auswählen oder einen neuen erstellen. Weitere Informationen finden Sie unter Hängen Sie einen Computer an einen EMR Studio-Workspace an.

3. Wählen Sie Editor öffnen, um eine neue Editor-Registerkarte in Ihrem Workspace zu öffnen.
4. Verfassen Sie Ihre SQL Abfrage auf der Registerkarte „Editor“.
5. Wählen Sie Ausführen aus.
6. Sehen Sie sich Ihre Abfrageergebnisse unter Ergebnisvorschau an. SQLDer Explorer zeigt standardmäßig die ersten 100 Ergebnisse an. Sie können eine andere Anzahl von Ergebnissen für die Anzeige auswählen (bis zu 1 000), indem Sie das Dropdownmenü Vorschau der ersten 100 Abfrageergebnisse verwenden.
7. Wählen Sie Ergebnisse herunterladen, um Ihre Ergebnisse im CSV Format herunterzuladen. Sie können bis zu 1 000 Ergebniszeilen herunterladen.

Hängen Sie einen Computer an einen EMR Studio-Workspace an

Amazon EMR Studio führt Notebook-Befehle mithilfe eines Kernels auf einem EMR Cluster aus. Bevor Sie einen Kernel auswählen können, sollten Sie den Workspace an einen Cluster anhängen, der EC2 Amazon-Instances verwendet, an einen EMR EKS Amazon-On-Cluster oder an eine EMR serverlose Anwendung. EMRStudio ermöglicht es Ihnen, Workspaces an neue oder bestehende Cluster anzuhängen, und bietet Ihnen die Flexibilität, Cluster zu ändern, ohne den Workspace zu schließen.

In diesem Abschnitt werden die folgenden Themen behandelt, die Ihnen bei der Arbeit mit und der Bereitstellung von Clustern für EMR Studio helfen sollen:

- Einen EC2 Amazon-Cluster an einen EMR Studio-Arbeitsbereich anhängen
- Hängen Sie einen EMR Amazon EKS On-Cluster an einen EMR Studio-Arbeitsbereich an
- Eine Amazon EMR Serverless-Anwendung an einen EMR Studio-Arbeitsbereich anhängen
- Erstellen Sie einen neuen EMR Cluster und fügen Sie ihn einem Studio-Workspace hinzu EMR
- Trennen Sie einen Computer von einem Studio-Workspace EMR

Einen EC2 Amazon-Cluster an einen EMR Studio-Arbeitsbereich anhängen

Sie können einen auf Amazon laufenden EMR Cluster EC2 an einen Workspace anhängen, wenn Sie den Workspace erstellen, oder einen Cluster an einen vorhandenen Workspace anhängen. Wenn Sie einen neuen Cluster erstellen und anhängen möchten, lesen Sie Erstellen Sie einen neuen EMR Cluster und fügen Sie ihn einem Studio-Workspace hinzu EMR.

> **Note**
>
> Ein Workspace in einem Studio, für den IAM Identity Center Trusted Identity Propagation aktiviert ist, kann nur an einen EMR Cluster angehängt werden, für den Identity Center aktiviert ist, dessen Sicherheitskonfiguration Identity Center aktiviert ist.

On create

Beim Erstellen eines Workspace eine Verbindung zu einem EMR Amazon-Compute-Cluster herstellen

1. Stellen Sie im Dialogfeld Workspace erstellen sicher, dass Sie bereits ein Subnetz für den neuen Workspace ausgewählt haben. Erweitern Sie den Abschnitt Erweiterte Konfiguration.
2. Wählen Sie Workspace an einen EMR Cluster anhängen.
3. Wählen Sie in der EMRCluster-Dropdown-Liste einen vorhandenen EMR Cluster aus, der an den Workspace angehängt werden soll.

Nachdem Sie einen Cluster angehängt haben, beenden Sie die Erstellung des Workspace. Wenn Sie den neuen Workspace zum ersten Mal öffnen und das EMRCluster-Panel auswählen, sollte Ihr ausgewählter Cluster als Anhang angezeigt werden.

On launch

Stellen Sie eine Verbindung zu einem EMR Amazon-Compute-Cluster her, wenn Sie den Workspace starten

1. Navigieren Sie zur Workspaces-Liste und wählen Sie die Zeile für den Workspace aus, den Sie starten möchten. Wählen Sie dann Workspace starten > Mit Optionen starten aus.
2. Wählen Sie einen EMR Cluster aus, der an Ihren Workspace angehängt werden soll.

Nachdem Sie einen Cluster angehängt haben, beenden Sie die Erstellung des Workspace. Wenn Sie den neuen Workspace zum ersten Mal öffnen und das EMRCluster-Panel auswählen, sollte Ihr ausgewählter Cluster als Anhang angezeigt werden.

In JupyterLab

Einen Workspace an einen EMR Amazon-Compute-Cluster anhängen in JupyterLab

1. Wählen Sie Ihren Workspace und dann Workspace starten > Schnellstart.
2. Öffnen Sie im Inneren JupyterLab die Registerkarte Cluster in der linken Seitenleiste.
3. Wählen Sie das Drop-down-Menü „EMROn EC2 Cluster“ oder wählen Sie ein EMR EKS Amazon-On-Cluster aus.
4. Wählen Sie Anfügen, um den Cluster an Ihren Workspace anzufügen.

Nachdem Sie einen Cluster angehängt haben, beenden Sie die Erstellung des Workspace. Wenn Sie den neuen Workspace zum ersten Mal öffnen und das EMRCluster-Panel auswählen, sollte Ihr ausgewählter Cluster als Anhang angezeigt werden.

In the Workspace UI

Hängen Sie über die Workspace-Benutzeroberfläche einen Workspace an einen EMR Amazon-Compute-Cluster an

1. Wählen Sie in dem Workspace, den Sie an einen Cluster anhängen möchten, das EMR Cluster-Symbol in der linken Seitenleiste, um das Cluster-Panel zu öffnen.
2. Erweitern Sie unter Clustertyp das Drop-down-Menü und wählen Sie EMRCluster on aus. EC2
3. Wählen Sie Cluster in der Dropdown-Liste aus. Möglicherweise müssen Sie zuerst einen vorhandenen Cluster trennen, um die Dropdownliste für die Clusterauswahl zu aktivieren.
4. Wählen Sie Anfügen aus. Wenn der Cluster angehängt ist, sollte eine Erfolgsmeldung angezeigt werden.

Hängen Sie einen EMR Amazon EKS On-Cluster an einen EMR Studio-Arbeitsbereich an

Zusätzlich zur Verwendung von EMR Amazon-Clustern, die auf Amazon ausgeführt werdenEC2, können Sie einen Workspace an einen EMR EKS Amazon-On-Cluster anhängen, um Notebook-Code auszuführen. Weitere Informationen zu Amazon EMR on EKS finden Sie unter Was ist Amazon EMR on EKS.

Bevor Sie einen Workspace mit einem EMR Amazon EKS On-Cluster verbinden können, muss Ihnen Ihr Studio-Administrator Zugriffsberechtigungen erteilen.

> **Note**
>
> Sie können kein EMR EKS Amazon-On-Cluster in einem EMR Studio starten, das IAM Identity Center Trusted Identity Propagation verwendet.

On create

So fügen Sie beim Erstellen eines Workspace einen EMR EKS Amazon-On-Cluster hinzu

1. Erweitern Sie im Dialogfeld Workspace erstellen den Abschnitt Erweiterte Konfiguration.
2. Wählen Sie Workspace an einen EMR EKS Amazon-On-Cluster anhängen.
3. Wählen Sie unter Amazon EMR on EKS cluster einen Cluster aus der Drop-down-Liste aus.
4. Wählen Sie unter Endpunkt auswählen einen verwalteten Endpunkt aus, der an den Workspace angefügt werden soll. Ein verwalteter Endpunkt ist ein Gateway, über das EMR Studio mit dem von Ihnen ausgewählten Cluster kommunizieren kann.
5. Wählen Sie Workspace erstellen aus, um den Workspace-Erstellungsprozess abzuschließen und den ausgewählten Cluster anzuhängen.

Nachdem Sie einen Cluster angehängt haben, können Sie den Workspace-Erstellungsprozess abschließen. Wenn Sie den neuen Workspace zum ersten Mal öffnen und das EMRCluster-Panel auswählen, sollten Sie sehen, dass Ihr ausgewählter Cluster angehängt ist.

In the Workspace UI

So hängen Sie EMR über die Workspace-Benutzeroberfläche einen Amazon EKS On-Cluster an

1. Wählen Sie in dem Workspace, den Sie einem Cluster zuordnen möchten, das Cluster-Symbol in der EMR linken Seitenleiste, um das Cluster-Panel zu öffnen.
2. Erweitern Sie das Drop-down-Menü Clustertyp und wählen Sie EMRCluster on aus. EKS
3. Wählen Sie unter EMRCluster on EKS einen Cluster aus der Dropdownliste aus.
4. Wählen Sie unter Endpunkt einen verwalteten Endpunkt aus, der an den Workspace angehängt werden soll. Ein verwalteter Endpunkt ist ein Gateway, über das EMR Studio mit dem von Ihnen ausgewählten Cluster kommunizieren kann.

5. Wählen Sie Anfügen aus. Wenn der Cluster angehängt ist, sollte eine Erfolgsmeldung angezeigt werden.

Eine Amazon EMR Serverless-Anwendung an einen EMR Studio-Arbeitsbereich anhängen

Sie können einen Workspace an eine EMR serverlose Anwendung anhängen, um interaktive Workloads auszuführen. Weitere Informationen finden Sie unter Verwenden von Notebooks zum Ausführen interaktiver Workloads mit EMR Serverless über Studio. EMR

> **Note**
>
> Sie können eine EMR serverlose Anwendung nicht an ein EMR Studio anhängen, das IAM Identity Center Trusted Identity Propagation verwendet.

Example Hängen Sie einen Workspace an eine EMR serverlose Anwendung an in JupyterLab

Bevor Sie einen Workspace mit einer EMR serverlosen Anwendung verbinden können, muss Ihnen Ihr Kontoadministrator Zugriffsberechtigungen erteilen, wie unter Erforderliche Berechtigungen für interaktive Workloads beschrieben.

1. Navigieren Sie zu EMR Studio, wählen Sie Ihren Workspace aus und wählen Sie dann Workspace starten > Schnellstart aus.
2. Öffnen Sie dort in der linken Seitenleiste den Tab Cluster. JupyterLab
3. Wählen Sie EMRServerless als Rechenoption aus und wählen Sie dann eine EMR serverlose Anwendung und eine Runtime-Rolle aus.
4. Wählen Sie Anfügen, um den Cluster an Ihren Workspace anzufügen.

Wenn Sie jetzt diesen Workspace öffnen, sollten Sie sehen, dass Ihre ausgewählte Anwendung angefügt ist.

Erstellen Sie einen neuen EMR Cluster und fügen Sie ihn einem Studio-Workspace hinzu EMR

Fortgeschrittene EMR Studio-Benutzer können neue EMR Cluster bereitstellen, die auf Amazon laufenEC2, um sie mit einem Workspace zu verwenden. Auf dem neuen Cluster sind standardmäßig alle Big-Data-Anwendungen installiert, die für EMR Studio erforderlich sind.

Um Cluster zu erstellen, muss Ihnen Ihr Studio-Administrator zunächst mithilfe einer Sitzungsrichtlinie die Erlaubnis erteilen. Weitere Informationen finden Sie unter Erstellen Sie Berechtigungsrichtlinien für EMR Studio-Benutzer.

Sie können einen neuen Cluster im Dialogfeld Workspace erstellen oder im Bereich Cluster in der Workspace-Benutzeroberfläche erstellen. In beiden Fällen haben Sie zwei Möglichkeiten zum Erstellen eines Clusters:

1. EMRCluster erstellen — Erstellen Sie einen EMR Cluster, indem Sie den EC2 Amazon-Instance-Typ und die Anzahl auswählen.
2. Eine Cluster-Vorlage verwenden – Stellen Sie einen Cluster bereit, indem Sie eine vordefinierte Cluster-Vorlage auswählen. Diese Option wird angezeigt, wenn Sie berechtigt sind, Clustervorlagen zu verwenden.

> **Note**
>
> Wenn Sie die Verbreitung vertrauenswürdiger Identitäten mit IAM Identity Center für Ihr Studio aktiviert haben, müssen Sie eine Vorlage verwenden, um einen Cluster zu erstellen.

Um einen EMR Cluster zu erstellen, indem Sie eine Clusterkonfiguration angeben

1. Wählen Sie einen Startpunkt aus.

Zu …	Vorgehensweise
Erstellen Sie den Cluster, wenn Sie einen Workspace mit dem Dialogfeld Workspace erstellen.	Erweitern Sie im Dialogfeld Einen Workspace erstellen den Abschnitt Erweiterte Konfiguration und wählen Sie EMRCluster erstellen aus.

Zu …	Vorgehensweise
Erstellen Sie den Cluster über das EMRCluster-Panel in der Workspace-Benutzeroberfläche, nachdem Sie einen Workspace erstellt haben.	Wählen Sie in der linken Seitenleiste eines geöffneten Workspace die Registerkarte EMRCluster, erweitern Sie den Abschnitt Erweiterte Konfiguration und wählen Sie Cluster erstellen aus.

2. Geben Sie einen Clusternamen ein. Wenn Sie den Cluster benennen, können Sie ihn später in der EMR Studio-Cluster-Liste leichter finden.
3. Wählen Sie für die EMRAmazon-Version eine EMR Amazon-Release-Version für den Cluster aus.
4. Wählen Sie unter Instance den Typ und die Anzahl der EC2 Amazon-Instances für den Cluster aus. Weitere Informationen zur Auswahl von Instance-Typen finden Sie unter EC2Amazon-Instances konfigurieren. Genau eine Instance wird als Primärknoten verwendet.
5. Wählen Sie ein Subnetz aus, in dem EMR Studio den neuen Cluster starten kann. Jede Subnetzoption wurde von Ihrem Studio-Administrator vorab genehmigt, und Ihr Workspace sollte in der Lage sein, eine Verbindung zu einem Cluster in einem beliebigen aufgelisteten Subnetz herzustellen.
6. Wählen Sie ein S3 URI für die Protokollspeicherung.
7. Wählen Sie Create EMR Cluster aus, um den Cluster bereitzustellen. Wenn Sie das Dialogfeld Workspace erstellen verwenden, wählen Sie Workspace erstellen aus, um den Workspace zu erstellen und den Cluster bereitzustellen. Nachdem EMR Studio den neuen Cluster bereitgestellt hat, wird der Cluster an den Workspace angehängt.

So erstellen Sie einen Cluster mit einer Cluster-Vorlage

1. Wählen Sie einen Startpunkt aus.

Zu …	Vorgehensweise
Erstellen Sie den Cluster, wenn Sie einen Workspace mit dem Dialogfeld Workspace erstellen.	Erweitern Sie den Abschnitt Erweiterte Konfiguration im Dialogfeld Workspace erstellen und wählen Sie Cluster-Vorlage verwenden aus.

Zu …	Vorgehensweise
Erstellen Sie den Cluster über das EMRCluster-Panel in der Workspace-Benutzeroberfläche.	Wählen Sie in der linken Seitenleiste eines geöffneten Workspace die Registerkarte EMRCluster, erweitern Sie den Abschnitt Erweiterte Konfiguration und wählen Sie dann Cluster-Vorlage aus.

2. Wählen Sie eine Cluster-Vorlage aus der Dropdown-Liste aus. Jede verfügbare Clustervorlage enthält eine kurze Beschreibung, die Ihnen bei der Auswahl hilft.
3. Die von Ihnen gewählte Cluster-Vorlage kann zusätzliche Parameter wie EMR Amazon-Release-Version oder Clustername enthalten. Sie können Werte auswählen oder einfügen oder die Standardwerte verwenden, die Ihr Administrator ausgewählt hat.
4. Wählen Sie ein Subnetz aus, in dem EMR Studio den neuen Cluster starten kann. Jede Subnetzoption wurde von Ihrem Studio-Administrator vorab genehmigt, und Ihr Workspace sollte in der Lage sein, eine Verbindung zu einem Cluster in einem beliebigen Subnetz herzustellen.
5. Wählen Sie Clustervorlage verwenden, um den Cluster bereitzustellen und an den Workspace anzuhängen. Es dauert einige Minuten, bis EMR Studio den Cluster erstellt hat. Wenn Sie das Dialogfeld Workspace erstellen verwenden, wählen Sie Workspace erstellen aus, um den Workspace zu erstellen und den Cluster bereitzustellen. Nachdem EMR Studio den neuen Cluster bereitgestellt hat, wird der Cluster an Ihren Workspace angehängt.

Trennen Sie einen Computer von einem Studio-Workspace EMR

Um den mit einem Workspace verbundenen Cluster auszutauschen, können Sie einen Cluster von der Workspace-Benutzeroberfläche trennen.

So trennen Sie einen Cluster von einem Workspace

1. Wählen Sie in dem Workspace, den Sie von einem Cluster trennen möchten, das Cluster-Symbol in der linken Seitenleiste, um das EMRCluster-Bedienfeld zu öffnen.
2. Wählen Sie unter Cluster auswählen die Option Trennen aus und warten Sie, bis EMR Studio den Cluster getrennt hat. Wenn der Cluster getrennt ist, sehen Sie eine Erfolgsmeldung.

Um eine EMR serverlose Anwendung von einem Studio-Arbeitsbereich zu trennen EMR

Um den mit einem Workspace verbundenen Compute auszutauschen, können Sie eine Anwendung von der Workspace-Benutzeroberfläche trennen.

1. Wählen Sie in dem Workspace, den Sie von einem Cluster trennen möchten, das EMRAmazon-Compute-Symbol in der linken Seitenleiste, um das Compute-Panel zu öffnen.
2. Wählen Sie unter „Datenverarbeitung auswählen“ die Option Trennen aus und warten Sie, bis EMR Studio die Anwendung getrennt hat. Wenn die Anwendung getrennt ist, sehen Sie eine Erfolgsmeldung.

Git-basierte Repositorys mit einem EMR Studio-Workspace verknüpfen

Über Git-Repositorys für Studio EMR

Sie können einem EMR Studio-Workspace maximal drei Git-Repositorys zuordnen. Standardmäßig können Sie in jedem Workspace aus einer Liste von Git-Repositorys wählen, die demselben AWS Konto wie das Studio zugeordnet sind. Sie können auch ein neues Git-Repository als Ressource für einen Workspace erstellen.

Sie können Git-Befehle wie die folgenden mit einem Terminalbefehl ausführen, während Sie mit dem Primärknoten eines Clusters verbunden sind.

```
!git pull origin <branch-name>
```

Sie können aber auch die jupyterlab-git-Erweiterung verwenden. Öffnen Sie es in der linken Seitenleiste, indem Sie das Git-Symbol auswählen. Informationen zur Jupyterlab-Git-Erweiterung für finden Sie unter jupyterlab-git. JupyterLab

Voraussetzungen

- Um ein Git-Repository mit einem Workspace zu verknüpfen, muss Ihr Studio so konfiguriert sein, dass die Verknüpfung mit Git-Repositorys zulässig ist. Ihr Studio-Administrator sollte folgende Schritte unternehmen, um Zugriff und Berechtigungen für Git-basierte Repositorys einrichten.
- Wenn Sie ein CodeCommit Repository verwenden, müssen Sie Git-Anmeldeinformationen und verwendenHTTPS. SSHSchlüssel und HTTPS mit dem AWS Command Line Interface Credential Helper werden nicht unterstützt. CodeCommit unterstützt auch keine persönlichen Zugriffstoken (PATs). Weitere Informationen finden Sie unter Using IAM with CodeCommit im

IAMBenutzerhandbuch und Setup für HTTPS Benutzer, die Git-Anmeldeinformationen verwenden, im AWS CodeCommit User Guide.

Anweisungen

So verknüpfen Sie ein zugeordnetes Git-Repository mit einem Workspace

1. Öffnen Sie den Workspace, den Sie mit einem Repository verknüpfen möchten, in der Workspaces-Liste im Studio.
2. Wählen Sie in der linken Seitenleiste das Amazon EMR Git Repository-Symbol, um das Git-Repository-Toolpanel zu öffnen.
3. Erweitern Sie unter Git-Repositorys die Drop-down-Liste und wählen Sie maximal drei Repositorys aus, die mit dem Workspace verknüpft werden sollen. EMRStudio registriert Ihre Auswahl und beginnt mit der Verknüpfung der einzelnen Repositorys.

Es kann einige Zeit dauern, bis der Verbindungsvorgang abgeschlossen ist. Sie können den Status für jedes Repository sehen, das Sie im Git-Repository-Toolpanel ausgewählt haben. Nachdem EMR Studio ein Repository mit einem Workspace verknüpft hat, sollten die Dateien, die zu diesem Repository gehören, im Dateibrowser-Bereich angezeigt werden.

Um einem Workspace ein neues Git-Repository als Ressource hinzuzufügen

1. Öffnen Sie den Workspace, den Sie mit einem Repository verknüpfen möchten, in der Workspaces-Liste im Studio.
2. Wählen Sie in der linken Seitenleiste das Amazon EMR Git Repository-Symbol, um das Git-Repository-Toolpanel zu öffnen.
3. Wählen Sie Neues Git-Repository hinzufügen.
4. Geben Sie unter Repository-Name einen aussagekräftigen Namen für das Repository in EMR Studio ein. Namen dürfen nur alphanumerische Zeichen, Bindestriche oder Unterstriche enthalten.
5. Geben Sie für URLGit-Repository den URL für das Repository ein. Wenn Sie ein CodeCommit Repository verwenden, wird dieses Repository kopiertURL, wenn Sie „Klonen“ URL und dann „Klonen“ wählenHTTPS. Beispiel, `https://git-codecommit.us-west-2.amazonaws.com/v1/repos/[MyCodeCommitRepoName]`.
6. Geben Sie für Branch den Namen eines vorhandenen Branches ein, den Sie auschecken möchten.

7. Wählen Sie Optionen für Git-Anmeldeinformationen gemäß den folgenden Richtlinien. EMRStudio greift mithilfe von Geheimnissen, die im Secrets Manager gespeichert sind, auf Ihre Git-Anmeldeinformationen zu.

> **Note**
>
> Wenn Sie ein GitHub Repository verwenden, empfehlen wir Ihnen, zur Authentifizierung ein persönliches Zugriffstoken (PAT) zu verwenden. Ab dem 13. August 2021 ist eine tokenbasierte Authentifizierung erforderlich und bei der Authentifizierung von Git-Vorgängen werden keine Passwörter mehr akzeptiert. GitHub Weitere Informationen finden Sie im Beitrag Token-Authentifizierungsanforderungen für Git-Operationen im GitHub Blog.

Option	Beschreibung
Erstellen eines neuen Secrets	Wählen Sie diese Option, um bestehende Git-Anmeldeinformationen mit einem neuen Geheimnis zu verknüpfen, das AWS Secrets Manager für Sie erstellt wird. Führen Sie basierend auf den Git-Anmeldeinformationen, die Sie für das Repository verwenden, einen der folgenden Schritte aus. Wenn Sie für den Zugriff auf das Repository einen Git-Benutzernamen mit Passwort verwenden, wählen Sie Benutzername und Passwort aus, geben Sie den Namen des Secrets ein, das in Secrets Manager verwendet werden soll, und geben Sie dann den Benutzernamen und das Passwort ein, die mit dem Secret verknüpft werden sollen. -ODER- Wenn Sie ein persönliches Zugriffstoken für den Zugriff auf das Repository verwenden, wählen Sie Persönliches Zugriffstoken (PAT) aus, geben Sie den geheimen Namen ein, der in Secrets Manager verwendet werden soll, und geben Sie dann Ihr persönliches Zugriffstoken ein. Weitere Informationen findest du unter Persönliches Zugriffstoken für die Befehlszeile erstellen für GitHub und Persönliche Zugriffstoken für Bitbucket. CodeCommit Repositorys unterstützen diese Option nicht.
Verwenden eines öffentlichen Repository ohne Anmeldeinformationen	Wählen Sie diese Option, um auf ein öffentliches Repository zuzugreifen.

Option	Beschreibung
Verwenden Sie ein vorhandenes Geheimnis AWS	Wählen Sie diese Option, wenn Sie Ihre Anmeldeinformationen bereits als Secret in Secrets Manager gespeichert haben, und wählen Sie dann den Namen des Secrets in der Liste aus. Wenn Sie ein Secret auswählen, das mit einem Git-Benutzernamen und -Passwort verknüpft ist, muss das Secret das Format `{"gitUsername": " MyUserName ", "gitPassword": " MyPassword "}` aufweisen.

8. Wählen Sie Repository hinzufügen, um das neue Repository zu erstellen. Nachdem EMR Studio das neue Repository erstellt hat, wird eine Erfolgsmeldung angezeigt. Das neue Repository erscheint in der Dropdown-Liste unter Git-Repositorys.
9. Um das neue Repository mit deinem Workspace zu verknüpfen, wähle es aus der Drop-down-Liste unter Git-Repositorys aus.

Es kann einige Zeit dauern, bis der Verbindungsvorgang abgeschlossen ist. Nachdem EMR Studio das neue Repository mit dem Workspace verknüpft hat, sollte im Dateibrowser-Bereich ein neuer Ordner mit demselben Namen wie Ihr Repository angezeigt werden.

Um ein anderes verknüpftes Repository zu öffnen, navigieren Sie im Dateibrowser zu seinem Ordner.

Verwenden Sie den Amazon Athena SQL Athena-Editor in Studio EMR

Übersicht

Sie können Amazon EMR Studio verwenden, um interaktive Abfragen auf Amazon Athena zu entwickeln und auszuführen. Das bedeutet, dass Sie SQL Analysen auf Athena von derselben EMR Studio-Oberfläche aus durchführen können, mit der Sie Ihre Spark-, Scala- und andere Workloads ausführen. Mit dieser Integration können Sie die automatische Vervollständigung verwenden, um schnell Abfragen zu entwickeln, Daten in Ihrem AWS Glue-Datenkatalog zu durchsuchen, gespeicherte Abfragen zu erstellen, Ihren Abfrageverlauf einzusehen und vieles mehr.

Weitere Informationen zur Verwendung von Amazon Athena finden Sie unter Using Athena SQL im Amazon Athena Athena-Benutzerhandbuch.

Verwenden Sie den SQL Athena-Editor in Studio EMR

Gehen Sie wie folgt vor, um interaktive Abfragen auf Amazon Athena von Ihrem EMR Studio aus zu entwickeln und auszuführen:

1. Fügen Sie der Benutzerrolle die erforderlichen Berechtigungen für die Benutzer hinzu, die auf die Workspaces in diesem Studio zugreifen. Die Berechtigungen sind in der AWS Identity and Access Management Berechtigungen für EMR Studio-Benutzer Tabelle in der Spalte Zugriff auf den Amazon Athena SQL Athena-Editor von Ihrem EMR Studio aus aufgeführt. Alternativ können Sie sich dafür entscheiden, den Inhalt der erweiterten Richtlinie aus dem zu kopierenBeispielbenutzerrichtlinien, um Benutzern vollständige Berechtigungen für EMR Studio-Funktionen, einschließlich dieser, zu gewähren.
2. Richten Sie ein EMR Studio ein und erstellen Sie es.
3. Navigieren Sie zu Ihrem Studio und wählen Sie in der Seitenleiste den Query Editor aus.

Sie sollten jetzt die bekannte Athena-Editor-Benutzeroberfläche sehen. Informationen zu den ersten Schritten und zur Verwendung von Athena SQL zur Ausführung interaktiver Abfragen finden Sie unter Erste Schritte und Verwendung von Athena SQL im Amazon Athena Athena-Benutzerhandbuch.

> **Note**
>
> Wenn Sie die Verbreitung vertrauenswürdiger Identitäten über IAM Identity Center für Ihr EMR Studio aktiviert haben, müssen Sie Athena-Arbeitsgruppen verwenden, um den Abfragezugriff zu steuern, und die Arbeitsgruppe, die Sie verwenden, muss auch die vertrauenswürdige Identitätsverbreitung verwenden. Schritte zur Einrichtung von Identity Center und zur Aktivierung der Verbreitung vertrauenswürdiger Identitäten für Ihre Arbeitsgruppe finden Sie unter Verwenden von IAM Identity Center-fähigen Athena-Arbeitsgruppen im Amazon Athena Athena-Benutzerhandbuch.

Überlegungen zur Verwendung des SQL Athena-Editors in Studio EMR

- Die Integration mit Athena ist in allen kommerziellen Regionen verfügbar, in denen EMR Studio und Athena verfügbar sind.

- Die folgenden Athena-Funktionen sind in EMR Studio nicht verfügbar:
  - Admin-Features wie das Erstellen oder Aktualisieren von Athena-Arbeitsgruppen, Datenquellen oder Kapazitätsreservierungen
  - Athena-for-Spark- oder Spark-Notebooks
  - DataZone Amazon-Integration
  - Kostenbasierter Optimierer () CBO
  - Step Functions

CodeWhisperer Amazon-Integration mit EMR Studio Workspaces

Übersicht

Sie können Amazon CodeWhisperer mit Amazon EMR Studio verwenden, um Empfehlungen in Echtzeit zu erhalten, während Sie Code einschreiben JupyterLab. CodeWhispererkann Ihre Kommentare vervollständigen, einzelne Codezeilen fertigstellen, line-by-line Empfehlungen aussprechen und vollständig formatierte Funktionen generieren.

> **Note**
>
> Wenn Sie Amazon EMR Studio verwenden, werden AWS möglicherweise Daten über Ihre Nutzung und Inhalte gespeichert, um den Service zu verbessern. Weitere Informationen und Anweisungen zum Deaktivieren der Datenweitergabe finden Sie unter Teilen Ihrer Daten mit AWS im CodeWhisperer Amazon-Benutzerhandbuch.

Überlegungen zur Verwendung CodeWhisperer mit Workspaces

- CodeWhisperer Die Integration ist genauso AWS-Regionen verfügbar, wie EMR Studio verfügbar ist, wie in den Überlegungen zu EMR Studio dokumentiert.
- Amazon EMR Studio verwendet automatisch den CodeWhisperer Endpunkt in USA Ost (Nord-Virginia) (us-east-1) für Empfehlungen, unabhängig von der Region, in der sich Ihr Studio befindet.
- CodeWhisperer unterstützt nur die Python-Sprache für die Codierung von ETL Skripten für Spark-Jobs in EMR Studio.
- Eine clientseitige Telemetrieoption quantifiziert Ihre Nutzung von. CodeWhisperer Diese Funktionalität wird von Studio nicht unterstützt. EMR

Berechtigungen erforderlich für CodeWhisperer

Um sie verwenden zu können CodeWhisperer, müssen Sie Ihrer IAM Benutzerrolle für Amazon EMR Studio die folgende Richtlinie hinzufügen:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
          "Sid": "CodeWhispererPermissions",
          "Effect": "Allow",
          "Action": [ "codewhisperer:GenerateRecommendations" ],
          "Resource": "*"
        }
     ]
}
```

CodeWhisperer Mit Workspaces verwenden

Um das CodeWhisperer Referenzprotokoll anzuzeigen JupyterLab, öffnen Sie das CodeWhispererFenster am unteren Rand des JupyterLab Fensters und wählen Sie „Code-Referenzprotokoll öffnen“.

Die folgende Liste enthält Tastenkombinationen, mit denen Sie mit CodeWhisperer Vorschlägen interagieren können:

- Empfehlungen pausieren — Verwende die Option Automatische Vorschläge pausieren in den CodeWhisperer Einstellungen.
- Eine Empfehlung akzeptieren – Drücken Sie die Tabulatortaste auf Ihrer Tastatur.
- Eine Empfehlung ablehnen – Drücken Sie die Escape-Taste auf Ihrer Tastatur.
- Empfehlungen durchsuchen – Verwenden Sie die Aufwärts- und Abwärtspfeile auf Ihrer Tastatur.
- Manueller Aufruf – Drücken Sie die Tasten Alt und C auf Ihrer Tastatur. Wenn Sie einen Mac verwenden, drücken Sie Cmd und C.

Sie können sie auch verwenden CodeWhisperer , um Einstellungen wie die Protokollebene zu ändern und Vorschläge für Codereferenzen zu erhalten. Weitere Informationen finden Sie unter Einrichtung CodeWhisperer mit JupyterLab und Funktionen im CodeWhisperer Amazon-Benutzerhandbuch.

Debuggen Sie Anwendungen und Jobs mit Studio EMR

Mit Amazon EMR Studio können Sie Datenanwendungsschnittstellen starten, um Anwendungen und Jobausführungen im Browser zu analysieren.

Sie können die persistenten Benutzeroberflächen außerhalb des Clusters für Amazon, die auf EC2 Clustern EMR ausgeführt werden, auch von der EMR Amazon-Konsole aus starten. Weitere Informationen finden Sie unter Persistente Anwendungsbenutzeroberflächen anzeigen.

> **Note**
>
> Abhängig von Ihren Browsereinstellungen müssen Sie möglicherweise Popups aktivieren, damit die Benutzeroberfläche einer Anwendung geöffnet werden kann.

Informationen zur Konfiguration und Verwendung der Anwendungsschnittstellen finden Sie unter The YARN Timeline Server, Monitoring and Instrumentation oder Tez UI Overview.

Debuggen Sie Amazon, EMR das auf EC2 Amazon-Jobs ausgeführt wird

Workspace UI

Starten Sie eine Cluster-Benutzeroberfläche aus einer Notebook-Datei

Wenn Sie die EMR Amazon-Release-Versionen 5.33.0 und höher verwenden, können Sie die Spark-Webbenutzeroberfläche (die Spark-Benutzeroberfläche oder den Spark History Server) von einem Notizbuch in Ihrem Workspace aus starten.

UIsArbeiten Sie im Cluster mit den PySpark Kernen, Spark oder SparkR. Die maximale sichtbare Dateigröße für Spark-Event- oder Container-Logs beträgt 10 MB. Wenn Ihre Protokolldateien 10 MB überschreiten, empfehlen wir Ihnen, zum Debuggen von Jobs den persistenten Spark History Server anstelle der Cluster-internen Spark-Benutzeroberfläche zu verwenden.

> **Important**
>
> Damit EMR Studio Benutzeroberflächen für Cluster-Anwendungen von einem Workspace aus starten kann, muss ein Cluster in der Lage sein, mit dem Amazon API Gateway zu kommunizieren. Sie müssen den EMR Cluster so konfigurieren, dass ausgehender Netzwerkverkehr zu Amazon API Gateway zugelassen wird, und sicherstellen, dass Amazon API Gateway vom Cluster aus erreichbar ist.

Die Spark-Benutzeroberfläche greift auf Container-Logs zu, indem sie Hostnamen auflöst. Wenn Sie einen benutzerdefinierten Domainnamen verwenden, müssen Sie sicherstellen, dass die Hostnamen Ihrer Clusterknoten von Amazon DNS oder dem von Ihnen angegebenen DNS Server aufgelöst werden können. Stellen Sie dazu die Dynamic Host Configuration Protocol (DHCP) -Optionen für die Amazon Virtual Private Cloud (VPC) ein, die Ihrem Cluster zugeordnet ist. Weitere Informationen zu DHCP Optionen finden Sie unter DHCPOptionssätze im Amazon Virtual Private Cloud Cloud-Benutzerhandbuch.

1. Öffnen Sie in Ihrem EMR Studio den Workspace, den Sie verwenden möchten, und stellen Sie sicher, dass er mit einem EMR Amazon-Cluster verbunden ist, auf dem ausgeführt wirdEC2. Detaillierte Anweisungen finden Sie unter Hängen Sie einen Computer an einen EMR Studio-Workspace an.
2. Öffnen Sie eine Notebook-Datei und verwenden Sie den PySpark, Spark- oder SparkR-Kernel. Um einen Kernel auszuwählen, wählen Sie den Kernel-Namen oben rechts in der Notebook-Symbolleiste, um das Dialogfeld Kernel auswählen zu öffnen. Der Name erscheint als Kein Kernel! wenn kein Kernel ausgewählt wurde.
3. Führen Sie Ihren Notebook-Code aus. Folgendes wird als Ausgabe im Notebook angezeigt, wenn Sie den Spark-Kontext starten. Es kann einige Sekunden dauern, bis es angezeigt wird. Wenn Sie den Spark-Kontext gestartet haben, können Sie den `%%info`-Befehl ausführen, um jederzeit auf einen Link zur Spark-Benutzeroberfläche zuzugreifen.

   > **Note**
   >
   > Wenn die Spark-UI-Links nicht funktionieren oder nach einigen Sekunden nicht angezeigt werden, erstellen Sie eine neue Notebook-Zelle und führen Sie den `%%info`-Befehl aus, um die Links neu zu generieren.

4. Um die Spark-Benutzeroberfläche zu starten, wählen Sie Link unter Spark-Benutzeroberfläche. Wenn Ihre Spark-Anwendung ausgeführt wird, wird die Spark-Benutzeroberfläche in einer neuen Registerkarte geöffnet. Wenn die Anwendung abgeschlossen ist, wird stattdessen der Spark History Server geöffnet.

   Nachdem Sie die Spark-Benutzeroberfläche gestartet haben, können Sie die URL im Browser ändern, um den YARN ResourceManager oder den Yarn Timeline Server zu öffnen. Fügen Sie nach `amazonaws.com` einen der folgenden Pfade hinzu.

Web-Benutzeroberfläche	Pfad	Beispiel geändert URL
YARN ResourceManager	/rm	https://*j-examplebby5ij*.emrappui-prod.*eu-west-1*.amazonaws.com/*rm*
Yarn-Timeline-Server	/yts	https://*j-examplebby5ij*.emrappui-prod.*eu-west-1*.amazonaws.com/*yts*
Spark History Server	/shs	https://*j-examplebby5ij*.emrappui-prod.*eu-west-1*.amazonaws.com/*shs*

Studio UI

Starten Sie den persistenten YARN Timeline Server, den Spark History Server oder die Tez-Benutzeroberfläche über die Studio-Benutzeroberfläche EMR

1. Wählen Sie in Ihrem EMR Studio EMREC2auf der linken Seite Amazon aus, um die Liste Amazon EMR on EC2 Clusters zu öffnen.
2. Filtern Sie die Clusterliste nach Name, Status oder ID, indem Sie Werte in das Suchfeld eingeben. Sie können auch nach Erstellungszeitraum suchen.
3. Wählen Sie einen Cluster aus und wählen Sie dann Anwendung starten UIs, um eine Anwendungsbenutzeroberfläche auszuwählen. Die Anwendungsbenutzeroberfläche wird in einer neuen Browser-Registerkarte geöffnet. Es kann einige Zeit dauern, bis sie geladen wird.

Debug EMR Studio läuft auf Serverless EMR

Ähnlich wie Amazon, EMR das auf Amazon läuftEC2, können Sie die Workspace-Benutzeroberfläche verwenden, um Ihre EMR serverlosen Anwendungen zu analysieren. Wenn Sie EMR Amazon-Versionen 6.14.0 und höher verwenden, können Sie von der Workspace-Benutzeroberfläche aus die Spark-Webbenutzeroberfläche (die Spark-Benutzeroberfläche oder den Spark History Server) von einem Notizbuch in Ihrem Workspace aus starten. Der Einfachheit halber stellen wir auch einen Link zum Treiberprotokoll zur Verfügung, über den Sie schnell auf die Spark-Treiberprotokolle zugreifen können.

Debuggen Sie Amazon EMR bei EKS Jobläufen mit dem Spark History Server

Wenn Sie einen Job Run an einen Amazon EKS On-Cluster senden, können Sie EMR über den Spark History Server auf die Logs für diesen Job zugreifen. Der Spark History Server bietet Tools für die Überwachung von Spark-Anwendungen, z. B. eine Liste von Scheduler-Phasen und -aufgaben, eine Zusammenfassung der RDD Größen und Speicherbelegung sowie Umgebungsinformationen. Sie können den Spark History Server for Amazon bei EKS Jobläufen EMR auf folgende Weise starten:

- Wenn Sie einen Job einreichen, der mit EMR Studio und einem EKS verwalteten EMR Amazon-Endpunkt ausgeführt wird, können Sie den Spark History Server von einer Notebook-Datei in Ihrem Workspace aus starten.
- Wenn Sie einen Job einreichen, der mit AWS CLI oder AWS SDK für EMR Amazon ausgeführt wirdEKS, können Sie den Spark History Server über die EMR Studio-Benutzeroberfläche starten.

Informationen zur Verwendung des Spark History Servers finden Sie unter Überwachung und Instrumentierung in der Apache-Spark-Dokumentation. Weitere Informationen zu Jobausführungen finden Sie unter Konzepte und Komponenten im Amazon EMR on EKS Development Guide.

So starten Sie den Spark History Server aus einer Notebook-Datei in Ihrem EMR Studio-Arbeitsbereich

1. Öffnen Sie einen Workspace, der mit einem EMR Amazon EKS On-Cluster verbunden ist.
2. Wählen Sie Ihre Notebook-Datei aus und öffnen Sie sie im Workspace.
3. Wählen Sie oben in der Notebook-Datei die Spark-Benutzeroberfläche, um den persistenten Spark-Geschichtsserver in einer neuen Registerkarte zu öffnen.

Um den Spark History Server über die EMR Studio-Benutzeroberfläche zu starten

> **Note**
>
> In der Jobliste in der EMR Studio-Benutzeroberfläche werden nur Jobausführungen angezeigt, die Sie mit AWS CLI oder AWS SDK für Amazon EMR am einreichenEKS.

1. Wählen Sie in Ihrem EMR Studio links EMREKSauf der Seite Amazon aus.
2. Suchen Sie nach dem EKS virtuellen EMR Amazon-Cluster, mit dem Sie Ihren Joblauf eingereicht haben. Sie können die Liste der Cluster nach Status oder ID filtern, indem Sie Werte in das Suchfeld eingeben.
3. Wählen Sie den Cluster aus, um seine Detailseite zu öffnen. Auf der Detailseite werden Informationen über den Cluster wie ID, Namespace und Status angezeigt. Auf der Seite wird auch eine Liste aller Auftragsausführungen angezeigt, die an diesen Cluster übermittelt wurden.
4. Wählen Sie auf der Cluster-Detailseite einen Auftrag aus, der debuggt werden soll.
5. Wählen Sie oben rechts in der Auftragsliste die Option Spark History Server starten, um die Anwendungsoberfläche in einer neuen Browser-Registerkarte zu öffnen.

Installieren Sie Kernel und Bibliotheken in einem EMR Studio-Arbeitsbereich

Jeder Amazon EMR Studio Workspace wird mit einer Reihe vorinstallierter Bibliotheken und Kernel geliefert.

Kernel und Bibliotheken auf Clustern, die auf Amazon laufen EC2

Sie können die Umgebung für EMR Studio auch auf folgende Weise anpassen, wenn Sie EMR Cluster verwenden, die auf Amazon ausgeführt werdenEC2:

- Jupyter-Notebook-Kernel und Python-Bibliotheken auf einem Cluster-Primärknoten installieren – Wenn Sie Bibliotheken mit dieser Option installieren, teilen sich alle Workspaces, die demselben Cluster zugeordnet sind, diese Bibliotheken gemeinsam. Sie können Kernel oder Bibliotheken von einer Notebook-Zelle aus installieren oder während Sie mit SSH dem primären Knoten eines Clusters verbunden sind.
- Verwenden Sie Bibliotheken für Notebooks – Wenn Workspace-Benutzer Bibliotheken von einer Notebook-Zelle aus installieren und verwenden, sind diese Bibliotheken nur für dieses Notebook verfügbar. Mit dieser Option können verschiedene Notebooks, die denselben Cluster verwenden, arbeiten, ohne sich Gedanken über widersprüchliche Bibliotheksversionen machen zu müssen.

EMRStudio-Arbeitsbereiche haben dieselbe grundlegende Architektur wie EMR Notebooks. Sie können Jupyter Notebook-Kernel und Python-Bibliotheken mit EMR Studio genauso installieren und verwenden wie mit Notebooks. EMR Detaillierte Anweisungen finden Sie unter Installieren und Verwenden von Kerneln und Bibliotheken.

Kernel und Bibliotheken bei Amazon EMR auf Clustern EKS

Amazon EMR on EKS Clusters umfasst die Kernel PySpark und Python 3.7 mit einer Reihe vorinstallierter Bibliotheken. Amazon EMR on EKS unterstützt die Installation zusätzlicher Bibliotheken oder Cluster nicht.

EMRAuf jedem Amazon EKS On-Cluster sind das folgende Python und die folgenden PySpark Bibliotheken installiert:

- Python – boto3, cffi, future, ggplot, jupyter, kubernetes, matplotlib, numpy, pandas, plotly, pycryptodomex, py4j, requests, scikit-learn, scipy, seaborn
- PySpark – ggplot, jupyter, matplotlib, numpy, pandas, plotly, pycryptodomex, py4j, requests, scikit-learn, scipy, seaborn

Kernel und Bibliotheken für EMR serverlose Anwendungen

In jeder EMR serverlosen Anwendung sind das folgende Python und die folgenden PySpark Bibliotheken installiert:

- Python – ggplot, matplotlib, numpy, pandas, plotly, bokeh, scikit-learn, scipy, seaborn
- PySpark – ggplot, matplotlib,numpy, pandas, plotly, bokeh, scikit-learn, scipy, seaborn

Verbessern Sie die Kernel mit Befehlen magic

Übersicht

EMRStudio und EMR Notebooks unterstützen magic Befehle. MagicBefehle oder magics sind Erweiterungen, die der IPython Kernel bereitstellt, um Sie beim Ausführen und Analysieren von Daten zu unterstützen. IPythonist eine interaktive Shell-Umgebung, die mit Python erstellt wurde.

Amazon unterstützt EMR auch ein PaketSparkmagic, das Spark-bezogene Kernel (PySparkSparkR- und Scala-Kernel) mit spezifischen magic Befehlen bereitstellt und Livy auf dem Cluster verwendet, um Spark-Jobs zu senden.

Sie können magic Befehle verwenden, solange Sie einen Python-Kernel in Ihrem EMR Notizbuch haben. In ähnlicher Weise unterstützt jeder Spark-bezogene Kernel Sparkmagic-Befehle.

Magic-Befehle, auch magic genannt, gibt es in zwei Varianten:

- Zeile magic – Diese magic-Befehle werden durch ein einzelnes `%`-Präfix gekennzeichnet und funktionieren in einer einzigen Codezeile
- Zelle magic – Diese magic-Befehle sind mit einem doppelten `%%`-Präfix gekennzeichnet und funktionieren auf mehreren Codezeilen

Alle verfügbaren magic finden Sie unter Liste der magic- und Sparkmagic-Befehle.

Überlegungen und Einschränkungen

- EMRServerless unterstützt `%%sh` das Ausführen `spark-submit` nicht. Die EMR Notebooks magic s werden nicht unterstützt.
- Amazon EMR auf EKS Clustern unterstützt keine Sparkmagic Befehle für EMR Studio. Das liegt daran, dass Spark-Kernel, die Sie mit verwalteten Endpunkten verwenden, in Kubernetes integriert sind und von Sparkmagic und Livy nicht unterstützt werden. Sie können die Spark-Konfiguration als Workaround direkt im SparkContext Objekt festlegen, wie das folgende Beispiel zeigt.

```
spark.conf.set("spark.driver.maxResultSize", '6g')
```

- Die folgenden magic Befehle und Aktionen sind verboten von AWS:
  - `%alias`
  - `%alias_magic`
  - `%automagic`
  - `%macro`
  - Ändern von `proxy_user` mit `%configure`
  - Ändern von `KERNEL_USERNAME` mit `%env` oder `%set_env`

Liste der magic- und Sparkmagic-Befehle

Verwenden Sie die folgenden Befehle, um die verfügbaren magic-Befehle aufzulisten:

- `%lsmagic` listet alle derzeit verfügbaren magic-Funktionen auf.
- `%%help` listet die derzeit verfügbaren SPARK-bezogenen magic-Funktionen auf, die vom Sparkmagic-Paket bereitgestellt werden.

`%%configure` wird verwendet, um Spark zu konfigurieren

Einer der nützlichsten Sparkmagic-Befehle ist der `%%configure`-Befehl, der die Parameter für die Sitzungserstellung konfiguriert. Mithilfe von `conf`-Einstellungen können Sie jede Spark-Konfiguration konfigurieren, die in der Konfigurationsdokumentation für Apache Spark erwähnt wird.

Example Fügen Sie externe JAR Dateien aus dem Maven-Repository oder Amazon S3 zu EMR Notebooks hinzu

Sie können den folgenden Ansatz verwenden, um jedem Spark-bezogenen Kernel, der von unterstützt wird, eine externe JAR Dateiabhängigkeit hinzuzufügen. Sparkmagic

```
%%configure -f
{"conf": {
    "spark.jars.packages": "com.jsuereth:scala-arm_2.11:2.0,ml.combust.bundle:bundle-
ml_2.11:0.13.0,com.databricks:dbutils-api_2.11:0.0.3",
    "spark.jars": "s3://DOC-EXAMPLE-BUCKET/my-jar.jar"
    }
}
```

Example : Konfigurieren von Hudi

Sie können den Notizbuch-Editor verwenden, um Ihr EMR Notizbuch für die Verwendung von Hudi zu konfigurieren.

```
%%configure
{ "conf": {
     "spark.jars": "hdfs://apps/hudi/lib/hudi-spark-bundle.jar,hdfs:///apps/hudi/lib/
spark-spark-avro.jar",
     "spark.serializer": "org.apache.spark.serializer.KryoSerializer",
     "spark.sql.hive.convertMetastoreParquet":"false"
     }
}
```

`%%sh` verwenden, um `spark-submit` auszuführen

Der `%%sh` magic führt Shell-Befehle in einem Unterprozess auf einer Instance Ihres verbundenen Clusters aus. In der Regel würden Sie einen der Spark-bezogenen Kernel verwenden, um Spark-Anwendungen auf Ihrem angeschlossenen Cluster auszuführen. Wenn Sie jedoch einen Python-Kernel verwenden möchten, um eine Spark-Anwendung einzureichen, können Sie Folgendes magic verwenden und den Bucket-Namen durch Ihren Bucket-Namen in Kleinbuchstaben ersetzen.

```
%%sh
spark-submit --master yarn --deploy-mode cluster s3://DOC-EXAMPLE-BUCKET/test.py
```

In diesem Beispiel benötigt der Cluster Zugriff auf den Speicherort von `s3://DOC-EXAMPLE-BUCKET/test.py`, andernfalls schlägt der Befehl fehl.

Sie können jeden Linux-Befehl mit `%%sh` magic verwenden. Wenn Sie Spark oder YARN Befehle ausführen möchten, verwenden Sie eine der folgenden Optionen, um einen `emr-notebook` Hadoop-Benutzer zu erstellen und dem Benutzer Berechtigungen zur Ausführung der Befehle zu gewähren:

- Sie können einen neuen Benutzer explizit erstellen, indem Sie die folgenden Befehle ausführen.

  ```
  hadoop fs -mkdir /user/emr-notebook
  hadoop fs -chown emr-notebook /user/emr-notebook
  ```

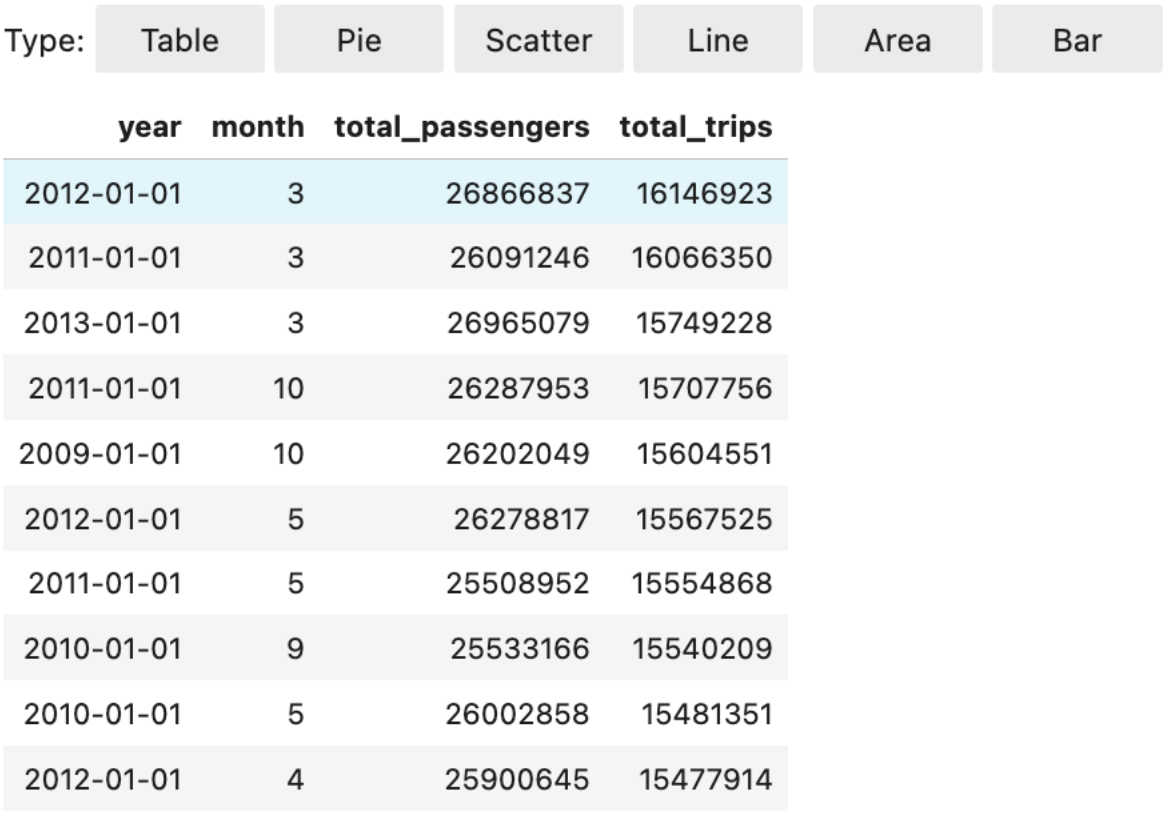
- Sie können den Benutzerwechsel in Livy aktivieren, wodurch der Benutzer automatisch erstellt wird. Weitere Informationen finden Sie unter Aktivieren des Identitätswechsels zur Überwachung von Spark-Benutzer- und -Aufgabenaktivitäten.

Wird zur Visualisierung von **`%%display`**-Spark-Datenrahmen verwendet

Sie können den verwenden, um einen `%%display`-magic-Spark-Datenrahmen zu visualisieren. Um diese magic zu verwenden, führen Sie den folgenden Befehl aus.

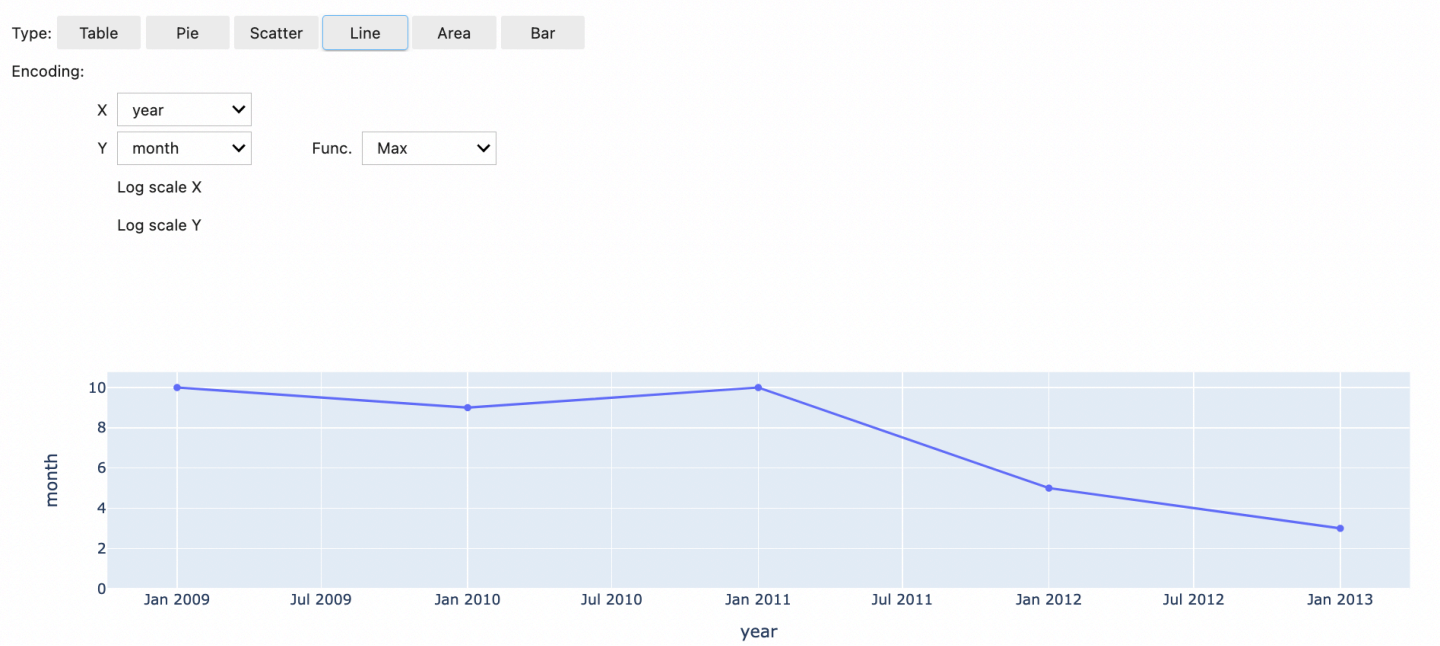
```
%%display df
```

Wählen Sie, ob Sie die Ergebnisse in einem Tabellenformat anzeigen möchten, wie das folgende Bild zeigt.



year	month	total_passengers	total_trips
2012-01-01	3	26866837	16146923
2011-01-01	3	26091246	16066350
2013-01-01	3	26965079	15749228
2011-01-01	10	26287953	15707756
2009-01-01	10	26202049	15604551
2012-01-01	5	26278817	15567525
2011-01-01	5	25508952	15554868
2010-01-01	9	25533166	15540209
2010-01-01	5	26002858	15481351
2012-01-01	4	25900645	15477914

Sie können sich auch dafür entscheiden, Ihre Daten mit fünf Arten von Diagrammen zu visualisieren. Zu Ihren Optionen gehören Kreis-, Streu-, Linien-, Flächen- und Balkendiagramme.



Verwenden Sie EMR Notebooks s magic

Amazon EMR bietet die folgenden EMR Notebooks magic an, die Sie mit Python3- und Spark-basierten Kerneln verwenden können:

- `%mount_workspace_dir` – Hängt Ihr Workspace-Verzeichnis in Ihren Cluster ein, sodass Sie Code aus anderen Dateien in Ihrem Workspace importieren und ausführen können

> **Note**
>
> Mit `%mount_workspace_dir` kann nur der Python-3-Kernel auf Ihre lokalen Dateisysteme zugreifen. Spark-Executoren haben mit diesem Kernel keinen Zugriff auf das bereitgestellte Verzeichnis.

- `%umount_workspace_dir` – Hängt Ihr Workspace-Verzeichnis von Ihrem Cluster ab
- `%generate_s3_download_url` – Generiert einen temporären Download-Link in Ihrer Notebook-Ausgabe für ein Amazon-S3-Objekt

Voraussetzungen

Bevor Sie EMR Notebooks magic s installieren, führen Sie die folgenden Aufgaben aus:

- Stellen Sie sicher, dass Servicerolle für EC2 Cluster-Instances (EC2Instance-Profil) Lesezugriff für Amazon S3 hat. Die `EMR_EC2_DefaultRole` mit der `AmazonElasticMapReduceforEC2Role` verwalteten Richtlinie erfüllt diese Anforderung. Wenn Sie eine benutzerdefinierte Rolle oder Richtlinie verwenden, stellen Sie sicher, dass sie über die erforderlichen S3-Berechtigungen verfügt.

  > **Note**
  >
  > EMRNotebooks magic werden auf einem Cluster als Notebook-Benutzer ausgeführt und verwenden das EC2 Instance-Profil, um mit Amazon S3 zu interagieren. Wenn Sie ein Workspace-Verzeichnis auf einem EMR Cluster bereitstellen, können alle Workspaces und EMR Notebooks, die berechtigt sind, eine Verbindung zu diesem Cluster herzustellen, auf das bereitgestellte Verzeichnis zugreifen.
  > Verzeichnisse werden standardmäßig schreibgeschützt bereitgestellt. Während `s3fs-fuse` und `goofys` Lese-/Schreibzugriffe ermöglichen, empfehlen wir dringend, die Bereitstellungsparameter nicht zu ändern, um Verzeichnisse im Lese-/Schreibmodus bereitzustellen. Wenn Sie Schreibzugriff zulassen, werden alle am Verzeichnis vorgenommenen Änderungen in den S3-Bucket geschrieben. Um ein versehentliches Löschen oder Überschreiben zu vermeiden, können Sie die Versionsverwaltung für Ihren S3-Bucket aktivieren. Weitere Informationen finden Sie unter Verwenden der Versionsverwaltung in S3-Buckets.

- Führen Sie eines der folgenden Skripts auf Ihrem Cluster aus, um die Abhängigkeiten für EMR Notebooks magic s zu installieren. Um ein Skript auszuführen, können Sie Benutzerdefinierte Bootstrap-Aktionen verwenden entweder den Anweisungen unter Befehle und Skripte auf einem EMR Amazon-Cluster ausführen folgen, wenn Sie bereits über einen laufenden Cluster verfügen.

  Sie können wählen, welche Abhängigkeit installiert werden soll. Sowohl s3fs-fuse als auch goofys sind Tools FUSE (Filesystem in Userspace), mit denen Sie einen Amazon S3 S3-Bucket als lokales Dateisystem auf einem Cluster mounten können. Das Tool bietet eine ähnliche Benutzererfahrung wie. `s3fs` POSIX Das `goofys` Tool ist eine gute Wahl, wenn Sie Leistung einem POSIX -konformen Dateisystem vorziehen.

  Die Amazon EMR 7.x-Serie verwendet Amazon Linux 2023, das keine EPEL Repositorys unterstützt. Wenn Sie Amazon EMR 7.x verwenden, folgen Sie zur Installation den GitHubs3fs-fuse-Anweisungen. `s3fs-fuse` Wenn Sie die Serien 5.x oder 6.x verwenden, verwenden Sie zur Installation die folgenden Befehle. `s3fs-fuse`

```
#!/bin/sh

# Install the s3fs dependency for EMR Notebooks magics
sudo amazon-linux-extras install epel -y
sudo yum install s3fs-fuse -y
```

ODER

```
#!/bin/sh

# Install the goofys dependency for EMR Notebooks magics
sudo wget https://github.com/kahing/goofys/releases/latest/download/goofys -P /usr/
bin/
sudo chmod ugo+x /usr/bin/goofys
```

Installieren Sie Notebooks s EMR magic

> **Note**
>
> Mit den EMR Amazon-Versionen 6.0 bis 6.9.0 und 5.0 bis 5.36.0 werden nur die `emr-notebooks-magics` Paketversionen 0.2.0 und höher unterstützt. `%mount_workspace_dir` magic

Gehen Sie wie folgt vor, um Notebooks s zu installieren. EMR magic

1. Führen Sie in Ihrem Notebook die folgenden Befehle aus, um das `emr-notebooks-magics`-Paket zu installieren.

   ```
   %pip install boto3 --upgrade
   %pip install botocore --upgrade
   %pip install emr-notebooks-magics --upgrade
   ```

2. Starten Sie Ihren Kernel neu, um die EMR Notebooks magic s zu laden.
3. Überprüfen Sie Ihre Installation mit dem folgenden Befehl, der den Ausgabehilfetext für `%mount_workspace_dir` anzeigen sollte.

   ```
   %mount_workspace_dir?
   ```

Ein Workspace-Verzeichnis mit **`%mount_workspace_dir`** mounten

`%mount_workspace_dir`magicDamit können Sie Ihr Workspace-Verzeichnis auf Ihrem EMR Cluster mounten, sodass Sie andere in Ihrem Verzeichnis gespeicherte Dateien, Module oder Pakete importieren und ausführen können.

Im folgenden Beispiel wird das gesamte Workspace-Verzeichnis auf einem Cluster bereitgestellt und das optionale *`<--fuse-type>`* Argument, um Goofys zum Mounten des Verzeichnisses zu verwenden.

```
%mount_workspace_dir . <--fuse-type goofys>
```

Um zu überprüfen, ob Ihr Workspace-Verzeichnis eingehängt ist, verwenden Sie das folgende Beispiel, um das aktuelle Arbeitsverzeichnis mit dem `ls`-Befehl anzuzeigen. Die Ausgabe sollte alle Dateien in Ihrem Workspace anzeigen.

```
%%sh
ls
```

Wenn Sie mit den Änderungen in Ihrem Workspace fertig sind, können Sie das Workspace-Verzeichnis mit dem folgenden Befehl unmounten:

> **Note**
>
> Ihr Workspace-Verzeichnis bleibt in Ihrem Cluster eingebunden, auch wenn der Workspace gestoppt oder getrennt wird. Sie müssen Ihr Workspace-Verzeichnis explizit unmounten.

```
%umount_workspace_dir
```

Herunterladen eines Amazon-S3-Objekts mit **`%generate_s3_download_url`**

Der `generate_s3_download_url` Befehl erstellt ein vorsigniertes Objekt URL für ein in Amazon S3 gespeichertes Objekt. Sie können das Presigned verwendenURL, um das Objekt auf Ihren lokalen Computer herunterzuladen. Sie könnten beispielsweise ausführen, `generate_s3_download_url` um das Ergebnis einer SQL Abfrage herunterzuladen, die Ihr Code in Amazon S3 schreibt.

Das vorsignierte URL ist standardmäßig 60 Minuten gültig. Sie können die Ablaufzeit ändern, indem Sie eine Anzahl von Sekunden für das `--expires-in`-Kennzeichen angeben. `--expires-in 1800`Erstellt beispielsweise eine, URL die 30 Minuten gültig ist.

Das folgende Beispiel generiert einen Download-Link für ein Objekt, indem der vollständige Amazon-S3-Pfad angegeben wird: *`s3://EXAMPLE-DOC-BUCKET/path/to/my/object`*.

```
%generate_s3_download_url s3://EXAMPLE-DOC-BUCKET/path/to/my/object
```

Um mehr über die Verwendung von `generate_s3_download_url` zu erfahren, führen Sie den folgenden Befehl aus, um den Hilfetext anzuzeigen.

```
%generate_s3_download_url?
```

Führen Sie ein Notebook im Headless-Modus mit **`%execute_notebook`**

Mit `%execute_notebook` magic können Sie ein anderes Notebook im Headless-Modus ausführen und die Ausgabe für jede Zelle anzeigen, die Sie ausgeführt haben. Dies magic erfordert zusätzliche Berechtigungen für die Instance-Rolle, die Amazon EMR und Amazon EC2 gemeinsam nutzen. Führen Sie den `%execute_notebook`?-Befehl aus, um weitere Informationen zur Gewährung zusätzlicher Berechtigungen zu erhalten.

Während eines Auftrags mit langer Laufzeit wechselt Ihr System möglicherweise aufgrund von Inaktivität in den Standbymodus oder verliert vorübergehend die Internetverbindung. Dadurch könnte die Verbindung zwischen Ihrem Browser und dem Jupyter Server unterbrochen werden. In diesem Fall verlieren Sie möglicherweise die Ausgabe der Zellen, die Sie vom Jupyter Server ausgeführt und gesendet haben.

Wenn Sie das Notebook im Headless-Modus mit ausführen `%execute_notebook`magic, erfasst EMR Notebooks die Ausgabe der Zellen, die ausgeführt wurden, auch wenn das lokale Netzwerk unterbrochen wird. EMRNotebooks speichert die Ausgabe inkrementell in einem neuen Notizbuch mit demselben Namen wie das Notizbuch, das Sie ausgeführt haben. EMRNotebooks platziert das Notizbuch dann in einem neuen Ordner innerhalb des Arbeitsbereichs. Headless-Läufe finden auf demselben Cluster statt und verwenden die Servicerolle `EMR_Notebook_DefaultRole`, aber zusätzliche Argumente können die Standardwerte ändern.

Verwenden Sie den folgenden Befehl, um ein Notebook im Headless-Modus auszuführen:

```
%execute_notebook <relative-file-path>
```

Verwenden Sie den folgenden Befehl, um eine Cluster-ID und eine Servicerolle für einen Headless-Lauf anzugeben:

```
%execute_notebook <notebook_name>.ipynb --cluster-id <emr-cluster-id> --service-role
 <emr-notebook-service-role>
```

Wenn Amazon EMR und Amazon sich eine Instance-Rolle EC2 teilen, sind für die Rolle die folgenden zusätzlichen Berechtigungen erforderlich:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "elasticmapreduce:StartNotebookExecution",
                "elasticmapreduce:DescribeNotebookExecution",
                "ec2:DescribeInstances"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:PassRole"
            ],
            "Resource": "arn:aws:iam::<AccoundId>:role/EMR_Notebooks_DefaultRole"
        }
    ]
}
```

> **Note**
>
> Um `%execute_notebook` magic zu verwenden, installieren Sie das `emr-notebooks-magics`-Paket, Version 0.2.3 oder höher.

Verwenden Sie mehrsprachige Notebooks mit Spark-Kerneln

Jeder Jupyter-Notebook-Kernel hat eine Standardsprache. Die Standardsprache des Spark-Kernels ist beispielsweise Scala, und die Standardsprache des PySpark Kernels ist Python. Mit Amazon EMR

6.4.0 und höher unterstützt EMR Studio mehrsprachige Notizbücher. Das bedeutet, dass jeder Kernel in EMR Studio zusätzlich zur Standardsprache die folgenden Sprachen unterstützen kann: Python, Spark, R und SparkSQL.

Um dieses Feature zu aktivieren, geben Sie am Anfang einer beliebigen Zelle einen der folgenden magic-Befehle an.

Sprache	Befehl
Python	`%%pyspark`
Scala	`%%scalaspark`
R	`%%rspark` Wird für interaktive Workloads mit EMR Serverless nicht unterstützt.
Spark SQL	`%%sql`

Wenn diese Befehle aufgerufen werden, führen sie die gesamte Zelle innerhalb derselben Spark-Sitzung mit dem Interpreter der entsprechenden Sprache aus.

Die `%%pyspark` Zelle magic ermöglicht es Benutzern, PySpark Code in alle Spark-Kernel zu schreiben.

```
%%pyspark
a = 1
```

Die `%%sql` Zelle magic ermöglicht es Benutzern, SQL Spark-Code in allen Spark-Kerneln auszuführen.

```
%%sql
SHOW TABLES
```

Die `%%rspark`-Zelle magic ermöglicht es Benutzern, SparkR in allen Spark-Kerneln auszuführen.

```
%%rspark
```

```
a <- 1
```

Die `%%scalaspark`-Zelle magic ermöglicht es Benutzern, Spark-Scala-Code in allen Spark-Kerneln auszuführen.

```
%%scalaspark
val a = 1
```

Teilen Sie Daten mit temporären Tabellen zwischen Sprachinterpretern

Mithilfe temporärer Tabellen können Sie Daten auch zwischen Sprachinterpretern austauschen. Das folgende Beispiel verwendet `%%pyspark` in einer Zelle, um eine temporäre Tabelle in Python zu erstellen, und verwendet `%%scalaspark` in der folgenden Zelle, um Daten aus dieser Tabelle in Scala zu lesen.

```
%%pyspark
df=spark.sql("SELECT * from nyc_top_trips_report LIMIT 20")
# create a temporary table called nyc_top_trips_report_view in python
df.createOrReplaceTempView("nyc_top_trips_report_view")
```

```
%%scalaspark
// read the temp table in scala
val df=spark.sql("SELECT * from nyc_top_trips_report_view")
df.show(5)
```

Übersicht über Amazon EMR Notebooks

> **Note**
>
> EMRNotebooks sind in der Konsole als EMR Studio-Workspaces verfügbar. Mit der Schaltfläche „Arbeitsbereich erstellen“ in der Konsole können Sie neue Notizbücher erstellen. Um auf Workspaces zuzugreifen oder diese zu erstellen, benötigen EMR Notebook-Benutzer zusätzliche IAM Rollenberechtigungen. Weitere Informationen finden Sie unter Amazon EMR Notebooks sind Amazon EMR Studio-Workspaces in der Konsole und EMRAmazon-Konsole.

Sie können Amazon EMR Notebooks zusammen mit EMR Amazon-Clustern verwenden, auf denen Apache Spark ausgeführt wird, um Jupyter Notebook und JupyterLab Schnittstellen innerhalb der Amazon-Konsole zu erstellen und zu öffnen. EMR Ein EMR Notebook ist ein „serverloses“ Notebook, mit dem Sie Abfragen und Code ausführen können. Im Gegensatz zu einem herkömmlichen Notizbuch wird der Inhalt eines EMR Notizbuches — die Gleichungen, Abfragen, Modelle, der Code und der erläuternde Text in Notizbuchzellen — in einem Client ausgeführt. Die Befehle werden mithilfe eines Kernels auf dem EMR Cluster ausgeführt. Notebook-Inhalte werden auch getrennt von den Cluster-Daten in Amazon S3 gespeichert, um eine sichere Speicherung und flexible Wiederverwendung zu gewährleisten.

Sie können einen Cluster starten, ein EMR Notebook zur Analyse anhängen und dann den Cluster beenden. Sie können auch ein Notebook schließen, das an einen ausgeführten Cluster angefügt ist, und zu einem anderen Cluster wechseln. Mehrere Benutzer können gleichzeitig Notebooks an denselben Cluster anfügen und in Amazon S3 Notebook-Dateien miteinander teilen. Diese Funktionen ermöglichen Ihnen die On-Demand-Ausführung von Clustern, um Kosten zu sparen und den Zeitaufwand für die Neukonfiguration von Notebooks für verschiedene Cluster und Datensätze zu reduzieren.

Sie können ein EMR Notebook auch programmgesteuert über Amazon ausführen EMRAPI, ohne mit der EMR Amazon-Konsole interagieren zu müssen („Headless Execution“). Sie müssen eine Zelle in das EMR Notizbuch aufnehmen, die über ein Parameter-Tag verfügt. Diese Zelle ermöglicht es einem Skript, neue Eingabewerte an das Notizbuch zu übergeben. Parametrisierte Notizbücher können mit unterschiedlichen Eingabewerten wiederverwendet werden. Es ist nicht erforderlich, Kopien desselben Notebooks zu erstellen, um es mit neuen Eingabewerten zu bearbeiten und auszuführen. Amazon EMR erstellt und speichert das Ausgabe-Notizbuch auf S3 für jeden Lauf des

parametrisierten Notebooks. APICodebeispiele für EMR Notebooks finden Sie unter. Beispielbefehle zur programmgesteuerten Ausführung von EMR Notebooks

> **Important**
>
> Die EMR Notebooks-Funktion unterstützt Cluster, die EMR Amazon-Versionen 5.18.0 und höher verwenden. Wir empfehlen, EMR Notebooks mit Clustern zu verwenden, die die neueste Version von Amazon oder mindestens 5.30.0EMR, 5.32.0 oder 6.2.0 verwenden. Mit diesen Versionen werden Jupyter-Kernel auf dem angefügten Cluster und nicht auf einer Jupyter-Instance ausgeführt werden. Dies verbessert die Leistung und erweitert Ihre Möglichkeiten von Kerneln und Bibliotheken zu verbessern. Weitere Informationen finden Sie unter Unterschiede in den Funktionalitäten nach Cluster-Release-Version.

Es fallen Gebühren für Amazon S3 S3-Speicher und für EMR Amazon-Cluster an.

Amazon EMR Notebooks sind als Amazon EMR Studio Workspaces in der Konsole verfügbar.

Der Übergang von EMR Notebooks zu Workspaces

In der neuen EMR Amazon-Konsole haben wir EMR Notebooks mit Amazon EMR Studio Workspaces zu einem einzigen Erlebnis zusammengeführt. Wenn Sie ein EMR Studio verwenden, können Sie verschiedene Workspaces erstellen und konfigurieren, um Notizbücher zu organisieren und auszuführen. Wenn Sie EMR Amazon-Notizbücher in der alten Konsole hatten, sind sie in der Konsole als EMR Studio-Workspaces verfügbar.

Amazon EMR hat diese neuen EMR Studio-Arbeitsbereiche für Sie erstellt. Die Anzahl der Studios, die wir erstellt haben, entspricht der Anzahl der unterschiedlichen StudiosVPCs, die Sie von EMR Notebooks aus verwenden. Wenn Sie beispielsweise eine Verbindung zu EMR Clustern herstellen, die sich VPCs von EMR Notebooks unterscheiden, haben wir zwei neue EMR Studios erstellt. Ihre Notebooks werden auf die neuen Studios verteilt.

> **Important**
>
> Wir haben die Option zum Erstellen neuer Notizbücher in der alten EMR Amazon-Konsole deaktiviert. Verwenden Sie stattdessen Create Workspace in der neuen EMR Amazon-Konsole.

Weitere Informationen zu Amazon EMR Studio Workspaces finden Sie unterInformationen über Workspace-Grundlagen. Einen konzeptionellen Überblick über EMR Studio finden Sie Workspaces auf der Wie Amazon EMR Studio funktioniert Seite.

Was müssen Sie als Nächstes tun?

Sie können Ihre vorhandenen Notizbücher zwar weiterhin in der alten Konsole verwenden, wir empfehlen jedoch, stattdessen Amazon EMR Studio Workspaces in der Konsole zu verwenden. Sie müssen zusätzliche Rollenberechtigungen konfigurieren, um die Funktionen in EMR Studio zu aktivieren, die in EMR Notebooks nicht verfügbar sind.

> **Note**
>
> Um bestehende EMR Notebooks als EMR Studio-Arbeitsbereiche anzuzeigen und neue Arbeitsbereiche zu erstellen, müssen Benutzer mindestens über `elasticmapreduce:CreateStudioPresignedUrl` Berechtigungen für ihre Rollen verfügen`elasticmapreduce:ListStudios`. Um auf alle EMR Studio-Funktionen zuzugreifen, finden Sie die EMRStudio-Funktionen für EMR Notebook-Benutzer aktivieren vollständige Liste der zusätzlichen Berechtigungen, die EMR Notebook-Benutzer benötigen.

Verbesserte Funktionen in EMR Studio, die über EMR Notebooks hinausgehen

Mit Amazon EMR Studio können Sie die folgenden Funktionen einrichten und verwenden, die bei EMR Notebooks nicht verfügbar sind:

- In Jupyterlab können Sie nach EMR Clustern suchen und Verbindungen zu ihnen herstellen
- Suchen Sie in Jupyterlab nach virtuellen Clustern in EMR Notebooks und verbinden Sie sie mit ihnen
- Verbindung zu Git-Repos aus Jupyterlab heraus

- Zusammenarbeit mit anderen Mitgliedern Ihres Teams beim Schreiben und Ausführen von Notebook-Code
- Durchsuchen Sie Daten mit dem Explorer SQL
- Bereitstellen von EMR Clustern mit Service Catalog

Eine vollständige Liste der Funktionen von Amazon EMR Studio finden Sie unterHauptfeatures von EMR Studio.

EMRStudio-Funktionen für EMR Notebook-Benutzer aktivieren

Die neuen EMR Studios, die wir im Rahmen dieser Zusammenführung erstellen werden, verwenden die bestehende `EMR_Notebooks_DefaultRole` IAM Rolle als EMR Studio-Servicerolle.

Benutzer, die von EMR Notebooks zu EMR Studio wechseln und die zusätzlichen Funktionen von EMR Studio nutzen möchten, benötigen mehrere neue Rollenberechtigungen. Fügen Sie den Rollen Ihrer EMR Notebook-Benutzer, die EMR Studio verwenden möchten, die folgenden Berechtigungen hinzu.

> **Note**
>
> Um bestehende EMR Notebooks als EMR Studio-Arbeitsbereiche anzuzeigen und neue Arbeitsbereiche zu erstellen, müssen Benutzer mindestens über `elasticmapreduce:CreateStudioPresignedUrl` Berechtigungen für ihre Rollen verfügen`elasticmapreduce:ListStudios`. Um alle EMR Studio-Funktionen nutzen zu können, fügen Sie alle unten aufgeführten Berechtigungen hinzu. Admin-Benutzer benötigen außerdem die Erlaubnis, ein EMR Studio zu erstellen und zu verwalten. Weitere Informationen finden Sie unter Administratorrechte zum Erstellen und Verwalten eines Studios EMR.

```
"elasticmapreduce:DescribeStudio",
"elasticmapreduce:ListStudios",
"elasticmapreduce:CreateStudioPresignedUrl",
"elasticmapreduce:UpdateEditor",
"elasticmapreduce:PutWorkspaceAccess",
"elasticmapreduce:DeleteWorkspaceAccess",
"elasticmapreduce:ListWorkspaceAccessIdentities",
"emr-containers:ListVirtualClusters",
```

```
"emr-containers:DescribeVirtualCluster",
"emr-containers:ListManagedEndpoints",
"emr-containers:DescribeManagedEndpoint",
"emr-containers:CreateAccessTokenForManagedEndpoint",
"emr-containers:ListJobRuns",
"emr-containers:DescribeJobRun",
"servicecatalog:SearchProducts",
"servicecatalog:DescribeProduct",
"servicecatalog:DescribeProductView",
"servicecatalog:DescribeProvisioningParameters",
"servicecatalog:ProvisionProduct",
"servicecatalog:UpdateProvisionedProduct",
"servicecatalog:ListProvisioningArtifacts",
"servicecatalog:DescribeRecord",
"servicecatalog:ListLaunchPaths",
"cloudformation:DescribeStackResources"
```

Die folgenden Berechtigungen sind ebenfalls erforderlich, um die Funktionen für die Zusammenarbeit in EMR Studio zu nutzen, waren jedoch für EMR Notebooks nicht erforderlich.

```
"sso-directory:SearchUsers",
"iam:GetUser",
"iam:GetRole",
"iam:ListUsers",
"iam:ListRoles",
"sso:GetManagedApplicationInstance"
```

Überlegungen zur Verwendung von EMR Notebooks

> **Note**
>
> EMRNotizbücher sind in der Konsole als EMR Studio-Arbeitsbereiche verfügbar. Mit der Schaltfläche „Arbeitsbereich erstellen“ in der Konsole können Sie neue Notizbücher erstellen. Um auf Workspaces zuzugreifen oder diese zu erstellen, benötigen EMR Notebook-Benutzer zusätzliche IAM Rollenberechtigungen. Weitere Informationen finden Sie unter Amazon EMR Notebooks sind Amazon EMR Studio-Workspaces in der Konsole und EMRAmazon-Konsole.

Beachten Sie die folgenden Anforderungen, wenn Sie Cluster erstellen und Lösungen mithilfe von EMR Notebooks entwickeln.

Cluster-Voraussetzungen

- Amazon EMR Block Public Access aktivieren — Durch den eingehenden Zugriff auf einen Cluster können Cluster-Benutzer Notebook-Kernel ausführen. Stellen Sie sicher, dass nur autorisierte Benutzer auf den Cluster zugreifen können. Wir empfehlen dringend, den Block Public Access aktiviert zu lassen und den eingehenden SSH Datenverkehr nur auf vertrauenswürdige Quellen zu beschränken. Weitere Informationen erhalten Sie unter Verwenden Sie Amazon, um EMR den öffentlichen Zugriff zu blockieren und Steuerung des Netzwerkverkehrs mit Sicherheitsgruppen.
- Kompatiblen Cluster verwenden – Ein Cluster, der an ein Notebook angefügt ist, muss die folgenden Voraussetzungen erfüllen:
  - Es werden nur Cluster unterstützt, EMR die mit Amazon erstellt wurden. Sie können unabhängig innerhalb von Amazon einen Cluster erstellen EMR und dann ein EMR Notebook anhängen, oder Sie können einen kompatiblen Cluster erstellen, wenn Sie ein EMR Notebook erstellen.
  - Es werden nur Cluster unterstützt, die mit der EMR Amazon-Version 5.18.0 und höher erstellt wurden. Siehe the section called “Unterschiede in den Funktionalitäten nach Cluster-Release-Version”.
  - Cluster, die mithilfe von EC2 Amazon-Instances mit AMD EPYC Prozessoren erstellt wurden — zum Beispiel die Instance-Typen m5a.* und r5a.* — werden nicht unterstützt.
  - EMRNotebooks funktionieren nur mit `VisibleToAllUsers` `true` Clustern, die mit der Einstellung auf erstellt wurden. `VisibleToAllUsers`ist `true` standardmäßig.
  - Der Cluster muss innerhalb eines EC2 - gestartet werdenVPC. Öffentliche und private Subnetze werden unterstützt. Die EC2 -Classic-Plattform wird nicht unterstützt.
  - Hadoop, Spark und Livy müssen auf dem Cluster installiert sein. Andere Anwendungen können installiert werden, aber EMR Notebooks unterstützt derzeit nur Spark-Cluster.

    > **Important**
    >
    > Für EMR Amazon-Release-Versionen 5.32.0 und höher oder 6.2.0 und höher muss auf Ihrem Cluster auch die Jupyter Enterprise Gateway-Anwendung ausgeführt werden, um mit Notebooks zu funktionieren. EMR

  - Cluster mit Kerberos-Authentifizierung werden nicht unterstützt.
  - Mit integrierte Cluster AWS Lake Formation unterstützen nur die Installation von Bibliotheken für Notebooks. Die Installation von Kerneln und Bibliotheken auf dem Cluster wird nicht unterstützt.
  - Cluster mit mehreren Primärknoten werden nicht unterstützt.

- Cluster, die EC2 Amazon-Instances verwenden, die auf AWS Graviton2 basieren, werden nicht unterstützt.

Unterschiede in den Funktionalitäten nach Cluster-Release-Version

Wir empfehlen dringend, EMR Notebooks mit Clustern zu verwenden, die mit den EMR Amazon-Release-Versionen 5.30.0, 5.32.0 oder höher oder 6.2.0 oder höher erstellt wurden. Mit diesen Versionen führt EMR Notebooks Kernel auf dem angeschlossenen EMR Amazon-Cluster aus. Kernel und Bibliotheken können direkt auf dem Cluster-Primärknoten installiert werden. Die Verwendung von EMR Notebooks mit diesen Cluster-Versionen hat die folgenden Vorteile:

- Verbesserte Leistung — Notebook-Kernel werden auf Clustern mit von Ihnen ausgewählten EC2 Instance-Typen ausgeführt. Frühere Versionen führen Kernel auf einer spezialisierten Instance aus, die nicht in der Größe geändert, auf die nicht zugegriffen und die nicht angepasst werden kann.
- Möglichkeit zum Hinzufügen und Anpassen von Kerneln – Sie können eine Verbindung zum Cluster herstellen, um Kernel-Pakete mit `conda` und `pip` zu installieren. Darüber hinaus wird die `pip`-Installation mithilfe von Terminal-Befehlen innerhalb von Notebook-Zellen unterstützt. In früheren Versionen waren nur vorinstallierte Kernel verfügbar (Python PySpark, Spark und SparkR). Weitere Informationen finden Sie unter Installieren von Kernels und Python-Bibliotheken auf einem Cluster-Primärknoten.
- Möglichkeit, Python-Bibliotheken zu installieren – Sie können Python-Bibliotheken mit `conda` und `pip` auf dem Cluster-Primärknoten installieren. Wir empfehlen die Verwendung von `conda`. In früheren Versionen wurden nur Bibliotheken für für Notebooks unterstützt. PySpark

Unterstützte EMR Notebooks-Funktionen nach Cluster-Version

Cluster-Version	Bibliotheken für Notebooks für PySpark	Kernel-Installation auf dem Cluster	Installation der Python-Bibliothek auf Primärknoten
Früher als 5.18.0	EMRNotebooks werden nicht unterstützt		
5.18.0–5.25.0	Nein	Nein	Nein
5.26.0–5-29.0	Ja	Nein	Nein
5.30.0	Ja	Ja	Ja

Cluster-Version	Bibliotheken für Notebooks für PySpark	Kernel-Installation auf dem Cluster	Installation der Python-Bibliothek auf Primärknoten
6.0.0	Nein	Nein	Nein
5.32.0 und höher und 6.2.0 und höher	Ja	Ja	Ja

Grenzwerte für gleichzeitig angeschlossene Notebooks EMR

Wenn Sie einen Cluster erstellen, der Notebooks unterstützt, sollten Sie den EC2 Instanztyp des primären Clusterknotens berücksichtigen. Die Speicherbeschränkungen dieser EC2 Instanz bestimmen die Anzahl der Notebooks, die gleichzeitig bereit sein können, Code und Abfragen auf dem Cluster auszuführen.

EC2Instanztyp des primären Knotens	Anzahl der EMR Notebooks
*.medium	2
*.large	4
*.xlarge	8
*.2xlarge	16
*.4xlarge	24
*.8xlarge	24
*.16xlarge	24

Jupyter Notebook und Python-Versionen

EMRAuf Notebooks werden Jupyter Notebook Version 6.0.2 und Python 3.6.5 ausgeführt, unabhängig von der EMR Amazon-Release-Version des angehängten Clusters.

Sicherheitsüberlegungen

Verwenden verschlüsselter S3-Standorte

Wenn Sie einen verschlüsselten Speicherort in Amazon S3 zum Speichern von Notebook-Dateien angeben, müssen Sie die Servicerolle für EMR Notebooks als Schlüsselbenutzer einrichten. Die Standard-Servicerolle ist `EMR_Notebooks_DefaultRole`. Wenn Sie einen AWS KMS Schlüssel für die Verschlüsselung verwenden, finden Sie weitere Informationen unter Verwenden von Schlüsselrichtlinien im AWS Key Management Service Entwicklerhandbuch und AWS KMS im Support-Artikel zum Hinzufügen von Schlüsselbenutzern.

Verwendung von Cookies mit Hosting-Domains

Um die Sicherheit der Anwendungen außerhalb der Konsole zu erhöhen, die Sie möglicherweise mit Amazon verwendenEMR, werden die Anwendungshosting-Domains in der Liste der öffentlichen Suffixe () registriert. PSL Zu diesen Hosting-Domains gehören beispielsweise die folgenden: `emrstudio-prod.us-east-1.amazonaws.com`, `emrnotebooks-prod.us-east-1.amazonaws.com`, `emrappui-prod.us-east-1.amazonaws.com`. Aus Sicherheitsgründen empfehlen wir Ihnen, Cookies mit einem `__Host-`-Präfix zu verwenden, falls Sie jemals sensible Cookies im Standard-Domainnamen einrichten müssen. Dies trägt dazu bei, Ihre Domain vor standortübergreifenden Anforderungsfälschungsversuchen zu schützen (). CSRF Weitere Informationen finden Sie auf der Set-Cookie-Seite im Mozilla Developer Network.

Erstellen eines Notebook

> **Note**
>
> EMRNotebooks sind in der Konsole als EMR Studio-Workspaces verfügbar. Mit der Schaltfläche „Arbeitsbereich erstellen“ in der Konsole können Sie neue Notizbücher erstellen. Um auf Workspaces zuzugreifen oder diese zu erstellen, benötigen EMR Notebook-Benutzer zusätzliche IAM Rollenberechtigungen. Weitere Informationen finden Sie unter Amazon EMR Notebooks sind Amazon EMR Studio-Workspaces in der Konsole und EMRAmazon-Konsole.

Sie erstellen ein EMR Notizbuch mit der alten EMR Amazon-Konsole. Das Erstellen von Notizbüchern mit dem AWS CLI oder Amazon EMR API wird nicht unterstützt.

Um ein EMR Notizbuch zu erstellen

1. Öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/elasticmapreduce/.
2. Wählen Sie Notebooks, Create notebook (Notebook erstellen).
3. Geben Sie einen Notebook name (Notebook-Namen) und optional eine Notebook description (Notebook-Beschreibung) ein.
4. Wenn Sie einen aktiven Cluster haben, den Sie an das Notebook anfügen möchten, behalten Sie die Standardeinstellung Vorhandenen Cluster auswählen bei und wählen Wählen aus. Wählen Sie anschließend einen Cluster aus der Liste und Cluster wählen aus. Informationen zu den Cluster-Anforderungen für EMR Notebooks finden Sie unterÜberlegungen zur Verwendung von EMR Notebooks.

   —oder—

   Wählen Sie Create a cluster (Cluster erstellen), geben Sie einen Clusternamen ein und wählen Sie Optionen gemäß den folgenden Richtlinien aus. Der Cluster wird standardmäßig VPC für das Konto mithilfe von On-Demand-Instances erstellt.

Einstellung	Beschreibung
Cluster name	Der Anzeigename, der zum Identifizieren des Clusters verwendet wird.
Veröffentlichung	Kann nicht geändert werden. Standardmäßig wird die neueste EMR Amazon-Release-Version (5.36.2) verwendet.
Anwendungen	Kann nicht geändert werden. Listet die Anwendungen auf, die auf dem Cluster installiert sind.
Instance	Geben Sie die Anzahl der Instances ein und wählen Sie den EC2 Instance-Typ aus. Eine Instance wird für den Primärknoten verwendet. Der Rest wird für Core-Knoten verwendet. Der Instance-Typ bestimmt die Anzahl der Notebooks, die gleichzeitig an den Cluster angefügt sein können. Weitere

Einstellung	Beschreibung
	Informationen finden Sie unter Grenzwerte für gleichzeitig angeschlossene Notebooks EMR.
EMRRolle	Behalten Sie die Standardeinstellung bei oder wählen Sie den Link, um eine benutzerdefinierte Servicerolle für Amazon anzugeben EMR. Weitere Informationen finden Sie unter Servicerolle für Amazon EMR (EMRRolle).
EC2Instanzprofil	Behalten Sie die Standardeinstellung bei oder wählen Sie den Link, um eine benutzerdefinierte Servicerolle für EC2 Instanzen anzugeben. Weitere Informationen finden Sie unter Servicerolle für EC2 Cluster-Instances (EC2Instance-Profil).
EC2key pair	Wählen Sie ein EC2 key pair, um eine Verbindung zu Cluster-Instances herstellen zu können. Weitere Informationen finden Sie unter Connect zum Primärknoten her mit SSH.
Automatische Beendigung	Die automatische Kündigung wird für die EMR Amazon-Versionen 5.30.0 und 6.1.0 und höher unterstützt. Aktivieren Sie das Kontrollkästchen, um die automatische Beendigung zu aktivieren, und geben Sie dann die Leerlaufzeit an, nach der der Cluster automatisch heruntergefahren werden soll. Weitere Informationen finden Sie unter Verwenden einer Richtlinie zur automatischen Beendigung.

5. Wählen Sie unter Security groups (Sicherheitsgruppen) die Option Use default security groups (Standardsicherheitsgruppen verwenden). Wählen Sie alternativ Sicherheitsgruppen auswählen

und wählen Sie benutzerdefinierte Sicherheitsgruppen aus, die VPC im Cluster verfügbar sind. Sie wählen eine Sicherheitsgruppe für die primäre Instance und eine andere für den Notebook-Service aus. Weitere Informationen finden Sie unter the section called “Sicherheitsgruppen für EMR Notebooks”.

6. Behalten Sie für AWS -Servicerolle die Standardeinstellung bei oder wählen Sie eine benutzerdefinierte Rolle aus der Liste aus. Die Client-Instance für das Notebook verwendet diese Rolle. Weitere Informationen finden Sie unter Servicerolle für EMR Notebooks.
7. Wählen Sie unter Notebook-Speicherort den Speicherort für die Notebook-Datei in Amazon S3 aus oder geben Sie einen eigenen Speicherort an. Wenn der Bucket und der Ordner nicht existieren, EMR erstellt Amazon sie.

   Amazon EMR erstellt einen Ordner mit der Notizbuch-ID als Ordnernamen und speichert das Notizbuch in einer Datei mit dem Namen`NotebookName`.ipynb. Wenn Sie zum Beispiel den Amazon-S3-Speicherort `s3://MyBucket/MyNotebooks` für ein Notebook mit dem Namen `MyFirstEMRManagedNotebook` angeben, wird die Notebook-Datei unter `s3://MyBucket/MyNotebooks/`*`NotebookID`*`/MyFirstEMRManagedNotebook.ipynb` gespeichert.

   Wenn Sie einen verschlüsselten Speicherort in Amazon S3 angeben, müssen Sie Servicerolle für EMR Notebooks als Schlüsselbenutzer einrichten. Die Standard-Servicerolle ist `EMR_Notebooks_DefaultRole`. Wenn Sie einen AWS KMS Schlüssel für die Verschlüsselung verwenden, finden Sie weitere Informationen unter Verwenden von Schlüsselrichtlinien AWS KMS im AWS Key Management Service Entwicklerhandbuch und im Support-Artikel zum Hinzufügen von Schlüsselbenutzern.
8. Wenn Sie Amazon ein Git-basiertes Repository hinzugefügt haben, EMR das Sie diesem Notizbuch zuordnen möchten, wählen Sie optional Git-Repository, wählen Sie Repository auswählen aus und wählen Sie dann ein Repository aus der Liste aus. Weitere Informationen finden Sie unter Git-basierte Repositorys mit Notebooks verknüpfen EMR.
9. Wählen Sie optional Tags und fügen Sie dann alle zusätzlichen Schlüssel-Wert-Tags für das Notebook hinzu.

   > **Important**
   >
   > Für den Zugriff wird ein Standard-Tag verwendet, bei dem die Schlüsselzeichenfolge auf `creatorUserID` und der Wert auf Ihre IAM Benutzer-ID gesetzt ist. Wir empfehlen, dieses Tag nicht zu ändern oder zu entfernen, da es für die Zugriffssteuerung verwendet

werden kann. Weitere Informationen finden Sie unter Verwenden Sie Cluster- und Notebook-Tags mit IAM Richtlinien für die Zugriffskontrolle.

10. Klicken Sie auf Create Notebook (Notebook erstellen).

Mit EMR Notizbüchern arbeiten

> **Note**
>
> EMRNotizbücher sind als EMR Studio-Arbeitsbereiche in der Konsole verfügbar. Mit der Schaltfläche „Arbeitsbereich erstellen“ in der Konsole können Sie neue Notizbücher erstellen. Um auf Workspaces zuzugreifen oder diese zu erstellen, benötigen EMR Notebook-Benutzer zusätzliche IAM Rollenberechtigungen. Weitere Informationen finden Sie unter Amazon EMR Notebooks sind Amazon EMR Studio-Workspaces in der Konsole und EMRAmazon-Konsole.

Nachdem Sie ein EMR Notizbuch erstellt haben, dauert es eine kurze Zeit, bis das Notizbuch gestartet wird. Der Status in der Liste Notebooks zeigt Starting (Wird gestartet) an. Sie können ein Notebook öffnen, wenn der Status Ready (Bereit) lautet. Es kann etwas länger dauern, bis ein Notebook den Status Ready (Bereit) anzeigt, wenn Sie einen Cluster mit diesem zusammen erstellt haben.

> **Tip**
>
> Aktualisieren Sie Ihren Browser oder wählen Sie das Aktualisierungssymbol über der Liste „Notebooks“, um den Notebookstatus zu aktualisieren.

Grundlegendes zum Notebook-Status

Ein EMR Notizbuch kann in der Notizbuchliste den folgenden Status haben.

Status	Bedeutung
Bereit	Sie können das Notebook mithilfe des Notebook-Editors öffnen. Wenn ein Notebook den Status Ready (Bereit) aufweist, können

Status	Bedeutung
	Sie es anhalten oder löschen. Um Cluster zu wechseln, müssen Sie das Notebook zuerst anhalten. Wenn ein Notebook mit dem Status Ready (Bereit) für einen langen Zeitraum inaktiv ist, wird es automatisch angehalten.
Wird gestartet	Das Notebook wird erstellt und an den Cluster angehängt. Während ein Notebook gestartet wird, können Sie den Notebook-Editor nicht öffnen, anhalten oder löschen und Cluster nicht wechseln.
Ausstehend	Das Notebook wurde erstellt und wartet darauf, dass die Integration mit dem Cluster abgeschlo ssen ist. Der Cluster stellt möglicherweise weiterhin Ressourcen bereit oder reagiert auf andere Anfragen. Sie können den Notebook-Editor mit dem Notebook im lokalen Modus öffnen. Code, der von Cluster-Prozessen abhängt, wird nicht ausgeführt und schlägt fehl.
Wird angehalten	Das Notebook wird heruntergefahren oder der Cluster, an den das Notebook angehängt ist, wird beendet. Während ein Notebook beendet wird, können Sie den Notebook-Editor nicht öffnen, anhalten oder löschen und Cluster nicht wechseln.
Angehalten	Das Notebook wurde heruntergefahren. Sie können das Notebook auf demselben Cluster starten, solange der Cluster noch ausgeführ t wird. Sie können Cluster wechseln und den Cluster löschen.

Status	Bedeutung
Löschen	Der Cluster wird aus der Liste der verfügbaren Cluster entfernt. Die Notebook-Datei *NotebookName* `.ipynb` verbleibt in Amazon S3 und es fallen weiterhin Gebühren für die Speicherung an.

Arbeiten mit dem Notebook-Editor

Ein Vorteil der Verwendung eines EMR Notebooks besteht darin, dass Sie das Notizbuch in Jupyter oder JupyterLab direkt von der Konsole aus starten können.

Bei EMR Notebooks ist der Notebook-Editor, auf den Sie von der EMR Amazon-Konsole aus zugreifen, der vertraute Open-Source-Jupyter Notebook-Editor oder. JupyterLab Da der Notebook-Editor in der EMR Amazon-Konsole gestartet wird, ist die Konfiguration des Zugriffs effizienter als bei einem Notebook, das auf einem EMR Amazon-Cluster gehostet wird. Sie müssen den Client eines Benutzers nicht so konfigurierenSSH, dass er über Sicherheitsgruppenregeln und Proxykonfigurationen auf das Internet zugreifen kann. Wenn ein Benutzer über ausreichende Berechtigungen verfügt, kann er einfach den Notebook-Editor in der EMR Amazon-Konsole öffnen.

In Amazon kann jeweils nur ein Benutzer ein EMR Notizbuch öffnenEMR. Wenn ein anderer Benutzer versucht, ein EMR Notizbuch zu öffnen, das bereits geöffnet ist, tritt ein Fehler auf.

> **Important**
>
> Amazon EMR erstellt URL für jede Notebook-Editor-Sitzung eine eindeutige Vorsignierung, die nur für kurze Zeit gültig ist. Wir empfehlen, den Notizbuch-Editor URL nicht mit anderen zu teilen. Dies stellt ein Sicherheitsrisiko dar, da die Empfänger Ihre Rechte zur Bearbeitung des Notizbuchs URL übernehmen und den Notizbuchcode für die gesamte Lebensdauer von ausführenURL. Wenn andere Benutzer Zugriff auf ein Notizbuch benötigen, gewähren Sie ihrem Benutzer über Berechtigungsrichtlinien Berechtigungen und stellen Sie sicher, dass die Servicerolle für EMR Notebooks Zugriff auf den Amazon S3 S3-Standort hat. Weitere Informationen erhalten Sie unter the section called “Sicherheit” und Servicerolle für EMR Notebooks.

Um den Notizbuch-Editor für ein EMR Notizbuch zu öffnen

1. Wählen Sie einen Notebook mit dem Status Ready (Bereit) oder Pending (Ausstehend) in der Liste Notebooks aus.
2. Wählen Sie Öffnen in JupyterLab oder Öffnen in Jupyter.

   Für den JupyterLab oder den Jupyter Notebook-Editor wird ein neuer Browser-Tab geöffnet.
3. Wählen Sie im Menü Kernel die Option Change Kernel (Kernel ändern) und wählen Sie dann den Kernel für Ihre Programmiersprache aus.

   Sie können jetzt Code innerhalb des Notebook-Editors schreiben und ausführen.

Speichern der Inhalte eines Notebooks

Wenn Sie im Notebook-Editor arbeiten, werden die Inhalte von Notebook-Zellen und Ausgaben automatisch regelmäßig in der Notebook-Datei in Amazon S3 gespeichert. Ein Notebook ohne Änderungen seit der letzten Bearbeitung von Zellen zeigt den Eintrag (autosaved) (automatisch gespeichert) neben dem Notebook-Namen im Editor an. Wenn Änderungen noch nicht gespeichert wurden, wird unsaved changes (nicht gespeicherte Änderungen) angezeigt.

Sie können ein Notebook manuell speichern. Wählen Sie im Menü Datei die Option Speichern und Checkpoint oder drücken Sie CTRL +S. Dadurch wird eine Datei mit dem Namen *NotebookName*`.ipynb` in einem Checkpoints-Ordner innerhalb des Notizbuchordners in Amazon S3 erstellt. Beispiel, `s3://`*MyBucket*`/`*MyNotebookFolder*`/`*NotebookID*`/checkpoints/`*NotebookName*`.ipynb`. Nur die aktuelle Prüfpunktdatei wird an diesem Speicherort gespeichert.

Wechseln von Clustern

Sie können den Cluster ändern, an den ein EMR Notizbuch angeschlossen ist, ohne den Inhalt des Notizbuchs selbst zu ändern. Sie können Cluster nur für Notebooks mit dem Status Stopped (Angehalten) wechseln.

Um den Cluster eines EMR Notebooks zu ändern

1. Wenn das Notebook, das Sie wechseln möchten, ausgeführt wird, wählen Sie dieses in der Liste Notebooks und anschließend Stop (Anhalten) aus.
2. Wenn das Notebook den Status Stopped (Angehalten) aufweist, wählen Sie das Notebook in der Liste Notebooks und anschließend View details (Details anzeigen) aus.

3. Wählen Sie Change cluster (Cluster wechseln).
4. Wenn Sie einen aktiven Cluster haben, auf dem Hadoop, Spark und Livy ausgeführt werden und an den Sie das Notebook anfügen möchten, behalten Sie die Standardeinstellung bei und wählen Sie einen Cluster aus der Liste aus. Es werden nur Cluster aufgeführt, die diesen Anforderungen entsprechen.

   –oder–

   Wählen Sie Create a cluster (Cluster erstellen) und anschließend die Clusteroptionen. Weitere Informationen finden Sie unter Cluster-Voraussetzungen.
5. Wählen Sie eine Option für Security groups (Sicherheitsgruppen) und anschließend Change cluster and start notebook (Cluster wechseln und Notebook starten).

Löschen von Notebooks und Notebook-Dateien

Wenn Sie ein EMR Notizbuch mit der EMR Amazon-Konsole löschen, löschen Sie das Notizbuch aus der Liste der verfügbaren Notizbücher. Notebook-Dateien verbleiben jedoch in Amazon S3 und es fallen weiterhin Speicherkosten an.

So löschen Sie ein Notizbuch und entfernen die zugehörigen Dateien

1. Öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/elasticmapreduce/.
2. Wählen Sie Notebooks, wählen Sie Ihr Notebook aus der Liste und anschließend View details (Details anzeigen) aus.
3. Wählen Sie das Ordnersymbol neben dem Speicherort des Notebooks und kopieren Sie das URL, was im Muster enthalten ist`s3://MyNotebookLocationPath/NotebookID/`.
4. Wählen Sie Löschen.

   Das Notebook wird aus der Liste entfernt und die Notebook-Details können nicht mehr angezeigt werden.
5. Befolgen Sie die Anweisungen für Wie kann ich Ordner aus einem S3-Bucket löschen? im Benutzerhandbuch für Amazon Simple Storage Service. Navigieren Sie zum Bucket und Ordner aus Schritt 3.

   –oder–

Wenn Sie das AWS CLI installiert haben, öffnen Sie eine Befehlszeile und geben Sie den Befehl am Ende dieses Absatzes ein. Ersetzen Sie den Amazon-S3-Speicherort mit dem oben kopierten Speicherort. Stellen Sie sicher, dass der mit den Zugriffsschlüsseln eines Benutzers konfiguriert AWS CLI ist, der berechtigt ist, den Amazon S3 S3-Standort zu löschen. Weitere Informationen finden Sie unter Konfigurieren der AWS CLI im AWS Command Line Interface -Leitfaden.

```
aws s3 rm s3://MyNotebookLocationPath/NotebookID
```

Freigeben von Notebook-Dateien

Jedes EMR Notizbuch wird in Amazon S3 als eine Datei mit dem Namen gespeichert*NotebookName*.ipynb. Solange eine Notizbuchdatei mit derselben Version von Jupyter Notebook kompatibel ist, auf der EMR Notebooks basiert, können Sie das Notizbuch als Notizbuch öffnen. EMR

Der einfachste Weg, eine Notebook-Datei von einem anderen Benutzer zu öffnen, besteht darin, die*.ipynb-Datei eines anderen Benutzers in Ihrem lokalen Dateisystem zu speichern und dann die Upload-Funktion in Jupyter und Editoren zu verwenden. JupyterLab

Mit diesem Verfahren können Sie Notizbücher verwenden, die von anderen geteilt wurden, EMR Notizbücher, die in der Jupyter-Community geteilt wurden, oder um ein Notizbuch wiederherzustellen, das von der Konsole gelöscht wurde, obwohl Sie die Notizbuchdatei noch haben.

Um eine andere Notizbuchdatei als Grundlage für ein Notizbuch zu verwenden EMR

1. Bevor Sie fortfahren, schließen Sie den Notizbuch-Editor für alle Notizbücher, mit denen Sie arbeiten werden, und beenden Sie dann das Notizbuch, falls es sich um ein EMR Notizbuch handelt.
2. Erstellen Sie ein EMR Notizbuch und geben Sie einen Namen dafür ein. Der Name, den Sie für das Notebook eingeben, wird der Name der Datei sein, die Sie ersetzen müssen. Der neue Dateiname muss genau mit diesem Dateinamen übereinstimmen.
3. Notieren Sie sich den Speicherort in Amazon S3, den Sie für das Notebook wählen. Die Datei, die Sie ersetzen, befindet sich in einem Ordner mit einem Pfad und Dateinamen, die dem folgenden Muster entsprechen: s3://*MyNotebookLocation*/*NotebookID*/*MyNotebookName*.ipynb.
4. Halten Sie das Notebook an.

5. Ersetzen Sie die alte Notebook-Datei im Amazon-S3-Speicherort mit der neuen Datei, die denselben Namen trägt.

   Der folgende AWS CLI Befehl für Amazon S3 ersetzt eine Datei, die auf einem lokalen Computer gespeichert wurde, der `SharedNotebook.ipynb` nach einem EMR Notizbuch benannt ist MyNotebook, mit dem Namen, einer ID von `e-12A3BCDEFJHIJKLMNO45PQRST` und erstellt mit dem in Amazon S3 `MyBucket/MyNotebooksFolder` angegebenen Namen. Weitere Informationen zur Verwendung der Amazon S3 Konsole zum Kopieren und Ersetzen von Dateien finden Sie unter Objekte hochladen, herunterladen und verwalten im Benutzerhandbuch für Amazon Simple Storage Service..

   ```
   aws s3 cp SharedNotebook.ipynb s3://MyBucket/
   MyNotebooksFolder/-12A3BCDEFJHIJKLMNO45PQRST/MyNotebook.ipynb
   ```

Beispielbefehle zur programmgesteuerten Ausführung von EMR Notebooks

> **Note**
>
> EMRNotebooks sind in der Konsole als EMR Studio-Workspaces verfügbar. Mit der Schaltfläche „Arbeitsbereich erstellen“ in der Konsole können Sie neue Notizbücher erstellen. Um auf Workspaces zuzugreifen oder diese zu erstellen, benötigen EMR Notebook-Benutzer zusätzliche IAM Rollenberechtigungen. Weitere Informationen finden Sie unter Amazon EMR Notebooks sind Amazon EMR Studio-Workspaces in der Konsole und EMRAmazon-Konsole.

Übersicht

Sie können EMR Notebooks mit Ausführung APIs über ein Skript oder über die Befehlszeile ausführen. Wenn Sie EMR Notebook-Ausführungen außerhalb der AWS Konsole starten, beenden, auflisten und beschreiben, können Sie ein Notebook programmgesteuert steuern. EMR Sie können verschiedene Parameterwerte an ein Notebook mit einer parametrisierten Notebookzelle übergeben. Dadurch entfällt die Notwendigkeit, für jeden neuen Satz von Parameterwerten eine Kopie des Notebooks zu erstellen. Weitere Informationen finden Sie unter EMRAPIAktionen von Amazon.

Sie können EMR Notizbuchausführungen mit CloudWatch Amazon-Ereignissen und AWS Lambda stapeln. Weitere Informationen finden Sie unter Verwendung AWS Lambda mit Amazon CloudWatch Events.

Rollenberechtigungen für die programmatische Ausführung

Um die programmgesteuerte Ausführung mit EMR Notebooks zu verwenden, müssen Sie Benutzerberechtigungen mit den folgenden Richtlinien konfigurieren:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowExecutionActions",
            "Effect": "Allow",
            "Action": [
                "elasticmapreduce:StartNotebookExecution",
                "elasticmapreduce:DescribeNotebookExecution",
                "elasticmapreduce:ListNotebookExecutions"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AllowPassingServiceRole",
            "Effect": "Allow",
            "Action": [
                "iam:PassRole"
            ],
            "Resource": "arn:aws:iam::account-id:role/EMR_Notebooks_DefaultRole"
        }
    ]
}
```

Wenn Sie Notebooks programmgesteuert auf einem EMR EMR Notebooks-Cluster ausführen, müssen Sie die folgenden zusätzlichen Berechtigungen hinzufügen:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowRetrievingManagedEndpointCredentials",
            "Effect": "Allow",
```

```
            "Action": [
                "emr-containers:GetManagedEndpointSessionCredentials"
            ],
            "Resource": [
                "arn:aws:emr-containers:region:account-id:/virtualclusters/virtual-
cluster-id/endpoints/managed-endpoint-id"
            ],
            "Condition": {
                "StringEquals": {
                    "emr-containers:ExecutionRoleArn": [
                        "arn:aws:iam::account-id:role/emr-on-eks-execution-role"
                    ]
                }
            }
        },
        {
            "Sid": "AllowDescribingManagedEndpoint",
            "Effect": "Allow",
            "Action": [
                "emr-containers:DescribeManagedEndpoint"
            ],
            "Resource": [
                "arn:aws:emr-containers:region:account-id:/virtualclusters/virtual-
cluster-id/endpoints/managed-endpoint-id"
            ]
        }
    ]
}
```

Einschränkungen bei der programmatischen Ausführung

- Pro Konto werden maximal 100 gleichzeitige Ausführungen unterstützt. AWS-Region
- Eine Ausführung wird beendet, wenn sie länger als 30 Tage läuft.
- Die programmatische Ausführung von Notebooks wird mit interaktiven Amazon EMR Serverless-Anwendungen nicht unterstützt.

Beispiele für die programmatische Ausführung von Notebooks EMR

Die folgenden Abschnitte enthalten mehrere Beispiele für die programmatische EMR Notebook-Ausführung mit Boto3 SDK (Python) und Ruby: AWS CLI

- Beispiele für CLI Befehle zur Ausführung von Notebooks
- Python-Beispiele für die Notebook-Ausführung
- Ruby-Beispiele für die Ausführung von Notebooks

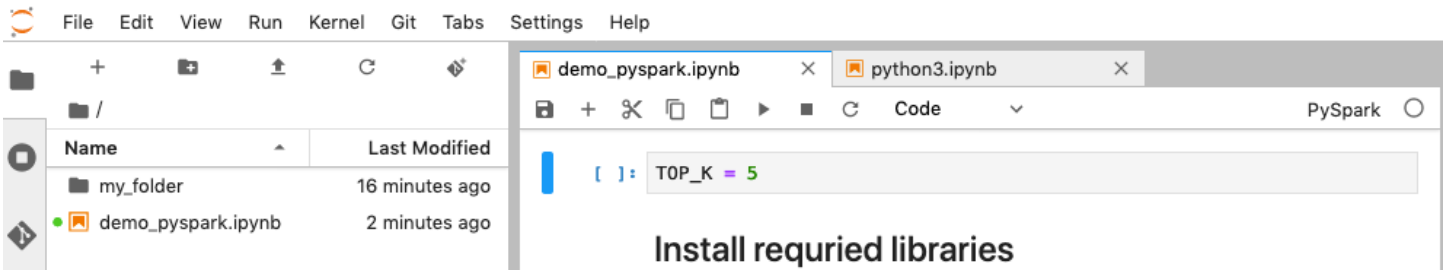
Sie können parametrisierte Notebooks auch als Teil von geplanten Workflows mit einem Orchestrierungstool wie Apache Airflow oder Amazon Managed Workflows for Apache Airflow () ausführen. MWAA Weitere Informationen finden Sie im Big Data-Blog unter Orchestrierung von Analyseaufträgen auf EMR Notebooks mithilfe von MWAA Notebooks.AWS

Beispiele für CLI Befehle zur Ausführung von Notebooks

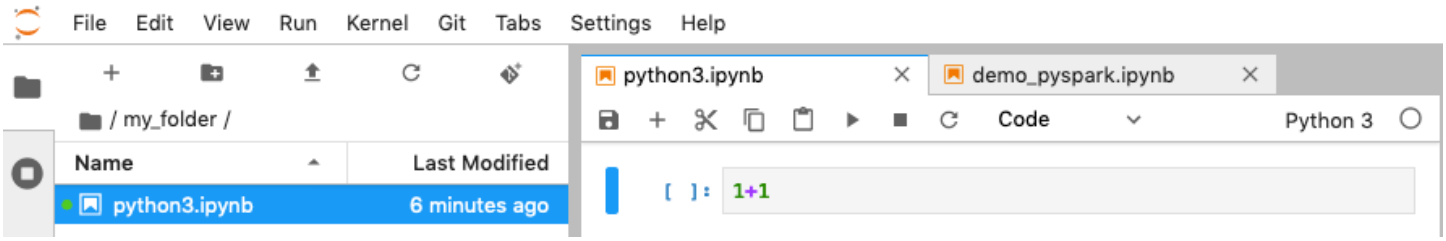
> **Note**
>
> EMRNotizbücher sind als EMR Studio-Arbeitsbereiche in der Konsole verfügbar. Mit der Schaltfläche „Arbeitsbereich erstellen“ in der Konsole können Sie neue Notizbücher erstellen. Um auf Workspaces zuzugreifen oder diese zu erstellen, benötigen EMR Notebook-Benutzer zusätzliche IAM Rollenberechtigungen. Weitere Informationen finden Sie unter Amazon EMR Notebooks sind Amazon EMR Studio-Workspaces in der Konsole und EMRAmazon-Konsole.

Im folgenden Beispiel wird das Demo-Notizbuch aus der EMR Notebooks-Konsole verwendet. Um das Notebook zu finden, verwenden Sie den Dateipfad relativ zum Home-Verzeichnis. In diesem Beispiel gibt es zwei Notebookdateien, die Sie ausführen können: `demo_pyspark.ipynb` und `my_folder/python3.ipynb`.

Der relative Pfad für die Datei `demo_pyspark.ipynb` ist `demo_pyspark.ipynb`, wie unten dargestellt.



Der relative Pfad für `python3.ipynb` ist `my_folder/python3.ipynb`, wie unten dargestellt.



Informationen zu den EMR API `NotebookExecution` Amazon-Aktionen finden Sie unter EMRAPIAmazon-Aktionen. .

Ein Notebook ausführen

Sie können den verwenden AWS CLI , um Ihr Notebook mit der `start-notebook-execution` Aktion auszuführen, wie die folgenden Beispiele zeigen.

Example — Ausführung eines EMR Notebooks in einem EMR Studio-Workspace mit einem Amazon-Cluster EMR (läuft auf AmazonEC2)

```
aws emr --region us-east-1 \
start-notebook-execution \
--editor-id e-ABCDEFG123456 \
--notebook-params '{"input_param":"my-value", "good_superhero":["superman", "batman"]}'
 \
--relative-path test.ipynb \
--notebook-execution-name my-execution \
--execution-engine '{"Id" : "j-1234ABCD123"}' \
--service-role EMR_Notebooks_DefaultRole

{
    "NotebookExecutionId": "ex-ABCDEFGHIJ1234ABCD"
}
```

Example — Ausführen eines EMR Notebooks in einem EMR Studio-Workspace mit einem EMR Notebooks-Cluster

```
aws emr start-notebook-execution \
    --region us-east-1 \
    --service-role EMR_Notebooks_DefaultRole \
    --environment-variables '{"KERNEL_EXTRA_SPARK_OPTS": "--conf
 spark.executor.instances=1", "KERNEL_LAUNCH_TIMEOUT": "350"}' \
    --output-notebook-format HTML \
```

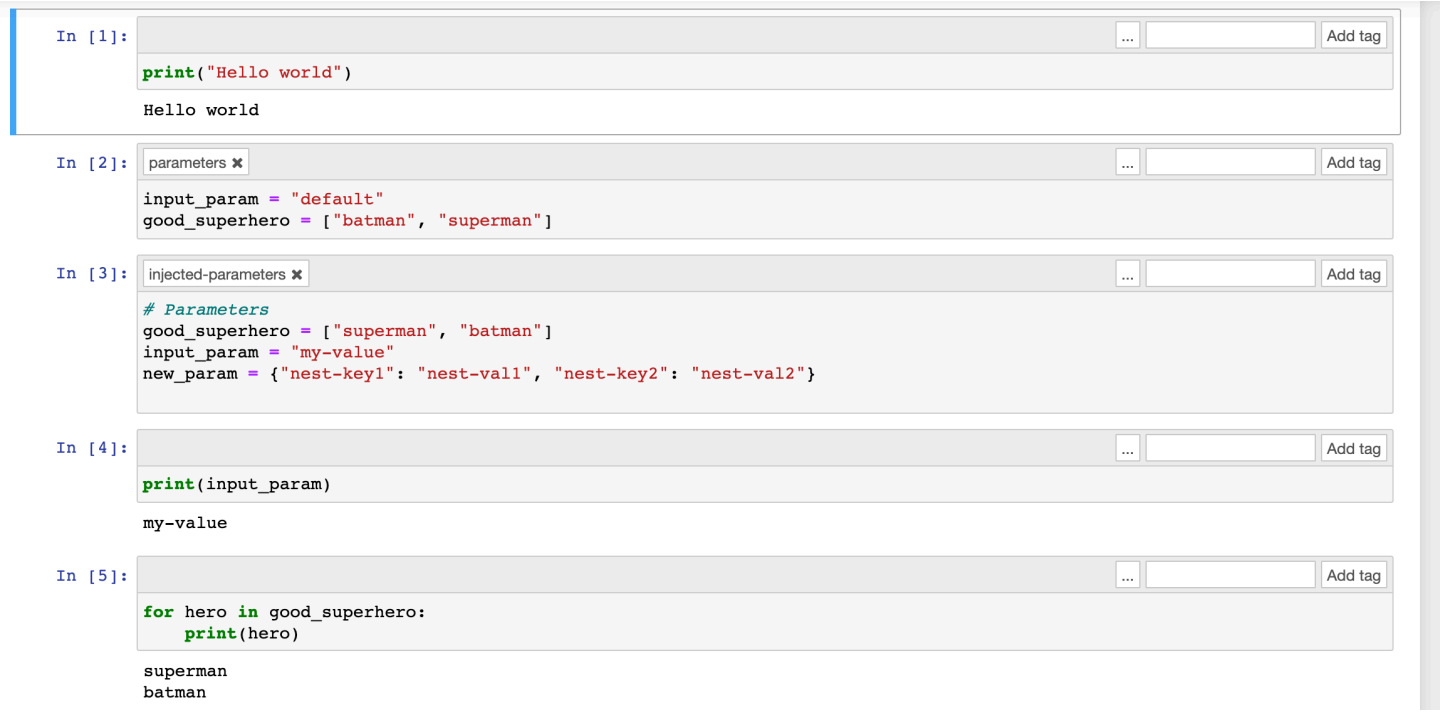
```
    --execution-engine Id=arn:aws:emr-containers:us-west-2:account-id:/
virtualclusters/ABCDEFG/
endpoints/ABCDEF,Type=EMR_ON_EKS,ExecutionRoleArn=arn:aws:iam::account-
id:role/execution-role \
    --editor-id e-ABCDEFG \
    --relative-path EMRonEKS-spark_python.ipynb
```

Example — Ausführung eines EMR Notebooks unter Angabe seines Amazon S3 S3-Standorts

```
aws emr start-notebook-execution \
    --region us-east-1 \
    --notebook-execution-name my-execution-on-emr-on-eks-cluster \
    --service-role EMR_Notebooks_DefaultRole \
    --environment-variables '{"KERNEL_EXTRA_SPARK_OPTS": "--conf
 spark.executor.instances=1", "KERNEL_LAUNCH_TIMEOUT": "350"}' \
    --output-notebook-format HTML \
    --execution-engine Id=arn:aws:emr-containers:us-west-2:account-id:/
virtualclusters/ABCDEF/
endpoints/ABCDEF,Type=EMR_ON_EKS,ExecutionRoleArn=arn:aws:iam::account-
id:role/execution-role \
    --notebook-s3-location '{"Bucket": "your-s3-bucket","Key": "s3-prefix-to-notebook-
location/EMRonEKS-spark_python.ipynb"}' \
    --output-notebook-s3-location '{"Bucket": "your-s3-bucket","Key": "s3-prefix-for-
storing-output-notebook"}'
```

Notebook-Ausgabe

Hier ist die Ausgabe eines Beispiel-Notebooks. Zelle 3 zeigt die neu eingegebenen Parameterwerte.



Ein Notebook beschreiben

Sie können die `describe-notebook-execution`-Aktion verwenden, um auf Informationen über eine bestimmte Notebook-Ausführung zuzugreifen.

```
aws emr --region us-east-1 \
describe-notebook-execution --notebook-execution-id ex-IZWZZVR9DKQ9WQ7VZWXJZR29UGHTE

{
    "NotebookExecution": {
        "NotebookExecutionId": "ex-IZWZZVR9DKQ9WQ7VZWXJZR29UGHTE",
        "EditorId": "e-BKTM2DIHXBEDRU44ANWRKIU8N",
        "ExecutionEngine": {
            "Id": "j-2QMOV6JAX1TS2",
            "Type": "EMR",
            "MasterInstanceSecurityGroupId": "sg-05ce12e58cd4f715e"
        },
        "NotebookExecutionName": "my-execution",
        "NotebookParams": "{\"input_param\":\"my-value\", \"good_superhero\":
[\"superman\", \"batman\"]}",
        "Status": "FINISHED",
        "StartTime": 1593490857.009,
        "Arn": "arn:aws:elasticmapreduce:us-east-1:123456789012:notebook-execution/ex-
IZWZZVR9DKQ9WQ7VZWXJZR29UGHTE",
```

```
        "LastStateChangeReason": "Execution is finished for cluster j-2QMOV6JAX1TS2.",
        "NotebookInstanceSecurityGroupId": "sg-0683b0a39966d4a6a",
        "Tags": []
    }
}
```

Ein Notebook stoppen

Wenn auf Ihrem Notebook eine Ausführung läuft, die Sie beenden möchten, können Sie dies mit dem `stop-notebook-execution`-Befehl tun.

```
# stop a running execution
aws emr --region us-east-1 \
stop-notebook-execution --notebook-execution-id ex-IZWZX78UVPAATC8LHJR129B1RBN4T


# describe it
aws emr --region us-east-1 \
describe-notebook-execution --notebook-execution-id ex-IZWZX78UVPAATC8LHJR129B1RBN4T

{
    "NotebookExecution": {
        "NotebookExecutionId": "ex-IZWZX78UVPAATC8LHJR129B1RBN4T",
        "EditorId": "e-BKTM2DIHXBEDRU44ANWRKIU8N",
        "ExecutionEngine": {
            "Id": "j-2QMOV6JAX1TS2",
            "Type": "EMR"
        },
        "NotebookExecutionName": "my-execution",
        "NotebookParams": "{\"input_param\":\"my-value\", \"good_superhero\":
[\"superman\", \"batman\"]}",
        "Status": "STOPPED",
        "StartTime": 1593490876.241,
        "Arn": "arn:aws:elasticmapreduce:us-east-1:123456789012:editor-execution/ex-
IZWZX78UVPAATC8LHJR129B1RBN4T",
        "LastStateChangeReason": "Execution is stopped for cluster j-2QMOV6JAX1TS2.
 Internal error",
        "Tags": []
    }
}
```

Listet die Ausführungen für ein Notebook nach Startzeit auf

Sie können einen `--from`-Parameter an `list-notebook-executions` übergeben, um die Ausführungen Ihres Notebooks nach Startzeit aufzulisten.

```
# filter by start time
aws emr --region us-east-1 \
list-notebook-executions --from 1593400000.000

{
    "NotebookExecutions": [
        {
            "NotebookExecutionId": "ex-IZWZX78UVPAATC8LHJR129B1RBN4T",
            "EditorId": "e-BKTM2DIHXBEDRU44ANWRKIU8N",
            "NotebookExecutionName": "my-execution",
            "Status": "STOPPED",
            "StartTime": 1593490876.241
        },
        {
            "NotebookExecutionId": "ex-IZWZZVR9DKQ9WQ7VZWXJZR29UGHTE",
            "EditorId": "e-BKTM2DIHXBEDRU44ANWRKIU8N",
            "NotebookExecutionName": "my-execution",
            "Status": "RUNNING",
            "StartTime": 1593490857.009
        },
        {
            "NotebookExecutionId": "ex-IZWZYRS0M14L5V95WZ9OQ399SKMNW",
            "EditorId": "e-BKTM2DIHXBEDRU44ANWRKIU8N",
            "NotebookExecutionName": "my-execution",
            "Status": "STOPPED",
            "StartTime": 1593490292.995
        },
        {
            "NotebookExecutionId": "ex-IZX009ZK83IVY5E33VH8MDMELVK8K",
            "EditorId": "e-BKTM2DIHXBEDRU44ANWRKIU8N",
            "NotebookExecutionName": "my-execution",
            "Status": "FINISHED",
            "StartTime": 1593489834.765
        },
        {
            "NotebookExecutionId": "ex-IZWZXOZF88JWDF9J09GJ91R57VI0N",
            "EditorId": "e-BKTM2DIHXBEDRU44ANWRKIU8N",
            "NotebookExecutionName": "my-execution",
```

```
            "Status": "FAILED",
            "StartTime": 1593488934.688
        }
    ]
}
```

Listet die Ausführungen für ein Notebook nach Startzeit und Status auf

Der `list-notebook-executions`-Befehl kann auch einen `--status`-Parameter verwenden, um die Ergebnisse zu filtern.

```
# filter by start time and status
aws emr --region us-east-1 \
list-notebook-executions --from 1593400000.000 --status FINISHED
{
    "NotebookExecutions": [
        {
            "NotebookExecutionId": "ex-IZWZZVR9DKQ9WQ7VZWXJZR29UGHTE",
            "EditorId": "e-BKTM2DIHXBEDRU44ANWRKIU8N",
            "NotebookExecutionName": "my-execution",
            "Status": "FINISHED",
            "StartTime": 1593490857.009
        },
        {
            "NotebookExecutionId": "ex-IZX009ZK83IVY5E33VH8MDMELVK8K",
            "EditorId": "e-BKTM2DIHXBEDRU44ANWRKIU8N",
            "NotebookExecutionName": "my-execution",
            "Status": "FINISHED",
            "StartTime": 1593489834.765
        }
    ]
}
```

Python-Beispiele für die Notebook-Ausführung

> **Note**
>
> EMRNotizbücher sind als EMR Studio-Arbeitsbereiche in der Konsole verfügbar. Mit der Schaltfläche „Arbeitsbereich erstellen“ in der Konsole können Sie neue Notizbücher erstellen. Um auf Workspaces zuzugreifen oder diese zu erstellen, benötigen EMR Notebook-Benutzer

zusätzliche IAM Rollenberechtigungen. Weitere Informationen finden Sie unter Amazon EMR Notebooks sind Amazon EMR Studio-Workspaces in der Konsole und EMRAmazon-Konsole.

Das folgende Codebeispiel ist eine SDK For-Python-Datei (Boto3)`demo.py`, die die Notebook-Ausführung zeigt. APIs

Informationen zu den EMR API `NotebookExecution` Amazon-Aktionen finden Sie unter EMRAPIAmazon-Aktionen.

```
import boto3,time

emr = boto3.client(
    'emr',
    region_name='us-west-1'
)

start_resp = emr.start_notebook_execution(
    EditorId='e-40AC8ZO6EGGCPJ4DLO48KGGGI',
    RelativePath='boto3_demo.ipynb',
    ExecutionEngine={'Id':'j-1HYZS6JQKV11Q'},
    ServiceRole='EMR_Notebooks_DefaultRole'
)

execution_id = start_resp["NotebookExecutionId"]
print(execution_id)
print("\n")

describe_response = emr.describe_notebook_execution(NotebookExecutionId=execution_id)

print(describe_response)
print("\n")

list_response = emr.list_notebook_executions()
print("Existing notebook executions:\n")
for execution in list_response['NotebookExecutions']:
    print(execution)
    print("\n")

print("Sleeping for 5 sec...")
time.sleep(5)

print("Stop execution " + execution_id)
```

```
emr.stop_notebook_execution(NotebookExecutionId=execution_id)
describe_response = emr.describe_notebook_execution(NotebookExecutionId=execution_id)
print(describe_response)
print("\n")
```

Hier ist die Ausgabe vom ausgeführten `demo.py`.

```
ex-IZX56YJDW1D29Q1PHR32WABU2SAPK

{'NotebookExecution': {'NotebookExecutionId': 'ex-IZX56YJDW1D29Q1PHR32WABU2SAPK',
 'EditorId': 'e-40AC8ZO6EGGCPJ4DLO48KGGGI', 'ExecutionEngine': {'Id':
 'j-1HYZS6JQKV11Q', 'Type': 'EMR'}, 'NotebookExecutionName': '', 'Status': 'STARTING',
 'StartTime': datetime.datetime(2020, 8, 19, 0, 49, 19, 418000, tzinfo=tzlocal()),
 'Arn': 'arn:aws:elasticmapreduce:us-west-1:123456789012:notebook-execution/ex-
IZX56YJDW1D29Q1PHR32WABU2SAPK', 'LastStateChangeReason': 'Execution is starting
 for cluster j-1HYZS6JQKV11Q.', 'Tags': []}, 'ResponseMetadata': {'RequestId':
 '70f12c5f-1dda-45b7-adf6-964987d373b7', 'HTTPStatusCode': 200, 'HTTPHeaders': {'x-
amzn-requestid': '70f12c5f-1dda-45b7-adf6-964987d373b7', 'content-type': 'application/
x-amz-json-1.1', 'content-length': '448', 'date': 'Wed, 19 Aug 2020 00:49:22 GMT'},
 'RetryAttempts': 0}}

Existing notebook executions:

{'NotebookExecutionId': 'ex-IZX56YJDW1D29Q1PHR32WABU2SAPK', 'EditorId':
 'e-40AC8ZO6EGGCPJ4DLO48KGGGI', 'NotebookExecutionName': '', 'Status': 'STARTING',
 'StartTime': datetime.datetime(2020, 8, 19, 0, 49, 19, 418000, tzinfo=tzlocal())}


{'NotebookExecutionId': 'ex-IZX5ABS5PR1E5AHMFYEMX3JJIORRB', 'EditorId':
 'e-40AC8ZO6EGGCPJ4DLO48KGGGI', 'NotebookExecutionName': '', 'Status': 'RUNNING',
 'StartTime': datetime.datetime(2020, 8, 19, 0, 48, 36, 373000, tzinfo=tzlocal())}


{'NotebookExecutionId': 'ex-IZX5GLVXIU1HNI8BWVW057F6MF4VE', 'EditorId':
 'e-40AC8ZO6EGGCPJ4DLO48KGGGI', 'NotebookExecutionName': '', 'Status': 'FINISHED',
 'StartTime': datetime.datetime(2020, 8, 19, 0, 45, 14, 646000, tzinfo=tzlocal()),
 'EndTime': datetime.datetime(2020, 8, 19, 0, 46, 26, 543000, tzinfo=tzlocal())}


{'NotebookExecutionId': 'ex-IZX5CV8YDUO8JAIWMXN2VH32RUIT1', 'EditorId':
 'e-40AC8ZO6EGGCPJ4DLO48KGGGI', 'NotebookExecutionName': '', 'Status': 'FINISHED',
```

```
 'StartTime': datetime.datetime(2020, 8, 19, 0, 43, 5, 807000, tzinfo=tzlocal()),
 'EndTime': datetime.datetime(2020, 8, 19, 0, 44, 31, 632000, tzinfo=tzlocal())}


{'NotebookExecutionId': 'ex-IZX5AS0PPW55CEDEURZ9NSOWSUJZ6', 'EditorId':
 'e-40AC8ZO6EGGCPJ4DLO48KGGGI', 'NotebookExecutionName': '', 'Status': 'FINISHED',
 'StartTime': datetime.datetime(2020, 8, 19, 0, 42, 29, 265000, tzinfo=tzlocal()),
 'EndTime': datetime.datetime(2020, 8, 19, 0, 43, 48, 320000, tzinfo=tzlocal())}


{'NotebookExecutionId': 'ex-IZX57YF5Q53BKWLR4I5QZ14HJ7DRS', 'EditorId':
 'e-40AC8ZO6EGGCPJ4DLO48KGGGI', 'NotebookExecutionName': '', 'Status': 'FINISHED',
 'StartTime': datetime.datetime(2020, 8, 19, 0, 38, 37, 81000, tzinfo=tzlocal()),
 'EndTime': datetime.datetime(2020, 8, 19, 0, 40, 39, 646000, tzinfo=tzlocal())}

Sleeping for 5 sec...
Stop execution ex-IZX56YJDW1D29Q1PHR32WABU2SAPK
{'NotebookExecution': {'NotebookExecutionId': 'ex-IZX56YJDW1D29Q1PHR32WABU2SAPK',
 'EditorId': 'e-40AC8ZO6EGGCPJ4DLO48KGGGI', 'ExecutionEngine': {'Id':
 'j-1HYZS6JQKV11Q', 'Type': 'EMR'}, 'NotebookExecutionName': '', 'Status': 'STOPPING',
 'StartTime': datetime.datetime(2020, 8, 19, 0, 49, 19, 418000, tzinfo=tzlocal()),
 'Arn': 'arn:aws:elasticmapreduce:us-west-1:123456789012:notebook-execution/ex-
IZX56YJDW1D29Q1PHR32WABU2SAPK', 'LastStateChangeReason': 'Execution is being stopped
 for cluster j-1HYZS6JQKV11Q.', 'Tags': []}, 'ResponseMetadata': {'RequestId':
 '2a77ef73-c1c6-467c-a1d1-7204ab2f6a53', 'HTTPStatusCode': 200, 'HTTPHeaders': {'x-
amzn-requestid': '2a77ef73-c1c6-467c-a1d1-7204ab2f6a53', 'content-type': 'application/
x-amz-json-1.1', 'content-length': '453', 'date': 'Wed, 19 Aug 2020 00:49:30 GMT'},
 'RetryAttempts': 0}}
```

Ruby-Beispiele für die Ausführung von Notebooks

> **Note**
>
> EMRNotizbücher sind in der Konsole als EMR Studio-Workspaces verfügbar. Mit der Schaltfläche „Arbeitsbereich erstellen“ in der Konsole können Sie neue Notizbücher erstellen. Um auf Workspaces zuzugreifen oder diese zu erstellen, benötigen EMR Notebook-Benutzer zusätzliche IAM Rollenberechtigungen. Weitere Informationen finden Sie unter Amazon EMR Notebooks sind Amazon EMR Studio-Workspaces in der Konsole und EMRAmazon-Konsole.

Im Folgenden finden Sie Ruby-Codebeispiele, die die Verwendung der Notebook-Ausführung API demonstrieren.

```
# prepare an Amazon EMR client

emr = Aws::EMR::Client.new(
  region: 'us-east-1',
  access_key_id: 'AKIA...JKPKA',
  secret_access_key: 'rLMeu...vU0OLrAC1',
)
```

Starten der Notebook-Ausführung und Abrufen der Ausführungs-ID

In diesem Beispiel sind es der Amazon S3 S3-Editor und das EMR Notizbuchs`3://mybucket/notebooks/e-EA8VGAA429FEQTC8HC9ZHWISK/test.ipynb`.

Informationen zu den EMR API `NotebookExecution` Amazon-Aktionen finden Sie unter EMRAPIAmazon-Aktionen.

```
start_response = emr.start_notebook_execution({
    editor_id: "e-EA8VGAA429FEQTC8HC9ZHWISK",
    relative_path: "test.ipynb",

    execution_engine: {id: "j-3U82I95AMALGE"},

    service_role: "EMR_Notebooks_DefaultRole",
})


notebook_execution_id = start_resp.notebook_execution_id
```

Beschreibung der Ausführung des Notebooks und Ausdrucken der Details

```
describe_resp = emr.describe_notebook_execution({
    notebook_execution_id: notebook_execution_id
})
puts describe_resp.notebook_execution
```

Die Ausgabe der obigen Befehle wird wie folgt aussehen.

```
{
```

```
:notebook_execution_id=>"ex-IZX3VTVZWVWPP27KUB90BZ7V9IEDG",
:editor_id=>"e-EA8VGAA429FEQTC8HC9ZHWISK",
:execution_engine=>{:id=>"j-3U82I95AMALGE", :type=>"EMR", :master_instance_security_group_id=>n
:notebook_execution_name=>"",
:notebook_params=>nil,
:status=>"STARTING",
:start_time=>2020-07-23 15:07:07 -0700,
:end_time=>nil,
:arn=>"arn:aws:elasticmapreduce:us-east-1:123456789012:notebook-execution/ex-
IZX3VTVZWVWPP27KUB90BZ7V9IEDG",
:output_notebook_uri=>nil,
:last_state_change_reason=>"Execution is starting for cluster
 j-3U82I95AMALGE.", :notebook_instance_security_group_id=>nil,
:tags=>[]
}
```

Notebookfilter

```
"EditorId": "e-XXXX",           [Optional]
"From" : "1593400000.000",    [Optional]
"To" :
```

Beenden der Notebook-Ausführung

```
stop_resp = emr.stop_notebook_execution({
    notebook_execution_id: notebook_execution_id
})
```

Aktivieren des Identitätswechsels zur Überwachung von Spark-Benutzer- und -Aufgabenaktivitäten

> **Note**
>
> EMRNotizbücher sind in der Konsole als EMR Studio-Workspaces verfügbar. Mit der Schaltfläche „Arbeitsbereich erstellen“ in der Konsole können Sie neue Notizbücher erstellen. Um auf Workspaces zuzugreifen oder diese zu erstellen, benötigen EMR Notebook-Benutzer zusätzliche IAM Rollenberechtigungen. Weitere Informationen finden Sie unter Amazon EMR Notebooks sind Amazon EMR Studio-Workspaces in der Konsole und EMRAmazon-Konsole.

EMRNotebooks ermöglicht es Ihnen, den Benutzerwechsel auf einem Spark-Cluster zu konfigurieren. Mit dieser Funktion können Sie die Auftragsaktivität nachverfolgen, die innerhalb des Notebook-Editors initiiert wurde. Darüber hinaus verfügt EMR Notebooks über ein integriertes Jupyter Notebook-Widget, mit dem Sie die Spark-Jobdetails zusammen mit der Abfrageausgabe im Notebook-Editor anzeigen können. Das Widget ist standardmäßig verfügbar und erfordert keine spezielle Konfiguration. Um die Verlaufsserver anzeigen zu können, muss Ihr Client jedoch so konfiguriert sein, dass er EMR Amazon-Webschnittstellen anzeigt, die auf dem primären Knoten gehostet werden.

Einrichten der Spark-Benutzerkennung

Standardmäßig stammen Spark-Aufträge, die Benutzer mit dem Notebook-Editor übermitteln, scheinbar aus einer unbestimmten `livy`-Benutzeridentität. Sie können eine Benutzerkennung für den Cluster konfigurieren, damit diese Aufträge stattdessen mit der Benutzeridentität verknüpft werden, die den Code ausgeführt hat. HDFSBenutzerverzeichnisse auf dem primären Knoten werden für jede Benutzeridentität erstellt, die Code im Notizbuch ausführt. Beispiel: Wenn der Benutzer `NbUser1` Code aus dem Notebook-Editor ausführt, können Sie eine Verbindung mit dem Primärknoten herstellen und sehen, dass `hadoop fs -ls /user` das Verzeichnis `/user/user_NbUser1` zeigt.

Sie können diese Funktion aktivieren, indem Sie Eigenschaften in den Konfigurationsklassifizierungen `core-site` und `livy-conf` festlegen. Diese Funktion ist standardmäßig nicht verfügbar, wenn Sie Amazon einen Cluster zusammen mit einem Notebook EMR erstellen lassen. Weitere Informationen zur Verwendung von Konfigurationsklassifizierungen zur Anpassung von Anwendungen finden Sie unter Konfiguration von Anwendungen im Amazon EMR Release Guide.

Verwenden Sie die folgenden Konfigurationsklassifizierungen und Werte, um die Benutzeridentität für Notebooks zu aktivieren: EMR

```
[
    {
        "Classification": "core-site",
        "Properties": {
          "hadoop.proxyuser.livy.groups": "*",
          "hadoop.proxyuser.livy.hosts": "*"
        }
    },
    {
        "Classification": "livy-conf",
```

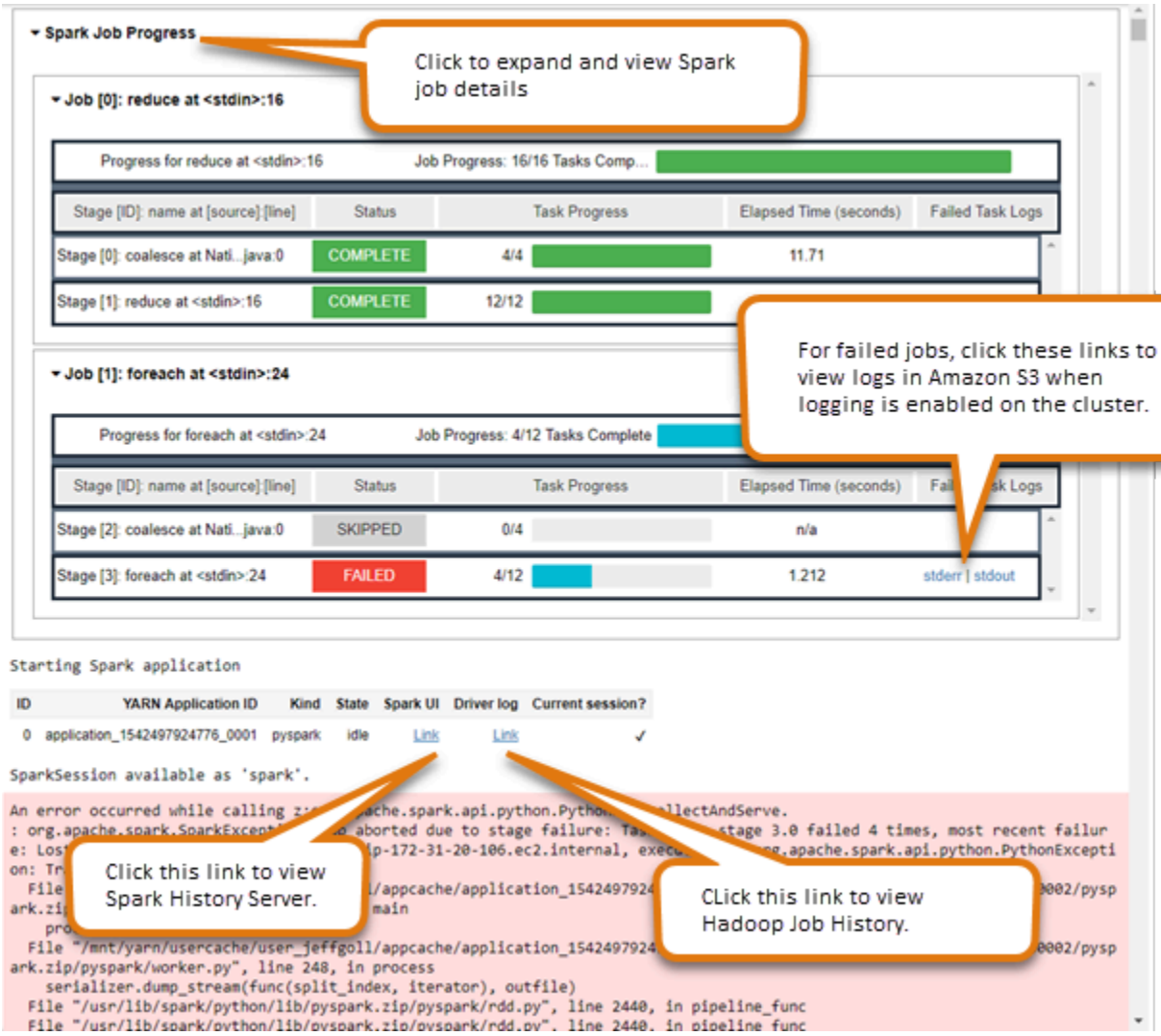
```
        "Properties": {
          "livy.impersonation.enabled": "true"
        }
    }
]
```

Verwenden des Spark-Widgets für die Auftragsüberwachung

Wenn Sie im Notebook-Editor Code ausführen, der Spark-Jobs auf dem EMR Cluster ausführt, enthält die Ausgabe ein Jupyter Notebook-Widget für die Spark-Jobüberwachung. Das Widget stellt Auftragsdetails, nützliche Links zur Spark-Verlaufsserverseite und zur Hadoop-Auftragsverlaufsseite sowie praktische Links zu Auftragsprotokollen in Amazon S3 für alle fehlgeschlagenen Aufträge bereit.

Um die Seiten des History-Servers auf dem primären Clusterknoten anzuzeigen, müssen Sie entsprechend einen SSH Client und einen Proxy einrichten. Weitere Informationen finden Sie unter Auf EMR Amazon-Clustern gehostete Weboberflächen anzeigen. Um Protokolle in Amazon S3 anzuzeigen, muss die Cluster-Protokollierung aktiviert sein. Dies ist die Standardeinstellung für neue Cluster. Weitere Informationen finden Sie unter In Amazon S3 archivierte Protokolldateien anzeigen.

Nachstehend finden Sie ein Beispiel für die Spark-Auftragsüberwachung.



EMRSicherheit und Zugriffskontrolle für Notebooks

> **Note**
>
> EMRNotebooks sind in der Konsole als EMR Studio-Workspaces verfügbar. Mit der Schaltfläche „Arbeitsbereich erstellen“ in der Konsole können Sie neue Notizbücher erstellen. Um auf Workspaces zuzugreifen oder diese zu erstellen, benötigen EMR Notebook-Benutzer zusätzliche IAM Rollenberechtigungen. Weitere Informationen finden Sie unter Amazon EMR Notebooks sind Amazon EMR Studio-Workspaces in der Konsole und EMRAmazon-Konsole.

Es stehen mehrere Funktionen zur Verfügung, mit denen Sie den Sicherheitsstatus von EMR Notebooks individuell anpassen können. Dadurch wird sichergestellt, dass nur autorisierte Benutzer

Zugriff auf ein EMR Notebook haben, mit Notebooks arbeiten und den Notebook-Editor verwenden können, um Code auf dem Cluster auszuführen. Diese Funktionen funktionieren zusammen mit den Sicherheitsfunktionen, die für Amazon EMR - und EMR Amazon-Cluster verfügbar sind. Weitere Informationen finden Sie unter Sicherheit bei Amazon EMR.

- Sie können AWS Identity and Access Management Richtlinienerklärungen zusammen mit Notebook-Tags verwenden, um den Zugriff einzuschränken. Weitere Informationen erhalten Sie unter So EMR arbeitet Amazon mit IAM und Beispiel für identitätsbasierte Richtlinienerklärungen für Notebooks EMR.
- EC2Amazon-Sicherheitsgruppen agieren als virtuelle Firewalls, die den Netzwerkverkehr zwischen der primären Instance des Clusters und dem Notebook-Editor steuern. Sie können Standardwerte verwenden oder diese Sicherheitsgruppen anpassen. Weitere Informationen finden Sie unter EC2Sicherheitsgruppen für EMR Notebooks angeben.
- Sie geben eine AWS Servicerolle an, die bestimmt, welche Berechtigungen ein EMR Notebook bei der Interaktion mit anderen AWS Diensten hat. Weitere Informationen finden Sie unter Servicerolle für EMR Notebooks.

Installieren und Verwenden von Kerneln und Bibliotheken

> **Note**
>
> EMRNotebooks sind in der Konsole als EMR Studio-Workspaces verfügbar. Mit der Schaltfläche „Arbeitsbereich erstellen“ in der Konsole können Sie neue Notizbücher erstellen. Um auf Workspaces zuzugreifen oder diese zu erstellen, benötigen EMR Notebook-Benutzer zusätzliche IAM Rollenberechtigungen. Weitere Informationen finden Sie unter Amazon EMR Notebooks sind Amazon EMR Studio-Workspaces in der Konsole und EMRAmazon-Konsole.

Jedes EMR Notebook wird mit einer Reihe vorinstallierter Bibliotheken und Kernel geliefert. Sie können zusätzliche Bibliotheken und Kernel in einem EMR Cluster installieren, wenn der Cluster Zugriff auf das Repository hat, in dem sich die Kernel und Bibliotheken befinden. Beispielsweise müssen Sie für Cluster in privaten Subnetzen möglicherweise die Netzwerkadressübersetzung (NAT) konfigurieren und einen Pfad angeben, über den der Cluster auf das öffentliche PyPI-Repository zugreifen kann, um eine Bibliothek zu installieren. Weitere Informationen zur Konfiguration des externen Zugriffs für verschiedene Netzwerkkonfigurationen finden Sie unter Szenarien und Beispiele im VPCAmazon-Benutzerhandbuch.

EMRServerlose Anwendungen verfügen über die folgenden vorinstallierten Bibliotheken für Python und: PySpark

- Python-Bibliotheken – ggplot, matplotlib, numpy, pandas, plotly, bokeh, scikit-learn, scipy, scipy
- PySpark Bibliotheken —ggplot,matplotlib,,numpy,pandas,plotly,bokeh, scikit-learn scipy scipy

Installieren von Kernels und Python-Bibliotheken auf einem Cluster-Primärknoten

Mit EMR Amazon-Release-Version 5.30.0 und höher, mit Ausnahme von 6.0.0, können Sie zusätzliche Python-Bibliotheken und -Kernel auf dem primären Knoten des Clusters installieren. Nach der Installation stehen diese Kernel und Bibliotheken allen Benutzern zur Verfügung, die ein an den Cluster angeschlossenes EMR Notebook ausführen. Auf diese Weise installierte Python-Bibliotheken sind nur für Prozesse verfügbar, die auf dem Primärknoten ausgeführt werden. Die Bibliotheken werden nicht auf Core- oder Aufgabenknoten installiert und sind für Executors, die auf diesen Knoten ausgeführt werden, nicht verfügbar.

> **Note**
>
> Für die EMR Amazon-Versionen 5.30.1, 5.31.0 und 6.1.0 müssen Sie zusätzliche Schritte unternehmen, um Kernel und Bibliotheken auf dem primären Knoten eines Clusters zu installieren.
> Um das Feature zu aktivieren, gehen Sie wie folgt vor:
>
> 1. Stellen Sie sicher, dass die der Servicerolle für Notebooks beigefügte Berechtigungsrichtlinie die folgende Aktion zulässtEMR:
>
>    ```
>    elasticmapreduce:ListSteps
>    ```
>
>   Weitere Informationen finden Sie unter Servicerolle für EMR Notebooks.
> 2. Verwenden Sie den AWS CLI , um einen Schritt auf dem Cluster auszuführen, der EMR Notebooks einrichtet, wie im folgenden Beispiel gezeigt. Sie müssen den Schrittnamen `EMRNotebooksSetup` verwenden. Ersetzen *`us-east-1`* mit der Region, in der sich Ihr Cluster befindet. Weitere Informationen finden Sie unter Hinzufügen von Schritten zu einem Cluster AWS CLI.
>
>    ```
>    aws emr add-steps --cluster-id MyClusterID --steps
>     Type=CUSTOM_JAR,Name=EMRNotebooksSetup,ActionOnFailure=CONTINUE,Jar=s3://us-
>    ```

```
east-1.elasticmapreduce/libs/script-runner/script-runner.jar,Args=["s3://
awssupportdatasvcs.com/bootstrap-actions/EMRNotebooksSetup/emr-notebooks-
setup.sh"]
```

Sie können Kernel und Bibliotheken mithilfe von `pip` oder `conda` im `/emr/notebook-env/bin`-Verzeichnis auf dem Primärknoten installieren.

Example – Installieren von Python-Bibliotheken

Führen Sie im Python3-Kernel den `%pip` Magic als Befehl in einer Notebook-Zelle aus, um Python-Bibliotheken zu installieren.

```
%pip install pmdarima
```

Möglicherweise müssen Sie den Kernel neu starten, um aktualisierte Pakete verwenden zu können. Sie können auch die `%%sh`-Spark-Magie zum Aufrufen von `pip` verwenden.

```
%%sh
/emr/notebook-env/bin/pip install -U matplotlib
/emr/notebook-env/bin/pip install -U pmdarima
```

Wenn Sie einen PySpark Kernel verwenden, können Sie entweder Bibliotheken auf dem Cluster mithilfe von `pip` Befehlen installieren oder Bibliotheken für Notebooks innerhalb eines Notebooks verwenden. PySpark

Um `pip` Befehle auf dem Cluster vom Terminal aus auszuführen, stellen Sie zunächst, wie die folgenden Befehle zeigenSSH, eine Verbindung zum Primärknoten her.

```
sudo pip3 install -U matplotlib
sudo pip3 install -U pmdarima
```

Alternativ können Sie Bibliotheken im Notebookbereich verwenden. Bei Bibliotheken für Notebooks ist Ihre Bibliotheksinstallation auf den Umfang Ihrer Sitzung beschränkt und erfolgt auf allen Spark-Executoren. Weitere Informationen finden Sie unter Verwenden von Notebook Bibliotheken.

Wenn Sie mehrere Python-Bibliotheken in einen PySpark Kernel packen möchten, können Sie auch eine isolierte virtuelle Python-Umgebung erstellen. Anwendungsbeispiele finden Sie unter VerwendenVirtualenv.

Um eine virtuelle Python-Umgebung in einer Sitzung zu erstellen, verwenden Sie die Spark-Eigenschaft `spark.yarn.dist.archives` aus dem `%%configure` magischen Befehl in der ersten Zelle in einem Notebook, wie das folgende Beispiel zeigt.

```
%%configure -f
{
   "conf": {
   "spark.yarn.appMasterEnv.PYSPARK_PYTHON":"./environment/bin/python",
   "spark.yarn.appMasterEnv.PYSPARK_DRIVER_PYTHON":"./environment/bin/python",
   "spark.yarn.dist.archives":"s3://DOC-EXAMPLE-BUCKET/prefix/
my_pyspark_venv.tar.gz#environment",
   "spark.submit.deployMode":"cluster"
   }
}
```

Auf ähnliche Weise können Sie eine Spark-Executor-Umgebung erstellen.

```
%%configure -f
{
   "conf": {
   "spark.yarn.appMasterEnv.PYSPARK_PYTHON":"./environment/bin/python",
   "spark.yarn.appMasterEnv.PYSPARK_DRIVER_PYTHON":"./environment/bin/python",
   "spark.executorEnv.PYSPARK_PYTHON":"./environment/bin/python",
   "spark.yarn.dist.archives":"s3://DOC-EXAMPLE-BUCKET/prefix/
my_pyspark_venv.tar.gz#environment",
   "spark.submit.deployMode":"cluster"
   }
}
```

Sie können es auch `conda` zur Installation von Python-Bibliotheken verwenden. Für die Verwendung von `conda` benötigen Sie keinen Sudo-Zugriff. Sie müssen sich mit SSH dem primären Knoten verbinden und dann vom Terminal `conda` aus starten. Weitere Informationen finden Sie unter Connect zum Primärknoten her mit SSH.

Example – Installieren eines Kernels

Das folgende Beispiel zeigt die Installation des Kotlin-Kernels mithilfe eines Terminalbefehls, während eine Verbindung zum Primärknoten eines Clusters besteht:

```
sudo /emr/notebook-env/bin/conda install kotlin-jupyter-kernel -c jetbrains
```

> **Note**
>
> Diese Anweisungen installieren keine Kernel-Abhängigkeiten. Wenn Ihr Kernel Abhängigkeiten von Drittanbietern hat, müssen Sie möglicherweise zusätzliche Einrichtungsschritte durchführen, bevor Sie den Kernel mit Ihrem Notebook verwenden können.

Überlegungen und Einschränkungen bei Bibliotheken für Notebooks

Beachten Sie bei der Verwendung von Bibliotheken im Format Notebook-Scoped Folgendes:

- Bibliotheken für Notebooks sind für Cluster verfügbar, die Sie mit EMR Amazon-Versionen 5.26.0 und höher erstellen.
- Bibliotheken für Notebooks sind nur für die Verwendung mit dem Kernel vorgesehen. PySpark
- Jeder Benutzer kann zusätzliche dedizierte Notebook-Bibliotheken innerhalb einer Notebook-Zelle installieren. Diese Bibliotheken stehen diesem Notebook-Benutzer nur während genau einer Notebook-Sitzung zur Verfügung. Wenn andere Benutzer dieselben Bibliotheken benötigen oder derselbe Benutzer dieselben Bibliotheken in einer anderen Sitzung benötigt, muss die Bibliothek neu installiert werden.
- Sie können nur die Bibliotheken deinstallieren, die mit dem installiert wurden. `install_pypi_package` API Sie können keine Bibliotheken deinstallieren, die auf dem Cluster vorinstalliert sind.
- Wenn dieselben Bibliotheken mit unterschiedlichen Versionen auf dem Cluster und als Notebook-Bibliotheken installiert sind, überschreibt die Version der Notebook-Bibliothek die Version der Cluster-Bibliothek.

Arbeiten mit Notebook-Bibliotheken

Um Bibliotheken zu installieren, muss Ihr EMR Amazon-Cluster Zugriff auf das PyPI-Repository haben, in dem sich die Bibliotheken befinden.

Die folgenden Beispiele zeigen einfache Befehle zum Auflisten, Installieren und Deinstallieren von Bibliotheken aus einer Notebook-Zelle heraus mithilfe des PySpark Kernels und. APIs Weitere Beispiele finden Sie im Beitrag Installieren von Python-Bibliotheken auf einem laufenden Cluster mit EMR Notebooks im AWS Big Data-Blog.

Example – Auflisten aktueller Bibliotheken

Der folgende Befehl listet die Python-Pakete auf, die für die aktuelle Spark-Notebook-Sitzung verfügbar sind. Hiermit werden Bibliotheken aufgelistet, die auf dem Cluster installiert sind, und Bibliotheken für Notebook-Bereiche.

```
sc.list_packages()
```

Example – Installieren der Celery-Bibliothek

Mit dem folgenden Befehl wird die Celery-Bibliothek als Notebook-Bibliothek installiert.

```
sc.install_pypi_package("celery")
```

Nach der Installation der Bibliothek bestätigt der folgende Befehl, dass die Bibliothek auf dem Spark-Treiber und den Executors verfügbar ist.

```
import celery
sc.range(1,10000,1,100).map(lambda x: celery.__version__).collect()
```

Example – Installieren der Arrow-Bibliothek unter Angabe der Version und des Repositorys

Mit dem folgenden Befehl wird die Arrow-Bibliothek als Bibliothek für Notebooks mit Angabe der Bibliotheksversion und des Repositorys installiert. URL

```
sc.install_pypi_package("arrow==0.14.0", "https://pypi.org/simple")
```

Example – Deinstallieren einer Bibliothek

Der folgende Befehl deinstalliert die Pfeilbibliothek und entfernt sie als Notebook-Bibliothek aus der aktuellen Sitzung.

```
sc.uninstall_package("arrow")
```

Git-basierte Repositorys mit Notebooks verknüpfen EMR

> **Note**
>
> EMRNotebooks sind in der Konsole als EMR Studio-Workspaces verfügbar. Mit der Schaltfläche „Arbeitsbereich erstellen“ in der Konsole können Sie neue Notizbücher erstellen.

Um auf Workspaces zuzugreifen oder diese zu erstellen, benötigen EMR Notebook-Benutzer zusätzliche IAM Rollenberechtigungen. Weitere Informationen finden Sie unter Amazon EMR Notebooks sind Amazon EMR Studio-Workspaces in der Konsole und EMRAmazon-Konsole.

Sie können Git-basierte Repositorys mit Ihren EMR Amazon-Notizbüchern verknüpfen, um Ihre Notizbücher in einer versionskontrollierten Umgebung zu speichern. Sie können einem Notebook bis zu drei Repositorys zuordnen. Folgende Git-basierte Services werden unterstützt:

- AWS CodeCommit
- GitHub
- Bitbucket
- GitLab

Die Verknüpfung von Git-basierten Repositorys mit Ihrem Notebook hat folgende Vorteile:

- Versionskontrolle – Sie können Codeänderungen in einem Versionskontrollsystem aufzeichnen, damit Sie den Verlauf Ihrer Änderungen überprüfen und selektiv rückgängig machen können.
- Zusammenarbeit – Kollegen, die in verschiedenen Notebooks arbeiten, können Code über Git-basierte Remote-Repositorys füreinander freigeben. Notebooks können Code aus Remote-Repositorys klonen oder zusammenführen und Änderungen in diese Remote-Repositorys zurückübertragen.
- Wiederverwendung von Code — Viele Jupyter-Notebooks, die Techniken der Datenanalyse oder des maschinellen Lernens demonstrieren, sind in öffentlich gehosteten Repositorys verfügbar, wie z. GitHub Sie können Ihre Notebooks mit einem Repository verknüpfen, um die in diesem Repository enthaltenen Jupyter-Notebooks wiederzuverwenden.

Um Git-basierte Repositorys mit EMR Notebooks zu verwenden, fügen Sie die Repositorys als Ressourcen in der EMR Amazon-Konsole hinzu, ordnen Anmeldeinformationen für Repositorys zu, die eine Authentifizierung erfordern, und verknüpfen sie mit Ihren Notebooks. Sie können eine Liste der Repositorys, die in Ihrem Konto gespeichert sind, und Details zu jedem Repository in der EMR Amazon-Konsole einsehen. Sie können ein vorhandenes Git-basiertes Repository mit einem Notebook verknüpfen, wenn Sie es erstellen.

Themen

- Voraussetzungen und Überlegungen

- Ein Git-basiertes Repository zu Amazon hinzufügen EMR
- Aktualisieren oder Löschen eines Git-basierten Repositorys
- Verknüpfen oder Aufheben der Verknüpfung eines Git-basierten Repositorys
- Erstellen eines neuen Notebooks mit einem zugehörigen Git-Repository
- Verwenden von Git-Repositorys in einem Notebook

Voraussetzungen und Überlegungen

> **Note**
>
> EMRNotizbücher sind in der Konsole als EMR Studio-Workspaces verfügbar. Mit der Schaltfläche „Arbeitsbereich erstellen“ in der Konsole können Sie neue Notizbücher erstellen. Um auf Workspaces zuzugreifen oder diese zu erstellen, benötigen EMR Notebook-Benutzer zusätzliche IAM Rollenberechtigungen. Weitere Informationen finden Sie unter Amazon EMR Notebooks sind Amazon EMR Studio-Workspaces in der Konsole und EMRAmazon-Konsole.

Beachten Sie Folgendes, wenn Sie planen, ein Git-basiertes Repository in EMR Notebooks zu integrieren.

AWS CodeCommit

Wenn Sie ein CodeCommit Repository verwenden, müssen Sie Git-Anmeldeinformationen und HTTPS mit verwenden CodeCommit. SSHSchlüssel und HTTPS mit dem AWS CLI Credential Helper werden nicht unterstützt. CodeCommit unterstützt keine persönlichen Zugriffstoken (PATs). Weitere Informationen finden Sie unter Verwenden IAM mit CodeCommit: Git-Anmeldeinformationen, SSH Schlüsseln und AWS Zugriffsschlüsseln im IAMBenutzerhandbuch und Einrichtung für HTTPS Benutzer, die Git-Anmeldeinformationen verwenden, im AWS CodeCommit Benutzerhandbuch.

Überlegungen zu Zugriff und Berechtigungen

Bevor Sie Ihrem Notebook ein Repository zuordnen, stellen Sie sicher, dass Ihr Cluster, Ihre IAM Rolle für EMR Notebooks und Ihre Sicherheitsgruppen über die richtigen Einstellungen und Berechtigungen verfügen. Sie können auch Git-basierte Repositorys konfigurieren, die Sie in einem privaten Netzwerk hosten, indem Sie den Anweisungen unter Ein privat gehostetes Git-Repository für Notebooks konfigurieren EMR folgen.

- Cluster-Internetzugriff – Die Netzwerkschnittstelle, die gestartet wird, hat nur eine private IP-Adresse. Das bedeutet, dass sich der Cluster, mit dem Ihr Notebook eine Verbindung herstellt, in einem privaten Subnetz mit einem Network Address Translation (NAT) -Gateway befinden muss oder über ein virtuelles privates Gateway auf das Internet zugreifen können muss. Weitere Informationen finden Sie unter VPCAmazon-Optionen.

  Die Sicherheitsgruppen für Ihr Notebook müssen eine Regel für ausgehenden Datenverkehr enthalten, sodass das Notebook Datenverkehr vom Cluster an das Internet weiterleiten kann. Wir empfehlen, eigene Sicherheitsgruppen zu erstellen. Weitere Informationen finden Sie unter EC2Sicherheitsgruppen für EMR Notebooks angeben.

  > **Important**
  >
  > Wenn die Netzwerkschnittstelle in ein öffentliches Subnetz gestartet wird, kann sie nicht über ein Internet-Gateway (IGW) mit dem Internet kommunizieren.

- Berechtigungen für AWS Secrets Manager — Wenn Sie Secrets Manager zum Speichern von Geheimnissen verwenden, die Sie für den Zugriff auf ein Repository verwenden, the section called “EMRRolle bei Notebooks” muss an sie eine Berechtigungsrichtlinie angehängt sein, die die `secretsmanager:GetSecretValue` Aktion ermöglicht.

Ein privat gehostetes Git-Repository für Notebooks konfigurieren EMR

Verwenden Sie die folgenden Anweisungen, um privat gehostete Repositorys für Notebooks zu konfigurieren. EMR Sie müssen eine Konfigurationsdatei mit Informationen zu Ihren DNS und Git-Servern bereitstellen. Amazon EMR verwendet diese Informationen, um EMR Notebooks zu konfigurieren, die den Datenverkehr an Ihre privat gehosteten Repositorys weiterleiten können.

Voraussetzungen

Bevor Sie ein privat gehostetes Git-Repository für EMR Notebooks konfigurieren, müssen Sie über Folgendes verfügen:

- Ein Amazon S3 Control Ort, an dem Dateien für Ihr EMR Notizbuch gespeichert werden.

Um ein oder mehrere privat gehostete Git-Repositorys für Notebooks zu konfigurieren EMR

1. Erstellen Sie eine Konfigurationsdatei mit der bereitgestellten Vorlage. Geben Sie für jeden Git-Server, den Sie in Ihrer Konfiguration angeben möchten, die folgenden Werte an:

   - **`DnsServerIpV4`**- Die IPv4 Adresse Ihres Servers. DNS Wenn Sie Werte für sowohl `DnsServerIpV4` als auch `GitServerIpV4List` angeben, hat der Wert für `DnsServerIpV4` Vorrang und wird zur Auflösung Ihres `GitServerDnsName` verwendet.

     > **Note**
     >
     > Um privat gehostete Git-Repositorys verwenden zu können, muss Ihr DNS Server eingehenden Zugriff von Notebooks zulassen. EMR Wir empfehlen Ihnen dringend, Ihren DNS Server vor anderen, unbefugten Zugriffen zu schützen.

   - **`GitServerDnsName`**- Der DNS Name Ihres Git-Servers. Zum Beispiel `"git.example.com"`.
   - **`GitServerIpV4List`**- Eine Liste von IPv4 Adressen, die zu deinen Git-Servern gehören.

```
[
    {
        "Type": "PrivatelyHostedGitConfig",
        "Value": [
            {
                "DnsServerIpV4": "<10.24.34.xxx>",
                "GitServerDnsName": "<enterprise.git.com>",
                "GitServerIpV4List": [
                    "<xxx.xxx.xxx.xxx>",
                    "<xxx.xxx.xxx.xxx>"
                ]
            },
            {
                "DnsServerIpV4": "<10.24.34.xxx>",
                "GitServerDnsName": "<git.example.com>",
                "GitServerIpV4List": [
                    "<xxx.xxx.xxx.xxx>",
                    "<xxx.xxx.xxx.xxx>"
                ]
            }
        ]
```

```
    }
]
```

2. Speichern Sie Ihre Konfigurationsdatei unter `configuration.json`.
3. Laden Sie die Konfigurationsdatei in den von Ihnen angegebenen Amazon-S3-Speicherort in einem Ordner mit dem Namen `life-cycle-configuration` hoch. Wenn Ihr Standard-S3-Speicherort beispielsweise `s3://DOC-EXAMPLE-BUCKET/notebooks` lautet, sollte sich Ihre Konfigurationsdatei unter `s3://DOC-EXAMPLE-BUCKET/notebooks/`**`life-cycle-configuration/configuration.json`** befinden.

   > **Important**
   >
   > Wir empfehlen dringend, den Zugriff auf Ihren `life-cycle-configuration` Ordner auf Ihre EMR Notebooks-Administratoren und auf die Servicerolle für EMR Notebooks zu beschränken. Sie sollten auch `configuration.json` vor unbefugtem Zugriff schützen. Anweisungen finden Sie unter Steuern des Zugriffs auf einen Bucket mit Benutzerrichtlinien oder Bewährte Sicherheitsmethoden für Amazon S3.

   Anweisungen zum Hochladen finden Sie unter Erstellen eines Ordners und Hochladen von Objekten im Benutzerhandbuch für Amazon Simple Storage Service.

Ein Git-basiertes Repository zu Amazon hinzufügen EMR

> **Note**
>
> EMRNotebooks sind in der Konsole als EMR Studio-Workspaces verfügbar. Mit der Schaltfläche „Arbeitsbereich erstellen“ in der Konsole können Sie neue Notizbücher erstellen. Um auf Workspaces zuzugreifen oder diese zu erstellen, benötigen EMR Notebook-Benutzer zusätzliche IAM Rollenberechtigungen. Weitere Informationen finden Sie unter Amazon EMR Notebooks sind Amazon EMR Studio-Workspaces in der Konsole und EMRAmazon-Konsole.

In den folgenden Abschnitten finden Sie Informationen zum Hinzufügen eines Git-basierten Repositorys zu einem EMR Notebook in der alten Konsole oder zu einem EMR Studio-Arbeitsbereich in der Konsole.

Console

Da EMR Notebooks in der neuen Konsole EMR Studio-Workspaces sind, kannst du den Anweisungen unter folgen, Git-basierte Repositorys mit einem EMR Studio-Workspace verknüpfen um deinem Workspace bis zu drei Git-Repositorys zuzuordnen.

Sie können aber auch die JupyterLab Git-Erweiterung verwenden. Wählen Sie das Git-Symbol in der linken Seitenleiste Ihres Jupyterlab-Notebooks, um auf die Erweiterung zuzugreifen. Informationen zur Erweiterung finden Sie im GitHub jupyterlab-git Repo.

Um ein Git-Repository mit einem Workspace zu verknüpfen, muss Ihr Studio-Administrator Schritte unternehmen, um das Studio so zu konfigurieren, dass die Verknüpfung mit Git-Repositorys zulässig ist. Weitere Informationen finden Sie unter Zugriff und Berechtigungen für Git-basierte Repositorys einrichten.

Aktualisieren oder Löschen eines Git-basierten Repositorys

> **Note**
>
> EMRNotebooks sind in der Konsole als Studio-Workspaces verfügbar. EMR Mit der Schaltfläche „Arbeitsbereich erstellen“ in der Konsole können Sie neue Notizbücher erstellen. Um auf Workspaces zuzugreifen oder diese zu erstellen, benötigen EMR Notebook-Benutzer zusätzliche IAM Rollenberechtigungen. Weitere Informationen finden Sie unter Amazon EMR Notebooks sind Amazon EMR Studio-Workspaces in der Konsole und EMRAmazon-Konsole.

In den folgenden Abschnitten finden Sie Informationen zum Löschen eines Git-basierten Repositorys aus einem EMR Notizbuch in der alten Konsole oder aus einem EMR Studio-Arbeitsbereich in der Konsole.

Console

Da EMR Notebooks in der neuen Konsole EMR Studio-Workspaces sind, findest du weitere Informationen Git-basierte Repositorys mit einem EMR Studio-Workspace verknüpfen zur Arbeit mit Git-Repositorys in deinem Workspace. Derzeit können Sie Git-Repositorys jedoch nicht aus Workspaces löschen.

Verknüpfen oder Aufheben der Verknüpfung eines Git-basierten Repositorys

> **Note**
>
> EMRNotebooks sind in der Konsole als EMR Studio-Workspaces verfügbar. Mit der Schaltfläche „Arbeitsbereich erstellen“ in der Konsole können Sie neue Notizbücher erstellen. Um auf Workspaces zuzugreifen oder diese zu erstellen, benötigen EMR Notebook-Benutzer zusätzliche IAM Rollenberechtigungen. Weitere Informationen finden Sie unter Amazon EMR Notebooks sind Amazon EMR Studio-Workspaces in der Konsole und EMRAmazon-Konsole.

Gehen Sie wie folgt vor, um ein Git-basiertes Repository mit einem EMR Notebook in der alten Konsole oder mit einem EMR Studio-Workspace in der Konsole zu verknüpfen oder die Verknüpfung aufzuheben.

Console

Da EMR Notebooks in der neuen Konsole EMR Studio-Workspaces sind, findest du weitere Informationen Git-basierte Repositorys mit einem EMR Studio-Workspace verknüpfen zur Arbeit mit Git-Repositorys in deinem Workspace. Derzeit können Sie Git-Repositorys jedoch nicht aus Workspaces löschen.

Grundlegendes zum Repository-Status

Ein Git-Repository kann beliebige der folgenden Status in der Repository-Liste aufweisen. Weitere Informationen zum Verknüpfen von EMR Notebooks mit Git-Repositorys finden Sie unterVerknüpfen oder Aufheben der Verknüpfung eines Git-basierten Repositorys.

Status	Bedeutung
Verknüpfen	Das Git-Repository wird mit dem Notebook verknüpft. Während der Status des Repositor ys Linking (Verknüpfung wird hergestellt) lautet, können Sie das Notebook anhalten.
Verknüpft	Das Git-Repository ist mit dem Notebook verknüpft. Solange das Repository den Status

Status	Bedeutung
	Linked (Verknüpft) aufweist, ist es mit dem Remote-Repository verbunden.
Verknüpfung fehlgeschlagen	Das Git-Repository konnte nicht mit dem Notebook verknüpft werden. Sie können erneut versuchen, es zu verknüpfen.
Verknüpfung aufheben	Die Verknüpfung des Git-Repositorys mit dem Notebook wird aufgehoben. Während der Status des Repositorys Unlinking (Verknüpf ung wird aufgehoben) lautet, können Sie das Notebook nicht anhalten. Durch das Aufheben der Verknüpfung eines Git-Repositorys mit einem Notebook wird nur die Verbindung mit dem Remote-Repository getrennt; es wird kein Code aus dem Notebook gelöscht.
Verknüpfung aufheben fehlgeschlagen	Das Git-Repository konnte die Verknüpfung mit dem Notebook nicht aufheben. Sie können erneut versuchen, die Verknüpfung aufzuhebe n.

Erstellen eines neuen Notebooks mit einem zugehörigen Git-Repository

> **Note**
>
> EMRNotizbücher sind in der Konsole als EMR Studio-Workspaces verfügbar. Mit der Schaltfläche „Arbeitsbereich erstellen“ in der Konsole können Sie neue Notizbücher erstellen. Um auf Workspaces zuzugreifen oder diese zu erstellen, benötigen EMR Notebook-Benutzer zusätzliche IAM Rollenberechtigungen. Weitere Informationen finden Sie unter Amazon EMR Notebooks sind Amazon EMR Studio-Workspaces in der Konsole und EMRAmazon-Konsole.

Um ein Notizbuch zu erstellen und es mit Git-Repositorys in der alten EMR Amazon-Konsole zu verknüpfen

1. Folgen Sie den Anweisungen unter Erstellen eines Notebook.
2. Wählen Sie für Security group (Sicherheitsgruppe) die Option Use your own security group (Eigene Sicherheitsgruppe verwenden) aus.

   > **Note**
   >
   > Die Sicherheitsgruppen für Ihr Notebook müssen eine Regel für ausgehenden Datenverkehr enthalten, sodass das Notebook Datenverkehr über den Cluster an das Internet weiterleiten kann. Wir empfehlen, eigene Sicherheitsgruppen zu erstellen. Weitere Informationen finden Sie unter EC2Sicherheitsgruppen für EMR Notebooks angeben.

3. Wählen Sie für Git repositories (Git-Repositorys) unter Choose repository (Repository wählen) das Repository aus, das dem Notebook zugeordnet werden soll.

   1. Wählen Sie ein Repository aus, das als Ressource in Ihrem Konto gespeichert ist, und wählen Sie dann Save (Speichern).
   2. Um ein neues Repository als Ressource in Ihrem Konto hinzuzufügen, wählen Sie add a new repository (Neues Repository hinzufügen). Führen Sie den Workflow Add repository (Repository hinzufügen) in einem neuen Fenster durch.

Verwenden von Git-Repositorys in einem Notebook

> **Note**
>
> EMRNotebooks sind in der Konsole als EMR Studio-Workspaces verfügbar. Mit der Schaltfläche „Arbeitsbereich erstellen“ in der Konsole können Sie neue Notizbücher erstellen. Um auf Workspaces zuzugreifen oder diese zu erstellen, benötigen EMR Notebook-Benutzer zusätzliche IAM Rollenberechtigungen. Weitere Informationen finden Sie unter Amazon EMR Notebooks sind Amazon EMR Studio-Workspaces in der Konsole und EMRAmazon-Konsole.

Wenn Sie ein Notizbuch öffnen, können Sie wählen, ob Sie in Jupyter öffnen JupyterLab oder In Jupyter öffnen möchten.

Wenn Sie das Notebook in Jupyter öffnen, wird eine Liste erweiterbarer Dateien und Ordner im Notebook angezeigt. Sie können Git-Befehle wie die folgenden manuell in einer Notebook-Zelle ausführen.

```
!git pull origin primary
```

Zum Öffnen der zusätzlichen Repositorys wechseln Sie in andere Ordner.

Wenn Sie das Notizbuch mit einer JupyterLab Oberfläche öffnen möchten, können Sie die vorinstallierte JupyterLab Git-Erweiterung verwenden. Weitere Informationen über die Erweiterung finden Sie unter jupyterlab-git.

Cluster planen und konfigurieren

In diesem Abschnitt werden die Konfigurationsoptionen und Anweisungen für die Planung, Konfiguration und den Start von Clustern mithilfe von Amazon erläutertEMR. Bevor Sie einen Cluster starten, treffen Sie Entscheidungen hinsichtlich Ihres Systems auf der Grundlage der Daten, die Sie verarbeiten, und Ihrer Anforderungen an Kosten, Geschwindigkeit, Kapazität, Verfügbarkeit, Sicherheit und Verwaltbarkeit. Ihre Entscheidungen betreffen u. a. Folgendes:

- Welche Region in einem Cluster ausgeführt wird, wo und wie Daten gespeichert werden und wie Ergebnisse ausgegeben werden. Siehe Cluster-Standort und Datenspeicher konfigurieren.
- Egal, ob Sie EMR Amazon-Cluster auf Outposts oder Local Zones ausführen. Weitere Informationen unter EMRCluster auf AWS Outposts oder EMRCluster in AWS Local Zones.
- Die Frage, ob ein Cluster von langer Dauer oder vorübergehend ist, und welche Software dort ausgeführt wird. Siehe Konfigurieren eines Clusters zum Fortfahren oder Beenden nach der Schrittausführung und Konfigurieren der Cluster-Software.
- Wenn ein Cluster einen einzelnen Primärknoten oder drei Primärknoten besitzt. Siehe Primärknoten planen und konfigurieren.
- Die Hardware- und Netzwerkoptionen, mit denen Kosten, Leistung und Verfügbarkeit Ihrer Anwendung optimiert werden. Siehe Cluster-Hardware und Netzwerken konfigurieren.
- Die Einrichtung von Clustern für eine leichtere Verwaltung sowie die Überwachung von Aktivitäten, Leistung und Zustand. Siehe Konfigurieren der Cluster-Protokollierung und des Debuggings und Tag-Cluster.
- Authentifizierung und Autorisierung des Zugriffs auf Cluster-Ressourcen und Verschlüsselung von Daten. Siehe Sicherheit bei Amazon EMR.
- Die Integration in andere Software und Services. Siehe Treiber und Drittanbieter-Anwendungsintegration.

Schnell einen Cluster starten

Um schnell einen Cluster mit der Konsole zu starten

1. Melden Sie sich bei https://console.aws.amazon.com/emr/Clustern an und öffnen Sie die EMR Amazon-Konsole. AWS Management Console

2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und dann Create cluster aus.
3. Geben Sie auf der Seite Cluster erstellen Werte für die bereitgestellten Felder ein oder wählen Sie sie aus. Das persistente Übersichtsfenster zeigt eine Echtzeitansicht Ihrer aktuell ausgewählten Clusteroptionen an. Wählen Sie im Übersichtsfenster eine Überschrift aus, um zum entsprechenden Abschnitt zu navigieren und Anpassungen vorzunehmen. Ihr Clustername darf die Zeichen <, >, $, | oder `(Backtick) nicht enthalten. Sie müssen alle erforderlichen Konfigurationen abschließen, bevor Sie Cluster erstellen auswählen können.
4. Wählen Sie Cluster erstellen aus, um die Konfiguration wie im Image gezeigt zu akzeptieren.
5. Dadurch wird die Cluster-Detailseite geöffnet. Suchen Sie den Cluster-Status neben dem Clusternamen. Der Status sollte sich während des Clustererstellungsprozesses von Startet zu Läuft zu Warten ändern. Möglicherweise müssen Sie das Aktualisierungssymbol oben rechts auswählen oder Ihren Browser aktualisieren, um Updates zu erhalten.

   Wenn sich der Status in Wartend ändert, ist Ihr Cluster betriebsbereit und bereit, Schritte und SSH Verbindungen anzunehmen.

Cluster-Standort und Datenspeicher konfigurieren

In diesem Abschnitt wird beschrieben, wie Sie die Region für einen Cluster konfigurieren, welche verschiedenen Dateisysteme verfügbar sind, wenn Sie Amazon verwenden, EMR und wie Sie sie verwenden. Außerdem erfahren Sie, wie Sie Daten vorbereiten oder bei EMR Bedarf auf Amazon hochladen und wie Sie einen Ausgabespeicherort für Protokolldateien und alle von Ihnen konfigurierten Ausgabedatendateien vorbereiten.

Themen

Wählen Sie eine AWS Region

Amazon Web Services wird auf Servern in Rechenzentren auf der ganzen Welt ausgeführt. Die Rechenzentren sind nach geografischer Region organisiert. Wenn Sie einen EMR Amazon-Cluster

starten, müssen Sie eine Region angeben. So können Sie eine Region auswählen, die die Latenz oder die Kosten verringert oder behördliche Vorschriften erfüllt. Eine Liste der von Amazon EMR unterstützten Regionen und Endpunkte finden Sie unter Regionen und Endpunkte in der. Allgemeine Amazon Web Services-Referenz

Um eine optimale Leistung zu erzielen, sollten Sie den Cluster in derselben Region starten wie Ihre Daten. Wenn sich der Amazon-S3-Bucket, in dem Ihre Eingabedaten gespeichert sind, beispielsweise in der Region USA West (Oregon) befindet, sollten Sie Ihren Cluster in der Region USA West (Oregon) starten, um Gebühren für die regionsübergreifende Datenübertragung zu vermeiden. Wenn Sie einen Amazon-S3-Bucket für den Empfang der Cluster-Ausgabe verwenden, sollten Sie diesen Bucket ebenfalls in der Region USA West (Oregon) erstellen.

Wenn Sie dem Cluster ein EC2 Amazon-Schlüsselpaar zuordnen möchten (erforderlich für die Anmeldung am Master-Knoten), muss das key pair in derselben Region wie der Cluster erstellt werden. SSH In ähnlicher Weise werden die Sicherheitsgruppen, die Amazon EMR zur Verwaltung des Clusters erstellt, in derselben Region wie der Cluster erstellt.

Wenn Sie sich AWS-Konto am oder nach dem 17. Mai 2017 für eine registriert haben, AWS Management Console ist die Standardregion, in der Sie auf eine Ressource zugreifen, USA Ost (Ohio) (us-east-2). Für ältere Konten ist die Standardregion entweder US West (Oregon) (us-west-2) oder USA Ost (Nord-Virginia) (us-east-1). Weitere Informationen finden Sie unter -Regionen und Endpunkte.

Einige AWS Funktionen sind nur in begrenzten Regionen verfügbar. Beispielsweise sind Cluster-Compute-Instances nur in der Region USA Ost (Nord-Virginia) verfügbar und die Region Asien-Pazifik (Sydney) unterstützt nur Hadoop 1.0.3 und höher. Prüfen Sie bei der Auswahl einer Region, dass die Feature, die Sie verwenden möchten, in dieser Region unterstützt werden.

Um eine optimale Leistung zu erzielen, verwenden Sie dieselbe Region für alle AWS Ressourcen, die mit dem Cluster verwendet werden. In der folgenden Tabelle werden die Regionsnamen den Services zugeordnet. Eine Liste der EMR Amazon-Regionen finden Sie unter AWS-Regionen und Endpunkte in der Allgemeine Amazon Web Services-Referenz.

Auswählen einer Region mithilfe der Konsole

Ihre Standardregion wird links neben Ihren Kontoinformationen in der Navigationsleiste angezeigt. Um die Region sowohl auf der neuen als auch auf der alten Konsole zu wechseln, wähle das Drop-down-Menü Region und wähle eine neue Option aus.

Geben Sie eine Region an mit AWS CLI

Geben Sie eine Standardregion in an, AWS CLI indem Sie entweder den aws configure Befehl oder die `AWS_DEFAULT_REGION` Umgebungsvariable verwenden. Weitere Informationen finden Sie unter Konfiguration der AWS Region im AWS Command Line Interface Benutzerhandbuch.

Wählen Sie eine Region mit einem SDK oder API

Um eine Region mithilfe von auszuwählenSDK, konfigurieren Sie Ihre Anwendung so, dass sie den Endpunkt dieser Region verwendet. Wenn Sie eine Client-Anwendung mithilfe von erstellen AWS SDK, können Sie den Client-Endpunkt ändern, indem Sie anrufen`setEndpoint`, wie im folgenden Beispiel gezeigt:

```
client.setEndpoint("elasticmapreduce.us-west-2.amazonaws.com");
```

Nachdem Ihre Anwendung durch das Festlegen des Endpunkts eine Region angegeben hat, können Sie die Availability Zone für die EC2 Instances Ihres Clusters festlegen. Availability Zones sind eindeutige geografische Standorte. Sie sollen vor Fehlern in anderen Availability Zones schützen und eine kostengünstige Netzwerkkonnektivität mit geringer Latenz zu anderen Availability Zones in der gleichen Region bereitstellen. Eine Region umfasst eine oder mehr Availability Zones. Um die Leistung zu optimieren und die Latenz zu verkürzen, sollten sich alle Ressourcen in derselben Availability Zone befinden wie der Cluster, der sie verwendet.

Mit Storage- und Dateisystemen arbeiten

Amazon EMR und Hadoop bieten eine Vielzahl von Dateisystemen, die Sie bei der Verarbeitung von Cluster-Schritten verwenden können. Sie geben an, welches Dateisystem verwendet werden soll, indem Sie das Präfix des für den Zugriff auf die Daten URI verwendeten Dateisystems angeben. Verweist beispielsweise `s3://amzn-s3-demo-bucket1/path` auf einen Amazon S3 S3-Bucket mitEMRFS. In der folgenden Tabelle werden die verfügbaren Dateisysteme sowie Empfehlungen zu ihrer Verwendung aufgeführt.

Amazon EMR und Hadoop verwenden bei der Verarbeitung eines Clusters in der Regel zwei oder mehr der folgenden Dateisysteme. HDFSund EMRFS sind die beiden wichtigsten Dateisysteme, die bei Amazon verwendet werdenEMR.

> **Important**
>
> Ab EMR Amazon-Version 5.22.0 EMR verwendet Amazon AWS Signature Version 4 ausschließlich zur Authentifizierung von Anfragen an Amazon S3. Frühere EMR Amazon-Versionen verwenden in einigen Fällen AWS Signature Version 2, sofern in den Versionshinweisen nicht angegeben ist, dass ausschließlich Signature Version 4 verwendet wird. Weitere Informationen finden Sie unter Authentifizieren von Anfragen (AWS Signature Version 4) und Authentifizieren von Anfragen (AWS Signature Version 2) im Amazon Simple Storage Service Developer Guide.

Dateisystem	Präfix	Beschreibung
HDFS	`hdfs://` (oder ohne Präfix)	HDFSist ein verteiltes, skalierbares und portables Dateisystem für Hadoop. Ein Vorteil von HDFS ist die Datensensibilität zwischen den Hadoop-Clusterknot en, die die Cluster verwalten, und den Hadoop-Cl usterknoten, die die einzelnen Schritte verwalten. Weitere Informationen finden Sie in der Hadoop-Do kumentation. HDFSwird von den Master- und Core-Knoten verwendet. Ein Vorteil ist, dass es schneller ist. Ein Nachteil ist, dass es ein flüchtiger Speicher ist. Dieser wird beim Beenden des Clusters verworfen. Es eignet sich am besten für die Zwischenspeicherung der Ergebnisse von zwischengeschalteten Auftragsv erlaufsschritten.
EMRFS	`s3://`	EMRFSist eine Implementierung des Hadoop-Da teisystems, das zum Lesen und Schreiben regulärer Dateien von Amazon EMR direkt in Amazon S3 verwendet wird. EMRFSbietet den Komfort, persisten te Daten in Amazon S3 zur Verwendung mit Hadoop zu speichern und bietet gleichzeitig Funktionen wie serverseitige Amazon S3 S3-Verschlüsselung, read-after-write Konsistenz und Listenkonsistenz.

Dateisystem	Präfix	Beschreibung
		Note Zuvor EMR verwendete Amazon die `s3a` Dateisysteme `s3n` und. Obwohl beide noch funktionieren, empfehlen wir Ihnen, das `s3` URI Schema zu verwenden, um die beste Leistung, Sicherheit und Zuverlässigkeit zu erzielen.
Lokales Dateisystem		Das lokale Dateisystem bezieht sich auf einen lokal verbundenen Datenträger. Wenn ein Hadoop-Cl uster erstellt wird, wird jeder Knoten aus einer EC2 Instanz erstellt, die über einen vorkonfigurierten Block von vorkonfiguriertem Festplattenspeicher verfügt, der als Instanzspeicher bezeichnet wird. Daten auf Instance-Speicher-Volumes bleiben nur während der Lebensdauer der jeweiligen Instanz erhalten. EC2 Instance-Speicher-Volumen eignen sich perfekt für die Speicherung von temporärer Daten, die sich ständig ändern (z. B. Puffer, Caches, Arbeitsdaten und andere temporäre Inhalte). Weitere Informationen finden Sie unter EC2Amazon-Instance-Speicher. Das lokale Dateisystem wird von verwendetHDFS, aber Python läuft auch vom lokalen Dateisystem aus, und Sie können wählen, ob Sie zusätzliche Anwendungsdateien auf Instance-Speicher-Volumes speichern möchten.

Dateisystem	Präfix	Beschreibung
Amazon-S3-Block-Dateisystem (veraltet)	`s3bfs://`	Das Amazon-S3-Block-Dateisystem ist ein veraltetes Dateispeichersystem. Es wird ausdrücklich von der Verwendung dieses Systems abgeraten. **Important** Wir empfehlen, dass Sie dieses Dateisystem nicht verwenden. Es kann dazu führen, dass der Cluster ausfällt. Möglicherweise ist es jedoch für ältere Anwendungen erforderlich.

Zugriff auf Dateisysteme

Sie geben das zu verwendende Dateisystem durch das Präfix des Uniform Resource Identifier (URI) an, der für den Zugriff auf die Daten verwendet wird. Die folgenden Verfahren veranschaulichen den Zugriff auf verschiedene Dateisystemarten.

Um auf ein lokales zuzugreifen HDFS

- Geben Sie das `hdfs:///` Präfix in der anURI. Amazon EMR löst Pfade auf, die kein Präfix im URI HDFS Lokalen angeben. Zum Beispiel URIs würden die beiden folgenden Antworten auf denselben Speicherort in HDFS aufgelöst.

```
hdfs:///path-to-data

/path-to-data
```

Um auf eine Fernbedienung zuzugreifen HDFS

- Nehmen Sie die IP-Adresse des Master-Knotens in die aufURI, wie in den folgenden Beispielen gezeigt.

```
hdfs://master-ip-address/path-to-data

master-ip-address/path-to-data
```

So greifen Sie auf Amazon S3 zu

- Verwenden Sie den `s3://`-Präfix.

```
s3://bucket-name/path-to-file-in-bucket
```

So greifen Sie auf das Amazon-S3-Block-Dateisystem zu

- Verwenden Sie dieses Dateisystem nur für ältere Anwendungen, die das Amazon-S3-Block-Dateisystem benötigen. Um mit diesem Dateisystem auf Daten zuzugreifen oder diese zu speichern, verwenden Sie das `s3bfs://` Präfix inURI.

  Das Amazon-S3-Block-Dateisystem ist ein veraltetes Dateisystem, das für Uploads zu Amazon S3 mit einer Größe von mehr als 5 GB verwendet wurde. Mit der mehrteiligen Upload-Funktion, die Amazon über AWS Java EMR bereitstelltSDK, können Sie Dateien mit einer Größe von bis zu 5 TB in das native Amazon S3 S3-Dateisystem hochladen, und das Amazon S3 S3-Blockdateisystem ist veraltet.

  > **Warning**
  >
  > Da dieses veraltete Dateisystem zu Race Conditions führen kann, die das Dateisystem beschädigen können, sollten Sie dieses Format vermeiden und stattdessen verwenden. EMRFS

```
s3bfs://bucket-name/path-to-file-in-bucket
```

Eingabedaten vorbereiten

Die meisten Clustern laden Eingabedaten und verarbeitet diese anschließend. Zum Laden von Daten müssen diese sich an einem Speicherort befinden, auf den der Cluster zugreifen kann und der ein Format hat, das der Cluster verarbeiten kann. Das gängigste Szenario ist das zum Hochladen von Eingabedaten in Amazon S3. Amazon EMR bietet Tools für Ihren Cluster zum Importieren oder Lesen von Daten aus Amazon S3.

Das Standardeingabeformat in Hadoop sind Textdateien. Sie können Hadoop jedoch anpassen und Tools zum Importieren von Daten in anderen Formaten verwenden.

Themen

Arten von Eingaben, die Amazon akzeptieren EMR kann

Das Standardeingabeformat für einen Cluster sind Textdateien, bei denen jede Zeile durch ein Zeilenvorschubzeichen (\ n) getrennt ist. Dies ist das am häufigsten verwendete Eingabeformat.

Wenn Ihre Eingabedaten in einem anderen Format geschrieben werden müssen als Standardtextdateien, können Sie die Hadoop-Benutzeroberfläche `InputFormat` verwenden, um andere Eingabetypen anzugeben. Sie können auch eine Unterklasse der `FileInputFormat`-Klasse für den Umgang mit benutzerdefinierten Datentypen verwenden. Weitere Informationen finden Sie unter http://hadoop.apache.org/docs/current/api/org/apache/hadoop/mapred/ InputFormat .html.

Wenn Sie Hive verwenden, können Sie einen Serializer/Deserializer (SerDe) verwenden, um Daten aus einem bestimmten Format einzulesen. HDFS Weitere Informationen finden Sie unter https://cwiki.apache.org/confluence/display/Hive/. SerDe

So erhalten Sie Daten in Amazon EMR

Amazon EMR bietet mehrere Möglichkeiten, Daten auf einen Cluster zu übertragen. Die gängigste Methode besteht darin, die Daten auf Amazon S3 hochzuladen und die integrierten Funktionen von Amazon EMR zu verwenden, um die Daten in Ihren Cluster zu laden. Sie können auch das Hadoop-Feature DistributedCache für den verteilten Cache verwenden, um Dateien von einem verteilten Dateisystem in das lokale Dateisystem zu übertragen. Die von Amazon bereitgestellte Implementierung von Hive EMR (Hive-Version 0.7.1.1 und höher) umfasst Funktionen, mit denen Sie

Daten zwischen DynamoDB und einem Amazon-Cluster importieren und exportieren können. EMR Wenn Sie große Datenmengen On-Premises verarbeiten müssen, kann der AWS Direct Connect - Service nützlich sein.

Themen

Daten aus Amazon S3 uploaden

Informationen zum Hochladen von Objekten in Amazon S3 finden Sie unter Ein Objekts zu Ihrem Bucket hinzufügen im Benutzerhandbuch zu Amazon Simple Storage Service. Weitere Informationen zur Verwendung von Amazon S3 mit Hadoop finden Sie unter http://wiki.apache.org/hadoop/AmazonS3.

Themen

Erstellen und Konfigurieren eines Amazon S3-Buckets

Amazon EMR verwendet das AWS SDK for Java zusammen mit Amazon S3, um Eingabedaten, Protokolldateien und Ausgabedaten zu speichern. Amazon S3 bezeichnet diese Speicherorte als Buckets. Buckets unterliegen bestimmten Einschränkungen und Beschränkungen, um Amazon S3 und DNS den Anforderungen zu entsprechen. Weitere Informationen finden Sie unter Bucket-Einschränkungen und -Limits im Benutzerhandbuch zu Amazon Simple Storage Service.

In diesem Abschnitt erfahren Sie, wie Sie Amazon S3 verwenden AWS Management Console , um Berechtigungen für einen Amazon S3 S3-Bucket zu erstellen und anschließend festzulegen.

Sie können auch mit Amazon S3 API oder Berechtigungen für einen Amazon S3-Bucket erstellen und festlegen AWS CLI. Sie können Curl auch zusammen mit einer Änderung verwenden, um die entsprechenden Authentifizierungsparameter für Amazon S3 zu übergeben.

Weitere Informationen finden Sie in den folgenden Ressourcen:

- Informationen zur Bucket-Erstellung mittels Konsole finden Sie unter Erstellen eines Buckets im Amazon-S3-Benutzerhandbuch.
- Informationen zum Erstellen und Arbeiten mit Buckets mithilfe von finden Sie unter Verwenden von S3-Befehlen auf hoher Ebene mit dem AWS Command Line Interface im Amazon S3 S3-Benutzerhandbuch. AWS CLI
- Informationen zum Erstellen eines Buckets mithilfe von finden Sie unter Beispiele für die Erstellung eines Buckets im Amazon Simple Storage Service-Benutzerhandbuch. SDK
- Informationen zum Arbeiten mit Buckets über Curl finden Sie unter Amazon-S3-Authentifizierungstool für Curl.
- Weitere Informationen zum Angeben regionsspezifischer Buckets finden Sie unter Zugreifen auf einen Bucket im Benutzerhandbuch für Amazon Simple Storage Service.
- Informationen zum Arbeiten mit Buckets unter Verwendung von Amazon S3 Access Points finden Sie unter Verwenden eines Alias im Bucket-Stil für Ihren Zugangspunkt im Amazon-S3-Benutzerhandbuch. Sie können Amazon S3 Access Points problemlos mit dem Alias von Amazon S3 Access Points anstelle des Amazon-S3-Bucket-Namens verwenden. Sie können den Alias Amazon S3 Access Point sowohl für bestehende als auch für neue Anwendungen verwenden, darunter Spark, Hive, Presto und andere.

> **Note**
>
> Wenn Sie die Protokollierung für einen Bucket aktivieren, werden nur Bucket-Zugriffsprotokolle aktiviert, keine EMR Amazon-Cluster-Protokolle.

Während der Bucket-Erstellung oder danach können Sie die entsprechenden Berechtigungen für den Zugriff auf den Bucket festlegen, abhängig von Ihrer Anwendung. Hierbei sollten Sie sich selbst (als Eigentümer) Lese- und Schreibzugriff und anderen autorisierten Benutzern Lesezugriff erteilen.

Erforderliche Amazon-S3-Buckets müssen vorhanden sein, bevor Sie einen Cluster erstellen können. Sie müssen alle erforderlichen Skripts und Daten auf Amazon S3 hochladen, auf die im Cluster

verwiesen wird. In der folgenden Tabelle werden Beispiele für Speicherorte für Daten, Skripts und Protokolldateien beschrieben.

Konfigurieren von mehrteiligen Uploads für Amazon S3

Amazon EMR unterstützt den mehrteiligen Amazon S3-Upload über das AWS SDK für Java. Mit dem mehrteiligen Upload können Sie ein einzelnes Objekt in mehreren Teilen hochladen. Sie können diese Objektteile unabhängig und in beliebiger Reihenfolge hochladen. Wenn die Übertragung eines Teils fehlschlägt, können Sie das Teil erneut übertragen, ohne dass dies Auswirkungen auf andere Teile hat. Nachdem alle Teile Ihres Objekts hochgeladen sind, fügt Amazon S3 diese Teile zusammen und erstellt das Objekt.

Weitere Informationen finden Sie unter Mehrteiliger Upload – Übersicht im Benutzerhandbuch zu Amazon Simple Storage Service.

Darüber hinaus EMR bietet Amazon Funktionen, mit denen Sie die Bereinigung fehlgeschlagener mehrteiliger Upload-Teile genauer steuern können.

In der folgenden Tabelle werden die EMR Amazon-Konfigurationseigenschaften für den mehrteiligen Upload beschrieben. Sie können diese mit der Konfigurationsklassifizierung `core-site` konfigurieren. Weitere Informationen finden Sie unter Anwendungen konfigurieren im EMRAmazon-Versionshandbuch.

Name des Konfigurationsparameters	Standardwert	Beschreibung
`fs.s3n.multipart.uploads.enabled`	`true`	Dieser Boolesche Typ gibt an, ob mehrteilige Uploads aktiviert werden sollen. Wenn die EMRFS konsistente Ansicht aktiviert ist, sind mehrteilige Uploads standardmäßig aktiviert und die Einstellung dieses Werts auf `false` wird ignoriert.
`fs.s3n.multipart.uploads.split.size`	`134217728`	Gibt die maximale Größe eines Teils in Byte an, bevor ein neuer Bauteil-Upload EMRFS gestartet wird, wenn mehrteilige Uploads aktiviert sind. Der Mindestwert ist `5242880` (5 MB). Wenn ein kleinerer Wert angegeben wird, wird `5242880` verwendet

Name des Konfigurationspara meters	Standardwert	Beschreibung
		. Der Höchstwert ist `5368709120` (5 GB). Wenn ein größerer Wert angegeben wird, wird `5368709120` verwendet. Wenn die EMRFS clientseitige Verschlüs selung deaktiviert ist und der Amazon S3 Optimized Committer ebenfalls deaktiviert ist, steuert dieser Wert auch die maximale Größe, die eine Datendatei vergrößern kann, bis mehrteilige Uploads anstelle einer `PutObject` Anfrage zum Hochladen der Datei EMRFS verwendet werden. Weitere Informationen finden Sie unter
`fs.s3n.ssl.enabled`	`true`	Dieser Boolesche Typ gibt an, ob HTTP oder HTTPS verwendet werden soll.
`fs.s3.buckets.crea te.enabled`	`false`	Ein boolescher Typ, der angibt, ob ein Bucket erstellt werden soll, wenn er nicht vorhanden ist. Wenn Sie dies auf `false` festlegen, wird eine Ausnahme für `CreateBucket` -Operationen ausgelöst.
`fs.s3.multipart.cl ean.enabled`	`false`	Ein boolescher Typ, der angibt, ob unvollstä ndige mehrteilige Uploads regelmäßig im Hintergrund bereinigt werden sollen.
`fs.s3.multipart.cl ean.age.threshold`	`604800`	Ein long-Typ, der das Mindestalter eines mehrteiligen Uploads in Sekunden angibt, bevor er zur Bereinigung vorgesehen wird. Die Standardeinstellung ist eine Woche.

Name des Konfigurationsparameters	Standardwert	Beschreibung
`fs.s3.multipart.clean.jitter.max`	`10000`	Eine integer-Typ, der den maximalen Betrag für zufällige Jitter-Verzögerungen in Sekunden angibt, die der festen Verzögerung von 15 Minuten hinzugefügt werden, bevor die nächste Bereinigung geplant wird.

So deaktivieren Sie mehrteilige Uploads

Console

Um mehrteilige Uploads mit der Konsole zu deaktivieren

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und anschließend Create cluster aus.
3. Geben Sie in Softwareeinstellungen bearbeiten die folgende Konfiguration ein: `classification=core-site,properties=[fs.s3n.multipart.uploads.enabled=false]`.
4. Wählen Sie alle anderen Optionen aus, die für Ihren Cluster gelten.
5. Um Ihren Cluster jetzt zu starten, wählen Sie Cluster erstellen aus.

CLI

Um den mehrteiligen Upload zu deaktivieren, verwenden Sie den AWS CLI

In diesem Verfahren wird erläutert, wie Sie mehrteilige Uploads mithilfe der AWS CLI deaktivieren. Um mehrteilige Uploads zu deaktivieren, geben Sie den Befehl `create-cluster` mit dem Parameter `--bootstrap-actions` ein.

1. Erstellen Sie eine Datei mit dem Namen `myConfig.json` und dem folgenden Inhalt und speichern Sie sie in dem Verzeichnis, in dem Sie den Befehl ausführen:

```
[
```

```
  {
    "Classification": "core-site",
    "Properties": {
      "fs.s3n.multipart.uploads.enabled": "false"
    }
  }
]
```

2. Geben Sie den folgenden Befehl ein und ersetzen Sie *myKey* mit dem Namen Ihres EC2 key pair.

> **Note**
>
> Linux-Zeilenfortsetzungszeichen (\) sind aus Gründen der Lesbarkeit enthalten. Sie können entfernt oder in Linux-Befehlen verwendet werden. Entfernen Sie sie unter Windows oder ersetzen Sie sie durch ein Caret-Zeichen (^).

```
aws emr create-cluster --name "Test cluster" \
--release-label emr-7.2.0 --applications Name=Hive Name=Pig \
--use-default-roles --ec2-attributes KeyName=myKey --instance-type m5.xlarge \
--instance-count 3 --configurations file://myConfig.json
```

API

Um den mehrteiligen Upload zu deaktivieren, verwenden Sie API

- Informationen zur programmgesteuerten Verwendung von mehrteiligen Amazon S3 S3-Uploads finden Sie unter Using the AWS SDK for Java for Multipart Upload im Amazon Simple Storage Service-Benutzerhandbuch.

  Weitere Informationen über das für Java finden Sie unter AWS SDK für Java.AWS SDK

Bewährte Methoden

Im Folgenden finden Sie Empfehlungen für die Verwendung von Amazon S3 S3-Buckets mit EMR Clustern.

Aktivieren von Versioning

Versioning ist eine empfohlene Konfiguration für Ihre Amazon S3-Buckets. Durch das Aktivieren von Versioning stellen Sie sicher, dass Sie auch versehentlich gelöschte oder überschriebene Daten wiederhergestellt werden können. Weitere Informationen finden Sie unter Verwenden von Versionsverwaltung im Benutzerhandbuch für Amazon Simple Storage Service.

Bereinigung mehrteiliger Uploads fehlgeschlagen

EMRCluster-Komponenten verwenden standardmäßig mehrteilige Uploads über Java mit Amazon S3APIs, um Protokolldateien zu schreiben und Daten in Amazon S3 auszugeben. AWS SDK Informationen zum Ändern der Eigenschaften dieser Konfiguration mithilfe von Amazon EMR finden Sie unterKonfigurieren von mehrteiligen Uploads für Amazon S3. Es kann vorkommen, dass das Hochladen einer großen Datei zu einem unvollständigen mehrteiligen Upload in Amazon S3 führt. Wenn ein mehrteiliger Upload nicht erfolgreich abgeschlossen werden kann, belegt der laufende Vorgang Ihren Bucket und es fallen Speichergebühren an. Wir empfehlen die folgenden Optionen, um eine übermäßige Dateispeicherung zu vermeiden:

- Verwenden Sie für Buckets, die Sie mit Amazon verwendenEMR, eine Lebenszyklus-Konfigurationsregel in Amazon S3, um unvollständige mehrteilige Uploads drei Tage nach dem Startdatum des Uploads zu entfernen. Mit Lebenszyklus-Konfigurationsregeln können Sie Speicherklasse und Lebensdauer von Objekten steuern. Weitere Informationen finden Sie unter Verwaltung des Objektlebenszyklus und Abbrechen unvollständiger mehrteiliger Uploads mit einer Bucket-Lebenszyklusrichtlinie.
- Aktivieren Sie EMR die mehrteilige Bereinigungsfunktion von Amazon, indem Sie andere `true` Bereinigungsparameter einstellen `fs.s3.multipart.clean.enabled` und anpassen. Diese Funktion ist bei einem hohen Volumen, einem großem Umfang und Clustern mit begrenzte Betriebszeit nützlich. In diesem Fall ist der `DaysAfterIntitiation`-Parameter einer Lebenszyklus-Konfigurationsregel möglicherweise zu lang, selbst wenn er auf das Minimum eingestellt ist, was zu Spitzen im Amazon-S3-Speicher führt. EMRDie mehrteilige Bereinigung von Amazon ermöglicht eine genauere Steuerung. Weitere Informationen finden Sie unter Konfigurieren von mehrteiligen Uploads für Amazon S3.

Versionsmarkierungen verwalten

Wir empfehlen Ihnen, eine Lebenszyklus-Konfigurationsregel in Amazon S3 zu aktivieren, um Markierungen zum Löschen abgelaufener Objekte für versionierte Buckets zu entfernen, die Sie mit Amazon verwenden. EMR Beim Löschen eines Objekts in einem versionierten Bucket wird eine

Löschmarkierung erstellt. Wenn anschließend alle vorherigen Versionen des Objekts ablaufen, verbleibt eine Löschmarkierung für abgelaufene Objekte im Bucket. Für Löschmarkierungen fallen zwar keine Gebühren an, aber das Entfernen abgelaufener Markierungen kann die Leistung von LIST Anfragen verbessern. Weitere Informationen finden Sie unter Lebenszykluskonfiguration für einen Bucket mit Versionsverwaltung im Benutzerhandbuch zu Amazon Simple Storage Service.

Bewährte Methoden zur Leistungssteigerung

Abhängig von Ihren Workloads können bestimmte Arten der Nutzung von EMR Clustern und Anwendungen auf diesen Clustern zu einer hohen Anzahl von Anfragen an einen Bucket führen. Weitere Informationen finden Sie unter Erwägungen zur Anforderungsrate und Leistung im Benutzerhandbuch zu Amazon Simple Storage Service.

Daten in Amazon S3 Express One Zone hochladen

Übersicht

Mit Amazon EMR 6.15.0 und höher können Sie Amazon EMR mit Apache Spark in Verbindung mit der Amazon S3 Express One Zone-Speicherklasse verwenden, um die Leistung Ihrer Spark-Jobs zu verbessern. EMRAmazon-Versionen 7.2.0 und höher unterstützen HBase auch Flink und Hive, sodass Sie auch von S3 Express One Zone profitieren können, wenn Sie diese Anwendungen verwenden. S3 Express One Zone ist eine S3-Speicherklasse für Anwendungen, die häufig mit Hunderttausenden Anfragen pro Sekunde auf Daten zugreifen. Zum Zeitpunkt seiner Veröffentlichung bietet S3 Express One Zone den Cloud-Objektspeicher mit der niedrigsten Latenz und der höchsten Leistung in Amazon S3.

Voraussetzungen

- Berechtigungen für S3 Express One Zone – Wenn S3 Express One Zone eine Aktion wie `GET`, `LIST` oder `PUT` für ein Amazon-S3-Objekt aufruft, ruft die Speicherklasse `CreateSession` in Ihrem Namen auf. Ihre IAM Richtlinie muss die `s3express:CreateSession` Genehmigung zulassen, damit der S3A Connector den aufrufen kann. `CreateSession` API Ein Beispielrichtlinie mit dieser Berechtigung finden Sie unter Erste Schritte mit Amazon S3 Express One Zone.
- S3A-Konnektor – Um Ihren Spark-Cluster für den Zugriff auf Daten aus einem Amazon-S3-Bucket zu konfigurieren, der die Speicherklasse S3 Express One Zone verwendet, müssen Sie den Apache-Hadoop-Konnektor S3A verwenden. Um den Connector zu verwenden, stellen Sie sicher, dass alle S3 das `s3a` Schema URIs verwenden. Wenn dies nicht der Fall ist, können Sie die Dateisystemimplementierung, die Sie für `s3`- und `s3n`-Schemata verwenden, ändern.

Um das s3-Schema zu ändern, geben Sie die folgenden Clusterkonfigurationen an:

```
[
  {
    "Classification": "core-site",
    "Properties": {
      "fs.s3.impl": "org.apache.hadoop.fs.s3a.S3AFileSystem",
      "fs.AbstractFileSystem.s3.impl": "org.apache.hadoop.fs.s3a.S3A"
    }
  }
]
```

Um das s3n-Schema zu ändern, geben Sie die folgenden Clusterkonfigurationen an:

```
[
  {
    "Classification": "core-site",
    "Properties": {
      "fs.s3n.impl": "org.apache.hadoop.fs.s3a.S3AFileSystem",
      "fs.AbstractFileSystem.s3n.impl": "org.apache.hadoop.fs.s3a.S3A"
    }
  }
]
```

Erste Schritte mit Amazon S3 Express One Zone

Themen

- Eine Berechtigungsrichtlinie erstellen
- Ihren Cluster erstellen und konfigurieren
- Konfigurationsübersicht

Eine Berechtigungsrichtlinie erstellen

Bevor Sie einen Cluster erstellen können, der Amazon S3 Express One Zone verwendet, müssen Sie eine IAM Richtlinie erstellen, die an das EC2 Amazon-Instance-Profil für den Cluster angehängt wird. Die Richtlinie muss über Berechtigungen für den Zugriff auf die Speicherklasse S3 Express One Zone verfügen. Die folgende Beispielrichtlinie zeigt, wie die erforderliche Berechtigung gewährt wird. Nachdem Sie die Richtlinie erstellt haben, fügen Sie die Richtlinie der Instance-Profilrolle hinzu, die Sie zur Erstellung Ihres EMR Clusters verwenden, wie im Ihren Cluster erstellen und konfigurieren Abschnitt beschrieben.

```
{
  "Version":"2012-10-17",
  "Statement": [
        {
            "Effect": "Allow",
            "Resource": "arn:aws:s3express:region-code:account-id:bucket/DOC-EXAMPLE-
BUCKET",
            "Action": [
                "s3express:CreateSession"
            ]
        }
    ]

 }
```

Ihren Cluster erstellen und konfigurieren

Erstellen Sie als Nächstes einen Cluster, auf dem SparkHBase, Flink oder Hive mit S3 Express One Zone ausgeführt wird. Die folgenden Schritte beschreiben einen allgemeinen Überblick über die Erstellung eines Clusters in der AWS Management Console:

1. Navigieren Sie zur EMR Amazon-Konsole und wählen Sie in der Seitenleiste Cluster aus. Wählen Sie dann Create cluster (Cluster erstellen) aus.
2. Wenn Sie Spark verwenden, wählen Sie EMR Amazon-Version `emr-6.15.0` oder höher. Wenn Sie Flink oder Hive verwendenHBase, wählen Sie `emr-7.2.0` eine höhere Version.
3. Wählen Sie die Anwendungen aus, die Sie in Ihren Cluster aufnehmen möchten, z. B. Spark oder HBase Flink.
4. Um Amazon S3 Express One Zone zu aktivieren, geben Sie im Abschnitt Softwareeinstellungen eine Konfiguration ein, die dem folgenden Beispiel ähnelt. Die Konfigurationen und empfohlenen Werte werden in dem Abschnitt Konfigurationsübersicht beschrieben, der diesem Verfahren folgt.

   ```
   [
     {
       "Classification": "core-site",
       "Properties": {
         "fs.s3a.aws.credentials.provider":
    "software.amazon.awssdk.auth.credentials.InstanceProfileCredentialsProvider",
         "fs.s3a.change.detection.mode": "none",
         "fs.s3a.endpoint.region": "aa-example-1",
         "fs.s3a.select.enabled": "false"
   ```

```
    }
  },
  {
    "Classification": "spark-defaults",
    "Properties": {
      "spark.sql.sources.fastS3PartitionDiscovery.enabled": "false"
    }
  }
]
```

5. Wählen Sie im EMR Abschnitt EC2Instanzprofil für Amazon aus, ob Sie eine vorhandene Rolle verwenden möchten, und verwenden Sie eine Rolle mit der angehängten Richtlinie, die Sie im obigen Eine Berechtigungsrichtlinie erstellen Abschnitt erstellt haben.
6. Konfigurieren Sie die restlichen Cluster-Einstellungen entsprechend Ihrer Anwendung und wählen Sie dann Create cluster (Cluster erstellen) aus.

Konfigurationsübersicht

In den folgenden Tabellen werden die Konfigurationen und vorgeschlagenen Werte beschrieben, die Sie angeben sollten, wenn Sie einen Cluster einrichten, der S3 Express One Zone mit Amazon verwendetEMR, wie im Ihren Cluster erstellen und konfigurieren Abschnitt beschrieben.

S3A-Konfigurationen

Parameter	Standardwert	Empfohlener Wert	Erklärung
`fs.s3a.aws.credentials.provider`	Wenn nicht angegeben, wird `AWSCredentialProviderList` in der folgenden Reihenfolge verwendet: `TemporaryAWSCredentialsProvider`, `SimpleAWSCredentia`	`software.amazon.awssdk.auth.credentials.InstanceProfileCredentialsProvider`	Die EMR Amazon-Instance-Profilrolle sollte die Richtlinie haben, die es dem S3A Dateisystem ermöglicht, aufzurufe `ns3express:CreateSession` . Andere Anmeldeinformationsanbieter sind ebenfalls möglich, wenn sie

Parameter	Standardwert	Empfohlener Wert	Erklärung
	`lsProvide r` , `Environme ntVariabl eCredenti alsProvid er` , `IAMInstan ceCredent ialsProvider` .		über die Berechtig ungen für S3 Express One Zone verfügen.
`fs.s3a.en dpoint.region`	Null	Der AWS-Region Ort, an dem Sie den Bucket erstellt haben.	Die Logik zur Regionsauflösung funktioniert nicht mit der Speicherklasse S3 Express One Zone.
`fs.s3a.se lect.enabled`	`true`	`false`	Amazon S3 `select` wird mit der Speicherk lasse S3 Express One Zone nicht unterstützt.
`fs.s3a.ch ange.dete ction.mode`	`server`	Keine	Die Änderungs erkennung von S3A erfolgt, indem MD5-basierte `etags` geprüft werden. Die Speicherklasse S3 Express One Zone unterstützt MD5 `checksums` nicht.

Spark-Konfigurationen

Parameter	Standardwert	Empfohlener Wert	Erklärung
`spark.sql.sources.fastS3PartitionDiscovery.enabled`	`true`	`false`	Die interne Optimierung verwendet einen API S3-Parameter, den die S3 Express One Zone-Speicherklasse nicht unterstützt.

Überlegungen

Beachten Sie Folgendes, wenn Sie Apache Spark auf Amazon EMR in die Speicherklasse S3 Express One Zone integrieren:

- Amazon S3 Express One Zone wird mit EMR Amazon-Versionen 6.15.0 und höher unterstützt.
- Der S3A-Anschluss ist erforderlich, um S3 Express One Zone mit Amazon EMR zu verwenden. Nur S3A verfügt über die Features und Speicherklassen, die für die Interaktion mit S3 Express One Zone erforderlich sind. Schritte zum Einrichten des Konnektors finden Sie unter the section called “Voraussetzungen”.
- Die Speicherklasse Amazon S3 Express One Zone wird nur mit Spark auf einem EMR Amazon-Cluster unterstützt, der auf Amazon läuftEC2.
- Die Speicherklasse Amazon S3 Express One Zone unterstützt nur die SSE-S3-Verschlüsselung. Weitere Informationen finden Sie unter Serverseitige Verschlüsselung mit verwalteten Amazon S3 S3-Schlüsseln (SSE-S3).
- Die Speicherklasse Amazon S3 Express One Zone unterstützte keine Schreibvorgänge mit dem S3A `FileOutputCommitter`. Schreibvorgänge mit dem S3A `FileOutputCommitter` in Buckets von S3 Express One Zone führen zu einem Fehler: InvalidStorageClass: The storage class you specified is not valid.
- Die Speicherklasse Amazon S3 Express One Zone wird mit Amazon EMR Serverless oder Amazon EMR on EKS nicht unterstützt.

Daten mit AWS DataSync hochladen

AWS DataSync ist ein Online-Datenübertragungsservice, der den Prozess der Übertragung von Daten zwischen Ihren lokalen Speicher- und Speicherdiensten oder zwischen AWS AWS Speicherdiensten vereinfacht, automatisiert und beschleunigt. DataSync unterstützt eine Vielzahl von lokalen Speichersystemen wie Hadoop Distributed File System (HDFS), NAS Dateiserver und selbstverwalteten Objektspeicher.

Die gängigste Methode, Daten in einen Cluster zu laden, besteht darin, die Daten auf Amazon S3 hochzuladen und die integrierten Funktionen von Amazon EMR zu verwenden, um die Daten in Ihren Cluster zu laden.

DataSync kann Ihnen helfen, die folgenden Aufgaben zu erledigen:

- Replizieren Sie HDFS auf Ihrem Hadoop-Cluster auf Amazon S3 für Geschäftskontinuität
- Kopieren Sie HDFS nach Amazon S3, um Ihre Data Lakes zu füllen
- Daten zwischen Ihrem Hadoop-Cluster HDFS und Amazon S3 zur Analyse und Verarbeitung übertragen

Um Daten in Ihren S3-Bucket hochzuladen, setzen Sie zunächst einen oder mehrere DataSync Agenten im selben Netzwerk ein, in dem sich Ihr lokaler Speicher befindet. Ein Agent ist eine virtuelle Maschine (VM), die zum Lesen von Daten oder zum Schreiben von Daten an einem selbstverwalteten Speicherort verwendet wird. Anschließend aktivieren Sie Ihre Agenten in dem AWS-Konto und AWS-Region wo sich Ihr S3-Bucket befindet.

Nachdem Ihr Agent aktiviert wurde, erstellen Sie einen Quellstandort für Ihren On-Premises-Speicher, einen Zielort für Ihren S3-Bucket und eine Aufgabe. Eine Aufgabe ist ein Satz von zwei Speicherorten (Quelle und Ziel) und eine Reihe von Standardoptionen, die Sie verwenden, um das Verhalten der Aufgabe zu steuern.

Schließlich führen Sie Ihre DataSync Aufgabe aus, um Daten von der Quelle zum Ziel zu übertragen.

Weitere Informationen finden Sie unter Erste Schritte mit AWS DataSync.

Dateien mit verteiltem Cache importieren

Themen

- Unterstützte Dateitypen
- Speicherort der zwischengespeicherten Dateien

- Auf zwischengespeicherte Dateien über Streaming-Anwendungen zugreifen
- Auf zwischengespeicherte Dateien über Streaming-Anwendungen zugreifen

DistributedCache ist ein Hadoop-Feature, das die Effizienz erhöhen kann, wenn eine Zuordnungs- oder Reduzierungs-Aufgabe Zugriff auf allgemeine Daten benötigt. Wenn Ihr Cluster von vorhandenen Anwendungen oder Binärdateien abhängt, die bei der Erstellung des Clusters nicht installiert sind, können Sie den DistributedCache zum Importieren dieser Dateien verwenden. Mit dieser Funktion kann ein Cluster-Knoten die importierten Dateien aus seinem lokalen Dateisystem lesen, anstatt die Dateien von anderen Cluster-Knoten abzurufen.

Weitere Informationen finden Sie unter http://hadoop.apache.org/docs/stable/api/org/apache/hadoop/filecache/ DistributedCache .html.

Sie rufen den DistributedCache beim Erstellen des Clusters auf. Die Dateien werden vor dem Starten des Hadoop-Auftrags nur für die Dauer des Auftrags im Cache zwischengespeichert. Sie können Dateien zwischenspeichern, die auf jedem Hadoop-kompatiblen Dateisystem gespeichert sind, z. B. HDFS oder Amazon S3. Die Standardgröße des Datei-Caches ist 10 GB. Zum Ändern der Größe des Caches konfigurieren Sie den Hadoop-Parameter `local.cache.size` mithilfe der Bootstrap-Aktion neu. Weitere Informationen finden Sie unter Erstellen von Bootstrap-Aktionen zur Installation zusätzlicher Software.

Unterstützte Dateitypen

Der DistributedCache lässt sowohl einzelne Dateien als auch Archive zu. Einzelne Dateien werden schreibgeschützt zwischengespeichert. Für ausführbare und Binärdateien werden Ausführungsberechtigungen festgelegt.

Archive sind eine oder mehrere Dateien, die mit einem Hilfsprogramm verpackt wurden, z. B. `gzip`. DistributedCache übergibt die komprimierten Dateien an jeden Core-Knoten und dekomprimiert das Archiv im Rahmen der Zwischenspeicherung. DistributedCacheunterstützt die folgenden Komprimierungsformate:

- zip
- tgz
- tar.gz
- tar
- jar

Speicherort der zwischengespeicherten Dateien

DistributedCache kopiert Dateien nur auf Core-Knoten. Wenn es im Cluster keine Core-Knoten gibt, kopiert der DistributedCache die Dateien zum Primärknoten.

Der DistributedCache weist die Cache-Dateien dem aktuellen Arbeitsverzeichnis des Mappers und Reducers mithilfe von symbolischen Links zu. Ein symbolischer Link (symlink) ist ein Alias für einen Dateispeicherort, nicht der tatsächliche Speicherort. Der Wert des Parameters, `yarn.nodemanager.local-dirs` in `yarn-site.xml`, gibt den Speicherort der temporären Dateien an. Amazon EMR legt diesen Parameter auf `/mnt/mapred` oder eine Variation basierend auf Instance-Typ und EMR Version fest. Eine Einstellung kann die Werte `/mnt/mapred` und `/mnt1/mapred` haben, da der Instance-Typ über zwei flüchtige Volumes verfügt. Cache-Dateien befinden sich in einem Unterverzeichnis des Speicherorts für temporäre Dateien unter `/mnt/mapred/taskTracker/archive`.

Wenn Sie eine einzelne Datei zwischenspeichern wird sie über den DistributedCache im Verzeichnis `archive` abgelegt. Wenn Sie ein Archiv zwischenspeichern, wird sie vom DistributedCache dekomprimiert und es wird im `/archive` ein Unterverzeichnis mit demselben Namen wie dem Archivdateinamen erstellt. Die einzelnen Dateien befinden sich im neuen Unterverzeichnis.

Sie können den DistributedCache nur bei Verwendung von Streaming verwenden.

Auf zwischengespeicherte Dateien über Streaming-Anwendungen zugreifen

Um aus Ihren Mapper- oder Reducer-Anwendungen auf die zwischengespeicherten Dateien zugreifen zu können, müssen Sie Ihrem Anwendungspfad das aktuelle Arbeitsverzeichnis (./) hinzufügen und die zwischengespeicherten Dateien so referenzieren, als würden sie sich im aktuellen Arbeitsverzeichnis befinden.

Auf zwischengespeicherte Dateien über Streaming-Anwendungen zugreifen

Sie können die AWS Management Console und die verwenden AWS CLI , um Cluster zu erstellen, die Distributed Cache verwenden.

Console

So geben Sie verteilte Cache-Dateien mithilfe der neuen Konsole an

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.

2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und anschließend Create cluster aus.
3. Wählen Sie unter Schritte die Option Schritt hinzufügen aus. Dadurch wird das Dialogfeld Schritt hinzufügen geöffnet. Geben Sie im Feld Argumente die Dateien und Archive an, die im Cache gespeichert werden sollen. Die Größe der Datei (oder Gesamtgröße der Dateien in einer Archivdatei) muss geringer sein als die zugewiesene Cachegröße.

   Wenn Sie eine einzelne Datei zum verteilten Cache hinzufügen möchten, geben Sie `-cacheFile` an, gefolgt vom Namen und Speicherort der Datei, dem Rautenzeichen (#) und dem Namen, den Sie der Datei geben möchten, wenn sie im lokalen Cache abgelegt wird. Im folgenden Beispiel wird gezeigt, wie eine einzelne Datei zum verteilten Cache hinzugefügt wird.

   ```
   -cacheFile \
   s3://DOC-EXAMPLE-BUCKET/file-name#cache-file-name
   ```

   Geben Sie `-cacheArchive` gefolgt von dem Speicherort der Dateien in Amazon S3, dem Rautenzeichen (#) und dann dem Namen ein, den Sie der Sammlung von Dateien im verteilten Cache geben möchten. Im folgenden Beispiel wird gezeigt, wie eine einzelne Datei zum verteilten Cache hinzugefügt wird.

   ```
   -cacheArchive \
   s3://DOC-EXAMPLE-BUCKET/archive-name#cache-archive-name
   ```

   Geben Sie die entsprechenden Werte in die anderen Dialogfelder ein. Die Optionen unterscheiden sich je nach Schritttyp. Um Ihren Schritt hinzuzufügen und das Dialogfeld zu verlassen, wählen Sie Schritt hinzufügen.
4. Wählen Sie alle anderen Optionen aus, die für Ihren Cluster gelten.
5. Um Ihren Cluster jetzt zu starten, wählen Sie Cluster erstellen aus.

CLI

Um verteilte Cache-Dateien mit dem zu spezifizieren AWS CLI

- Um einen Streaming-Schritt beim Erstellen eines Clusters zu senden, geben Sie den Befehl `create-cluster` mit dem Parameter `--steps` ein. Um verteilte Cachedateien mithilfe von

anzugeben AWS CLI, geben Sie beim Senden eines Streaming-Schritts die entsprechenden Argumente an.

Wenn Sie eine einzelne Datei zum verteilten Cache hinzufügen möchten, geben Sie `-cacheFile` an, gefolgt vom Namen und Speicherort der Datei, dem Rautenzeichen (#) und dem Namen, den Sie der Datei geben möchten, wenn sie im lokalen Cache abgelegt wird.

Geben Sie `-cacheArchive` gefolgt von dem Speicherort der Dateien in Amazon S3, dem Rautenzeichen (#) und dann dem Namen ein, den Sie der Sammlung von Dateien im verteilten Cache geben möchten. Im folgenden Beispiel wird gezeigt, wie eine einzelne Datei zum verteilten Cache hinzugefügt wird.

Weitere Informationen zur Verwendung von EMR Amazon-Befehlen in der AWS CLI finden Sie unterhttps://docs.aws.amazon.com/cli/latest/reference/emr.

Example 1

Geben Sie den folgenden Befehl zum Starten eines Clusters und zum Senden eines Streaming-Schritts ein, der `-cacheFile` zum Hinzufügen einer Datei, `sample_dataset_cached.dat`, zum Cache verwendet.

```
aws emr create-cluster --name "Test cluster" --release-label emr-4.0.0 --
applications Name=Hive Name=Pig --use-default-roles --ec2-attributes KeyName=myKey
 --instance-type m5.xlarge --instance-count 3 --steps Type=STREAMING,Name="Streaming
 program",ActionOnFailure=CONTINUE,Args=["--files","s3://my_bucket/my_mapper.py
 s3://my_bucket/my_reducer.py","-mapper","my_mapper.py","-reducer","my_reducer.py,"-
input","s3://my_bucket/my_input","-output","s3://my_bucket/my_output", "-
cacheFile","s3://my_bucket/sample_dataset.dat#sample_dataset_cached.dat"]
```

Wenn Sie die Instance-Anzahl ohne den `--instance-groups`-Parameter angeben, wird ein einzelner Primärknoten gestartet. Die verbleibenden Instances werden dabei als Core-Knoten gestartet. Alle Knoten verwenden den im Befehl angegebenen Instance-Typ.

Wenn Sie die EMR Standard-Servicerolle und das EC2 Instanzprofil noch nicht erstellt haben, geben Sie vor der Eingabe des `create-cluster` Unterbefehls ein, `aws emr create-default-roles` um sie zu erstellen.

Example 2

Der folgende Befehl erstellt einen Streaming-Cluster und verwendet `-cacheArchive`, um dem Cache ein Dateiarchiv hinzuzufügen.

```
aws emr create-cluster --name "Test cluster" --release-label emr-4.0.0 --
applications Name=Hive Name=Pig --use-default-roles --ec2-attributes KeyName=myKey
 --instance-type m5.xlarge --instance-count 3 --steps Type=STREAMING,Name="Streaming
 program",ActionOnFailure=CONTINUE,Args=["--files","s3://my_bucket/my_mapper.py
 s3://my_bucket/my_reducer.py","-mapper","my_mapper.py","-reducer","my_reducer.py,"-
input","s3://my_bucket/my_input","-output","s3://my_bucket/my_output", "-
cacheArchive","s3://my_bucket/sample_dataset.tgz#sample_dataset_cached"]
```

Wenn Sie die Instance-Anzahl ohne den `--instance-groups`-Parameter angeben, wird ein einzelner Primärknoten gestartet. Die verbleibenden Instances werden dabei als Core-Knoten gestartet. Alle Knoten verwenden den im Befehl angegebenen Instance-Typ.

Wenn Sie die EMR Standard-Servicerolle und das EC2 Standardinstanzprofil noch nicht erstellt haben, geben Sie vor der Eingabe des `create-cluster` Unterbefehls ein, `aws emr create-default-roles` um sie zu erstellen.

So verarbeiten Sie komprimierte Dateien

Hadoop überprüft die Dateierweiterung zur Erkennung von komprimierten Dateien. Die von Hadoop unterstützten Komprimierungstypen sind: gzip, bzip2 und. LZO Sie müssen keine zusätzlichen Schritte ausführen, um Dateien dieser Komprimierungstypen zu extrahieren, da Hadoop diesen Vorgang für Sie erledigt.

Um LZO Dateien zu indizieren, können Sie die Hadoop-Lzo-Bibliothek verwenden, die von hadoop-lzo heruntergeladen werden kann. https://github.com/kevinweil/ Beachten Sie, dass Amazon EMR keine Entwickelrunterstützung bei der Verwendung dieses Tools anbietet, da es sich um eine Bibliothek eines Drittanbieters handelt. Informationen zur Nutzung finden Sie in der Readme-Datei für hadoop-lzo.

DynamoDB-Daten in Hive importieren

Die von Amazon bereitgestellte Implementierung von Hive EMR umfasst Funktionen, mit denen Sie Daten zwischen DynamoDB und einem Amazon-Cluster importieren und exportieren können. EMR Dies ist nützlich, wenn Ihre Eingabedaten in DynamoDB gespeichert sind. Weitere Informationen

finden Sie unter Exportieren, Importieren, Abfragen und Verbinden von Tabellen in DynamoDB mithilfe von Amazon. EMR

Verbindung zu Daten mit AWS Direct Connect herstellen

AWS Direct Connect ist ein Service, mit dem Sie von Ihrem Rechenzentrum, Büro oder Ihrer Colocation-Umgebung aus eine private, dedizierte Netzwerkverbindung zu Amazon Web Services herstellen können. Wenn Sie über große Mengen an Eingabedaten verfügen, AWS Direct Connect kann die Verwendung Ihre Netzwerkkosten senken, den Bandbreitendurchsatz erhöhen und ein einheitlicheres Netzwerkerlebnis bieten als internetbasierte Verbindungen. Weitere Informationen finden Sie im AWS Direct Connect -Benutzerhandbuch.

Große Datenmengen mit AWS Snowball hochladen

AWS Snowball ist ein Service, mit dem Sie große Datenmengen schnell zwischen Amazon Simple Storage Service (Amazon S3) und Ihrem Datenspeicherort vor Ort übertragen faster-than-internet können. Snowball unterstützt zwei Auftragstypen: Importaufträte und Exportaufträge. Importaufträge beinhalten eine Datenübertragung von einer On-Premises-Quelle zu einem Amazon-S3-Bucket. Exportaufträge beinhalten eine Datenübertragung aus einem Amazon-S3-Bucket zu einer On-Premises-Quelle. Bei beiden Auftragstypen sichern und schützen Snowball-Geräte Ihre Daten, während regionale Spediteure sie zwischen Amazon S3 und Ihrem Datenspeicherort vor Ort transportieren. Snowball-Geräte sind physisch robust und werden durch die AWS Key Management Service (AWS KMS) geschützt. Weitere Informationen finden Sie im AWS Snowball -Edge-Entwicklerhandbuch.

Einen Ausgabespeicherort konfigurieren

Das gängigste Ausgabeformat eines EMR Amazon-Clusters sind Textdateien, entweder komprimiert oder unkomprimiert. Diese Dateien werden in der Regel in einen Amazon-S3-Bucket geschrieben. Dieser Bucket muss erstellt werden, bevor Sie den Cluster starten. Sie geben den S3-Bucket als Ausgabespeicherort an, wenn Sie den Cluster starten.

Weitere Informationen finden Sie unter den folgenden Themen:

Themen

- Erstellen und Konfigurieren eines Amazon S3-Buckets
- Welche Formate kann Amazon EMR zurückgeben?
- So schreiben Sie Daten in einen Amazon S3-Bucket, für den Sie keine Rechte haben
- Die Ausgabe Ihres Clusters komprimieren

Erstellen und Konfigurieren eines Amazon S3-Buckets

Amazon EMR (AmazonEMR) verwendet Amazon S3 zum Speichern von Eingabedaten, Protokolldateien und Ausgabedaten. Amazon S3 bezeichnet diese Speicherorte als Buckets. Buckets unterliegen bestimmten Einschränkungen und Beschränkungen, um Amazon S3 und DNS den Anforderungen zu entsprechen. Weitere Informationen finden Sie unter Bucket-Einschränkungen und -Limits im Amazon Simple Storage Service-Entwicklerhandbuch.

Um einen Amazon-S3-Bucket zu erstellen, befolgen Sie die Anweisungen auf der Seite Bucket erstellen im Benutzerhandbuch zu Amazon Simple Storage Service.

> **Note**
>
> Wenn Sie im Assistenten Bucket erstellen die Protokollierung aktivieren, werden nur Bucket-Zugriffsprotokolle aktiviert und nicht Cluster-Protokolle.

> **Note**
>
> Weitere Informationen zur Angabe regionsspezifischer Buckets finden Sie unter Buckets and Regions im Amazon Simple Storage Service Developer Guide und Available Region Endpoints for the. AWS SDKs

Nachdem Sie Ihren Bucket erstellt haben, können Sie die entsprechenden Zugriffsberechtigungen hierzu einrichten. Hierbei sollten Sie sich selbst (als Eigentümer) Lese- und Schreibzugriff erteilen. Wir empfehlen Ihnen dringend, bei der Konfiguration Ihres Buckets die bewährten Sicherheitsmethoden für Amazon S3 zu befolgen.

Erforderliche Amazon-S3-Buckets müssen vorhanden sein, bevor Sie einen Cluster erstellen können. Sie müssen alle erforderlichen Skripts und Daten auf Amazon S3 hochladen, auf die im Cluster verwiesen wird. In der folgenden Tabelle werden Beispiele für Speicherorte für Daten, Skripts und Protokolldateien beschrieben.

Informationen	Beispielspeicherort auf Amazon S3
Skript oder Programm	`s3://amzn-s3-demo-bucket1/script/MapperScript.py`

Informationen	Beispielspeicherort auf Amazon S3
Protokolldateien	`s3://amzn-s3-demo-bucket1/logs`
Eingabedaten	`s3://amzn-s3-demo-bucket1/input`
Ausgabedaten	`s3://amzn-s3-demo-bucket1/output`

Welche Formate kann Amazon EMR zurückgeben?

Das Standardausgabeformat für einen Cluster ist Text mit Schlüssel-Wert-Paaren, die in einzelne Zeilen der Textdateien geschrieben werden. Dies ist das am häufigsten verwendete Ausgabeformat.

Wenn Ihre Ausgabedaten in einem anderen Format geschrieben werden müssen als Standardtextdateien, können Sie die Hadoop-Benutzeroberfläche `OutputFormat` verwenden, um andere Ausgabetypen anzugeben. Sie können auch eine Unterklasse der `FileOutputFormat`-Klasse für den Umgang mit benutzerdefinierten Datentypen verwenden. Weitere Informationen finden Sie unter http://hadoop.apache.org/docs/current/api/org/apache/hadoop/mapred/ OutputFormat .html.

Wenn Sie einen Hive-Cluster starten, können Sie einen Serializer/Deserializer (SerDe) verwenden, um Daten in einem bestimmten Format auszugeben. HDFS Weitere Informationen finden Sie unter https://cwiki.apache.org/confluence/display/Hive/. SerDe

So schreiben Sie Daten in einen Amazon S3-Bucket, für den Sie keine Rechte haben

Wenn Sie eine Datei in einen Amazon Simple Storage Service (Amazon S3)-Bucket schreiben, können standardmäßig nur Sie die Datei lesen. Es wird davon ausgegangen, dass Sie Dateien in Ihre Buckets schreiben. Diese Standardeinstellung dient dem Schutz Ihrer Dateien.

Wenn Sie jedoch einen Cluster ausführen und möchten, dass die Ausgabe in den Amazon S3 S3-Bucket eines anderen AWS Benutzers schreibt und Sie möchten, dass dieser andere AWS Benutzer diese Ausgabe lesen kann, müssen Sie zwei Dinge tun:

- Bitten Sie den anderen AWS Benutzer, Ihnen Schreibberechtigungen für seinen Amazon S3 S3-Bucket zu gewähren. Der Cluster, den Sie starten, wird unter Ihren AWS Anmeldeinformationen ausgeführt, sodass alle Cluster, die Sie starten, auch in den Bucket dieses anderen AWS Benutzers schreiben können.
- Legen Sie Leseberechtigungen für den anderen AWS Benutzer für die Dateien fest, die Sie oder der Cluster in den Amazon S3 S3-Bucket schreiben. Die einfachste Methode, diese

Leseberechtigungen festzulegen, besteht darin, vorgefertigte Zugriffskontrolllisten (ACLs) zu verwenden. Dabei handelt es sich um eine Reihe von vordefinierten Zugriffsrichtlinien, die von Amazon S3 definiert wurden.

Informationen darüber, wie der andere AWS Benutzer Ihnen Berechtigungen zum Schreiben von Dateien in den Amazon S3 S3-Bucket des anderen Benutzers gewähren kann, finden Sie unter Bearbeiten von Bucket-Berechtigungen im Amazon Simple Storage Service-Benutzerhandbuch.

Damit Ihr Cluster ACLs beim Schreiben von Dateien in Amazon S3 die Option „Gespeichert“ verwendet, setzen Sie die `fs.s3.canned.acl` Cluster-Konfigurationsoption auf „Gespeichert“, ACL um sie zu verwenden. In der folgenden Tabelle sind die aktuell definierten gespeicherten Dateien aufgeführtACLs.

Gescannt ACL	Beschreibung
`AuthenticatedRead`	Gibt an, dass dem Eigentümer `Permission.FullControl` und dem Berechtigungsempfänger der Gruppe `GroupGrantee.AuthenticatedUsers` der Zugriff `Permission.Read` gewährt wird.
`BucketOwnerFullControl`	Gibt an, dass dem Bucket-Eigentümer `Permission.FullControl` gewährt wird. Der Bucket-Eigentümer muss nicht unbedingt derselbe wie der Objekteigentümer sein.
`BucketOwnerRead`	Gibt an, dass dem Bucket-Eigentümer `Permission.Read` gewährt wird. Der Bucket-Eigentümer muss nicht unbedingt derselbe wie der Objekteigentümer sein.
`LogDeliveryWrite`	Gibt an, dass dem Eigentümer `Permission.FullControl` und dem Berechtigungsempfänger der Gruppe `GroupGrantee.LogDelivery` der Zugriff `Permission.Write` gewährt wird, damit Zugriffsprotokolle bereitgestellt werden können.
`Private`	Gibt an, dass dem Eigentümer `Permission.FullControl` gewährt wird.

Gescannt ACL	Beschreibung
`PublicRead`	Gibt an, dass dem Eigentümer `Permission.FullControl` und dem Berechtigungsempfänger der Gruppe `GroupGrantee.AllUsers` der Zugriff `Permission.Read` gewährt wird.
`PublicReadWrite`	Gibt an, dass dem Eigentümer `Permission.FullControl` und dem Berechtigungsempfänger der Gruppe `GroupGrantee.AllUsers` die Zugriffsberechtigungen `Permission.Read` und `Permission.Write` gewährt wird.

Die Cluster-Konfigurationsoptionen können auf vielfältige Weise festgelegt werden, je nach Typ des ausgeführten Clusters. Die folgenden Verfahren zeigen die Festlegung der Option für allgemeine Anwendungsfälle.

So schreiben Sie Dateien mithilfe von canned ACLs in Hive

- Stellen Sie in der Hive-Befehlszeile die `fs.s3.canned.acl` Konfigurationsoption auf „Gespeichert“ ein. ACL Sie möchten, dass der Cluster für Dateien festgelegt wird, die er in Amazon S3 schreibt. Um auf die Hive-Befehlszeile zuzugreifen, stellen Sie eine Verbindung zum Master-Knoten her und geben Sie SSH Hive an der Hadoop-Befehlszeile ein. Weitere Informationen finden Sie unter Connect zum Primärknoten her mit SSH.

  Im folgenden Beispiel wird die Konfigurationsoption `fs.s3.canned.acl` auf `BucketOwnerFullControl` festgelegt. Dadurch erhält der Eigentümer des Amazon-S3-Buckets vollständige Kontrolle über die Datei. Beachten Sie: Der Festlegungsbefehl erfordert die Beachtung der Groß- und Kleinschreibung und enthält keine Anführungszeichen oder Leerzeichen.

  ```
  hive> set fs.s3.canned.acl=BucketOwnerFullControl;
  create table acl (n int) location 's3://acltestbucket/acl/';
  insert overwrite table acl select count(*) from acl;
  ```

Die beiden letzten Zeilen des Beispiels erstellen eine Tabelle, die in Amazon S3 gespeichert wird, und schreiben Daten in die Tabelle.

So schreiben Sie Dateien mithilfe von Canned in Pig ACLs

- Stellen Sie in der Pig-Befehlszeile die `fs.s3.canned.acl` Konfigurationsoption auf „Gespeichert“ ein. ACL Sie möchten, dass der Cluster für Dateien festgelegt wird, die er in Amazon S3 schreibt. Um auf die Pig-Befehlszeile zuzugreifenSSH, stellen Sie eine Verbindung zum Master-Knoten her und geben Sie Pig an der Hadoop-Befehlszeile ein. Weitere Informationen finden Sie unter Connect zum Primärknoten her mit SSH.

  Im folgenden Beispiel wird die `fs.s3.canned.acl` Konfigurationsoption auf gesetzt BucketOwnerFullControl, wodurch der Besitzer des Amazon S3 S3-Buckets die vollständige Kontrolle über die Datei hat. Beachten Sie, dass der Befehl set ein Leerzeichen vor dem gespeicherten ACL Namen enthält und keine Anführungszeichen enthält.

  ```
  pig> set fs.s3.canned.acl BucketOwnerFullControl;
  store some data into 's3://acltestbucket/pig/acl';
  ```

Um Dateien mit dem Befehl „Einscannen“ ACLs in einem benutzerdefinierten Format zu schreiben JAR

- Legen Sie mit Hadoop die Konfigurationsoption `fs.s3.canned.acl` mit -D-Flag fest. Das wird im Beispiel unten veranschaulicht.

  ```
  hadoop jar hadoop-examples.jar wordcount
  -Dfs.s3.canned.acl=BucketOwnerFullControl s3://mybucket/input s3://mybucket/output
  ```

Die Ausgabe Ihres Clusters komprimieren

Themen

- Kompression der Ausgabedaten

- Intermediäre Datenkompression
- Die Snappy-Bibliothek mit Amazon verwenden EMR

Kompression der Ausgabedaten

Dies komprimiert die Ausgabe Ihres Hadoop-Auftrags. Wenn Sie TextOutputFormat das Ergebnis verwenden, handelt es sich um eine GZIP-komprimierte Textdatei. Wenn Sie schreiben, SequenceFiles ist das Ergebnis eine SequenceFile , die intern komprimiert ist. Dies kann aktiviert werden, indem Sie die Konfigurationseinstellung "mapred.output.compress" auf "true" setzen.

Wenn Sie einen Streaming-Auftrag ausführen, können Sie dies aktivieren, indem Sie dem Streaming-Auftrag diese Argumente übergeben.

```
-jobconf mapred.output.compress=true
```

Sie können auch mit einer Bootstrap-Aktion alle Auftragsausgaben automatisch komprimieren. So können Sie dies mit dem Ruby-Client ausführen:

```
--bootstrap-actions s3://elasticmapreduce/bootstrap-actions/configure-hadoop \
--args "-s,mapred.output.compress=true"
```

Wenn Sie eine Custom Jar schreiben, können Sie die Ausgabekompression mit folgender Zeile bei der Erstellung Ihres Auftrags aktivieren:

```
FileOutputFormat.setCompressOutput(conf, true);
```

Intermediäre Datenkompression

Wenn Ihr Auftrag eine erhebliche Menge Daten von den Mappern zu den Reducern verlagert, können Sie eine Leistungsverbesserung durch Aktivierung der intermediären Kompression feststellen. Sie komprimieren die Zuweisungsausgabe und dekomprimieren sie, wenn sie auf dem Core-Knoten eingeht. Die Konfigurationseinstellung ist "mapred.compress.map.output". Sie können sie ähnlich wie die Ausgabekompression aktivieren.

Wenn Sie eine Custom Jar schreiben, verwenden Sie den folgenden Befehl:

```
conf.setCompressMapOutput(true);
```

Die Snappy-Bibliothek mit Amazon verwenden EMR

Snappy ist eine Komprimierungs- und Dekomprimierungsbibliothek, die für höhere Geschwindigkeit optimiert ist. Es ist auf Amazon EMR AMIs Version 2.0 und höher verfügbar und wird als Standard für die Zwischenkomprimierung verwendet. Weitere Informationen zu Snappy finden Sie unter http://code.google.com/p/snappy/.

Primärknoten planen und konfigurieren

Wenn Sie einen EMR Amazon-Cluster starten, können Sie wählen, ob Sie einen oder drei primäre Knoten in Ihrem Cluster haben möchten. Hochverfügbarkeit für Instance-Flotten wird mit den EMR Amazon-Versionen 5.36.1, 5.36.2, 6.8.1, 6.9.1, 6.10.1, 6.11.1, 6.12.0 und höher unterstützt. Für Instance-Gruppen wird Hochverfügbarkeit mit EMR Amazon-Versionen 5.23.0 und höher unterstützt. Um die Cluster-Verfügbarkeit weiter zu verbessern, EMR kann Amazon Amazon EC2 Placement-Gruppen verwenden, um sicherzustellen, dass die primären Knoten auf unterschiedlicher zugrunde liegender Hardware platziert werden. Weitere Informationen finden Sie unter EMRAmazon-Integration mit EC2 Platzierungsgruppen.

Ein EMR Amazon-Cluster mit mehreren Primärknoten bietet die folgenden Vorteile:

- Der Primärknoten ist nicht länger eine einzelne Fehlerquelle. Wenn ein Primärknoten ausfällt, verwendet der Cluster die beiden anderen Primärknoten und wird ohne Unterbrechung weiter ausgeführt. In der Zwischenzeit ersetzt Amazon den ausgefallenen Primärknoten EMR automatisch durch einen neuen, der mit derselben Konfiguration und denselben Bootstrap-Aktionen bereitgestellt wird.
- Amazon EMR aktiviert die Hadoop-Hochverfügbarkeitsfunktionen HDFS NameNode YARN ResourceManager und unterstützt Hochverfügbarkeit für einige andere Open-Source-Anwendungen.

  Weitere Informationen darüber, wie ein EMR Amazon-Cluster mit mehreren Primärknoten Open-Source-Anwendungen und andere EMR Amazon-Funktionen unterstützt, finden Sie unterUnterstützte Anwendungen und Features.

> **Note**
>
> Der Cluster kann sich nur in einer einzigen Availability Zone oder einem einzigen Subnetz befinden.

Dieser Abschnitt enthält Informationen zu den unterstützten Anwendungen und Funktionen eines EMR Amazon-Clusters mit mehreren Primärknoten sowie zu Konfigurationsdetails, bewährten Methoden und Überlegungen zum Starten des Clusters.

Themen

Unterstützte Anwendungen und Features

Dieses Thema enthält Informationen zu den Hadoop-Hochverfügbarkeitsfunktionen von HDFS NameNode und YARN ResourceManager in einem EMR Amazon-Cluster und darüber, wie die Hochverfügbarkeitsfunktionen mit Open-Source-Anwendungen und anderen Amazon-Funktionen funktionieren. EMR

Hochverfügbarkeit HDFS

Ein EMR Amazon-Cluster mit mehreren Primärknoten ermöglicht die HDFS NameNode Hochverfügbarkeitsfunktion in Hadoop. Weitere Informationen finden Sie unter HDFSHochverfügbarkeit.

In einem EMR Amazon-Cluster sind zwei oder mehr separate Knoten konfiguriert als NameNodes. Einer NameNode befindet sich in einem `active` Staat und die anderen befinden sich in einem `standby` Staat. Wenn der Knoten mit `active` NameNode ausfällt, EMR startet Amazon einen automatischen HDFS Failover-Prozess. Ein Knoten mit `standby` NameNode wird `active` und übernimmt alle Client-Operationen im Cluster. Amazon EMR ersetzt den ausgefallenen Knoten durch einen neuen, der sich dann wieder als `standby` Knoten verbindet.

> **Note**
>
> In den EMR Amazon-Versionen 5.23.0 bis einschließlich 5.30.1 laufen nur zwei der drei Primärknoten. HDFS NameNode

Wenn Sie herausfinden möchten, welcher NameNode ist`active`, können Sie damit eine Verbindung SSH zu einem beliebigen Primärknoten im Cluster herstellen und den folgenden Befehl ausführen:

```
hdfs haadmin -getAllServiceState
```

In der Ausgabe werden die Knoten, auf denen installiert NameNode ist, und ihr Status aufgeführt. Zum Beispiel

```
ip-##-#-#-##1.ec2.internal:8020 active
ip-##-#-#-##2.ec2.internal:8020 standby
ip-##-#-#-##3.ec2.internal:8020 standby
```

Hohe Verfügbarkeit YARN ResourceManager

Ein EMR Amazon-Cluster mit mehreren Primärknoten ermöglicht die YARN ResourceManager Hochverfügbarkeitsfunktion in Hadoop. Weitere Informationen finden Sie unter ResourceManager Hochverfügbarkeit.

YARN ResourceManager Läuft in einem EMR Amazon-Cluster mit mehreren Primärknoten auf allen drei Primärknoten. Einer ResourceManager befindet sich im `active` Bundesstaat, und die anderen beiden befinden sich im `standby` Bundesstaat. Wenn der primäre Knoten mit `active` ResourceManager ausfällt, EMR startet Amazon einen automatischen Failover-Prozess. Ein primärer Knoten mit a `standby` ResourceManager übernimmt alle Operationen. Amazon EMR ersetzt den ausgefallenen Primärknoten durch einen neuen, der dann wieder dem ResourceManager Quorum als beitritt. `standby`

Sie können eine Verbindung zu „http://“ herstellen*`master-public-dns-name`*:8088/cluster“ für jeden primären Knoten, wodurch Sie automatisch zum Ressourcenmanager weitergeleitet werden. `active` Um herauszufinden, welcher Ressourcenmanager das ist`active`, stellen Sie eine Verbindung SSH zu einem beliebigen primären Knoten im Cluster her. Führen Sie anschließend den folgenden Befehl aus, um die Liste der drei Primärknoten und deren Status abzurufen:

```
yarn rmadmin -getAllServiceState
```

Unterstützte Anwendungen in einem EMR Amazon-Cluster mit mehreren Primärknoten

Sie können die folgenden Anwendungen auf einem EMR Amazon-Cluster mit mehreren Primärknoten installieren und ausführen. Für jede Anwendung variiert der Failover-Prozess des Primärknotens.

Anwendung	Verfügbarkeit während Failover für den Primärknoten	Hinweise
Flink	Verfügbarkeit nicht durch Failover für den Primärknoten betroffen	Flink-Jobs bei Amazon EMR werden als YARN Anwendungen ausgeführt. Flink JobManagers läuft wie ApplicationMasters auf YARN den Kernknoten. Der JobManager wird durch den Failover-Prozess des Primärknotens nicht beeinflusst. Wenn Sie EMR Amazon-Version 5.27.0 oder früher verwenden, JobManager handelt es sich um einen einzigen Fehlerpunkt. Wenn der JobManager fehlschlägt, gehen alle Jobstatus verloren und die laufenden Jobs werden nicht wieder aufgenommen. Sie können JobManager Hochverfügbarkeit aktivieren, indem Sie die Anzahl der Anwendungsversuche, das Checkpointing und die Aktivierung ZooKeeper als Statusspeicher für Flink konfigurieren. Weitere Informationen finden Sie unter Konfiguration von Flink auf einem EMR Amazon-Cluster mit mehreren Primärknoten. Ab EMR Amazon-Version 5.28.0 ist keine manuelle Konfiguration erforderlich, um JobManager Hochverfügbarkeit zu aktivieren.
Ganglia	Verfügbarkeit nicht durch Failover für den Primärknoten betroffen	Ganglia ist auf allen Primärknoten verfügbar. Daher kann Ganglia während des Failover-

Anwendung	Verfügbarkeit während Failover für den Primärknoten	Hinweise
		Prozesses für den Primärknoten weiter ausgeführt werden.
Hadoop	Hohe Verfügbarkeit	HDFS NameNode und führt YARN ResourceManager automatisch ein Failover zum Standby-Knoten durch, wenn der aktive Primärknoten ausfällt.
HBase	Hohe Verfügbarkeit	HBaseführt automatisch ein Failover zum Standby-Knoten durch, wenn der aktive Primärknoten ausfällt. Wenn Sie die Verbindung HBase über einen REST oder Thrift-Server herstellen, müssen Sie zu einem anderen Primärknoten wechseln, wenn der aktive Primärknoten ausfällt.
HCatalog	Verfügbarkeit nicht durch Failover für den Primärknoten betroffen	HCatalogbasiert auf dem Hive-Metastore, der sich außerhalb des Clusters befindet. HCatalogbleibt während des Failover-Prozesses für den primären Knoten verfügbar.
JupyterHub	Hohe Verfügbarkeit	JupyterHub ist auf allen drei primären Instanzen installiert. Es wird dringend empfohlen, die Notebook-Persistenz zu konfigurieren, um Notebookverlust bei einem Ausfall des Primärknotens zu verhindern. Weitere Informationen finden Sie unter Konfigurieren von Persistenz für Notebooks in Amazon S3.

Anwendung	Verfügbarkeit während Failover für den Primärknoten	Hinweise
Livy	Hohe Verfügbarkeit	Livy wird auf allen drei Primärknoten installiert. Bei einem Ausfall des aktiven Primärknotens verlieren Sie den Zugriff auf die aktuelle Livy-Sitzung und müssen eine neue Livy-Sitzung auf einem anderen Primärknoten oder auf dem neuen Ersatzknoten erstellen.
Mahout	Verfügbarkeit nicht durch Failover für den Primärknoten betroffen	Da Mahout keinen Daemon besitzt, hat der Failover-Prozess für den Primärknoten keine Auswirkungen.
MXNet	Verfügbarkeit nicht durch Failover für den Primärknoten betroffen	Da MXNet es keinen Daemon gibt, ist er vom Failover-Prozess des Primärknotens nicht betroffen.
Phoenix	Hochverfügbarkeit	Phoenix QueryServer läuft nur auf einem der drei Primärknoten. Phoenix ist auf allen drei Mastern so konfiguriert, dass er den Phoenix verbindet. QueryServer Sie können die private IP des Phoenix QueryServer anhand der `/etc/phoenix/conf/phoenix-env.sh` - Datei finden
Pig	Verfügbarkeit nicht durch Failover für den Primärknoten betroffen	Da Pig keinen Daemon besitzt, hat der Failover-Prozess für den Primärknoten keine Auswirkungen.
Spark	Hohe Verfügbarkeit	Alle Spark-Anwendungen werden in YARN Containern ausgeführt und können auf Failover des Primärknotens genauso reagieren wie HochverfügbarkeitsfunktionenYARN.

Anwendung	Verfügbarkeit während Failover für den Primärknoten	Hinweise
Sqoop	Hohe Verfügbarkeit	Standardmäßig speichern sqoop-job und sqoop-metastore Daten (Auftragsbeschreib ungen) auf der lokalen Festplatte des Masters, der den Befehl ausführt. Wenn Sie Metastore-Daten in einer externen Datenbank speichern möchten, schlagen Sie bitte in der Apache Sqoop-Dokumentation nach.
Tez	Hohe Verfügbarkeit	Da Tez-Container darauf laufenYARN, verhält sich Tez genauso wie YARN während des Failover-Prozesses für den primären Knoten.
TensorFlow	Verfügbarkeit nicht durch Failover für den Primärknoten betroffen	Da TensorFlow es keinen Daemon gibt, ist er vom Failover-Prozess des Primärknotens nicht betroffen.
Zeppelin	Hohe Verfügbarkeit	Zeppelin wird auf allen drei Primärknoten installiert. Zeppelin speichert standardmäßig Notizen und Interpreter-Konfigurationen, um Datenverlust zu HDFS verhindern. Interpret ersitzungen sind über alle drei Primär-Instances vollständig isoliert. Sitzungsdaten gehen beim Master-Ausfall verloren. Es wird empfohlen, dieselbe Notiz nicht gleichzeitig auf verschied enen Primär-Instances zu ändern.

Anwendung	Verfügbarkeit während Failover für den Primärknoten	Hinweise
ZooKeeper	Hohe Verfügbarkeit	ZooKeeper ist die Grundlage der HDFS automatischen Failover-Funktion. ZooKeeper bietet einen hochverfügbaren Dienst zur Verwaltung von Koordinationsdaten, zur Benachrichtigung von Clients über Änderungen an diesen Daten und zur Überwachung von Clients im Hinblick auf Ausfälle. Weitere Informationen finden Sie unter HDFSAutomatisches Failover.

Um die folgenden Anwendungen in einem EMR Amazon-Cluster mit mehreren Primärknoten auszuführen, müssen Sie eine externe Datenbank konfigurieren. Die externe Datenbank befindet sich außerhalb des Clusters. Daher sind Daten während des Failover-Prozesses für den Primärknoten persistent. Für die folgenden Anwendungen werden die Servicekomponenten während des Primärknoten-Failoverprozesses automatisch wiederhergestellt, aktive Aufträge können jedoch fehlschlagen und müssen erneut versucht werden.

Anwendung	Verfügbarkeit während Failover für den Primärknoten	Hinweise
Hive	Hohe Verfügbarkeit, jedoch ausschließlich für Service-Komponenten	Ein externer Metastore für Hive ist erforderlich. Dabei muss es sich um einen SQL externen Metastore von My external handeln, da Postgre SQL für Multi-Master-Cluster nicht unterstützt wird. Weiter Informationen finden Sie unter Konfigurieren eines externen Metastores für Hive.
Hue	Hohe Verfügbarkeit, jedoch ausschließlich für Service-Komponenten	Eine externe Datenbank für Hue ist erforderlich. Weitere Informationen finden Sie unter

Anwendung	Verfügbarkeit während Failover für den Primärknoten	Hinweise
		Verwenden von Hue mit einer Remote-Datenbank in Amazon RDS.
Oozie	Hohe Verfügbarkeit, jedoch ausschließlich für Service-Komponenten	Eine externe Datenbank für Oozie ist erforderlich. Weitere Informationen finden Sie unter Verwenden von Oozie mit einer Remote-Datenbank in Amazon RDS. Oozie-Server und Oozie-Client sind auf allen drei Primärknoten installiert. Die oozie-clients sind so konfiguriert, dass sie standardmäßig eine Verbindung mit dem richtigen oozie-server herstellen.
PrestoDB oder Presto / Trino SQL	Hohe Verfügbarkeit, jedoch ausschließlich für Service-Komponenten	Ein externer Hive-Metastore für PrestoDB (Presto SQL auf Amazon EMR 6.1.0-6.3.0 oder Trino auf Amazon 6.4.0 und höher) ist erforderlich. EMR Sie können Presto mit dem AWS Glue Data Catalog oder eine externe My SQL database for Hive verwenden. Presto CLI ist auf allen drei Primärknoten installiert, sodass Sie damit von jedem der Primärknoten aus auf den Presto Coordinator zugreifen können. Der Presto Coordinator ist nur auf einem Primärknoten installiert. Sie können den DNS Namen des primären Knotens ermitteln, auf dem der Presto Coordinator installiert ist, indem Sie Amazon anrufen EMR `describe-cluster` API und den zurückgegebenen Wert des `MasterPublicDnsName` Felds in der Antwort lesen.

> **Note**
>
> Wenn ein primärer Knoten ausfällt, beendet Ihre Java Database Connectivity (JDBC) oder Open Database Connectivity (ODBC) ihre Verbindung zum primären Knoten. Sie können eine Verbindung mit einem der verbleibenden Primärknoten herstellen und Ihre Arbeit fortsetzen, da der Hive-Metastore-Daemon auf allen Primärknoten ausgeführt wird. Sie können auch warten, bis der ausgefallene Primärknoten ersetzt wird.

So funktionieren EMR Amazon-Funktionen in einem Cluster mit mehreren Primärknoten

Verbindung zu Primärknoten herstellen mit SSH

Sie können sich mit jedem der drei Primärknoten in einem EMR Amazon-Cluster auf dieselbe Weise verbinden wie mit SSH einem einzelnen Primärknoten. Weitere Informationen finden Sie unter Connect dem Primärknoten herstellen mit SSH.

Wenn ein primärer Knoten ausfällt, wird Ihre SSH Verbindung zu diesem primären Knoten beendet. Um Ihre Arbeit fortzusetzen, können Sie eine Verbindung mit einem der beiden anderen Primärknoten herstellen. Alternativ können Sie auf den neuen Primärknoten zugreifen, nachdem Amazon den ausgefallenen Knoten durch einen neuen EMR ersetzt hat.

> **Note**
>
> Die private IP-Adresse des ersetzenden Primärknoten ist mit der privaten IP-Adresse des vorherigen Primärknotens identisch. Die öffentliche IP-Adresse des ersetzenden Primärknotens wird möglicherweise geändert. Sie können die neuen IP-Adressen in der Konsole oder mithilfe des `describe-cluster` Befehls in der abrufen AWS CLI. NameNode läuft nur auf zwei der primären Knoten. Sie können jedoch `hdfs` CLI Befehle ausführen und Jobs ausführen, um HDFS auf alle drei primären Knoten zuzugreifen.

Arbeiten mit Schritten in einem EMR Amazon-Cluster mit mehreren Primärknoten

Sie können Schritte auf dieselbe Weise an einen EMR Amazon-Cluster mit mehreren Primärknoten senden, wie Sie mit Schritten in einem Cluster mit einem einzigen primären Knoten arbeiten. Weitere Informationen finden Sie unter Übermitteln von Arbeit an einen Cluster.

Im Folgenden finden Sie Überlegungen zur Arbeit mit Schritten in einem EMR Amazon-Cluster mit mehreren Primärknoten:

- Wenn ein primärer Knoten ausfällt, werden die Schritte, die auf dem primären Knoten ausgeführt werden, als gekennzeichnetFAILED. Alle lokal geschriebenen Daten gehen verloren. Der Status spiegelt jedoch FAILED möglicherweise nicht den tatsächlichen Status der Schritte wider.
- Wenn ein laufender Schritt eine YARN Anwendung gestartet hat, obwohl der primäre Knoten ausfällt, kann der Schritt aufgrund des automatischen Failovers des primären Knotens fortgesetzt und erfolgreich ausgeführt werden.
- Sie sollten den Status von Schritten anhand der Ausgaben der Aufgaben überprüfen. MapReduce Jobs verwenden beispielsweise eine _SUCCESS Datei, um festzustellen, ob der Job erfolgreich abgeschlossen wurde.
- Es wird empfohlenCONTINUE, den ActionOnFailure Parameter auf oder CANCEL _ AND _ WAIT statt auf TERMINATE _ JOB _ FLOW oder TERMINATE _ festzulegenCLUSTER.

Automatischer Beendigungsschutz

Amazon aktiviert EMR automatisch den Kündigungsschutz für alle Cluster mit mehreren Primärknoten und überschreibt alle Einstellungen für die Schrittausführung, die Sie bei der Erstellung des Clusters angeben. Sie können den Kündigungsschutz deaktivieren, nachdem der Cluster gestartet wurde. Siehe Konfigurieren des Beendigungsschutzes für aktive Cluster. Um einen Cluster mit mehreren Primärknoten herunterzufahren, müssen Sie zunächst die Clusterattribute ändern, um den Kündigungsschutz zu deaktivieren. Detaillierte Anweisungen finden Sie unter Einen EMR Amazon-Cluster mit mehreren Primärknoten beenden.

Weitere Informationen zum Beendigungsschutz finden Sie unter Verwenden des Beendigungsschutzes.

Nicht unterstützte Funktionen in einem EMR Amazon-Cluster mit mehreren Primärknoten

Die folgenden EMR Amazon-Funktionen sind derzeit in einem EMR Amazon-Cluster mit mehreren Primärknoten nicht verfügbar:

- EMRNotizbücher
- Zugriff auf den permanenten Spark History Server mit nur einem Klick
- Persistente Anwendungsbenutzeroberflächen

- Der Ein-Klick-Zugriff auf persistente Anwendungsbenutzeroberflächen ist derzeit für EMR Amazon-Cluster mit mehreren Primärknoten oder für EMR Amazon-Cluster, die in AWS Lake Formation integriert sind, nicht verfügbar.

> **Note**
>
> Um die Kerberos-Authentifizierung in Ihrem Cluster zu verwenden, müssen Sie ein externes System konfigurieren. KDC
> Ab EMR Amazon-Version 5.27.0 können Sie die HDFS transparente Verschlüsselung auf einem EMR Amazon-Cluster mit mehreren Primärknoten konfigurieren. Weitere Informationen finden Sie unter Transparente Verschlüsselung HDFS bei Amazon EMR.

Starten Sie einen EMR Amazon-Cluster mit mehreren Primärknoten

Dieses Thema enthält Konfigurationsdetails und Beispiele für den Start eines EMR Amazon-Clusters mit mehreren Primärknoten.

> **Note**
>
> Amazon aktiviert EMR automatisch den Kündigungsschutz für alle Cluster mit mehreren Primärknoten und überschreibt alle Einstellungen für die automatische Terminierung, die Sie bei der Erstellung des Clusters angeben. Um einen Cluster mit mehreren Primärknoten herunterzufahren, müssen Sie zunächst die Clusterattribute ändern, um den Kündigungsschutz zu deaktivieren. Detaillierte Anweisungen finden Sie unter Einen EMR Amazon-Cluster mit mehreren Primärknoten beenden.

Voraussetzungen

- Sie können einen EMR Amazon-Cluster mit mehreren Primärknoten in öffentlichen und privaten VPC Subnetzen starten. EC2-Classic wird nicht unterstützt. Um einen EMR Amazon-Cluster mit mehreren Primärknoten in einem öffentlichen Subnetz zu starten, müssen Sie den Instances in diesem Subnetz den Empfang einer öffentlichen IP-Adresse ermöglichen, indem Sie IPv4 in der Konsole Automatisch zuweisen auswählen oder den folgenden Befehl ausführen. Ersetzen *22XXXX01* mit Ihrer Subnetz-ID.

```
aws ec2 modify-subnet-attribute --subnet-id subnet-22XXXX01 --map-public-ip-on-launch
```

- Um Hive, Hue oder Oozie auf einem EMR Amazon-Cluster mit mehreren Primärknoten auszuführen, müssen Sie einen externen Metastore erstellen. Weitere Informationen finden Sie unter Konfiguration eines externen Metastores für Hive, Verwenden von Hue mit einer Remote-Datenbank in Amazon oder Apache RDS Oozie.
- Um die Kerberos-Authentifizierung in Ihrem Cluster zu verwenden, müssen Sie eine externe Authentifizierung konfigurieren. KDC Weitere Informationen finden Sie unter Konfiguration von Kerberos auf Amazon Amazon. EMR

Starten Sie einen EMR Amazon-Cluster mit mehreren Primärknoten

Sie können einen Cluster mit mehreren Primärknoten starten, wenn Sie Instance-Gruppen oder Instance-Flotten verwenden. Wenn Sie Instance-Gruppen mit mehreren Primärknoten verwenden, müssen Sie für die Primärknoten-Instance-Gruppe den Wert `3` für die Zahl der Instances angeben. Wenn Sie Instance-Flotten mit mehreren Primärknoten verwenden, müssen Sie die `TargetOnDemandCapacity` von `3`, die `TargetSpotCapacity` von `0` für die primäre Instance-Flotte und die `WeightedCapacity` von `1` für jeden Instance-Typ angeben, den Sie für die primäre Flotte konfigurieren.

Die folgenden Beispiele zeigen, wie der Cluster mit dem Standard AMI - oder einem benutzerdefinierten Cluster sowohl AMI mit Instance-Gruppen als auch mit Instance-Flotten gestartet wird:

> **Note**
>
> Sie müssen die Subnetz-ID angeben, wenn Sie einen EMR Amazon-Cluster mit mehreren Primärknoten mit dem AWS CLI starten. Ersetzen *22XXXX01* and *22XXXX02* mit Ihrer Subnetz-ID in den folgenden Beispielen.

Default AMI, instance groups

Example Beispiel — Starten eines EMR Amazon-Instance-Gruppen-Clusters mit mehreren Primärknoten unter Verwendung eines Standardknotens AMI

```
aws emr create-cluster \
```

```
--name "ha-cluster" \
--release-label emr-6.15.0 \
--instance-groups InstanceGroupType=MASTER,InstanceCount=3,InstanceType=m5.xlarge
 InstanceGroupType=CORE,InstanceCount=4,InstanceType=m5.xlarge \
--ec2-attributes
 KeyName=ec2_key_pair_name,InstanceProfile=EMR_EC2_DefaultRole,SubnetId=subnet-22XXXX01
 \
--service-role EMR_DefaultRole \
--applications Name=Hadoop Name=Spark
```

Default AMI, instance fleets

Example Beispiel — Starten eines EMR Amazon-Instance-Flotten-Clusters mit mehreren Primärknoten unter Verwendung eines Standardknotens AMI

```
aws emr create-cluster \
--name "ha-cluster" \
--release-label emr-6.15.0 \
--instance-fleets '[
    {
        "InstanceFleetType": "MASTER",
        "TargetOnDemandCapacity": 3,
        "TargetSpotCapacity": 0,
        "LaunchSpecifications": {
            "OnDemandSpecification": {
                "AllocationStrategy": "lowest-price"
            }
        },
        "InstanceTypeConfigs": [
            {
                "WeightedCapacity": 1,
                "BidPriceAsPercentageOfOnDemandPrice": 100,
                "InstanceType": "m5.xlarge"
            },
            {
                "WeightedCapacity": 1,
                "BidPriceAsPercentageOfOnDemandPrice": 100,
                "InstanceType": "m5.2xlarge"
            },
            {
                "WeightedCapacity": 1,
                "BidPriceAsPercentageOfOnDemandPrice": 100,
                "InstanceType": "m5.4xlarge"
            }
```

```
        ],
        "Name": "Master - 1"
    },
    {
        "InstanceFleetType": "CORE",
        "TargetOnDemandCapacity": 5,
        "TargetSpotCapacity": 0,
        "LaunchSpecifications": {
            "OnDemandSpecification": {
                "AllocationStrategy": "lowest-price"
            }
        },
        "InstanceTypeConfigs": [
            {
                "WeightedCapacity": 1,
                "BidPriceAsPercentageOfOnDemandPrice": 100,
                "InstanceType": "m5.xlarge"
            },
            {
                "WeightedCapacity": 2,
                "BidPriceAsPercentageOfOnDemandPrice": 100,
                "InstanceType": "m5.2xlarge"
            },
            {
                "WeightedCapacity": 4,
                "BidPriceAsPercentageOfOnDemandPrice": 100,
                "InstanceType": "m5.4xlarge"
            }
        ],
        "Name": "Core - 2"
    }
]' \
 --ec2-attributes '{"InstanceProfile":"EMR_EC2_DefaultRole","SubnetIds":
["subnet-22XXXX01", "subnet-22XXXX02"]}' \
--service-role EMR_DefaultRole \
--applications Name=Hadoop Name=Spark
```

Custom AMI, instance groups

Example Beispiel — Starten eines EMR Amazon-Instance-Gruppen-Clusters mit mehreren Primärknoten mithilfe eines benutzerdefinierten AMI

```
aws emr create-cluster \
--name "custom-ami-ha-cluster" \
```

```
--release-label emr-6.15.0 \
--instance-groups InstanceGroupType=MASTER,InstanceCount=3,InstanceType=m5.xlarge
 InstanceGroupType=CORE,InstanceCount=4,InstanceType=m5.xlarge \
--ec2-attributes
 KeyName=ec2_key_pair_name,InstanceProfile=EMR_EC2_DefaultRole,SubnetId=subnet-22XXXX01
 \
--service-role EMR_DefaultRole \
--applications Name=Hadoop Name=Spark \
--custom-ami-id ami-MyAmiID
```

Custom AMI, instance fleets

Example Beispiel — Starten eines EMR Amazon-Instance-Flottenclusters mit mehreren Primärknoten mithilfe eines benutzerdefinierten AMI

```
aws emr create-cluster \
--name "ha-cluster" \
--release-label emr-6.15.0 \
--instance-fleets '[
    {
        "InstanceFleetType": "MASTER",
        "TargetOnDemandCapacity": 3,
        "TargetSpotCapacity": 0,
        "LaunchSpecifications": {
            "OnDemandSpecification": {
                "AllocationStrategy": "lowest-price"
            }
        },
        "InstanceTypeConfigs": [
            {
                "WeightedCapacity": 1,
                "BidPriceAsPercentageOfOnDemandPrice": 100,
                "InstanceType": "m5.xlarge"
            },
            {
                "WeightedCapacity": 1,
                "BidPriceAsPercentageOfOnDemandPrice": 100,
                "InstanceType": "m5.2xlarge"
            },
            {
                "WeightedCapacity": 1,
                "BidPriceAsPercentageOfOnDemandPrice": 100,
                "InstanceType": "m5.4xlarge"
            }
```

```
        ],
        "Name": "Master - 1"
    },
    {
        "InstanceFleetType": "CORE",
        "TargetOnDemandCapacity": 5,
        "TargetSpotCapacity": 0,
        "LaunchSpecifications": {
            "OnDemandSpecification": {
                "AllocationStrategy": "lowest-price"
            }
        },
        "InstanceTypeConfigs": [
            {
                "WeightedCapacity": 1,
                "BidPriceAsPercentageOfOnDemandPrice": 100,
                "InstanceType": "m5.xlarge"
            },
            {
                "WeightedCapacity": 2,
                "BidPriceAsPercentageOfOnDemandPrice": 100,
                "InstanceType": "m5.2xlarge"
            },
            {
                "WeightedCapacity": 4,
                "BidPriceAsPercentageOfOnDemandPrice": 100,
                "InstanceType": "m5.4xlarge"
            }
        ],
        "Name": "Core - 2"
    }
]' \
--ec2-attributes '{"InstanceProfile":"EMR_EC2_DefaultRole","SubnetIds":
["subnet-22XXXX01", "subnet-22XXXX02"]}' \
--service-role EMR_DefaultRole \
--applications Name=Hadoop Name=Spark \
--custom-ami-id ami-MyAmiID
```

Einen EMR Amazon-Cluster mit mehreren Primärknoten beenden

Um einen EMR Amazon-Cluster mit mehreren Primärknoten zu beenden, müssen Sie den Kündigungsschutz deaktivieren, bevor Sie den Cluster beenden, wie das folgende Beispiel zeigt. Ersetzen *j-3KVTXXXXXX7UG* mit Ihrer Cluster-ID.

```
aws emr modify-cluster-attributes --cluster-id j-3KVTXXXXXX7UG --no-termination-
protected
aws emr terminate-clusters --cluster-id j-3KVTXXXXXX7UG
```

EMRAmazon-Integration mit EC2 Platzierungsgruppen

Wenn Sie einen Amazon-Cluster mit EMR mehreren primären Knoten auf Amazon startenEC2, haben Sie die Möglichkeit, Platzierungsgruppenstrategien zu verwenden, um festzulegen, wie die Primärknoten-Instances zum Schutz vor Hardwareausfällen bereitgestellt werden sollen.

Platzierungsgruppenstrategien werden ab EMR Amazon-Version 5.23.0 als Option für Cluster mit mehreren primären Knoten unterstützt. Derzeit werden nur Primärkotentypen von der Platzierungsgruppenstrategie unterstützt, und die `SPREAD`-Strategie wird auf diese Primärknoten angewendet. Bei dieser `SPREAD`-Strategie wird eine kleine Gruppe von Instances auf separater zugrundeliegender Hardware platziert, um den Verlust mehrerer Primärknoten im Falle eines Hardwarefehlers zu verhindern. Beachten Sie, dass eine Anforderung zum Starten einer Instance fehlschlagen kann, wenn es nicht genügend eindeutige Hardware zur Erfüllung der Anforderung gibt. Weitere Informationen zu EC2 Platzierungsstrategien und Einschränkungen finden Sie unter Platzierungsgruppen im EC2Benutzerhandbuch für Linux-Instances.

Amazon gibt ein anfängliches Limit EC2 von 500 Clustern mit aktivierter Platzierungsgruppenstrategie, die pro AWS Region gestartet werden können. Wenden Sie sich an den AWS Support, um eine Erhöhung der Anzahl der zulässigen Platzierungsgruppen zu beantragen. Sie können EC2 Platzierungsgruppen identifizieren, die Amazon EMR erstellt, indem Sie das Schlüssel-Wert-Paar verfolgen, das Amazon mit der EMR Amazon-Platzierungsgruppenstrategie EMR verknüpft. Weitere Informationen zu EC2 Cluster-Instance-Tags finden Sie unter. Cluster-Instances in Amazon anzeigen EC2

Die von der Platzierungsgruppe verwaltete Richtlinie an Amazon anhängen EMRrole

Die Platzierungsgruppenstrategie erfordert eine verwaltete Richtlinie namens`AmazonElasticMapReducePlacementGroupPolicy`, die es Amazon ermöglicht, Platzierungsgruppen auf Amazon EMR zu erstellen, zu löschen und zu beschreibenEC2. Sie

müssen `AmazonElasticMapReducePlacementGroupPolicy` sich der Service-Rolle für Amazon zuordnen, EMR bevor Sie einen EMR Amazon-Cluster mit mehreren Primärknoten starten.

Sie können die `AmazonEMRServicePolicy_v2` verwaltete Richtlinie alternativ der EMR Amazon-Servicerolle anstelle der verwalteten Richtlinie für die Platzierungsgruppe zuordnen. `AmazonEMRServicePolicy_v2`ermöglicht den gleichen Zugriff auf Platzierungsgruppen bei Amazon EC2 wie der`AmazonElasticMapReducePlacementGroupPolicy`. Weitere Informationen finden Sie unter Servicerolle für Amazon EMR (EMRRolle).

Bei der `AmazonElasticMapReducePlacementGroupPolicy` verwalteten Richtlinie handelt es sich um den folgenden JSON Text, der von Amazon erstellt und verwaltet wirdEMR.

> **Note**
>
> Da die `AmazonElasticMapReducePlacementGroupPolicy` verwaltete Richtlinie automatisch aktualisiert wird, kann es sein, dass die hier gezeigte Richtlinie aktualisiert wird out-of-date. Verwenden Sie die AWS Management Console, um die aktuelle Richtlinie einzusehen.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Resource":"*",
         "Effect":"Allow",
         "Action":[
            "ec2:DeletePlacementGroup",
            "ec2:DescribePlacementGroups"
         ]
      },
      {
         "Resource":"arn:aws:ec2:*:*:placement-group/pg-*",
         "Effect":"Allow",
         "Action":[
            "ec2:CreatePlacementGroup"
         ]
      }
   ]
}
```

Starten Sie einen EMR Amazon-Cluster mit mehreren Primärknoten mithilfe der Platzierungsgruppenstrategie

Um einen EMR Amazon-Cluster mit mehreren primären Knoten mit einer Platzierungsgruppenstrategie zu starten, fügen Sie die von der Platzierungsgruppe verwaltete Richtlinie der EMR Amazon-Rolle `AmazonElasticMapReducePlacementGroupPolicy` hinzu. Weitere Informationen finden Sie unter Die von der Platzierungsgruppe verwaltete Richtlinie an Amazon anhängen EMRrole.

Jedes Mal, wenn Sie diese Rolle verwenden, um einen EMR Amazon-Cluster mit mehreren primären Knoten zu starten, EMR versucht Amazon, einen Cluster mit einer `SPREAD` Strategie zu starten, die auf seine primären Knoten angewendet wird. Wenn Sie eine Rolle verwenden, der die Richtlinie zur Verwaltung der Platzierungsgruppe nicht `AmazonElasticMapReducePlacementGroupPolicy` zugeordnet ist, EMR versucht Amazon, einen EMR Amazon-Cluster mit mehreren primären Knoten ohne Platzierungsgruppenstrategie zu starten.

Wenn Sie einen EMR Amazon-Cluster mit mehreren Primärknoten starten, wobei der `placement-group-configs` Parameter Amazon EMRAPI oder verwendetCLI, startet Amazon den Cluster EMR nur, wenn Amazon die Platzierungsgruppen-verwaltete Richtlinie `AmazonElasticMapReducePlacementGroupPolicy` angehängt EMRrole hat. Wenn Amazon die Richtlinie EMRrole nicht angehängt hat, schlägt der Start des EMR Amazon-Clusters mit mehreren Primärknoten fehl.

Amazon EMR API

Example Beispiel — Verwenden Sie eine Platzierungsgruppenstrategie, um einen Instance-Gruppen-Cluster mit mehreren Primärknoten von Amazon aus zu starten EMR API

Wenn Sie die RunJobFlow Aktion verwenden, um einen EMR Amazon-Cluster mit mehreren Primärknoten zu erstellen, legen Sie die `PlacementGroupConfigs` Eigenschaft wie folgt fest. Derzeit wird die `MASTER`-Instance-Rolle automatisch `SPREAD` als Platzierungsgruppenstrategie verwendet.

```
{
   "Name":"ha-cluster",
   "PlacementGroupConfigs":[
      {
         "InstanceRole":"MASTER"
      }
   ],
```

```
   "ReleaseLabel": emr-6.15.0,
   "Instances":{
      "ec2SubnetId":"subnet-22XXXX01",
      "ec2KeyName":"ec2_key_pair_name",
      "InstanceGroups":[
         {
            "InstanceCount":3,
            "InstanceRole":"MASTER",
            "InstanceType":"m5.xlarge"
         },
         {
            "InstanceCount":4,
            "InstanceRole":"CORE",
            "InstanceType":"m5.xlarge"
         }
      ]
   },
   "JobFlowRole":"EMR_EC2_DefaultRole",
   "ServiceRole":"EMR_DefaultRole"
}
```

- Ersetzen *`ha-cluster`* mit dem Namen Ihres Hochverfügbarkeitsclusters.
- Ersetzen *`subnet-22XXXX01`* mit Ihrer Subnetz-ID.
- Ersetzen Sie die *`ec2_key_pair_name`* mit dem Namen Ihres EC2 key pair für diesen Cluster. EC2Das key pair ist optional und nur erforderlich, wenn Sie es für den SSH Zugriff auf Ihren Cluster verwenden möchten.

AWS CLI

Example Beispiel – Verwenden Sie eine Platzierungsgruppenstrategie, um einen Instance-Flotten-Cluster mit mehreren Primärknoten über die AWS Command Line Interface zu starten

Wenn Sie die RunJobFlow Aktion verwenden, um einen EMR Amazon-Cluster mit mehreren Primärknoten zu erstellen, legen Sie die `PlacementGroupConfigs` Eigenschaft wie folgt fest. Derzeit wird die `MASTER`-Instance-Rolle automatisch `SPREAD` als Platzierungsgruppenstrategie verwendet.

```
aws emr create-cluster \
--name "ha-cluster" \
--placement-group-configs InstanceRole=MASTER \
--release-label emr-6.15.0 \
```

```
--instance-fleets '[
    {
        "InstanceFleetType": "MASTER",
        "TargetOnDemandCapacity": 3,
        "TargetSpotCapacity": 0,
        "LaunchSpecifications": {
            "OnDemandSpecification": {
                "AllocationStrategy": "lowest-price"
            }
        },
        "InstanceTypeConfigs": [
            {
                "WeightedCapacity": 1,
                "BidPriceAsPercentageOfOnDemandPrice": 100,
                "InstanceType": "m5.xlarge"
            },
            {
                "WeightedCapacity": 1,
                "BidPriceAsPercentageOfOnDemandPrice": 100,
                "InstanceType": "m5.2xlarge"
            },
            {
                "WeightedCapacity": 1,
                "BidPriceAsPercentageOfOnDemandPrice": 100,
                "InstanceType": "m5.4xlarge"
            }
        ],
        "Name": "Master - 1"
    },
    {
        "InstanceFleetType": "CORE",
        "TargetOnDemandCapacity": 5,
        "TargetSpotCapacity": 0,
        "LaunchSpecifications": {
            "OnDemandSpecification": {
                "AllocationStrategy": "lowest-price"
            }
        },
        "InstanceTypeConfigs": [
            {
                "WeightedCapacity": 1,
                "BidPriceAsPercentageOfOnDemandPrice": 100,
                "InstanceType": "m5.xlarge"
            },
```

```
            {
                "WeightedCapacity": 2,
                "BidPriceAsPercentageOfOnDemandPrice": 100,
                "InstanceType": "m5.2xlarge"
            },
            {
                "WeightedCapacity": 4,
                "BidPriceAsPercentageOfOnDemandPrice": 100,
                "InstanceType": "m5.4xlarge"
            }
        ],
        "Name": "Core - 2"
    }
]' \
--ec2-attributes '{
    "KeyName": "ec2_key_pair_name",
    "InstanceProfile": "EMR_EC2_DefaultRole",
    "SubnetIds": [
        "subnet-22XXXX01",
        "subnet-22XXXX02"
    ]
}' \
--service-role EMR_DefaultRole \
--applications Name=Hadoop Name=Spark
```

- Ersetzen *`ha-cluster`* mit dem Namen Ihres Hochverfügbarkeitsclusters.
- Ersetzen Sie das *`ec2_key_pair_name`* mit dem Namen Ihres EC2 key pair für diesen Cluster. EC2Das key pair ist optional und nur erforderlich, wenn Sie es für den SSH Zugriff auf Ihren Cluster verwenden möchten.
- Ersetzen *`subnet-22XXXX01`* and *`subnet-22XXXX02`*mit Ihrem SubnetzIDs.

Starten Sie einen Cluster mit mehreren Primärknoten ohne eine Platzierungsgruppenstrategie

Damit ein Cluster mit mehreren Primärknoten ohne die Platzierungsgruppenstrategie starten kann, müssen Sie einen der folgenden Schritte ausführen:

- Entfernen Sie die von der Platzierungsgruppe verwaltete Richtlinie `AmazonElasticMapReducePlacementGroupPolicy` aus AmazonEMRrole, oder

- Starten Sie einen Cluster mit mehreren Primärknoten mit dem `placement-group-configs` Parameter, indem Sie Amazon verwenden EMRAPI oder `NONE` als Platzierungsgruppenstrategie CLI wählen.

Amazon EMR API

Example — Starten eines Clusters mit mehreren Primärknoten ohne Platzierungsgruppenstrategie mithilfe von AmazonEMRAPI.

Wenn Sie die RunJobFlow Aktion verwenden, um einen Cluster mit mehreren Primärknoten zu erstellen, legen Sie die `PlacementGroupConfigs` Eigenschaft wie folgt fest.

```
{
   "Name":"ha-cluster",
   "PlacementGroupConfigs":[
      {
         "InstanceRole":"MASTER",
         "PlacementStrategy":"NONE"
      }
   ],
   "ReleaseLabel":"emr-5.30.1",
   "Instances":{
      "ec2SubnetId":"subnet-22XXXX01",
      "ec2KeyName":"ec2_key_pair_name",
      "InstanceGroups":[
         {
            "InstanceCount":3,
            "InstanceRole":"MASTER",
            "InstanceType":"m5.xlarge"
         },
         {
            "InstanceCount":4,
            "InstanceRole":"CORE",
            "InstanceType":"m5.xlarge"
         }
      ]
   },
   "JobFlowRole":"EMR_EC2_DefaultRole",
   "ServiceRole":"EMR_DefaultRole"
}
```

- Ersetzen *`ha-cluster`* mit dem Namen Ihres Hochverfügbarkeitsclusters.

- Ersetzen *subnet-22XXXX01* mit Ihrer Subnetz-ID.
- Ersetzen Sie die *ec2_key_pair_name* mit dem Namen Ihres EC2 key pair für diesen Cluster. EC2Das key pair ist optional und nur erforderlich, wenn Sie es für den SSH Zugriff auf Ihren Cluster verwenden möchten.

Amazon EMR CLI

Example — Starten eines Clusters mit mehreren Primärknoten ohne Platzierungsgruppenstrategie mithilfe von AmazonEMRCLI.

Wenn Sie die RunJobFlow Aktion verwenden, um einen Cluster mit mehreren Primärknoten zu erstellen, legen Sie die `PlacementGroupConfigs` Eigenschaft wie folgt fest.

```
aws emr create-cluster \
--name "ha-cluster" \
--placement-group-configs InstanceRole=MASTER,PlacementStrategy=NONE \
--release-label emr-5.30.1 \
--instance-groups InstanceGroupType=MASTER,InstanceCount=3,InstanceType=m5.xlarge
 InstanceGroupType=CORE,InstanceCount=4,InstanceType=m5.xlarge \
--ec2-attributes
 KeyName=ec2_key_pair_name,InstanceProfile=EMR_EC2_DefaultRole,SubnetId=subnet-22XXXX01
 \
--service-role EMR_DefaultRole \
--applications Name=Hadoop Name=Spark
```

- Ersetzen *ha-cluster* mit dem Namen Ihres Hochverfügbarkeitsclusters.
- Ersetzen *subnet-22XXXX01* mit Ihrer Subnetz-ID.
- Ersetzen Sie die *ec2_key_pair_name* mit dem Namen Ihres EC2 key pair für diesen Cluster. EC2Das key pair ist optional und nur erforderlich, wenn Sie es für den SSH Zugriff auf Ihren Cluster verwenden möchten.

Überprüfen Sie die Konfiguration der Platzierungsgruppenstrategie, die an den Cluster mit mehreren Primärknoten angehängt ist

Sie können den Amazon EMR Describe-Cluster verwendenAPI, um die Konfiguration der Platzierungsgruppenstrategie zu sehen, die dem Cluster mit mehreren primären Knoten zugeordnet ist.

Example

```
aws emr describe-cluster --cluster-id "j-xxxxx"
{
   "Cluster":{
      "Id":"j-xxxxx",
      ...
      ...
      "PlacementGroups":[
         {
            "InstanceRole":"MASTER",
            "PlacementStrategy":"SPREAD"
         }
      ]
   }
}
```

Überlegungen und bewährte Methoden

Beachten Sie Folgendes, wenn Sie einen EMR Amazon-Cluster mit mehreren Primärknoten erstellen:

> **Important**
>
> Um EMR Hochverfügbarkeitscluster mit mehreren Primärknoten zu starten, empfehlen wir dringend, die neueste EMR Amazon-Version zu verwenden. Dadurch wird sichergestellt, dass Sie ein Höchstmaß an Resilienz und Stabilität für Ihre Hochverfügbarkeits-Cluster erhalten.

- Hochverfügbarkeit für Instance-Flotten wird mit den EMR Amazon-Versionen 5.36.1, 5.36.2, 6.8.1, 6.9.1, 6.10.1, 6.11.1, 6.12.0 und höher unterstützt. Für Instance-Gruppen wird Hochverfügbarkeit mit EMR Amazon-Versionen 5.23.0 und höher unterstützt. Weitere Informationen finden Sie unter Über Amazon EMR Releases.
- Auf Clustern mit hoher Verfügbarkeit unterstützt Amazon EMR nur den Start von Primärknoten mit On-Demand-Instances. Dadurch wird die höchste Verfügbarkeit für Ihren Cluster gewährleistet.
- Sie können immer noch mehrere Instance-Typen für die primäre Flotte angeben, aber alle Primärknoten von Hochverfügbarkeits-Clustern werden mit demselben Instance-Typ gestartet, einschließlich Ersatz-Instances für fehlerhafte Primärknoten.

- Um den Betrieb fortzusetzen, müssen bei einem Hochverfügbarkeits-Cluster mit mehreren Primärknoten zwei von drei Primärknoten fehlerfrei sein. Wenn also zwei Primärknoten gleichzeitig ausfallen, fällt Ihr EMR Cluster aus.
- Alle EMR Cluster, einschließlich Hochverfügbarkeitscluster, werden in einer einzigen Availability Zone gestartet. Daher können sie Ausfälle in der Availability Zone nicht tolerieren. Beim Ausfall einer Availability Zone verlieren Sie den Zugriff auf den Cluster.
- Wenn Sie beim Starten eines Clusters innerhalb einer Instance-Flotte eine benutzerdefinierte Servicerolle oder -richtlinie verwenden, können Sie die `ec2:DescribeInstanceTypeOfferings` Berechtigung hinzufügen, damit Amazon nicht unterstützte Availability Zones (AZ) herausfiltern EMR kann. Wenn Amazon diejenigen EMR herausfiltertAZs, die keine Instance-Typen von Primärknoten unterstützen, EMR verhindert Amazon, dass Cluster-Starts aufgrund nicht unterstützter primärer Instance-Typen fehlschlagen. Weitere Informationen finden Sie unter Instance-Typ wird nicht unterstützt.
- Amazon garantiert EMR keine Hochverfügbarkeit für andere Open-Source-Anwendungen als die, die in Unterstützte Anwendungen in einem EMR Amazon-Cluster mit mehreren Primärknoten spezifiziert sind.
- In den EMR Amazon-Versionen 5.23.0 bis 5.36.2 werden nur zwei der drei primären Knoten für einen Instance-Gruppen-Cluster ausgeführt. HDFS NameNode
- In EMR Amazon-Versionen 6.x und höher werden alle drei primären Knoten für eine Instance-Gruppe ausgeführtHDFS NameNode.

Überlegungen für das Konfigurieren von Subnetzen:

- Ein EMR Amazon-Cluster mit mehreren Primärknoten kann sich nur in einer Availability Zone oder einem Subnetz befinden. Amazon EMR kann einen ausgefallenen Primärknoten nicht ersetzen, wenn das Subnetz vollständig ausgelastet oder im Falle eines Failovers überbelegt ist. Um dieses Szenario zu vermeiden, wird empfohlen, einem EMR Amazon-Cluster ein ganzes Subnetz zuzuweisen. Darüber hinaus sollten Sie sicherstellen, dass im Subnetz eine ausreichende Zahl von privaten IP-Adressen verfügbar ist.

Überlegungen für das Konfigurieren von Core-Knoten:

- Um sicherzustellen, dass die Core-Knoten ebenfalls hoch verfügbar sind, sollten Sie mindestens vier Core-Knoten starten. Wenn Sie sich dafür entscheiden, einen kleineren Cluster mit drei oder weniger Kernknoten zu starten, sollten Sie mindestens vier auswählen, `dfs.replication`

`parameter` um eine ausreichende 2 HDFS DFS Replikation zu gewährleisten. Weitere Informationen finden Sie unter HDFSKonfiguration.

> ⚠ Warning
>
> 1. Die Einstellung `dfs.replication` 1 in Clustern mit weniger als vier Knoten kann zu HDFS Datenverlust führen, wenn ein einzelner Knoten ausfällt. Wir empfehlen, für Produktionsworkloads einen Cluster mit mindestens vier Core-Knoten zu verwenden.
> 2. Amazon EMR erlaubt Clustern nicht, Kernknoten nach unten zu skalierend`fs.replication`. Bei `dfs.replication = 2` z. B. beträgt die Mindestanzahl von Core-Knoten 2.
> 3. Wenn Sie verwaltete Skalierung oder Auto-Scaling verwenden oder die Größe Ihres Clusters manuell ändern möchten, empfehlen wir Ihnen, `dfs.replication` auf 2 oder höher einzustellen.

Überlegungen zum Einrichten von Alarmen für Metriken:

- Amazon stellt EMR keine anwendungsspezifischen Metriken zu HDFS oder bereit. YARN Sie sollten Alarme einrichten, um die Instance-Zahl der Primärknoten zu überwachen. Konfigurieren Sie die Alarme anhand der folgenden CloudWatch Amazon-Metriken: `MultiMasterInstanceGroupNodesRunning`MultiMasterInstanceGroupNodesRunningPercent oder`MultiMasterInstanceGroupNodesRequested`. CloudWatch benachrichtigt Sie, falls der Primärknoten ausfällt oder ausgetauscht wird.
  - Wenn `MultiMasterInstanceGroupNodesRunningPercentage` kleiner als 1,0 und größer als 0,5 ist, ist im Cluster möglicherweise ein Primärknoten ausgefallen. In dieser Situation EMR versucht Amazon, einen Primärknoten zu ersetzen.
  - Wenn `MultiMasterInstanceGroupNodesRunningPercentage` kleiner als 0,5 ist, sind im Cluster möglicherweise zwei Primärknoten ausgefallen. In diesem Fall ist das Quorum verloren und der Cluster kann nicht wiederhergestellt werden. Sie müssen Daten manuell aus diesem Cluster migrieren.

  Weitere Informationen finden Sie unter Einrichten von Alarmen für Metriken.

EMRCluster auf AWS Outposts

Ab Amazon EMR 5.28.0 können Sie EMR Cluster erstellen und ausführen. AWS Outposts AWS Outposts ermöglicht native AWS Dienste, Infrastrukturen und Betriebsmodelle in lokalen Einrichtungen. In AWS Outposts Umgebungen können Sie dieselben AWS APIs Tools und dieselbe Infrastruktur verwenden wie in der AWS Cloud. Amazon EMR on AWS Outposts ist ideal für Workloads mit niedriger Latenz, die in unmittelbarer Nähe zu lokalen Daten und Anwendungen ausgeführt werden müssen. Weitere Informationen zu AWS Outposts finden Sie im AWS Outposts Benutzerhandbuch.

Voraussetzungen

Im Folgenden sind die Voraussetzungen für die Nutzung von Amazon EMR auf AWS Outposts:

- Sie müssen AWS Outposts in Ihrem lokalen Rechenzentrum installiert und konfiguriert haben.
- Sie müssen über eine zuverlässige Netzwerkverbindung zwischen Ihrer Outpost-Umgebung und einer AWS Region verfügen.
- Sie müssen über ausreichende Kapazität für von Amazon EMR unterstützte Instance-Typen in Ihrem Outpost verfügen.

Einschränkungen

Im Folgenden sind die Einschränkungen bei der Nutzung von EMR Amazon aufgeführt AWS Outposts:

- On-Demand-Instances sind die einzige unterstützte Option für EC2 Amazon-Instances. Spot-Instances sind für Amazon EMR am nicht verfügbar AWS Outposts.
- Wenn Sie zusätzliche EBS Amazon-Speichervolumen benötigen, wird nur General Purpose SSD (GP2) unterstützt.
- Wenn Sie die EMR Amazon-Versionen 5.28 bis 6.x verwenden AWS Outposts , können Sie nur S3-Buckets verwenden, die Objekte in einem AWS-Region von Ihnen angegebenen speichern. Mit Amazon EMR 7.0.0 und höher AWS Outposts wird Amazon EMR on auch mit dem S3A Dateisystem-Client, Präfix, unterstützt. `s3a://`
- Nur die folgenden Instance-Typen werden von Amazon EMR am unterstützt AWS Outposts:

Instance-Klasse	Instance-Typen
Allgemeine Zwecke	`m5.xlarge | m5.2xlarge | m5.4xlarge | m5.12xlarge | m5.24xlarge | m5d.xlarge | m5d.2xlarge | m5d.4xlarge | m5d.12xlarge | m5d.24xlarge`
Für Datenverarbeitung optimiert	`c5.xlarge | c5.2xlarge | c5.4xlarge | c5.18xlarge | c5d.xlarge | c5d.2xlarge | c5d.4xlarge | c5d.18xlarge`
RAM-optimiert	`r5.xlarge | r5.2xlarge | r5.4xlarge | r5.12xlarge | r5d.xlarge | r5d.2xlarge | r5d.4xlarge | r5d.12xlarge | r5d.24xlarge`
Speicheroptimiert	`i3en.xlarge | i3en.2xlarge | i3en.3xlarge | i3en.6xlarge | i3en.12xlarge | i3en.24xlarge`

Überlegungen zur Netzwerkkonnektivität

- Wenn die Netzwerkverbindung zwischen Ihrem Outpost und seiner AWS Region unterbrochen wird, laufen Ihre Cluster weiter. Sie können jedoch keine neuen Cluster erstellen oder neue Aktionen für vorhandene Cluster ausführen, bis die Verbindung wiederhergestellt wurde. Bei Instance-Fehlern wird die Instance nicht automatisch ersetzt. Darüber hinaus werden Aktionen wie das Hinzufügen von Schritten zu einem laufenden Cluster, das Überprüfen des Ausführungsstatus der Schritte und das Senden von CloudWatch Metriken und Ereignissen verzögert.
- Wir empfehlen Ihnen, eine zuverlässige und hochverfügbare Netzwerkkonnektivität zwischen Ihrem Outpost und der AWS Region bereitzustellen. Wenn die Netzwerkverbindung zwischen Ihrem Outpost und seiner AWS Region für mehr als ein paar Stunden unterbrochen wird, laufen Cluster, für die der Terminierungsschutz aktiviert ist, weiter, und Cluster, die den Terminierungsschutz deaktiviert haben, können beendet werden.
- Falls die Netzwerkkonnektivität aufgrund einer routinemäßigen Wartung beeinträchtigt wird, empfehlen wir die proaktive Aktivierung des Beendigungsschutzes. Generell bedeutet die Unterbrechung der Konnektivität, dass externe Abhängigkeiten, die nicht lokal im Outpost oder

Kundennetzwerk sind, nicht zugänglich sind. Dazu gehören Amazon S3, DynamoDB, die mit EMRFS Consistency View verwendet werden, und Amazon, RDS wenn eine Instance in der Region für einen EMR Amazon-Cluster mit mehreren Primärknoten verwendet wird.

Erstellen eines EMR Amazon-Clusters auf AWS Outposts

Das Erstellen eines EMR Amazon-Clusters auf AWS Outposts ähnelt dem Erstellen eines EMR Amazon-Clusters in der AWS Cloud. Wenn Sie einen EMR Amazon-Cluster auf erstellen AWS Outposts, müssen Sie ein EC2 Amazon-Subnetz angeben, das Ihrem Outpost zugeordnet ist.

Ein Amazon VPC kann sich über alle Availability Zones in einer AWS Region erstrecken. AWS Outposts sind Erweiterungen von Availability Zones, und Sie können ein VPC Amazon-Konto so erweitern, dass es sich über mehrere Availability Zones und zugehörige Outpost-Standorte erstreckt. Wenn Sie Ihren Outpost konfigurieren, ordnen Sie ihm ein Subnetz zu, um Ihre regionale VPC Umgebung auf Ihre lokale Einrichtung auszudehnen. Outpost-Instances und zugehörige Dienste werden als Teil Ihrer Region angezeigtVPC, ähnlich einer Availability Zone mit zugehörigen Subnetzen. Weitere Informationen finden Sie im AWS Outposts -Benutzerhandbuch.

Konsole

Um einen neuen EMR Amazon-Cluster AWS Outposts mit dem zu erstellen AWS Management Console, geben Sie ein EC2 Amazon-Subnetz an, das mit Ihrem Outpost verknüpft ist.

Console

Um einen Cluster AWS Outposts mit der Konsole zu erstellen

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und anschließend Create cluster aus.
3. Wählen Sie unter Clusterkonfiguration die Option Instance-Gruppen oder Instance-Flotten aus. Wählen Sie dann im Dropdownmenü Instanztyp auswählen einen EC2 Instanztyp aus oder wählen Sie Aktionen und dann EBSVolumes hinzufügen aus. Amazon EMR on AWS Outposts unterstützt begrenzte EBS Amazon-Volumen- und Instance-Typen.
4. Wählen Sie unter Netzwerk ein EC2 Subnetz mit einer Outpost-ID in diesem Format aus: op-123456789.
5. Wählen Sie alle anderen Optionen aus, die für Ihren Cluster gelten.

6. Um Ihren Cluster jetzt zu starten, wählen Sie Cluster erstellen aus.

CLI

Um einen Cluster mit dem zu erstellen AWS OutpostsAWS CLI

- Um einen neuen EMR Amazon-Cluster AWS Outposts mit dem zu erstellen AWS CLI, geben Sie ein EC2 Subnetz an, das Ihrem Outpost zugeordnet ist, wie im folgenden Beispiel. Ersetzen *subnet-22XXXX01* mit Ihrer eigenen EC2 Amazon-Subnetz-ID.

```
aws emr create-cluster \
--name "Outpost cluster" \
--release-label emr-7.2.0 \
--applications Name=Spark \
--ec2-attributes KeyName=myKey SubnetId=subnet-22XXXX01 \
--instance-type m5.xlarge --instance-count 3 --use-default-roles
```

EMRCluster in AWS Local Zones

Ab EMR Amazon-Version 5.28.0 können Sie EMR Amazon-Cluster in einem Local Zones-Subnetz als logische Erweiterung einer AWS Region, die AWS Local Zones unterstützt, erstellen und ausführen. Eine lokale Zone ermöglicht es, EMR Amazon-Funktionen und eine Untergruppe von AWS Diensten, wie Rechen- und Speicherdienste, näher an den Benutzern zu platzieren, um einen Zugriff mit sehr geringer Latenz auf lokal ausgeführte Anwendungen zu ermöglichen. Eine Liste der verfügbaren Local Zones finden Sie unter AWS Local Zones. Informationen zum Zugriff auf verfügbare AWS Local Zones finden Sie unter Regionen, Availability Zones und lokale Zonen.

Unterstützte Instance-Typen

Die folgenden Instance-Typen sind für EMR Amazon-Cluster in Local Zones verfügbar. Die Verfügbarkeit des Instance-Typs kann je nach Region variieren.

Instance-Klasse	Instance-Typen
Allgemeine Zwecke	m5.xlarge | m5.2xlarge | m5.4xlarge | m5.12xlarge | m5.24xlarge | m5d.xlarge | m5d.2xlarge | m5d.4xlarge | m5d.12xlarge | m5d.24xlarge

Instance-Klasse	Instance-Typen
Für Datenverarbeitung optimiert	c5.xlarge | c5.2xlarge | c5.4xlarge | c5.9xlarge | c5.18xlarge | c5d.xlarge | c5d.2xlarge | c5d.4xlarge| c5d.9xlarge | c5d.18xlarge
RAM-optimiert	r5.xlarge | r5.2xlarge | r5.4xlarge | r5.12xlarge | r5d.xlarge | r5d.2xlarge | r5d.4xlarge | r5d.12xlarge | r5d.24xlarge
Speicheroptimiert	i3en.xlarge | i3en.2xlarge | i3en.3xlarge | i3en.6xlarge | i3en.12xlarge | i3en.24xlarge

Erstellen eines EMR Amazon-Clusters in Local Zones

Erstellen Sie einen EMR Amazon-Cluster in AWS Local Zones, indem Sie den EMR Amazon-Cluster in einem VPC Amazon-Subnetz starten, das mit einer lokalen Zone verknüpft ist. Sie können auf den Cluster mit dem Local-Zone-Namen zugreifen, z. B. us-west-2-lax-1a in der USA West (Oregon)-Konsole.

Local Zones unterstützen derzeit keine EMR Amazon-Notebooks oder direkte Verbindungen zu Amazon EMR über den VPC Schnittstellenendpunkt (AWS PrivateLink).

Console

Um mit der Konsole einen Cluster in einer lokalen Zone zu erstellen

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und anschließend Create cluster aus.
3. Wählen Sie unter Netzwerk ein EC2 Subnetz mit einer lokalen Zonen-ID in diesem Format aus: subnet 123abc | us-west-2-lax-1a.
4. Wählen Sie einen Instance-Typ oder fügen Sie EBS Amazon-Speicher-Volumes für einheitliche Instance-Gruppen oder Instance-Flotten hinzu.
5. Wählen Sie alle anderen Optionen aus, die für Ihren Cluster gelten.
6. Um Ihren Cluster jetzt zu starten, wählen Sie Cluster erstellen aus.

CLI

Um einen Cluster in einer lokalen Zone mit dem zu erstellen AWS CLI

- Verwenden Sie den Befehl create-cluster zusammen mit dem Befehl SubnetId for the Local Zone, wie im folgenden Beispiel gezeigt. Ersetzen Sie subnet-22 XXXX1234567 durch die Lokale Zone SubnetId und ersetzen Sie gegebenenfalls andere Optionen. Weitere Informationen finden Sie unter https://docs.aws.amazon.com/cli/latest/reference/emr/create-cluster.html.

```
aws emr create-cluster \
--name "Local Zones cluster" \
--release-label emr-5.29.0 \
--applications Name=Spark \
--ec2-attributes KeyName=myKey,SubnetId=subnet-22XXXX1234567 \
--instance-type m5.xlarge --instance-count 3 --use-default-roles
```

Docker konfigurieren

Amazon EMR 6.x unterstützt Hadoop 3, wodurch Container entweder direkt YARN NodeManager auf dem EMR Amazon-Cluster oder in einem Docker-Container gestartet werden können. Docker-Container bieten benutzerdefinierte Ausführungsumgebungen, in denen Anwendungscode ausgeführt wird. Die benutzerdefinierte Ausführungsumgebung ist von der Ausführungsumgebung der YARN NodeManager und anderer Anwendungen isoliert.

Docker-Container können spezielle Bibliotheken enthalten, die von der Anwendung verwendet werden, und sie können verschiedene Versionen von systemeigenen Tools und Bibliotheken wie R und Python bereitstellen. Sie können die vertrauten Docker-Tools verwenden, um Bibliotheken und Laufzeitabhängigkeiten für Ihre Anwendungen zu definieren.

Amazon EMR 6.x-Cluster sind standardmäßig so konfiguriert, dass YARN Anwendungen wie Spark mithilfe von Docker-Containern ausgeführt werden können. Um die Containerkonfiguration anzupassen, bearbeiten Sie die Docker-Unterstützungsoptionen, die in den im `/etc/hadoop/conf`-Verzeichnis verfügbaren Dateien „`yarn-site.xml`“ und „`container-executor.cfg`“ definiert sind. Weitere Informationen zu den einzelnen Konfigurationsoptionen und deren Verwendung finden Sie unter Starten von Anwendungen mithilfe von Docker-Containern.

Sie können Docker verwenden, wenn Sie eine Aufgabe absenden. Verwenden Sie die folgenden Variablen, um die Docker-Laufzeit und das Docker-Image anzugeben.

- `YARN_CONTAINER_RUNTIME_TYPE=docker`
- `YARN_CONTAINER_RUNTIME_DOCKER_IMAGE={DOCKER_IMAGE_NAME}`

Wenn Sie Docker-Container zum Ausführen Ihrer YARN Anwendungen verwenden, wird das Docker-Image YARN heruntergeladen, das Sie beim Absenden Ihres Jobs angeben. Damit dieses Docker-Image aufgelöst werden kann, muss es mit einer Docker-Registrierung konfiguriert werden. YARN Die Konfigurationsoptionen für eine Docker-Registrierung hängen davon ab, ob Sie den Cluster über ein öffentliches oder privates Subnetz bereitstellen.

Docker-Registrierungen

Eine Docker-Registrierung ist ein Speicher- und Verteilungssystem für Docker-Images. Für Amazon empfehlen EMR wir die Verwendung von AmazonECR, einer vollständig verwalteten Docker-Container-Registry, mit der Sie Ihre eigenen benutzerdefinierten Images erstellen und diese in einer hochverfügbaren und skalierbaren Architektur hosten können.

Überlegungen zur Bereitstellung

Für Docker-Registrierungen ist Netzwerkzugriff von jedem Host im Cluster erforderlich. Das liegt daran, dass jeder Host Bilder aus der Docker-Registry herunterlädt, wenn Ihre YARN Anwendung auf dem Cluster ausgeführt wird. Diese Anforderungen an die Netzwerkkonnektivität können Ihre Wahl der Docker-Registry einschränken, je nachdem, ob Sie Ihren EMR Amazon-Cluster in einem öffentlichen oder privaten Subnetz bereitstellen.

Public subnet (Öffentliches Subnetz)

Wenn EMR Cluster in einem öffentlichen Subnetz bereitgestellt werden, YARN NodeManager können die laufenden Knoten direkt auf jede Registrierung zugreifen, die über das Internet verfügbar ist.

Privates Subnetz

Wenn EMR Cluster in einem privaten Subnetz bereitgestellt werden, haben die laufenden Knoten YARN NodeManager keinen direkten Zugriff auf das Internet. Docker-Images können in Amazon gehostet ECR und über AWS PrivateLink abgerufen werden.

Weitere Informationen dazu, wie Sie AWS PrivateLink den Zugriff auf Amazon ECR in einem privaten Subnetzszenario zulassen, finden Sie unter Einrichtung AWS PrivateLink für Amazon ECS und Amazon ECR.

Konfigurieren von Docker-Registrierungen

Um Docker-Registries mit Amazon zu verwendenEMR, müssen Sie Docker so konfigurieren, dass es der spezifischen Registrierung vertraut, die Sie zum Auflösen von Docker-Images verwenden möchten. Die Standardvertrauensregistrierungen sind „local“ (privat) und „centos“. Um andere öffentliche Repositorien oder Amazon zu verwendenECR, können Sie die `docker.trusted.registries` Einstellungen bei der `/etc/hadoop/conf/container-executor.cfg` Verwendung der EMR Klassifizierung API mit dem `container-executor` Klassifizierungsschlüssel überschreiben.

Das folgende Beispiel zeigt, wie der Cluster so konfiguriert wird, dass er sowohl einem benannten `your-public-repo` öffentlichen Repository als auch einem ECR Registrierungsendpunkt vertraut. `123456789123.dkr.ecr.us-east-1.amazonaws.com` Wenn Sie diesen Endpunkt verwendenECR, ersetzen Sie diesen Endpunkt durch Ihren spezifischen ECR Endpunkt.

```
[
  {
    "Classification": "container-executor",
    "Configurations": [
        {
            "Classification": "docker",
            "Properties": {
                "docker.trusted.registries": "local,centos,your-public-
repo,123456789123.dkr.ecr.us-east-1.amazonaws.com",
                "docker.privileged-containers.registries": "local,centos,your-public-
repo,123456789123.dkr.ecr.us-east-1.amazonaws.com"
            }
        }
    ]
  }
]
```

Um einen Amazon EMR 6.0.0-Cluster mit dieser Konfiguration mithilfe von AWS Command Line Interface (AWS CLI) zu starten, erstellen Sie eine Datei `container-executor.json` mit dem Namen des Inhalts der vorherigen JSON Container-Executor-Konfiguration. Verwenden Sie dann die folgenden Befehle, um den Cluster zu starten.

```
export KEYPAIR=<Name of your Amazon EC2 key-pair>
export SUBNET_ID=<ID of the subnet to which to deploy the cluster>
export INSTANCE_TYPE=<Name of the instance type to use>
```

```
export REGION=<Region to which to deploy the cluster>

aws emr create-cluster \
    --name "EMR-6.0.0" \
    --region $REGION \
    --release-label emr-6.0.0 \
    --applications Name=Hadoop Name=Spark \
    --service-role EMR_DefaultRole \
    --ec2-attributes KeyName=$KEYPAIR,InstanceProfile=EMR_EC2_DefaultRole,SubnetId=
$SUBNET_ID \
    --instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=
$INSTANCE_TYPE InstanceGroupType=CORE,InstanceCount=2,InstanceType=$INSTANCE_TYPE \
    --configuration file://container-executor.json
```

Konfiguration YARN für den Zugriff ECR auf Amazon unter EMR 6.0.0 und früher

Wenn Sie neu bei Amazon sindECR, folgen Sie den Anweisungen unter Erste Schritte mit Amazon ECR und stellen Sie sicher, dass Sie ECR von jeder Instance in Ihrem EMR Amazon-Cluster aus Zugriff auf Amazon haben.

In EMR Version 6.0.0 und früher müssen Sie zunächst Anmeldeinformationen generieren, um ECR mit dem Docker-Befehl auf Amazon zuzugreifen. Um zu überprüfen, ob auf Bilder von Amazon zugegriffen werden YARN kannECR, verwenden Sie die Container-Umgebungsvariable, `YARN_CONTAINER_RUNTIME_DOCKER_CLIENT_CONFIG` um einen Verweis auf die von Ihnen generierten Anmeldeinformationen zu übergeben.

Führen Sie den folgenden Befehl auf einem der Core-Nodes aus, um die Login-Zeile für Ihr ECR Konto abzurufen.

```
aws ecr get-login --region us-east-1 --no-include-email
```

Der `get-login` Befehl generiert den richtigen CLI Docker-Befehl, der ausgeführt werden muss, um Anmeldeinformationen zu erstellen. Kopieren Sie die Ausgabe von `get-login` und führen Sie sie aus.

```
sudo docker login -u AWS -p <password> https://<account-id>.dkr.ecr.us-
east-1.amazonaws.com
```

Mit diesem Befehl wird eine `config.json`-Datei im `/root/.docker`-Ordner generiert. Kopieren Sie diese Datei in, HDFS damit Jobs, die an den Cluster gesendet wurden, sie zur Authentifizierung bei Amazon ECR verwenden können.

Führen Sie die folgenden Befehle aus, um die `config.json`-Datei in Ihr Startverzeichnis zu kopieren.

```
mkdir -p ~/.docker
sudo cp /root/.docker/config.json ~/.docker/config.json
sudo chmod 644 ~/.docker/config.json
```

Führen Sie die folgenden Befehle aus, um die Datei config.json HDFS so zu speichern, dass sie von Jobs verwendet werden kann, die auf dem Cluster ausgeführt werden.

```
hadoop fs -put ~/.docker/config.json /user/hadoop/
```

YARNkann ECR als Docker-Image-Registry zugreifen und während der Auftragsausführung Container abrufen.

Nachdem Sie die Docker-Registrierungen und konfiguriert habenYARN, können Sie YARN Anwendungen mithilfe von Docker-Containern ausführen. Weitere Informationen finden Sie unter Spark-Anwendungen mit Docker mithilfe von Amazon EMR 6.0.0 ausführen.

In EMR 6.1.0 und höher müssen Sie die Authentifizierung bei Amazon ECR nicht manuell einrichten. Wenn im `container-executor` Klassifikationsschlüssel eine ECR Amazon-Registrierung erkannt wird, wird die ECR auto Amazon-Authentifizierungsfunktion aktiviert und YARN wickelt den Authentifizierungsprozess ab, wenn Sie einen Spark-Job mit einem ECR Bild einreichen. Sie können überprüfen, ob die automatische Authentifizierung aktiviert ist, indem Sie in `yarn.nodemanager.runtime.linux.docker.ecr-auto-authentication.enabled` der YARN-Seite einchecken. Die automatische Authentifizierung ist aktiviert und die YARN Authentifizierungseinstellung ist auf gesetzt, `true` wenn die eine ECR Registrierung `docker.trusted.registries` enthältURL.

Voraussetzungen für die Verwendung der automatischen Authentifizierung bei Amazon ECR

- EMRVersion 6.1.0 oder höher
- ECRDie in der Konfiguration enthaltene Registrierung befindet sich in derselben Region wie der Cluster

- IAMRolle mit Berechtigungen zum Abrufen eines Autorisierungstoken und zum Abrufen eines beliebigen Images

Weitere Informationen finden Sie unter Einrichtung ECR bei Amazon.

Wie aktiviere ich die automatische Authentifizierung

Gehen Sie wie folgt vor, Konfigurieren von Docker-Registrierungen um eine ECR Amazon-Registrierung als vertrauenswürdige Registrierung festzulegen, und stellen Sie sicher, dass sich das ECR Amazon-Repository und der Cluster in derselben Region befinden.

Um diese Funktion auch dann zu aktivieren, wenn die ECR Registrierung nicht in der vertrauenswürdigen Registrierung festgelegt ist, verwenden Sie die Konfigurationsklassifizierung für die Einstellung `yarn.nodemanager.runtime.linux.docker.ecr-auto-authentication.enabled` auf`true`.

Wie deaktiviere ich die automatische Authentifizierung

Standardmäßig ist die automatische Authentifizierung deaktiviert, wenn in der vertrauenswürdigen ECR Registrierung keine Amazon-Registrierung erkannt wird.

Um die automatische Authentifizierung zu deaktivieren, auch wenn die ECR Amazon-Registrierung in der vertrauenswürdigen Registrierung festgelegt ist, verwenden Sie die Konfigurationsklassifizierung für die Einstellung `yarn.nodemanager.runtime.linux.docker.ecr-auto-authentication.enabled` auf`false`.

Wie überprüft man, ob die automatische Authentifizierung in einem Cluster aktiviert ist

Verwenden Sie einen Text-Editor wie `vi` auf dem Hauptknoten, um den Inhalt der Datei `vi /etc/hadoop/conf.empty/yarn-site.xml` anzuzeigen. Überprüfen Sie den Wert von `yarn.nodemanager.runtime.linux.docker.ecr-auto-authentication.enabled`.

Steuern der Cluster-Beendigung

In diesem Abschnitt werden Ihre Optionen zum Herunterfahren von EMR Amazon-Clustern beschrieben. Es behandelt automatische Kündigung und Kündigungsschutz sowie deren Interaktion mit anderen EMR Amazon-Funktionen.

Sie können einen EMR Amazon-Cluster auf folgende Weise herunterfahren:

- Kündigung nach der Ausführung des letzten Schritts – Erstellen Sie einen vorübergehenden Cluster, der nach Abschluss aller Schritte heruntergefahren wird.
- Automatische Kündigung (nach Inaktivität) – Erstellen Sie einen Cluster mit einer automatischen Terminierungsrichtlinie, der nach einer bestimmten Leerlaufzeit heruntergefahren wird. Weitere Informationen finden Sie unter Verwenden einer Richtlinie zur automatischen Beendigung.
- Manuelles Beenden – Erstellen Sie einen Cluster mit langer Laufzeit, der so lange läuft, bis Sie ihn bewusst beenden. Informationen zum manuellen Beenden eines Clusters finden Sie unter Einen Cluster beenden.

Sie können auch einen Kündigungsschutz für einen Cluster einrichten, um zu verhindern, dass EC2 Instances versehentlich oder versehentlich heruntergefahren werden.

Wenn Amazon Ihren Cluster EMR herunterfährt, werden alle EC2 Amazon-Instances im Cluster heruntergefahren. Daten im Instance-Speicher und auf den EBS Volumes sind nicht mehr verfügbar und können nicht wiederhergestellt werden. Es ist von kritischer Bedeutung, das Beenden von Clustern zu verstehen und zu kontrollieren, um eine Strategie für die Verwaltung und Bewahrung von Daten erstellen zu können, bei der die Daten zu Amazon S3 geschrieben und die Kosten abgewogen werden.

Themen

Konfigurieren eines Clusters zum Fortfahren oder Beenden nach der Schrittausführung

In diesem Thema werden die Unterschiede zwischen der Verwendung eines Clusters mit langer Laufzeit und der Erstellung eines transienten Clusters erläutert, der nach der Ausführung des letzten Schritts heruntergefahren wird. Außerdem wird beschrieben, wie die Schrittausführung für einen Cluster konfiguriert wird.

So erstellen Sie einen langlebigen Cluster

Standardmäßig haben Cluster, die Sie mit der Konsole oder der AWS CLI erstellen, eine lange Laufzeit. Cluster mit langer Laufzeit laufen weiter, akzeptieren Arbeit und es fallen Gebühren an, bis Sie Maßnahmen ergreifen, um sie herunterzufahren.

Ein Cluster mit langer Laufzeit ist in folgenden Situationen wirksam:

- Wenn Sie interaktiv oder automatisch Daten abfragen müssen.
- Wenn Sie kontinuierlich mit Big-Data-Anwendungen interagieren müssen, die auf dem Cluster gehostet werden.
- Wenn Sie regelmäßig einen Datensatz verarbeiten, der so groß oder so häufig ist, dass es ineffizient ist, jedes Mal neue Cluster zu starten und Daten zu laden.

Sie können auch einen Kündigungsschutz für einen Cluster mit langer Laufzeit einrichten, um zu verhindern, dass EC2 Instanzen versehentlich oder versehentlich heruntergefahren werden. Weitere Informationen finden Sie unter Verwenden des Beendigungsschutzes.

> **Note**
>
> Amazon aktiviert EMR automatisch den Kündigungsschutz für alle Cluster mit mehreren Primärknoten und überschreibt alle Einstellungen für die Schrittausführung, die Sie bei der Erstellung des Clusters angeben. Sie können den Kündigungsschutz deaktivieren, nachdem der Cluster gestartet wurde. Siehe Konfigurieren des Beendigungsschutzes für aktive Cluster. Um einen Cluster mit mehreren Primärknoten herunterzufahren, müssen Sie zunächst die Clusterattribute ändern, um den Kündigungsschutz zu deaktivieren. Detaillierte Anweisungen finden Sie unter Einen EMR Amazon-Cluster mit mehreren Primärknoten beenden.

Einen Cluster so konfigurieren, dass er nach der Ausführung des Schritts beendet wird

Wenn Sie die Beendigung nach der Schrittausführung konfigurieren, startet der Cluster, führt Bootstrap-Aktionen aus und führt dann die von Ihnen angegebenen Schritte aus. Sobald der letzte Schritt abgeschlossen ist, EMR beendet Amazon die EC2 Amazon-Instances des Clusters. Bei Clustern, die Sie mit Amazon starten, ist EMR API die Step-Ausführung standardmäßig aktiviert.

Die Beendigung nach der Schrittausführung ist für Cluster wirksam, die eine periodische Verarbeitungsaufgabe ausführen, beispielsweise einen täglichen Datenverarbeitungslauf. Mit der

schrittweisen Ausführung können Sie außerdem sicherstellen, dass Ihnen nur die Zeit in Rechnung gestellt wird, die für die Verarbeitung Ihrer Daten erforderlich ist. Weitere Informationen zu den Schritten finden Sie unter Übermitteln von Arbeit an einen Cluster.

Console

Um die Beendigung nach der schrittweisen Ausführung mit der Konsole zu aktivieren

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und anschließend Create cluster aus.
3. Wählen Sie unter Schritte die Option Schritt hinzufügen aus. Geben Sie im Dialogfeld Schritt hinzufügen die entsprechenden Feldwerte ein. Die Optionen unterscheiden sich je nach Schritttyp. Um Ihren Schritt hinzuzufügen und das Dialogfeld zu verlassen, wählen Sie Schritt hinzufügen.
4. Aktivieren Sie unter Clusterbeendigung das Kontrollkästchen Cluster nach Abschluss des letzten Schritts beenden.
5. Wählen Sie alle anderen Optionen aus, die für Ihren Cluster gelten.
6. Um Ihren Cluster jetzt zu starten, wählen Sie Cluster erstellen aus.

AWS CLI

Um die Beendigung nach der Ausführung des Schritts zu aktivieren, verwenden Sie AWS CLI

- Geben Sie den `--auto-terminate`-Parameter an, wenn Sie den `create-cluster`-Befehl verwenden, um einen vorübergehenden Cluster zu erstellen.

  Das folgende Beispiel veranschaulicht die Verwendung des `--auto-terminate`-Parameters. Sie können den folgenden Befehl eingeben und ihn ersetzen *`myKey`* mit dem Namen Ihres EC2 key pair.

  > **Note**
  >
  > Linux-Zeilenfortsetzungszeichen (\) sind aus Gründen der Lesbarkeit enthalten. Sie können entfernt oder in Linux-Befehlen verwendet werden. Entfernen Sie sie unter Windows oder ersetzen Sie sie durch ein Caret-Zeichen (^).

```
aws emr create-cluster --name "Test cluster" --release-label emr-7.2.0 \
--applications Name=Hive Name=Pig --use-default-roles --ec2-attributes
 KeyName=myKey \
--steps Type=PIG,Name="Pig Program",ActionOnFailure=CONTINUE,\
Args=[-f,s3://mybucket/scripts/pigscript.pig,-p,\
INPUT=s3://mybucket/inputdata/,-p,OUTPUT=s3://mybucket/outputdata/,\
$INPUT=s3://mybucket/inputdata/,$OUTPUT=s3://mybucket/outputdata/]
--instance-type m5.xlarge --instance-count 3 --auto-terminate
```

API

So deaktivieren Sie die Kündigung nach der Ausführung des Schritts beim Start von Amazon EMR API im Cluster

1. Wenn Sie die RunJobFlowAktion verwenden, um einen Cluster zu erstellen, setzen Sie die KeepJobFlowAliveWhenNoStepsEigenschaft auf`false`.
2. So ändern Sie Ihre Konfiguration für die Beendigung nach der Ausführung des Schritts mit dem Start des Amazon-Clusters nach EMR API dem Start des Clusters:

   SetKeepJobFlowAliveWhenNoSteps Aktion verwenden.

Verwenden einer Richtlinie zur automatischen Beendigung

Mit einer Richtlinie zur automatischen Terminierung können Sie die Clusterbereinigung orchestrieren, ohne ungenutzte Cluster überwachen und manuell beenden zu müssen. Wenn Sie einem Cluster eine automatische Terminierungsrichtlinie hinzufügen, geben Sie die Leerlaufzeit an, nach der der Cluster automatisch heruntergefahren werden soll.

Je nach Release-Version EMR verwendet Amazon unterschiedliche Kriterien, um einen Cluster als inaktiv zu kennzeichnen. In der folgenden Tabelle wird beschrieben, wie Amazon EMR den Cluster-Leerlauf bestimmt.

Wenn Sie …	Ein Cluster gilt als inaktiv, wenn …
EMRAmazon-Versionen 5.34.0 und höher und 6.4.0 und höher	• Es gibt keine aktiven Anwendungen YARN •

Wenn Sie …	Ein Cluster gilt als inaktiv, wenn …
	HDFSDie Auslastung liegt unter 10% • Es gibt keine aktiven EMR Notebook- oder EMR Studio-Verbindungen • Es werden keine Benutzeroberflächen für Cluster-Anwendungen verwendet • Es gibt keine ausstehenden Schritte
EMRAmazon-Versionen 5.30.0 — 5.33.0 und 6.1.0 — 6.3.0	• Es YARN gibt keine aktiven Anwendungen • Der Cluster hat keine aktiven Spark–Aufträge **Note** Amazon EMR markiert einen Cluster als inaktiv und kann den Cluster automatisch beenden, auch wenn Sie einen aktiven Python3-Kernel haben. Das liegt daran, dass bei der Ausführung eines Python3-Kernels kein Spark-Job auf dem Cluster gesendet wird. Um die automatische Terminierung mit einem Python3-Kernel zu verwenden, empfehlen wir die Verwendung von EMR Amazon-Version 6.4.0 oder höher.

Note

EMRAmazon-Versionen 6.4.0 und höher unterstützen eine Cluster-Datei zur Erkennung von Aktivitäten auf dem primären Knoten:. `/emr/metricscollector/isbusy` Wenn Sie einen

Cluster verwenden, um Shell-Skripts oder andere YARN Anwendungen auszuführen, können Sie Amazon in regelmäßigen Abständen berühren oder aktualisieren, `isbusy` um Amazon mitzuteilenEMR, dass sich der Cluster nicht im Leerlauf befindet.

Sie können beim Erstellen eines Clusters eine automatische Terminierungsrichtlinie anhängen oder einem vorhandenen Cluster eine Richtlinie hinzufügen. Um die automatische Kündigung zu ändern oder zu deaktivieren, können Sie die Richtlinie aktualisieren oder entfernen.

Überlegungen

Berücksichtigen Sie die folgenden Features und Einschränkungen, bevor Sie eine Richtlinie zum automatischen Beenden verwenden:

- Im Folgenden AWS-Regionen ist die EMR automatische Kündigung von Amazon mit Amazon EMR 6.14.0 und höher verfügbar:
  - Asien-Pazifik (Hyderabad) (ap-south-2)
  - Asien-Pazifik (Jakarta) (ap-southeast-3)
  - Europa (Spanien) (eu-south-2)
- Im Folgenden AWS-Regionen ist die EMR automatische Kündigung von Amazon mit Amazon EMR 5.30.0 und 6.1.0 und höher verfügbar:
  - USA Ost (Nord-Virginia): (us-east-1)
  - USA Ost (Ohio): (us-east-2)
  - USA West (Oregon): (us-west-2)
  - USA West (Nordkalifornien) (us-west-1)
  - Afrika (Kapstadt) (af-south-1)
  - Asien-Pazifik (Hongkong) (ap-east-1)
  - Asien-Pazifik (Mumbai): (ap-south-1)
  - Asien-Pazifik (Seoul): (ap-northeast-2)
  - Asien-Pazifik (Singapur): (ap-southeast-1)
  - Asien-Pazifik (Sydney): (ap-southeast-2)
  - Asien-Pazifik (Tokyo) (ap-northeast-1)
  - Kanada (Zentral): (ca-central-1)
  - Südamerika (São Paulo) (sa-east-1)

  - Europa (Frankfurt) (eu-central-1)
  - Europa (Irland) (eu-west-1)
  - Europa (London) (eu-west-2)
  - Europa (Mailand) (eu-south-1)
  - Europa (Paris) (eu-west-3)
  - Europa (Stockholm) (eu-north-1)
  - China (Peking) (cn-north-1)
  - China (Ningxia) (cn-northwest-1)
  - AWS GovCloud (US-Ost) (-1) us-gov-east
  - AWS GovCloud (US-West) (us-gov-west-1)
- Das Leerlauf-Timeout ist standardmäßig auf 60 Minuten (eine Stunde) eingestellt, wenn Sie keinen Wert angeben. Sie können ein minimales Timeout für den Leerlauf von einer Minute und ein maximales Timeout für den Leerlauf von 7 Tagen angeben.
- Bei EMR Amazon-Versionen 6.4.0 und höher ist die automatische Terminierung standardmäßig aktiviert, wenn Sie einen neuen Cluster mit der EMR Amazon-Konsole erstellen.
- Amazon EMR veröffentlicht hochauflösende Amazon CloudWatch Metriken, wenn Sie die automatische Terminierung für einen Cluster aktivieren. Sie können diese Metriken verwenden, um Cluster-Aktivität und Inaktivität zu verfolgen. Weitere Informationen finden Sie unter Cluster-Kapazitätsmetriken.
- Die automatische Terminierung wird nicht unterstützt, wenn Sie nicht YARN basierte Anwendungen wie Presto, Trino oder verwenden. HBase
- Um die automatische Terminierung zu verwenden, muss der Metrics-Collector-Prozess in der Lage sein, eine Verbindung zum öffentlichen API Endpunkt für die automatische Terminierung in Gateway herzustellen. API Wenn Sie einen privaten DNS Namen mit verwenden Amazon Virtual Private Cloud, funktioniert die automatische Terminierung nicht richtig. Um sicherzustellen, dass die automatische Beendigung funktioniert, empfehlen wir Ihnen, eine der folgenden Maßnahmen zu ergreifen:
  - Entfernen Sie den API VPC Gateway-Schnittstellenendpunkt von Ihrem AmazonVPC.
  - Folgen Sie den Anweisungen unter Warum erhalte ich die Fehlermeldung HTTP 403 Forbidden, wenn ich von einem APIs aus eine Verbindung zu meinem API Gateway herstelleVPC? um die Einstellung für private DNS Namen zu deaktivieren.
  - Starten Sie Ihren Cluster stattdessen in einem privaten Subnetz. Weitere Informationen finden Sie im Thema Private Subnetze.

- (EMR5.30.0 und höher) Wenn Sie die Standardregel „Alle ausgehenden Nachrichten zulassen“ für die primäre Sicherheitsgruppe auf 0.0.0.0/ entfernen, müssen Sie eine Regel hinzufügen, die ausgehende TCP Konnektivität zu Ihrer Sicherheitsgruppe für den Dienstzugriff auf Port 9443 zulässt. Ihre Sicherheitsgruppe für den Dienstzugriff muss auch eingehenden TCP Datenverkehr über Port 9443 von der primären Sicherheitsgruppe zulassen. Weitere Informationen zur Konfiguration von Sicherheitsgruppen finden Sie unter Von Amazon EMR verwaltete Sicherheitsgruppe für die primäre Instance (private Subnetze).

Berechtigungen zur Verwendung der automatischen Beendigung

Bevor Sie Richtlinien zur automatischen Kündigung für Amazon anwenden und verwalten könnenEMR, müssen Sie die in der folgenden Beispielrichtlinie aufgeführten IAM Berechtigungen den IAM Ressourcen zuordnen, die Ihren EMR Cluster verwalten.

```
{
  "Version": "2012-10-17",
  "Statement": {
      "Sid": "AllowAutoTerminationPolicyActions",
      "Effect": "Allow",
      "Action": [
        "elasticmapreduce:PutAutoTerminationPolicy",
        "elasticmapreduce:GetAutoTerminationPolicy",
        "elasticmapreduce:RemoveAutoTerminationPolicy"
      ],
      "Resource": "<your-resources>"
    }
}
```

Eine Richtlinie zur automatischen Beendigung anhängen, aktualisieren oder entfernen

Dieser Abschnitt enthält Anweisungen, die Ihnen helfen, eine Richtlinie zur automatischen Kündigung an einen EMR Amazon-Cluster anzuhängen, zu aktualisieren oder zu entfernen. Bevor Sie mit Richtlinien zur automatischen Kündigung arbeiten, stellen Sie sicher, dass Sie über die erforderlichen IAM Berechtigungen verfügen. Siehe Berechtigungen zur Verwendung der automatischen Beendigung.

Console

Um eine automatische Terminierungsrichtlinie anzuhängen, wenn Sie einen Cluster mit der Konsole erstellen

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und anschließend Create cluster aus.
3. Wählen Sie unter Clusterbeendigung die Option Cluster nach Leerlauf beenden aus.
4. Geben Sie die Anzahl der Stunden und Minuten im Leerlauf an, die vergehen können, bis der Cluster automatisch beendet wird. Die standardmäßige Leerlaufzeit beträgt eine Stunde.
5. Wählen Sie alle anderen Optionen aus, die für Ihren Cluster gelten.
6. Um Ihren Cluster jetzt zu starten, wählen Sie Cluster erstellen aus.

Um eine automatische Terminierungsrichtlinie auf einem laufenden Cluster mit der Konsole anzuhängen, zu aktualisieren oder zu entfernen

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und dann den Cluster aus, den Sie aktualisieren möchten.
3. Suchen Sie auf der Registerkarte Eigenschaften der Cluster-Detailseite nach Clusterbeendigung und wählen Sie Bearbeiten aus.
4. Wählen oder deaktivieren Sie Automatische Beendigung aktivieren, um das Feature ein- oder auszuschalten. Wenn Sie die automatische Terminierung aktivieren, geben Sie die Anzahl der Stunden und Minuten im Leerlauf an, die vergehen können, bis der Cluster automatisch beendet wird. Wählen Sie dann zur Bestätigung Änderungen speichern aus.

AWS CLI

Bevor Sie beginnen

Bevor Sie mit Richtlinien zur automatischen Kündigung arbeiten, empfehlen wir Ihnen, auf die neueste Version von AWS CLI zu aktualisieren. Anweisungen finden Sie unter Installieren, Aktualisieren und Deinstallieren von AWS CLI.

Um eine automatische Beendigungsrichtlinie anzuhängen oder zu aktualisieren, verwenden Sie AWS CLI

- Sie können den `aws emr put-auto-termination-policy`-Befehl verwenden, um eine automatische Beendigungssrichtlinie für einen Cluster anzuhängen oder zu aktualisieren.

  Das folgende Beispiel spezifiziert 3600 Sekunden für *`IdleTimeout`*. Wenn Sie es nicht spezifizieren *`IdleTimeout`*, ist der Standardwert auf eine Stunde eingestellt.

```
aws emr put-auto-termination-policy \
--cluster-id <your-cluster-id> \
--auto-termination-policy IdleTimeout=3600
```

  > **Note**
  >
  > Linux-Zeilenfortsetzungszeichen (\) sind aus Gründen der Lesbarkeit enthalten. Sie können entfernt oder in Linux-Befehlen verwendet werden. Entfernen Sie sie unter Windows oder ersetzen Sie sie durch ein Caret-Zeichen (^).

  Sie können auch einen Wert für `--auto-termination-policy` angeben, wenn Sie den `aws emr create-cluster`-Befehl verwenden. Weitere Informationen zur Verwendung von EMR Amazon-Befehlen in der AWS CLI finden Sie in der AWS CLI Befehlsreferenz.

Um eine automatische Kündigungsrichtlinie zu entfernen mit dem AWS CLI

- Verwenden Sie den `aws emr remove-auto-termination-policy`-Befehl, um eine automatische Beendigungsrichtlinie aus einem Cluster zu entfernen. Weitere Informationen zur Verwendung von EMR Amazon-Befehlen in der AWS CLI finden Sie in der AWS CLI Befehlsreferenz.

```
aws emr remove-auto-termination-policy --cluster-id <your-cluster-id>
```

Verwenden des Beendigungsschutzes

Der Terminierungsschutz schützt Ihre Cluster vor einer versehentlichen Kündigung. Dies kann besonders bei Clustern mit langer Laufzeit, die kritische Workloads verarbeiten, nützlich sein. Wenn

der Beendigungsschutz für einen langlebigen Cluster aktiviert ist, können Sie den Cluster weiter beenden, müssen jedoch zunächst den Beendigungsschutz explizit aus dem Cluster entfernen. Dadurch wird sichergestellt, dass EC2 Instances nicht versehentlich oder irrtümlich heruntergefahren werden. Sie können den Beendigungsschutz aktivieren, wenn Sie einen Cluster erstellen. Sie können die Einstellung auf einem ausgeführten Cluster ändern.

Wenn der Kündigungsschutz aktiviert ist, EMR API funktioniert die `TerminateJobFlows` Aktion im Amazon nicht. Benutzer können den Cluster nicht mit diesem API oder dem `terminate-clusters` Befehl von beenden AWS CLI. Der API gibt einen Fehler zurück und CLI beendet den Vorgang mit einem Rückgabecode ungleich Null. Wenn Sie die EMR Amazon-Konsole verwenden, um einen Cluster zu beenden, werden Sie zu einem zusätzlichen Schritt aufgefordert, den Kündigungsschutz auszuschalten.

> **Warning**
>
> Der Kündigungsschutz garantiert nicht, dass Daten im Falle eines menschlichen Fehlers oder einer Behelfslösung erhalten bleiben, z. B. wenn ein Neustartbefehl über die Befehlszeile ausgegeben wird, während eine Verbindung zur Instance bestehtSSH, wenn eine Anwendung oder ein Skript, das auf der Instance ausgeführt wird, einen Neustartbefehl ausgibt oder wenn Amazon EC2 oder Amazon verwendet EMR API wird, um den Kündigungsschutz zu deaktivieren. Dies gilt auch, wenn Sie EMR Amazon-Versionen 7.1 und höher ausführen und eine Instance fehlerhaft und nicht wiederherstellbar ist. Selbst wenn der Kündigungsschutz aktiviert ist, können im Instance-Speicher gespeicherte Daten, einschließlich HDFS Daten, verloren gehen. Schreiben Sie die Datenausgabe an Amazon-S3-Standorte und erstellen Sie Backup-Strategien, die Ihren Anforderungen an die Geschäftskontinuität entsprechen.

Der Beendigungsschutz wirkt sich nicht auf Ihre Fähigkeit aus, Cluster-Ressourcen mit einer der folgenden Aktionen zu skalieren:

- Manuelles Ändern der Größe eines Clusters mit dem AWS Management Console oder AWS CLI. Weitere Informationen finden Sie unter Manuelle Größenanpassung eines aktiven Clusters.
- Entfernen von Instances aus einer Core- oder Aufgaben-Instance-Gruppe unter Verwendung einer Abwärtsskalierungsrichtlinie mit Auto Scaling. Weitere Informationen finden Sie unter Verwenden der automatischen Skalierung mit einer benutzerdefinierten Richtlinie für Instance-Gruppen.

- Entfernen von Instances aus einer Instance-Flotte durch Reduzierung der Zielkapazität. Weitere Informationen finden Sie unter Instance-Flotten-Optionen.

Kündigungsschutz und Amazon EC2

Die Einstellung für den Kündigungsschutz in einem EMR Amazon-Cluster entspricht dem `DisableApiTermination` Attribut für alle EC2 Amazon-Instances im Cluster. Wenn Sie beispielsweise den Kündigungsschutz in einem EMR Cluster aktivieren, setzt `DisableApiTermination` Amazon EMR automatisch für alle EC2 Instances innerhalb des EMR Clusters auf true. Das Gleiche gilt, wenn Sie den Kündigungsschutz deaktivieren. Amazon setzt EMR `DisableApiTermination` automatisch für alle EC2 Instances innerhalb des EMR Clusters auf False. Wenn Sie einen Cluster von Amazon beenden oder herunterskalieren EMR und die EC2 Amazon-Einstellungen für eine EC2 Instance in Konflikt geraten, EMR priorisiert Amazon die EMR Amazon-Einstellung vor den `DisableApiTermination` Einstellungen `DisableApiStop` und in Amazon EC2 und beendet die EC2 Instance weiterhin.

Sie können beispielsweise die EC2 Amazon-Konsole verwenden, um den Kündigungsschutz für eine EC2 Amazon-Instance in einem EMR Cluster mit deaktiviertem Kündigungsschutz zu aktivieren. Wenn Sie den Cluster mit der EMR Amazon-Konsole, der AWS CLI oder Amazon beenden oder herunterskalieren EMRAPI, EMR überschreibt Amazon die `DisableApiTermination` Einstellung, setzt sie auf False und beendet die Instance zusammen mit anderen Instances.

Sie können die EC2 Amazon-Konsole auch verwenden, um den Stop-Schutz für eine EC2 Amazon-Instance in einem EMR Cluster mit deaktiviertem Kündigungsschutz zu aktivieren. Wenn Sie den Cluster beenden oder herunterskalieren, EMR setzt Amazon in Amazon `DisableApiStop` auf False EC2 und beendet die Instance zusammen mit anderen Instances.

Amazon EMR überschreibt die `DisableApiStop` Einstellung nur, wenn Sie einen Cluster beenden oder herunterskalieren. Wenn Sie den Kündigungsschutz in einem EMR Cluster aktivieren oder deaktivieren, EMR ändert Amazon die `disableApiStop` Einstellung für keine der EC2 Instances im jeweiligen EMR Cluster.

> **Important**
>
> Wenn Sie eine Instance als Teil eines EMR Amazon-Clusters mit Kündigungsschutz erstellen und die Amazon- EC2 API oder AWS CLI -Befehle verwenden, um die Instance so zu ändern, dass das `DisableApiTermination` heißt`false`, und dann die Amazon- EC2 API oder

AWS CLI -Befehle den `TerminateInstances` Vorgang ausführen, wird die EC2 Amazon-Instance beendet.

Kündigungsschutz und fehlerhafte Knoten YARN

Amazon überprüft EMR regelmäßig den Apache YARN Hadoop-Status von Knoten, die auf Kern- und EC2 Task-Amazon-Instances in einem Cluster ausgeführt werden. Der Gesundheitsstatus wird vom NodeManager Health Checker Service gemeldet. Wenn ein Knoten meldet`UNHEALTHY`, fügt der EMR Amazon-Instance-Controller den Knoten zu einer Denylist hinzu und weist ihm keine YARN Container zu, bis er wieder fehlerfrei ist. Abhängig vom Status des Kündigungsschutzes, des Austauschs fehlerhafter Knoten und der EMR Amazon-Release-Version ersetzt Amazon entweder EMR die fehlerhafte Instance oder beendet die Zuweisung von Controllern zur Instance.

Kündigungsschutz und Kündigung nach Ausführung des Schritts

Wenn Sie die Kündigung nach der Ausführung des Schritts aktivieren und gleichzeitig den Kündigungsschutz aktivieren, EMR ignoriert Amazon den Kündigungsschutz.

Wenn Sie Schritte an einen Cluster übermitteln, können Sie die Eigenschaft `ActionOnFailure` festlegen, um zu bestimmen, was passiert, wenn die Ausführung eines Schritts aufgrund eines Fehlers nicht abgeschlossen werden kann. Die möglichen Werte für diese Einstellung sind `TERMINATE_CLUSTER` (`TERMINATE_JOB_FLOW` mit früheren Versionen), `CANCEL_AND_WAIT` und `CONTINUE`. Weitere Informationen finden Sie unter Übermitteln von Arbeit an einen Cluster.

Wenn ein Schritt fehlschlägt, der mit der `ActionOnFailure` Einstellung auf konfiguriert ist`CANCEL_AND_WAIT`, und wenn die Beendigung nach der Ausführung des Schritts aktiviert ist, wird der Cluster beendet, ohne dass nachfolgende Schritte ausgeführt werden.

Wenn ein Schritt fehlschlägt, für den `ActionOnFailure` auf `TERMINATE_CLUSTER` festgelegt wurde, können Sie anhand der folgenden Tabelle mit Einstellungen das Ergebnis ermitteln.

ActionOnFailure	Beendigung nach der Ausführung des Schritts	Termination protection	Ergebnis
	Enabled	Disabled	Cluster wird beendet

ActionOnFailure	Beendigung nach der Ausführung des Schritts	Termination protection	Ergebnis
`TERMINATE_CLUSTER`	Aktiviert	Aktiviert	Cluster wird beendet
	Disabled	Aktiviert	Cluster wird weiter ausgeführt
	Disabled	Disabled	Cluster wird beendet

Beendigungsschutz und Spot Instances

Der EMR Kündigungsschutz von Amazon verhindert nicht, dass eine Amazon EC2 Spot-Instance beendet wird, wenn der Spot-Preis über den maximalen Spot-Preis steigt.

Konfigurieren des Beendigungsschutzes beim Starten eines Clusters

Sie können den Kündigungsschutz aktivieren oder deaktivieren, wenn Sie einen Cluster über die Konsole AWS CLI, die oder die API starten.

Für Cluster mit einem Knoten lauten die Standardeinstellungen für den Kündigungsschutz wie folgt:

- Starten eines Clusters über Amazon EMR Console — Termination Protection ist standardmäßig deaktiviert.
- Das Starten eines Clusters mit AWS CLI `aws emr create-cluster` —Termination Protection ist deaktiviert, sofern nicht anders angegeben`--termination-protected`.
- Einen Cluster per Amazon EMR API RunJobFlowCommand—Termination Protection starten ist deaktiviert, sofern der `TerminationProtected` boolesche Wert nicht auf gesetzt ist. `true`

Für Hochverfügbarkeitscluster lauten die Standardeinstellungen für den Kündigungsschutz wie folgt:

- Starten eines Clusters über Amazon EMR Console — Termination Protection ist standardmäßig aktiviert.
- Das Starten eines Clusters mit AWS CLI `aws emr create-cluster` —Termination Protection ist deaktiviert, sofern nicht anders `--termination-protected` angegeben.

- Einen Cluster per Amazon EMR API RunJobFlowCommand—Termination Protection starten ist deaktiviert, sofern der `TerminationProtected` boolesche Wert nicht auf gesetzt ist. `true`

Console

Um den Kündigungsschutz ein- oder auszuschalten, wenn Sie einen Cluster mit der Konsole erstellen

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und anschließend Create cluster aus.
3. Wählen Sie für die EMRRelease-Version emr-6.6.0 oder höher aus.
4. Vergewissern Sie sich, dass unter Clusterbeendigung und Knotenaustausch die Option Kündigungsschutz verwenden vorausgewählt ist, oder löschen Sie die Auswahl, um sie auszuschalten.
5. Wählen Sie alle anderen Optionen aus, die für Ihren Cluster gelten.
6. Um Ihren Cluster jetzt zu starten, wählen Sie Cluster erstellen aus.

AWS CLI

Um den Kündigungsschutz ein- oder auszuschalten, wenn Sie einen Cluster mit dem AWS CLI

- Mit dem AWS CLI können Sie einen Cluster mit aktiviertem Kündigungsschutz mit dem `create-cluster` Befehl mit dem `--termination-protected` Parameter starten. Der Beendigungsschutz ist standardmäßig deaktiviert.

  Im folgenden Beispiel wird ein Cluster mit aktiviertem Beendigungsschutz erstellt:

  > **Note**
  >
  > Linux-Zeilenfortsetzungszeichen (\) sind aus Gründen der Lesbarkeit enthalten. Sie können entfernt oder in Linux-Befehlen verwendet werden. Entfernen Sie sie unter Windows oder ersetzen Sie sie durch ein Caret-Zeichen (^).

```
aws emr create-cluster --name "TerminationProtectedCluster" --release-
label emr-7.2.0 \
--applications Name=Hadoop Name=Hive Name=Pig \
--use-default-roles --ec2-attributes KeyName=myKey --instance-type m5.xlarge \
--instance-count 3 --termination-protected
```

Weitere Informationen zur Verwendung von EMR Amazon-Befehlen in der AWS CLI finden Sie unterhttps://docs.aws.amazon.com/cli/latest/reference/emr.

Konfigurieren des Beendigungsschutzes für aktive Cluster

Sie können den Beendigungsschutz für einen aktiven Cluster mithilfe der Konsole oder AWS CLI konfigurieren.

Console

So schalten Sie den Kündigungsschutz für einen laufenden Cluster mit der Konsole ein oder aus

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und dann den Cluster aus, den Sie aktualisieren möchten.
3. Suchen Sie auf der Registerkarte Eigenschaften der Cluster-Detailseite nach Clusterbeendigung und wählen Sie Bearbeiten aus.
4. Aktivieren oder deaktivieren Sie das Kontrollkästchen Beendigungsschutz verwenden, um das Feature ein- oder auszuschalten. Wählen Sie dann zur Bestätigung Änderungen speichern aus.

AWS CLI

Um den Kündigungsschutz für einen laufenden Cluster ein- oder auszuschalten, verwenden Sie AWS CLI

- Um den Beendigungsschutz für einen ausgeführten Cluster über die AWS CLI zu aktivieren, verwenden Sie den Befehl `modify-cluster-attributes` mit dem Parameter `--`

`termination-protected`. Um ihn zu deaktivieren, verwenden Sie den Parameter `--no-termination-protected`.

Das folgende Beispiel aktiviert den Kündigungsschutz auf dem Cluster mit der ID *j-3KVTXXXXXX7UG*:

```
aws emr modify-cluster-attributes --cluster-id j-3KVTXXXXXX7UG --termination-protected
```

Im folgenden Beispiel wird der Beendigungsschutz für dasselbe Cluster deaktiviert:

```
aws emr modify-cluster-attributes --cluster-id j-3KVTXXXXXX7UG --no-termination-protected
```

Fehlerhafte Knoten werden ersetzt

Amazon verwendet EMR regelmäßig den NodeManager Health Checker-Service in Apache Hadoop, um den Status der Kernknoten in Ihren Amazon EMR on Amazon-Clustern zu überwachen. EC2 Wenn ein Knoten nicht optimal funktioniert, meldet der Health Checker diesen Knoten an den EMR Amazon-Controller. Der EMR Amazon-Controller fügt den Knoten einer Denylist hinzu und verhindert so, dass der Knoten neue YARN Anwendungen empfängt, bis sich der Status des Knotens verbessert. Ein häufiger Grund dafür, dass ein Knoten fehlerhaft wird, ist die Überlastung der Festplatte. Weitere Informationen zur Identifizierung fehlerhafter Knoten und zur Wiederherstellung finden Sie unter Ressourcenfehler.

Sie können wählen, ob Amazon EMR fehlerhafte Knoten beenden oder sie im Cluster belassen soll. Wenn Sie den Austausch fehlerhafter Knoten deaktivieren, bleiben die fehlerhaften Knoten in der Denylist und werden weiterhin zur Clusterkapazität gezählt. Sie können zur Konfiguration und Wiederherstellung weiterhin eine Verbindung zu Ihrer Amazon EC2 Core-Instance herstellen, sodass Sie die Größe Ihres Clusters ändern können, um Kapazität zu erhöhen. Beachten Sie, EMR dass Amazon fehlerhafte Knoten ersetzt, auch wenn der Kündigungsschutz aktiviert ist.

Wenn der Austausch fehlerhafter Knoten aktiviert ist, beendet Amazon den EMR fehlerhaften Kernknoten und stellt eine neue Instance bereit, die auf der Anzahl der Instances in der Instance-Gruppe oder der Zielkapazität für Instance-Flotten basiert. Wenn mehrere oder alle Kernknoten länger als 45 Minuten fehlerhaft sind, ersetzt Amazon EMR die Knoten ordnungsgemäß.

> **Important**
>
> Um zu vermeiden, dass HDFS Daten dauerhaft verloren gehen, wenn Amazon EMR eine fehlerhafte Core-Instance ordnungsgemäß ersetzt, empfehlen wir, dass Sie Ihre Daten immer sichern.

Amazon EMR veröffentlicht Amazon CloudWatch Events für den Austausch fehlerhafter Knoten, sodass Sie verfolgen können, was mit Ihren fehlerhaften Core-Instances passiert. Weitere Informationen finden Sie unter Ereignisse beim Austausch fehlerhafter Knoten.

Standardeinstellungen für den Austausch von Knoten und den Kündigungsschutz

Unhealthy Node Replacement ist für alle EMR Amazon-Releases verfügbar, aber die Standardeinstellungen hängen von der von Ihnen gewählten Release-Bezeichnung ab. Sie können jede dieser Einstellungen ändern, indem Sie beim Erstellen eines neuen Clusters den Austausch fehlerhafter Knoten konfigurieren oder indem Sie jederzeit zur Cluster-Konfiguration wechseln.

Wenn Sie einen Einzelknoten-Cluster oder einen Hochverfügbarkeitscluster erstellen, auf dem Amazon EMR Version 7.0 oder niedriger ausgeführt wird, hängt die Standardeinstellung für den Austausch fehlerhafter Knoten vom Kündigungsschutz ab:

- Durch die Aktivierung des Kündigungsschutzes wird der Austausch fehlerhafter Knoten deaktiviert.
- Durch die Deaktivierung des Terminierungsschutzes wird der Austausch fehlerhafter Knoten ermöglicht.

Konfiguration des Austauschs fehlerhafter Knoten beim Start eines Clusters

Sie können den Austausch fehlerhafter Knoten aktivieren oder deaktivieren, wenn Sie einen Cluster mit der Konsole AWS CLI, dem oder dem starten. API

Die Standardeinstellung für den Austausch fehlerhafter Knoten hängt davon ab, wie Sie den Cluster starten:

- EMRAmazon-Konsole — Der Austausch fehlerhafter Knoten ist standardmäßig aktiviert.
- AWS CLI `aws emr create-cluster`— Der Austausch fehlerhafter Knoten ist standardmäßig aktiviert, sofern Sie nichts anderes angeben. `--no-unhealthy-node-replacement`

- EMRRunJobFlow APIAmazon-Befehl — Austausch ungesunder Knoten ist standardmäßig aktiviert, sofern Sie den `UnhealthyNodeReplacement` booleschen Wert nicht auf oder setzen. `True` `False`

Console

Um den Austausch fehlerhafter Knoten ein oder aus zu schalten, wenn Sie mit der Konsole einen Cluster erstellen

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und anschließend Create cluster aus.
3. Wählen Sie für die EMRRelease-Version das gewünschte EMR Amazon-Release-Label aus.
4. Vergewissern Sie sich, dass unter Clusterbeendigung und Austausch von Knoten die Option Austausch fehlerhafter Knoten (empfohlen) vorausgewählt ist, oder löschen Sie die Auswahl, um sie auszuschalten.
5. Wählen Sie alle anderen Optionen aus, die für Ihren Cluster gelten.
6. Um Ihren Cluster jetzt zu starten, wählen Sie Cluster erstellen aus.

AWS CLI

Um den Austausch fehlerhafter Knoten zu aktivieren oder zu deaktivieren, wenn Sie einen Cluster mit dem AWS CLI

- Mit dem können Sie einen Cluster starten AWS CLI, bei dem der Austausch fehlerhafter Knoten mit dem `create-cluster` Befehl mit dem `--unhealthy-node-replacement` Parameter aktiviert ist. Der Austausch fehlerhafter Knoten ist standardmäßig aktiviert.

  Im folgenden Beispiel wird ein Cluster erstellt, bei dem der Austausch fehlerhafter Knoten aktiviert ist:

> **Note**
>
> Linux-Zeilenfortsetzungszeichen (\) sind aus Gründen der Lesbarkeit enthalten. Sie können entfernt oder in Linux-Befehlen verwendet werden. Entfernen Sie sie unter Windows oder ersetzen Sie sie durch ein Caret-Zeichen (^).

```
aws emr create-cluster --name "SampleCluster" --release-label emr-7.2.0 \
--applications Name=Hadoop Name=Hive Name=Pig \
--use-default-roles --ec2-attributes KeyName=myKey --instance-type m5.xlarge \
--instance-count 3 --unhealthy-node-replacement
```

Weitere Informationen zur Verwendung von EMR Amazon-Befehlen finden Sie unter EMR AWS CLI Amazon-Befehle. AWS CLI

Konfiguration eines fehlerhaften Knotenaustauschs in einem laufenden Cluster

Sie können den Austausch fehlerhafter Knoten für einen laufenden Cluster mithilfe der Konsole, der oder der ein- oder ausschalten. AWS CLI API

Console

Um den Austausch fehlerhafter Knoten für einen laufenden Cluster mit der Konsole ein- oder auszuschalten

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und dann den Cluster aus, den Sie aktualisieren möchten.
3. Suchen Sie auf der Cluster-Detailseite auf der Registerkarte Eigenschaften nach Clusterbeendigung und Knotenersatz und wählen Sie Bearbeiten aus.
4. Aktivieren oder deaktivieren Sie das Kontrollkästchen für fehlerhaften Knotenaustausch, um die Funktion ein- oder auszuschalten. Wählen Sie dann zur Bestätigung Änderungen speichern aus.

AWS CLI

Um den Austausch fehlerhafter Knoten für einen laufenden Cluster ein- oder auszuschalten, verwenden Sie den AWS CLI

- Um den Austausch fehlerhafter Knoten in einem laufenden Cluster mit dem zu aktivieren AWS CLI, verwenden Sie den `modify-cluster-attributes` Befehl mit dem `--unhealthy-node-replacement` Parameter. Um ihn zu deaktivieren, verwenden Sie den Parameter `--no-unhealthy-node-replacement`.

  Im folgenden Beispiel wird der Austausch fehlerhafter Knoten auf dem Cluster mit der ID aktiviert *`j-3KVTXXXXXX7UG`*:

  ```
  aws emr modify-cluster-attributes --cluster-id j-3KVTXXXXXX7UG --unhealthy-node-replacement
  ```

  Im folgenden Beispiel wird der Austausch fehlerhafter Knoten auf demselben Cluster deaktiviert:

  ```
  aws emr modify-cluster-attributes --cluster-id j-3KVTXXXXXX7UG --no-unhealthy-node-replacement
  ```

Arbeiten mit Amazon Linux AMIs in Amazon EMR

Amazon Linux Amazon-Maschinenabbilder (AMIs)

Amazon EMR verwendet ein Amazon Linux Amazon Machine Image (AMI), um EC2 Amazon-Instances zu initialisieren, wenn Sie einen Cluster erstellen und starten. Das AMI enthält das Amazon Linux-Betriebssystem, andere Software und die Konfigurationen, die für jede Instance zum Hosten Ihrer Cluster-Anwendungen erforderlich sind.

Wenn Sie einen Cluster erstellen, EMR verwendet Amazon standardmäßig ein standardmäßiges Amazon Linux, das speziell für AMI die von Ihnen verwendete EMR Amazon-Release-Version erstellt wurde. Weitere Informationen zum standardmäßigen Amazon Linux AMI finden Sie unterVerwenden des standardmäßigen Amazon Linux AMI für Amazon EMR. Wenn Sie Amazon EMR 5.7.0 oder höher verwenden, können Sie AMI anstelle des standardmäßigen Amazon Linux AMI für Amazon ein benutzerdefiniertes Amazon Linux angeben. EMR Eine benutzerdefinierte Option AMI ermöglicht es Ihnen, das Root-Geräte-Volume zu verschlüsseln und Anwendungen und Konfigurationen als

Alternative zur Verwendung von Bootstrap-Aktionen anzupassen. Sie können AMI für jeden Instance-Typ in der Instance-Gruppen- oder Instance-Flottenkonfiguration eines EMR Amazon-Clusters einen benutzerdefinierten Wert angeben. Die AMI Unterstützung mehrerer benutzerdefinierter Optionen gibt Ihnen die Flexibilität, mehr als einen Architekturtyp in einem Cluster zu verwenden. Siehe Verwenden Sie ein benutzerdefiniertes AMI.

Amazon fügt EMR automatisch ein Amazon EBS General SSD Purpose-Volume als Root-Gerät für alle AMIs an. EBS-unterstützt AMIs die Leistung. Weitere Informationen zu Amazon Linux AMIs finden Sie unter Amazon Machine Images (AMI). Weitere Informationen zum Instance-Speicher für EMR Amazon-Instances finden Sie unterInstance-Speicher.

Verwenden des standardmäßigen Amazon Linux AMI für Amazon EMR

Jede EMR Amazon-Release-Version verwendet ein standardmäßiges Amazon Linux AMI für Amazon, EMR sofern Sie keine benutzerdefinierte Version angebenAMI. Ab den Versionen Amazon EMR 5.36, Amazon EMR 6.6 und Amazon EMR 7.0 besteht das Standardverhalten bei der Aktualisierung von Amazon Linux 2 (AL2für EMR 5.x und 6.x, AL2 023 für EMR 7.x) in einer EMR Amazon-Standardeinstellung AMI darin, automatisch die neueste Amazon Linux-Version für das Standard-Amazon anzuwenden. EMR AMI

Automatische Amazon Linux-Updates für EMR Amazon-Releases

Wenn Sie einen Cluster mit der neuesten Patch-Version von Amazon EMR 7.0 oder höher, 6.6 oder höher oder 5.36 oder höher starten, EMR verwendet Amazon die neueste Amazon Linux-Version als Standard-Amazon EMRAMI. Beispielsweise:

- Wenn es eine Version `x.x.0` und eine `x.x.1` Version gibt, erhält die `x.x.0` Version beim `x.x.1` Start AMI keine Updates mehr.
- Ebenso werden `x.x.1` beim `x.x.2` Start AMI keine Updates mehr abgerufen.
- Später, wenn `x.y.0` es veröffentlicht `x.x.[latest]` wird, erhält es weiterhin AMI Updates`x.y.[latest]`.

Um zu sehen, ob Sie die neueste Patch-Version verwenden, die durch die Zahl nach dem zweiten Dezimalpunkt (`6.8.1`) für eine EMR Amazon-Version gekennzeichnet ist, sehen Sie sich die verfügbaren Versionen im Amazon EMR Release Guide an, überprüfen Sie das Drop-down-Menü für EMRAmazon-Versionen, wenn Sie einen Cluster in der Konsole erstellen, oder verwenden Sie die `ListReleaseLabels`APIAktion oder. `list-release-labels`CLI Um auf dem Laufenden zu

bleiben, wenn wir eine neue EMR Amazon-Version auf den Markt bringen, abonnieren Sie den RSS Feed unter Was ist neu? Seite im Versionshandbuch.

Wenn Sie möchten, können Sie Ihren Cluster mit der Amazon Linux-Version starten, mit der die EMR Amazon-Version zuerst ausgeliefert wurde. Weitere Informationen zum Spezifizieren der Amazon-Linux-Version für Ihren Cluster finden Sie unter Ändern der Amazon Linux-Version beim Erstellen eines EMR Clusters.

Standard-Amazon-Linux-Versionen

Themen

- Standard AMIs für Amazon EMR 7.0 und höher
- Standard AMIs für Amazon EMR 6.6 und höher
- Standard AMIs für Amazon EMR 5.x

Standard AMIs für Amazon EMR 7.0 und höher

In der folgenden Tabelle sind Amazon Linux-Informationen für die neueste Patch-Version der EMR Amazon-Versionen 7.0 und höher aufgeführt.

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2023.5.2 240708.(	6.1.96-102.177.amzn2023	23. Juli 2024	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• eu-central-2 • eu-south-1 • eu-south-2 • ap-east-1 • ap-south-1 • ap-south-2 • ap-southeast-3 • ap-southeast-4 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-central-1 • me-south-1 • ca-central-1 • il-central-1 • ca-west-1 • us-gov-east-1 • us-gov-west-1 • cn-north-1 • cn-northeast-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2023.3.2 240304.(	6.1.79-99.164.amzn2023	12. März 2024	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 • eu-south-1 • eu-south-2 • ap-east-1 • ap-south-1 • ap-south-2 • ap-southeast-3 • ap-southeast-4 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-central-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• me-south-1 • ca-central-1 • il-central-1 • ca-west-1 • us-gov-east-1 • us-gov-west-1 • cn-north-1 • cn-northeast-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2023.3.2 240219.(	6.1.77-99.164.amzn2023	1. März 2024	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 • eu-south-1 • eu-south-2 • ap-east-1 • ap-south-1 • ap-south-2 • ap-southeast-3 • ap-southeast-4 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-central-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• me-south-1 • ca-central-1 • il-central-1 • ca-west-1 • us-gov-east-1 • us-gov-west-1 • cn-north-1 • cn-northeast-1

OsRelea Label (AL- Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2023.3.2 240205.(	6.1.75-99.163.amzn2023	19. Februar 2024	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 • eu-south-1 • eu-south-2 • ap-east-1 • ap-south-1 • ap-south-2 • ap-southeast-3 • ap-southeast-4 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-central-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• me-south-1 • ca-central-1 • il-central-1 • ca-west-1 • us-gov-east-1 • us-gov-west-1 • cn-north-1 • cn-northeast-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2023.3.2 240122.(	6.1.72-96.amzn2023	5. Februar 2024	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 • eu-south-1 • eu-south-2 • ap-east-1 • ap-south-1 • ap-south-2 • ap-southeast-3 • ap-southeast-4 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-central-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• me-south-1 • ca-central-1 • il-central-1 • ca-west-1 • us-gov-east-1 • us-gov-west-1 • cn-north-1 • cn-northeast-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2023.3.2 240108.(	6.1.72-96.amzn2023	24. Januar 2024	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 • eu-south-1 • eu-south-2 • ap-east-1 • ap-south-1 • ap-south-2 • ap-southeast-3 • ap-southeast-4 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-central-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• me-south-1 • ca-central-1 • il-central-1 • ca-west-1 • us-gov-east-1 • us-gov-west-1 • cn-north-1 • cn-northeast-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2023.3.2 231211.4	6.1.66-91.160.amzn2023	19. Dezember 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 • eu-south-1 • eu-south-2 • ap-east-1 • ap-south-1 • ap-south-2 • ap-southeast-3 • ap-southeast-4 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-central-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• me-south-1 • ca-central-1 • il-central-1 • us-gov-east-1 • us-gov-west-1 • cn-north-1 • cn-northeast-1

Standard AMIs für Amazon EMR 6.6 und höher

In der folgenden Tabelle sind Amazon Linux-Informationen für die neueste Patch-Version der EMR Amazon-Versionen 6.6.x und höher aufgeführt.

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2024 709.1	4,14.348	23. Juli 2024	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• `eu-centra l-2` (6.10.1+) • `eu-south-1` • `eu-south-2` (6.10.1+) • `ap-east-1` • `ap-south-1` • `ap-south-2` (6.10.1+) • `ap-southeast-3` • `ap-southe ast-4` (6.8.1+ und 5.36.1) • `ap-northeast-1` • `ap-northeast-2` • `ap-northeast-3` • `ap-southeast-1` • `ap-southeast-2` • `af-south-1` • `sa-east-1` • `me-centra l-1` (6.10.1+) • `me-south-1` • `ca-central-1` • `il-central-1` (6.8.1+ und 5.36.1) • `ca-west-1` (6.9.1+ und 5.36.1) • `us-gov-east-1`

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• `us-gov-west-1` • `cn-north-1` • `cn-northeast-1`

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2024 223.0	4,14.336	8. März 2024	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-centra l-2 (6.10.1+) • eu-south-1 • eu-south-2 (6.10.1+) • ap-east-1 • ap-south-1 • ap-south-2 (6.10.1+) • ap-southeast-3 • ap-southe ast-4 (6.8.1+ und 5.36.1) • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• `sa-east-1` • `me-centra l-1` (6.10.1+) • `me-south-1` • `ca-central-1` • `il-central-1` (6.8.1+ und 5.36.1) • `ca-west-1` (6.9.1+ und 5.36.1) • `us-gov-east-1` • `us-gov-west-1` • `cn-north-1` • `cn-northeast-1`

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2024 131.0	4,14.336	14. Februar 2024	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-centra l-2 (6.10.1+) • eu-south-1 • eu-south-2 (6.10.1+) • ap-east-1 • ap-south-1 • ap-south-2 (6.10.1+) • ap-southeast-3 • ap-southe ast-4 (6.8.1+ und 5.36.1) • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• `sa-east-1` • `me-centra l-1` (6.10.1+) • `me-south-1` • `ca-central-1` • `il-central-1` (6.8.1+ und 5.36.1) • `ca-west-1` (6.9.1+ und 5.36.1) • `us-gov-east-1` • `us-gov-west-1` • `cn-north-1` • `cn-northeast-1`

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2024 124.0	4,14.336	7. Februar 2024	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-centra l-2 (6.10.1+) • eu-south-1 • eu-south-2 (6.10.1+) • ap-east-1 • ap-south-1 • ap-south-2 (6.10.1+) • ap-southeast-3 • ap-southe ast-4 (6.8.1+ und 5.36.1) • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• `sa-east-1` • `me-centra l-1` (6.10.1+) • `me-south-1` • `ca-central-1` • `il-central-1` (6.8.1+ und 5.36.1) • `ca-west-1` (6.9.1+ und 5.36.1) • `us-gov-east-1` • `us-gov-west-1` • `cn-north-1` • `cn-northeast-1`

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2024 109.0	4,14.334	24. Januar 2024	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-centra l-2 (6.10.1+) • eu-south-1 • eu-south-2 (6.10.1+) • ap-east-1 • ap-south-1 • ap-south-2 (6.10.1+) • ap-southeast-3 • ap-southe ast-4 (6.8.1+ und 5.36.1) • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• `sa-east-1` • `me-centra l-1` (6.10.1+) • `me-south-1` • `ca-central-1` • `il-central-1` (6.8.1+ und 5.36.1) • `ca-west-1` (6.9.1+ und 5.36.1) • `us-gov-east-1` • `us-gov-west-1` • `cn-north-1` • `cn-northeast-1`

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2023 218.0	4,14.330	2. Januar 2024	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 (6.10+) • eu-south-1 • eu-south-2 (6.10+) • ap-east-1 • ap-south-1 • ap-south-2 (6.10+) • ap-southeast-3 • ap-southeast-4 (6.8+ und 5.36.1) • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• `me-central-1` (6.10+) • `me-south-1` • `ca-central-1` • `il-central-1` (6.8+ und 5.36.1) • `us-gov-east-1` • `us-gov-west-1` • `cn-north-1` • `cn-northeast-1`

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2023 206.0	4,14.330	22. Dezember 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 (6.10+) • eu-south-1 • eu-south-2 (6.10+) • ap-east-1 • ap-south-1 • ap-south-2 (6.10+) • ap-southeast-3 • ap-southeast-4 (6.8+ und 5.36.1) • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• `me-central-1` (6.10+) • `me-south-1` • `ca-central-1` • `il-central-1` (6.8+ und 5.36.1) • `us-gov-east-1` • `us-gov-west-1` • `cn-north-1` • `cn-northeast-1`

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2023 116.0	4,14.328	11. Dezember 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 (6.10+) • eu-south-1 • eu-south-2 (6.10+) • ap-east-1 • ap-south-1 • ap-south-2 (6.10+) • ap-southeast-3 • ap-southeast-4 (6.8+ und 5.36.1) • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• `me-central-1` (6.10+) • `me-south-1` • `ca-central-1` • `il-central-1` (6.8+ und 5.36.1) • `us-gov-east-1` • `us-gov-west-1` • `cn-north-1` • `cn-northeast-1`

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2023 101.0	4,14.327	17. November 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 (6.10+) • eu-south-1 • eu-south-2 (6.10+) • ap-east-1 • ap-south-1 • ap-south-2 (6.10+) • ap-southeast-3 • ap-southeast-4 (6.8+ und 5.36.1) • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• `me-central-1` (6.10+) • `me-south-1` • `ca-central-1` • `il-central-1` (6.8+ und 5.36.1) • `us-gov-east-1` • `us-gov-west-1` • `cn-north-1` • `cn-northeast-1`

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2023 020.1	4,14.326	07. November 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 (6.10+) • eu-south-1 • eu-south-2 (6.10+) • ap-east-1 • ap-south-1 • ap-south-2 (6.10+) • ap-southeast-3 • ap-southeast-4 (6.8+ und 5.36.1) • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• `me-central-1` (6.10+) • `me-south-1` • `ca-central-1` • `il-central-1` (6.8+ und 5.36.1) • `us-gov-east-1` • `us-gov-west-1` • `cn-north-1` • `cn-northeast-1`

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2023 012.1	4,14.326	26. Oktober 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 (6.10+) • eu-south-1 • eu-south-2 (6.10+) • ap-east-1 • ap-south-1 • ap-south-2 (6.10+) • ap-southeast-3 • ap-southeast-4 (6.8+ und 5.36.1) • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• `me-central-1` (6.10+) • `me-south-1` • `ca-central-1` • `il-central-1` (6.8+ und 5.36.1) • `us-gov-east-1` • `us-gov-west-1` • `cn-north-1` • `cn-northeast-1`

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2023 926.0	4,14.322	19. Oktober 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 (6.10+) • eu-south-1 • eu-south-2 (6.10+) • ap-east-1 • ap-south-1 • ap-south-2 (6.10+) • ap-southeast-3 • ap-southeast-4 (6.8+ und 5.36.1) • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• `me-central-1` (6.10+) • `me-south-1` • `ca-central-1` • `il-central-1` (6.8+ und 5.36.1) • `us-gov-east-1` • `us-gov-west-1` • `cn-north-1` • `cn-northeast-1`

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2023 8906,0	4,14.322	04. Oktober 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 (6.10+) • eu-south-1 • eu-south-2 (6.10+) • ap-east-1 • ap-south-1 • ap-south-2 (6.10+) • ap-southeast-3 • ap-southeast-4 (6.8+ und 5.36.1) • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• `me-central-1` (6.10+) • `me-south-1` • `ca-central-1` • `il-central-1` (6.9+ und 5.36.1)

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2023 822.0	4,14.322	30. August 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 (6.10+) • eu-south-1 • eu-south-2 (6.10+) • ap-east-1 • ap-south-1 • ap-south-2 (6.10+) • ap-southeast-3 • ap-southeast-4 (6.8+ und 5.36.1) • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• `me-central-1` (6.10+) • `me-south-1` • `ca-central-1` • `il-central-1` (6.9+ und 5.36.1)

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2023 808,0	4,14.320	24. August 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 (6.10+) • eu-south-1 • eu-south-2 (6.10+) • ap-east-1 • ap-south-1 • ap-south-2 (6.10+) • ap-southeast-3 • ap-southeast-4 (6.8+ und 5.36.1) • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1

OsRelease Label (AL-Version)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• `me-central-1` (6.10+) • `me-south-1` • `ca-central-1` • `il-central-1` (6.9+ und 5.36.1) • `us-gov-east-1` • `us-gov-west-1` • `cn-north-1` • `cn-northeast-1`

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2023 727.0	4,14.320	14. August 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 (6.10+) • eu-south-1 • eu-south-2 (6.10+) • ap-east-1 • ap-south-1 • ap-south-2 (6.10+) • ap-southeast-3 • ap-southeast-4 (6.8+ und 5.36.1) • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• `me-central-1` (6.10+) • `me-south-1` • `ca-central-1` • `il-central-1` (6.9+ und 5.36.1)

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2023 719,0	4,14.320	02. August 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 (6.10+) • eu-south-1 • eu-south-2 (6.10+) • ap-east-1 • ap-south-1 • ap-south-2 (6.10+) • ap-southeast-3 • ap-southeast-4 (6.8+ und 5.36.1) • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• `me-central-1` (6.10+) • `me-south-1` • `ca-central-1` • `il-central-1` (6.9+ und 5.36.1)

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2023 628.0	4,14.318	12. Juli 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 (6.10+) • eu-south-1 • eu-south-2 (6.10+) • ap-east-1 • ap-south-1 • ap-south-2 (6.10+) • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-central-1 (6.10+)

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• me-south-1 • ca-central-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2023 612,0	4,14.314	23. Juni 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 (6.10+) • eu-south-1 • eu-south-2 (6.10+) • ap-east-1 • ap-south-1 • ap-south-2 (6.10+) • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-central-1 (6.10+)

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• `me-south-1` • `ca-central-1`

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2023 504.1	4,14.313	16. Mai 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 (6.10+) • eu-south-1 • eu-south-2 (6.10+) • ap-east-1 • ap-south-1 • ap-south-2 (6.10+) • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-central-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• me-south-1 • ca-central-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2023 418,0	4,14.311	3. Mai 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 (nur 6.10) • eu-south-1 • eu-south-2 (nur 6.10) • ap-east-1 • ap-south-1 • ap-south-2 (nur 6.10) • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• me-central-1 • me-south-1 • ca-central-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2023 404.1	4,14.311	18. April 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 • eu-south-1 • eu-south-2 • ap-east-1 • ap-south-1 • ap-south-2 • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-central-1 • me-south-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• `ca-central-1`
2.0.2023 404.0	4,14.311	10. April 2023	• `us-east-1` • `eu-west-3`

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2023 320.0	4,14.309	30. März 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 • eu-south-1 • eu-south-2 • ap-east-1 • ap-south-1 • ap-south-2 • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-central-1 • me-south-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• `ca-central-1`

OsRelease Label (AL-Version)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.20230307,0	4,14.305	15. März 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 • eu-south-1 • eu-south-2 • ap-east-1 • ap-south-1 • ap-south-2 • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-central-1 • me-south-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• `ca-central-1`

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2023 207,0	4,14.304	03. März 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 • eu-south-1 • eu-south-2 • ap-east-1 • ap-south-1 • ap-south-2 • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-central-1 • me-south-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• ca-central-1
2.0.2023 119.1	4,14.301	9. Februar 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-south-1 • ap-east-1 • ap-south-1 • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-south-1 • ca-central-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2022 210.1	4.14.301	12. Januar 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-south-1 • ap-east-1 • ap-south-1 • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-south-1 • ca-central-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2022 103.3	4,14.296	5. Dezember 2022	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-south-1 • ap-east-1 • ap-south-1 • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-south-1 • ca-central-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2022 004,0	4,14.294	02. November 2022	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-south-1 • ap-east-1 • ap-south-1 • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-south-1 • ca-central-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2022 912.1	4,14.291	7. Oktober 2022	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-south-1 • ap-east-1 • ap-south-1 • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-south-1 • ca-central-1
2.0.2022 805.0	4,14.287	30. August 2022	• us-west-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2022 719.0	4,14.287	10. August 2022	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-south-1 • ap-east-1 • ap-south-1 • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-south-1 • ca-central-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2022 426,0	4,14.281	10. Juni 2022	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-south-1 • ap-east-1 • ap-south-1 • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-south-1 • ca-central-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2022 406.1	4,14.275	2. Mai 2022	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-south-1 • ap-east-1 • ap-south-1 • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-south-1 • ca-central-1

Standard AMIs für Amazon EMR 5.x

In der folgenden Tabelle sind Amazon Linux-Informationen für die neueste Patch-Version der Amazon EMR 5.x-Versionen 5.36 und höher aufgeführt.

OsRelease Label (AL-Version)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.20240709.1	4,14.348	23. Juli 2024	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-central-2 (6.10.1+) • eu-south-1 • eu-south-2 (6.10.1+) • ap-east-1 • ap-south-1 • ap-south-2 (6.10.1+) • ap-southeast-3 • ap-southeast-4 (6.8.1+ und 5.36.1) • ap-northeast-1 • ap-northeast-2

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
			• ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-centra l-1 (6.10.1+) • me-south-1 • ca-central-1 • il-central-1 (6.8.1+ und 5.36.1) • ca-west-1 (6.9.1+ und 5.36.1) • us-gov-east-1 • us-gov-west-1 • cn-north-1 • cn-northeast-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2023 504.1	4,14.313	16. Mai 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • ca-central-1 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-south-1 • ap-east-1 • ap-south-1 • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-south-1 • me-central-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2023 418,0	4,14.311	3. Mai 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • ca-central-1 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-south-1 • ap-east-1 • ap-south-1 • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-south-1 • me-central-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2023 404.1	4,14.311	18. April 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • ca-central-1 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-south-1 • ap-east-1 • ap-south-1 • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-south-1
2.0.2023 404.0	4,14.311	10. April 2023	• us-east-1 • eu-west-3

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2023 320.0	4,14.309	30. März 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • ca-central-1 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-south-1 • ap-east-1 • ap-south-1 • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-south-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2023 307,0	4,14.305	15. März 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • ca-central-1 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-south-1 • ap-east-1 • ap-south-1 • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-south-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2023 207,0	4,14.304	03. März 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-south-1 • ap-east-1 • ap-south-1 • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-south-1 • ca-central-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2022 210.1	4.14.301	12. Januar 2023	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-south-1 • ap-east-1 • ap-south-1 • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-south-1 • ca-central-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2022 103.3	4,14.296	5. Dezember 2022	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-south-1 • ap-east-1 • ap-south-1 • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-south-1 • ca-central-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2022 004,0	4,14.294	02. November 2022	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-south-1 • ap-east-1 • ap-south-1 • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-south-1 • ca-central-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2022 912.1	4,14.291	7. Oktober 2022	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-south-1 • ap-east-1 • ap-south-1 • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-south-1 • ca-central-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2022 719.0	4,14.287	10. August 2022	• us-west-1 • eu-west-3 • eu-north-1 • eu-central-1 • ap-south-1 • me-south-1

OsRelea Label (AL-Versi on)	AL-Kernel-Version	Verfügbarkeitsdatum	AWS-Regionen
2.0.2022 426,0	4,14.281	14. Juni 2022	• us-east-1 • us-east-2 • us-west-1 • us-west-2 • eu-north-1 • eu-west-1 • eu-west-2 • eu-west-3 • eu-central-1 • eu-south-1 • ap-east-1 • ap-south-1 • ap-southeast-3 • ap-northeast-1 • ap-northeast-2 • ap-northeast-3 • ap-southeast-1 • ap-southeast-2 • af-south-1 • sa-east-1 • me-south-1 • ca-central-1

Überlegungen zu Softwareupdates

Beachten Sie die folgenden Standardverhalten für Softwareupdates:

Amazon EMR 7.x — Amazon Linux 2023

EMRAmazon-Versionen 7.0 und höher laufen auf Amazon Linux 2023 (AL2023). Das Standardverhalten für AL2 023 ist das Sperren AMIs auf eine bestimmte Version des Amazon Linux-Software-Repositorys. Daher werden Sicherheitsupdates nicht bei jedem Start eines Clusters angewendet. Stattdessen ist das Standardverhalten für Amazon EMR 7.x-Versionen so, dass die neueste Version AL2 023 für die Standard-Amazon-Version EMR AMI nur dann automatisch angewendet wird, wenn Sie den Cluster erstellen. Um die neuesten Sicherheitsupdates zu erhalten, empfehlen wir Ihnen, Ihren Cluster regelmäßig neu zu erstellen.

Amazon EMR 5.x und 6.x — Amazon Linux und Amazon Linux 2

Wenn bei EMR Amazon-Versionen unter 7.0 eine EC2 Amazon-Instance zum ersten Mal in einem Cluster gestartet wird, der auf dem standardmäßigen Amazon Linux (AL) oder Amazon Linux 2 (AL2) AMI für Amazon basiertEMR, sucht sie in den aktivierten Paket-Repositorys für AL und Amazon EMR nach Softwareupdates, die für die Release-Version gelten. Wie bei anderen AL und AL2 Instances werden kritische und wichtige Sicherheitsupdates aus diesen Repositorys automatisch installiert.

Beachten Sie außerdem, dass Sie in Ihrer Netzwerkkonfiguration Amazon Linux-Repositorys in Amazon S3 zulassen HTTP und HTTPS auf diese zugreifen müssen. Andernfalls schlagen Sicherheitsupdates fehl. Weitere Informationen finden Sie unter Amazon Linux — Package Repository im EC2Amazon-Benutzerhandbuch. Standardmäßig sind andere Softwarepakete und Kernel-Updates, die einen Neustart erfordernCUDA, einschließlich NVIDIA und, vom automatischen Download beim ersten Start ausgeschlossen.

Amazon EMR 5.35.0 und niedriger und 6.5.0 und niedriger — Amazon Linux ist an die AMI Amazon-Release-Version gebunden EMR

Für Amazon EMR 5.35.0 und niedriger sowie 6.5.0 und niedriger AMI basiert die Standardeinstellung auf den meisten up-to-date Amazon-Linux-Versionen, die zum Zeitpunkt der Amazon-Veröffentlichung AMI verfügbar waren. EMR AMIDas wurde auf Kompatibilität mit den Big-Data-Anwendungen und EMR Amazon-Funktionen getestet, die in dieser Release-Version enthalten sind.

Jede EMR Amazon-Release-Version EMR 5.35.0 und niedriger sowie 6.5.0 und niedriger von Amazon ist aus Kompatibilitätsgründen an die jeweils zugewiesene Amazon AMI Linux-Version „gesperrt“. Aus diesem Grund empfehlen wir Ihnen, die neueste EMR Amazon-Release-Version zu verwenden, es sei denn, Sie benötigen aus Kompatibilitätsgründen eine niedrigere Version und können nicht migrieren. Wenn Sie aus EMR Kompatibilitätsgründen eine niedrigere Version von Amazon verwenden müssen, empfehlen wir Ihnen, die neueste Version einer Serie zu verwenden.

Wenn Sie beispielsweise die Reihe 5.12 verwenden müssen, sollten Sie 5.12.2 und nicht 5.12.0 oder 5.12.1 verwenden. Wenn in einer Reihe eine neue Version verfügbar wird, sollten Sie eine Migration Ihrer Anwendungen auf die neue Version in Betracht ziehen.

Weitere Informationen zum Verhalten bei automatischen Updates, das mit Amazon EMR 5.36.0 und höher und 6.6.0 und höher eingeführt wurde, finden Sie unter. Automatische Amazon Linux-Updates für EMR Amazon-Releases

Das standardmäßige Startverhalten schließt Kernel-Updates aus

Wenn eine EC2 Amazon-Instance in einem Cluster, der auf dem standardmäßigen Amazon Linux AMI für Amazon basiert, zum ersten Mal EMR startet, überprüft sie die aktivierten Paket-Repositorys für Amazon Linux und Amazon auf Softwareupdates, die EMR für die AMI Version gelten. Wie bei anderen EC2 Amazon-Instances werden kritische und wichtige Sicherheitsupdates aus diesen Repositorys automatisch installiert.

Wenn Sie jedoch eine ältere Version von Amazon Linux verwendenAMI, wird das neueste Sicherheitsupdate möglicherweise nicht automatisch installiert. Das liegt daran, dass die Repositorys, auf die Ihr EMR Cluster verweist, für jede Version von Amazon Linux AMI festgelegt sind.

Beachten Sie außerdem, dass Sie in Ihrer Netzwerkkonfiguration Amazon Linux-Repositorys in Amazon S3 zulassen HTTP und HTTPS auf diese zugreifen müssen. Andernfalls schlagen Sicherheitsupdates fehl. Weitere Informationen finden Sie unter Amazon Linux — Package Repository im EC2Amazon-Benutzerhandbuch. Standardmäßig sind andere Softwarepakete und Kernel-Updates, die einen Neustart erfordernCUDA, einschließlich NVIDIA und, vom automatischen Download beim ersten Start ausgeschlossen.

> **Important**
>
> EMRCluster, auf denen AL2 023 ausgeführt wird, verwenden das Standardverhalten von Amazon Linux, und Ihre Amazon Machine Images (AMIs) sind an eine bestimmte Version des Amazon Linux-Repositorys gebunden. Standardmäßig erhalten Ihre Cluster beim Start nicht automatisch Software-Sicherheitsupdates. Ihre Cluster enthalten nur die Updates, die in der Version AL2 023 verfügbar warenAMI, die Sie bei der Erstellung Ihres Clusters ausgewählt haben. Weitere Informationen finden Sie unter Aktualisieren von Amazon Linux 2023 im Benutzerhandbuch für Amazon Linux 2023.

> **Important**
>
> EMRCluster, auf denen Amazon Linux oder Amazon Linux 2 Amazon Machine Images (AMIs) ausgeführt werden, verwenden das Standardverhalten von Amazon Linux und laden wichtige und kritische Kernel-Updates, die einen Neustart erfordern, nicht automatisch herunter und installieren sie. Dies ist dasselbe Verhalten wie bei anderen EC2 Amazon-Instances, auf denen das standardmäßige Amazon Linux ausgeführt wirdAMI. Wenn neue Amazon Linux-Softwareupdates, die einen Neustart erfordern (wie Kernel und CUDA Updates)NVIDIA, verfügbar werden, nachdem eine EMR Amazon-Version verfügbar wird, laden EMR Cluster-Instances, die standardmäßig ausgeführt werden, diese Updates AMI nicht automatisch herunter und installieren sie. Um Kernel-Updates zu erhalten, können Sie Ihr Amazon so anpassen EMR AMI, dass es das neueste Amazon Linux verwendet AMI.

Der Cluster wird mit oder ohne Updates gestartet

Beachten Sie, dass die Cluster-Instance ihren Start trotzdem abschließt, wenn Softwareupdates nicht installiert werden können, weil Paket-Repositorys beim ersten Clusterstart nicht erreichbar sind. Beispielsweise sind Repositorys möglicherweise nicht erreichbar, weil S3 vorübergehend nicht verfügbar ist, oder Sie haben VPC möglicherweise Firewall-Regeln so konfiguriert, dass sie den Zugriff blockieren.

`sudo yum update` nicht ausführen

Wenn Sie eine Verbindung zu einer Cluster-Instance herstellenSSH, enthalten die ersten Zeilen der Bildschirmausgabe einen Link zu den Versionshinweisen für das Amazon Linux, AMI das die Instance verwendet, einen Hinweis auf die neueste Amazon AMI Linux-Version, einen Hinweis auf die Anzahl der Pakete, die für Updates aus den aktivierten Repositorys verfügbar sind, und eine Anweisung zur Ausführung`sudo yum update`.

> **Important**
>
> Wir empfehlen dringend, dass Sie die Ausführung nicht `sudo yum update` auf Cluster-Instances durchführen, weder während Sie mit einer Bootstrap-Aktion SSH verbunden sind noch wenn Sie eine Bootstrap-Aktion verwenden. Dies kann zu Inkompatibilitäten führen, da alle Pakete unterschiedslos installiert werden.

Bewährte Methoden für Softwareupdates

Bewährte Methoden für die Verwaltung von Software-Updates

- Wenn Sie eine niedrigere Version von Amazon verwendenEMR, sollten Sie eine Migration auf die neueste Version in Betracht ziehen und testen, bevor Sie Softwarepakete aktualisieren.
- Wenn Sie auf eine höhere Version migrieren oder Softwarepakete upgraden, sollten Sie die Implementierung zunächst in einer Umgebung außerhalb der Produktion testen. Die Option, Cluster mit der EMR Amazon-Konsole zu klonen, ist dafür hilfreich.
- Evaluieren Sie Softwareupdates für Ihre Anwendungen und für Ihre Version von Amazon Linux AMI auf individueller Basis. Testen und installieren Sie nur Pakete in Produktionsumgebungen, die Ihrer Meinung nach für Sicherheit, Anwendungsfunktionalität oder Leistung unbedingt notwendig sind.
- Achten Sie im Amazon-Linux-Sicherheitszentrum auf Updates.
- Vermeiden Sie die Installation von Paketen, indem Sie mithilfe von... eine Verbindung zu einzelnen Cluster-Instances herstellenSSH. Verwenden Sie stattdessen eine Bootstrap-Aktion, um Pakete auf allen Cluster-Instances wie notwendig zu installieren und zu aktualisieren. Hierzu müssen Sie einen Cluster beenden und neu starten. Weitere Informationen finden Sie unter Erstellen von Bootstrap-Aktionen zur Installation zusätzlicher Software.

Verwenden Sie ein benutzerdefiniertes AMI

Wenn Sie Amazon EMR 5.7.0 oder höher verwenden, können Sie AMI anstelle des standardmäßigen Amazon Linux AMI für Amazon ein benutzerdefiniertes Amazon Linux angeben. EMR Ein benutzerdefinierter AMI Code ist nützlich, wenn Sie Folgendes tun möchten:

- Installieren Sie Anwendungen vorab und führen Sie weitere Anpassungen aus, statt Bootstrap-Aktionen zu verwenden. Dies kann die Cluster-Startzeit verbessern und den Startup-Workflow optimieren. Weitere Informationen sowie ein Beispiel finden Sie unter Ein benutzerdefiniertes Amazon Linux AMI aus einer vorkonfigurierten Instance erstellen.
- Implementierung komplexerer Cluster- und Knoten-Konfigurationen als von Bootstrap-Aktionen zugelassen.
- Verschlüsseln Sie die EBS Root-Geräte-Volumes (Boot-Volumes) der EC2 Instances in Ihrem Cluster, wenn Sie eine EMR Amazon-Version unter 5.24.0 verwenden. Wie bei der Standardeinstellung AMI beträgt die Mindestgröße des Root-Volumes für ein benutzerdefiniertes AMI Volume 10 GiB für EMR Amazon-Versionen 6.9 und niedriger und 15 GiB für EMR Amazon-

Versionen 6.10 und höher. Weitere Informationen finden Sie unter Benutzerdefiniertes Volume AMI mit einem verschlüsselten EBS Amazon-Root-Geräte-Volume erstellen.

> **Note**
>
> Ab EMR Amazon-Version 5.24.0 können Sie eine Sicherheitskonfigurationsoption verwenden, um EBS Root-Geräte und Speichervolumes zu verschlüsseln, wenn Sie dies AWS KMS als Ihren Schlüsselanbieter angeben. Weitere Informationen finden Sie unter Verschlüsselung lokaler Datenträger.

Ein benutzerdefinierter AMI Code muss in derselben AWS Region existieren, in der Sie den Cluster erstellen. Es sollte auch der EC2 Instanzarchitektur entsprechen. Eine m5.xlarge-Instance hat beispielsweise eine x86_64-Architektur. Um eine m5.xlarge mithilfe einer benutzerdefinierten Datei bereitzustellenAMI, AMI sollte Ihre benutzerdefinierte Datei daher auch über eine x86_64-Architektur verfügen. Um eine m6g.xlarge-Instanz bereitzustellen, die über eine arm64-Architektur verfügt, sollte Ihre benutzerdefinierte Instanz ebenfalls über eine arm64-Architektur verfügen. AMI Weitere Informationen zur Identifizierung eines Linux AMI für Ihren Instance-Typ finden Sie unter Find a Linux AMI im EC2Amazon-Benutzerhandbuch.

> **Important**
>
> EMRCluster, auf denen Amazon Linux oder Amazon Linux 2 Amazon Machine Images (AMIs) ausgeführt werden, verwenden das Standardverhalten von Amazon Linux und laden wichtige und kritische Kernel-Updates, die einen Neustart erfordern, nicht automatisch herunter und installieren sie. Dies ist dasselbe Verhalten wie bei anderen EC2 Amazon-Instances, auf denen das standardmäßige Amazon Linux ausgeführt wirdAMI. Wenn neue Amazon Linux-Softwareupdates, die einen Neustart erfordern (wie Kernel und CUDA Updates)NVIDIA, verfügbar werden, nachdem eine EMR Amazon-Version verfügbar wird, laden EMR Cluster-Instances, die standardmäßig ausgeführt werden, diese Updates AMI nicht automatisch herunter und installieren sie. Um Kernel-Updates zu erhalten, können Sie Ihr Amazon so anpassen EMR AMI, dass es das neueste Amazon Linux verwendet AMI.

Ein benutzerdefiniertes Amazon Linux AMI aus einer vorkonfigurierten Instance erstellen

Die grundlegenden Schritte zur Vorinstallation von Software und zur Durchführung anderer Konfigurationen zur Erstellung eines benutzerdefinierten Amazon Linux AMI für Amazon EMR lauten wie folgt:

- Starten Sie eine Instance vom Amazon Linux-Basisserver ausAMI.
- Stellen Sie eine Verbindung mit der Instance her, um Software zu installieren und andere Anpassungen vorzunehmen.
- Erstellen Sie ein neues Image (AMISnapshot) der von Ihnen konfigurierten Instance.

Nachdem Sie das Abbild auf der Grundlage Ihrer benutzerdefinierten Instance erstellt haben, können Sie es auf ein verschlüsseltes Ziel kopieren, wie im Abschnitt Benutzerdefiniertes Volume AMI mit einem verschlüsselten EBS Amazon-Root-Geräte-Volume erstellen beschrieben.

Tutorial: AMI Aus einer Instanz erstellen, auf der benutzerdefinierte Software installiert ist

Um eine EC2 Instance zu starten, die auf dem neuesten Amazon Linux basiert AMI

1. Verwenden Sie den AWS CLI , um den folgenden Befehl auszuführen, der eine Instance aus einer vorhandenen erstelltAMI. *MyKeyName*Ersetzen Sie durch das key pair, das Sie für die Verbindung mit der Instance verwenden, und *MyAmiId* mit der ID eines entsprechenden Amazon LinuxAMI. Die neuesten AMI IDs Informationen finden Sie unter Amazon Linux AMI.

   > **Note**
   >
   > Linux-Zeilenfortsetzungszeichen (\) sind aus Gründen der Lesbarkeit enthalten. Sie können entfernt oder in Linux-Befehlen verwendet werden. Entfernen Sie sie unter Windows oder ersetzen Sie sie durch ein Caret-Zeichen (^).

   ```
   aws ec2 run-instances --image-id MyAmiID \
   --count 1 --instance-type m5.xlarge \
   --key-name MyKeyName --region us-west-2
   ```

   Der Ausgabewert `InstanceId` wird im nächsten Schritt als *MyInstanceId* verwendet.

2. Führen Sie den folgenden Befehl aus:

```
aws ec2 describe-instances --instance-ids MyInstanceId
```

Der Ausgabewert `PublicDnsName` wird im nächsten Schritt verwendet, um eine Verbindung mit der Instance herzustellen.

So stellen Sie eine Verbindung mit der Instance her und installieren Software

1. Verwenden Sie eine SSH Verbindung, mit der Sie Shell-Befehle auf Ihrer Linux-Instance ausführen können. Weitere Informationen finden Sie unter Herstellen einer Verbindung zu Ihrer Linux-Instance mithilfe SSH im EC2Amazon-Benutzerhandbuch.
2. Führen Sie alle erforderlichen Anpassungen durch. Beispielsweise:

```
sudo yum install MySoftwarePackage
sudo pip install MySoftwarePackage
```

So erstellen Sie einen Snapshot vom benutzerdefinierten Abbild

- Nachdem Sie die Instance angepasst haben, verwenden Sie den `create-image` Befehl, um eine AMI aus der Instance zu erstellen.

```
aws ec2 create-image --no-dry-run --instance-id MyInstanceId --name MyEmrCustomAmi
```

Der Ausgabewert `imageID` wird verwendet, wenn Sie den Cluster starten oder einen verschlüsselten Snapshot erstellen. Weitere Informationen erhalten Sie unter Verwenden Sie einen einzelnen benutzerdefinierten AMI Code in einem EMR Cluster und Benutzerdefiniertes Volume AMI mit einem verschlüsselten EBS Amazon-Root-Geräte-Volume erstellen.

So verwenden Sie ein benutzerdefiniertes Objekt AMI in einem EMR Amazon-Cluster

Sie können einen Custom verwendenAMI, um einen EMR Amazon-Cluster auf zwei Arten bereitzustellen:

- Verwenden Sie einen einzigen benutzerdefinierten Code AMI für alle EC2 Instances im Cluster.

- Verwenden Sie unterschiedliche benutzerdefinierte AMIs Einstellungen für die verschiedenen EC2 Instanztypen, die im Cluster verwendet werden.

Sie können bei der Bereitstellung eines EMR Clusters nur eine der beiden Optionen verwenden, und Sie können sie nicht mehr ändern, nachdem der Cluster gestartet wurde.

Überlegungen zur Verwendung von einzelnen oder mehreren benutzerdefinierten Optionen AMIs in einem EMR Amazon-Cluster

Überlegungen	Einzeln, benutzerdefiniert AMI	Mehrfach benutzerdefiniert AMIs
Verwenden Sie sowohl x86- als auch Graviton2-Prozessoren mit Custom AMIs im selben Cluster	✗ Nicht unterstützt	✓ Wird unterstützt
AMIDie Anpassung variiert je nach Instanztyp	✗ Nicht unterstützt	✓ Wird unterstützt
Ändern Sie die benutzerdefinierte AMIs Einstellung, wenn Sie einem laufenden Cluster neue Task-Instanz-Gruppen/-Flotten hinzufügen. Hinweis: Sie können den Benutzerstandard vorhandener AMI Instanzgruppen/Flotten nicht ändern.	✗ Nicht unterstützt	✓ Wird unterstützt
Verwenden Sie die AWS Konsole, um einen Cluster zu starten	✓ Wird unterstützt	✗ Nicht unterstützt
Wird verwendet AWS CloudFormation , um einen Cluster zu starten	✓ Wird unterstützt	✓ Wird unterstützt

Verwenden Sie einen einzelnen benutzerdefinierten AMI Code in einem EMR Cluster

Verwenden Sie eine der folgenden Methoden, um bei der Erstellung eines Clusters eine benutzerdefinierte AMI ID anzugeben:

- AWS Management Console
- AWS CLI
- Amazon EMR SDK
- Amazon EMR API RunJobFlow
- AWS CloudFormation (siehe die `CustomAmiID` Eigenschaft unter Cluster InstanceGroupConfig, Cluster InstanceTypeConfig InstanceGroupConfig, Ressource oder Ressource InstanceFleetConfig - InstanceTypeConfig)

Amazon EMR console

Um einen einzelnen benutzerdefinierten Wert AMI von der Konsole aus anzugeben

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und anschließend Create cluster aus.
3. Suchen Sie unter Name und Anwendungen nach Betriebssystemoptionen. Wählen Sie Benutzerdefiniert AMI und geben Sie Ihre AMI ID in das AMI Feld Benutzerdefiniert ein.
4. Wählen Sie alle anderen Optionen aus, die für Ihren Cluster gelten.
5. Um Ihren Cluster jetzt zu starten, wählen Sie Cluster erstellen aus.

AWS CLI

Um einen einzelnen benutzerdefinierten Wert anzugeben, AMI verwenden Sie AWS CLI

- Verwenden Sie den `--custom-ami-id` Parameter, um die AMI ID anzugeben, wenn Sie den `aws emr create-cluster` Befehl ausführen.

  Das folgende Beispiel spezifiziert einen Cluster, der ein einzelnes benutzerdefiniertes AMI mit einem 20-GiB-Startvolume verwendet. Weitere Informationen finden Sie unter Anpassen des EBS Amazon-Root-Geräte-Volumes.

> **Note**
>
> Linux-Zeilenfortsetzungszeichen (\) sind aus Gründen der Lesbarkeit enthalten. Sie können entfernt oder in Linux-Befehlen verwendet werden. Entfernen Sie sie unter Windows oder ersetzen Sie sie durch ein Caret-Zeichen (^).

```
aws emr create-cluster --name "Cluster with My Custom AMI" \
--custom-ami-id MyAmiID --ebs-root-volume-size 20 \
--release-label emr-5.7.0 --use-default-roles \
--instance-count 2 --instance-type m5.xlarge
```

Verwenden Sie mehrere benutzerdefinierte AMIs in einem EMR Amazon-Cluster

Um einen Cluster mit mehreren benutzerdefinierten Clustern zu erstellenAMIs, verwenden Sie eine der folgenden Optionen:

- AWS CLIVersion 1.20.21 oder höher
- AWS SDK
- Amazon EMR RunJobFlowin der EMRAPIAmazon-Referenz
- AWS CloudFormation (siehe die `CustomAmiID` Eigenschaft unter Cluster InstanceGroupConfig, Cluster InstanceTypeConfig InstanceGroupConfig, Ressource oder Ressource InstanceFleetConfig - InstanceTypeConfig)

Die AWS Management Console unterstützt derzeit nicht die Erstellung eines Clusters mit mehreren benutzerdefinierten ClusternAMIs.

Example — Verwenden Sie den AWS CLI, um einen Instanzgruppen-Cluster mit mehreren benutzerdefinierten Clustern zu erstellen AMIs

Mit der AWS CLI Version 1.20.21 oder höher können Sie dem gesamten Cluster einen einzelnen benutzerdefinierten AMI Code zuweisen, oder Sie können jedem Instanzknoten in Ihrem Cluster mehrere benutzerdefinierte AMIs Knoten zuweisen.

Das folgende Beispiel zeigt einen einheitlichen Instance-Gruppen-Cluster, der mit zwei Instance-Typen (m5.xlarge) erstellt wurde, die für alle Knotentypen (Primär, Core, Aufgabe) verwendet werden. Jeder Knoten hat mehrere benutzerdefinierte Knoten. AMIs Das Beispiel veranschaulicht mehrere Funktionen der mehrfachen benutzerdefinierten AMI Konfiguration:

- Auf Clusterebene wurde kein benutzerdefinierter Wert AMI zugewiesen. Dadurch sollen Konflikte zwischen mehreren benutzerdefinierten AMIs und einzelnen benutzerdefinierten Elementen vermieden werdenAMI, die dazu führen würden, dass der Clusterstart fehlschlägt.
- Der Cluster kann AMIs über mehrere benutzerdefinierte Knoten verfügen, die sich auf primäre, zentrale und einzelne Taskknoten verteilen. Dies ermöglicht individuelle AMI Anpassungen, wie z. B. vorinstallierte Anwendungen, ausgefeilte Cluster-Konfigurationen und verschlüsselte EBS Amazon-Root-Geräte-Volumes.
- Der Kernknoten der Instanzgruppe kann nur einen Instance-Typ und einen entsprechenden benutzerdefinierten Instance-Typ haben. AMI Ebenso kann der primäre Knoten nur einen Instanztyp und einen entsprechenden benutzerdefinierten Instanztyp habenAMI.
- Der Cluster kann mehrere Aufgabenknoten haben.

```
aws emr create-cluster --instance-groups
InstanceGroupType=PRIMARY,InstanceType=m5.xlarge,InstanceCount=1,CustomAmiId=ami-123456
InstanceGroupType=CORE,InstanceType=m5.xlarge,InstanceCount=1,CustomAmiId=ami-234567
InstanceGroupType=TASK,InstanceType=m6g.xlarge,InstanceCount=1,CustomAmiId=ami-345678
InstanceGroupType=TASK,InstanceType=m5.xlarge,InstanceCount=1,CustomAmiId=ami-456789
```

Example — Verwenden Sie die AWS CLI Version 1.20.21 oder höher, um einem laufenden Instanzgruppen-Cluster mit mehreren Instanztypen und mehreren benutzerdefinierten Instanztypen einen Task-Knoten hinzuzufügen AMIs

Mit der AWS CLI Version 1.20.21 oder höher können Sie einer Instanzgruppe, die Sie einem AMIs laufenden Cluster hinzufügen, mehrere benutzerdefinierte Instanzen hinzufügen. Das `CustomAmiId`-Argument kann zusammen mit dem `add-instance-groups`-Befehl verwendet werden, wie im folgenden Beispiel gezeigt. Beachten Sie, dass dieselbe mehrfache benutzerdefinierte AMI ID (ami-123456) in mehr als einem Knoten verwendet wird.

```
aws emr create-cluster --instance-groups
InstanceGroupType=PRIMARY,InstanceType=m5.xlarge,InstanceCount=1,CustomAmiId=ami-123456
InstanceGroupType=CORE,InstanceType=m5.xlarge,InstanceCount=1,CustomAmiId=ami-123456
InstanceGroupType=TASK,InstanceType=m5.xlarge,InstanceCount=1,CustomAmiId=ami-234567
```

```
{
    "ClusterId": "j-123456",
    ...
}

aws emr add-instance-groups --cluster-id j-123456 --instance-groups
 InstanceGroupType=Task,InstanceType=m6g.xlarge,InstanceCount=1,CustomAmiId=ami-345678
```

Example - Verwenden Sie die AWS CLI Version 1.20.21 oder höher, um einen Instance-Flottencluster, mehrere benutzerdefinierte Instance-Typen, AMIs On-Demand-Primärinstanzen, On-Demand-Core, mehrere Core- und Task-Knoten zu erstellen

```
aws emr create-cluster --instance-fleets
InstanceFleetType=PRIMARY,TargetOnDemandCapacity=1,InstanceTypeConfigs=['{InstanceType=m5.xlarg
 CustomAmiId=ami-123456}']
InstanceFleetType=CORE,TargetOnDemandCapacity=1,InstanceTypeConfigs=['{InstanceType=m5.xlarge,C
{InstanceType=m6g.xlarge, CustomAmiId=ami-345678}']
InstanceFleetType=TASK,TargetSpotCapacity=1,InstanceTypeConfigs=['{InstanceType=m5.xlarge,Custo
{InstanceType=m6g.xlarge, CustomAmiId=ami-567890}']
```

Example - Verwenden Sie die AWS CLI Version 1.20.21 oder höher, um Taskknoten zu einem laufenden Cluster mit mehreren Instance-Typen und mehreren benutzerdefinierten Instance-Typen hinzuzufügen AMIs

```
aws emr create-cluster --instance-fleets
InstanceFleetType=PRIMARY,TargetOnDemandCapacity=1,InstanceTypeConfigs=['{InstanceType=m5.xlarg
 CustomAmiId=ami-123456}']
InstanceFleetType=CORE,TargetOnDemandCapacity=1,InstanceTypeConfigs=['{InstanceType=m5.xlarge,C
{InstanceType=m6g.xlarge, CustomAmiId=ami-345678}']

{
    "ClusterId": "j-123456",
    ...
}

aws emr add-instance-fleet --cluster-id j-123456 --instance-fleet
InstanceFleetType=TASK,TargetSpotCapacity=1,InstanceTypeConfigs=['{InstanceType=m5.xlarge,Custo
{InstanceType=m6g.xlarge, CustomAmiId=ami-345678}']
```

Verwaltung von AMI Paket-Repository-Updates

Beim ersten Start stellt Amazon Linux standardmäßig eine AMIs Verbindung zu Paket-Repositorys her, um Sicherheitsupdates zu installieren, bevor andere Dienste gestartet werden. Je nach Ihren Anforderungen können Sie diese Updates deaktivieren, wenn Sie ein benutzerdefiniertes Update AMI für Amazon angebenEMR. Die Option zum Deaktivieren dieser Funktion ist nur verfügbar, wenn Sie eine benutzerdefinierte Funktion verwendenAMI. Standardmäßig werden Amazon-Linux-Kernel-Updates und andere Softwarepakete, die einen Neustart erfordern, nicht aktualisiert. Beachten Sie, dass Ihre Netzwerkkonfiguration Amazon Linux-Repositorys in Amazon S3 zulassen HTTP und HTTPS zu diesen gelangen muss, da andernfalls Sicherheitsupdates nicht erfolgreich sein werden.

> **Warning**
>
> Wir empfehlen dringend, dass Sie sich dafür entscheiden, alle installierten Pakete beim Neustart zu aktualisieren, wenn Sie ein benutzerdefiniertes Paket angeben. AMI Wenn Sie keine Pakete aktualisieren, entstehen zusätzliche Sicherheitsrisiken.

Mit dem AWS Management Console können Sie die Option zum Deaktivieren von Updates auswählen, wenn Sie Benutzerdefiniert wählenAMI.

Mit dem können Sie angeben AWS CLI`--repo-upgrade-on-boot NONE`, `--custom-ami-id` wann Sie den create-cluster Befehl verwenden möchten.

Bei Amazon EMR API können Sie `NONE` den RepoUpgradeOnBootParameter angeben.

Benutzerdefiniertes Volume AMI mit einem verschlüsselten EBS Amazon-Root-Geräte-Volume erstellen

Um das EBS Amazon-Root-Geräte-Volume eines Amazon Linux AMI for Amazon zu verschlüsselnEMR, kopieren Sie ein Snapshot-Image von einem unverschlüsselten AMI auf ein verschlüsseltes Ziel. Informationen zum Erstellen verschlüsselter EBS Volumes finden Sie unter EBSAmazon-Verschlüsselung im EC2Amazon-Benutzerhandbuch. Die Quelle AMI für den Snapshot kann das Amazon Linux-Basissystem seinAMI, oder Sie können einen Snapshot aus einem von der Amazon Linux-Basis AMI abgeleiteten Snapshot kopierenAMI, den Sie angepasst haben.

> **Note**
>
> Ab EMR Amazon-Version 5.24.0 können Sie eine Sicherheitskonfigurationsoption verwenden, um EBS Root-Geräte und Speichervolumes zu verschlüsseln, wenn Sie dies AWS KMS als Ihren Schlüsselanbieter angeben. Weitere Informationen finden Sie unter Verschlüsselung lokaler Datenträger.

Sie können einen externen Schlüsselanbieter oder einen AWS KMS Schlüssel verwenden, um das Root-Volume zu verschlüsseln. EBS Die von Amazon EMR verwendete Servicerolle (normalerweise die Standardeinstellung`EMR_DefaultRole`) muss mindestens berechtigt sein, das Volume zu ver- und entschlüsseln, EMR damit Amazon einen Cluster mit dem erstellen kann. AMI Bei der Verwendung AWS KMS als Schlüsselanbieter bedeutet dies, dass die folgenden Aktionen zulässig sein müssen:

- `kms:encrypt`
- `kms:decrypt`
- `kms:ReEncrypt*`
- `kms:CreateGrant`
- `kms:GenerateDataKeyWithoutPlaintext"`
- `kms:DescribeKey"`

Hierfür fügen Sie am einfachsten die Rolle als Schlüsselbenutzer wie im folgenden Tutorial beschrieben hinzu. Die folgende Richtlinienanweisung dient als Beispiel für den Fall, dass Sie Rollenrichtlinien anpassen müssen.

```
{
"Version": "2012-10-17",
"Statement": [
   {
   "Sid": "EmrDiskEncryptionPolicy",
   "Effect": "Allow",
   "Action": [
      "kms:Encrypt",
      "kms:Decrypt",
      "kms:ReEncrypt*",
      "kms:CreateGrant",
      "kms:GenerateDataKeyWithoutPlaintext",
```

```
      "kms:DescribeKey"
      ],
   "Resource": [
      "*"
      ]
   }
]
}
```

Tutorial: Ein benutzerdefiniertes Volume AMI mit einem verschlüsselten Root-Geräte-Volume mithilfe eines KMS Schlüssels erstellen

Der erste Schritt in diesem Beispiel besteht darin, den ARN KMS Schlüssel zu finden oder einen neuen zu erstellen. Weitere Informationen zum Erstellen von -Schlüsseln finden Sie unter Erstellen von Schlüsseln im AWS Key Management Service -Entwicklerhandbuch. Im folgenden Verfahren wird gezeigt, wie Sie als Schlüsselbenutzer die Standard-Servicerolle `EMR_DefaultRole` zur Schlüsselrichtlinie hinzufügen. Notieren Sie sich den ARNWert für den Schlüssel, während Sie ihn erstellen oder bearbeiten. Den ARN höheren Wert verwendest du, wenn du den erstellstAMI.

Um die Servicerolle für Amazon EC2 zur Liste der Benutzer von Verschlüsselungsschlüsseln mit der Konsole hinzuzufügen

1. Melden Sie sich bei der Konsole AWS Key Management Service (AWS KMS) an AWS Management Console und öffnen Sie sie unter https://console.aws.amazon.com/kms.
2. Um das zu ändern AWS-Region, verwenden Sie die Regionsauswahl in der oberen rechten Ecke der Seite.
3. Wählen Sie den Alias des KMS Schlüssels, den Sie verwenden möchten.
4. Wählen Sie auf der Seite mit den Schlüsseldetails unter Key Users (Schlüsselbenutzer(die Option Add (Hinzufügen) aus.
5. Wählen Sie im Dialogfeld „Anhängen“ die EMR Amazon-Servicerolle aus. Der Name der Standardrolle lautet `EMR_DefaultRole`.
6. Wählen Sie Anfügen aus.

Um ein verschlüsseltes AMI mit dem zu erstellen AWS CLI

- Verwenden Sie den `aws ec2 copy-image` Befehl von AWS CLI , um ein Volume AMI mit einem verschlüsselten EBS Root-Gerät und dem von Ihnen geänderten Schlüssel zu erstellen.

Ersetzen Sie den angegebenen `--kms-key-id` Wert durch den vollständigen Wert ARN des Schlüssels, den Sie unten erstellt oder geändert haben.

> **Note**
>
> Linux-Zeilenfortsetzungszeichen (\) sind aus Gründen der Lesbarkeit enthalten. Sie können entfernt oder in Linux-Befehlen verwendet werden. Entfernen Sie sie unter Windows oder ersetzen Sie sie durch ein Caret-Zeichen (^).

```
aws ec2 copy-image --source-image-id MyAmiId \
--source-region us-west-2 --name MyEncryptedEMRAmi \
--encrypted --kms-key-id arn:aws:kms:us-west-2:12345678910:key/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
```

Die Ausgabe des Befehls enthält die ID des AMI von Ihnen erstellten Clusters, die Sie angeben können, wenn Sie einen Cluster erstellen. Weitere Informationen finden Sie unter Verwenden Sie einen einzelnen benutzerdefinierten AMI Code in einem EMR Cluster. Sie können dies auch anpassen, AMI indem Sie Software installieren und andere Konfigurationen durchführen. Weitere Informationen finden Sie unter Ein benutzerdefiniertes Amazon Linux AMI aus einer vorkonfigurierten Instance erstellen.

Bewährte Methoden und Überlegungen

Beachten Sie FolgendesEMR, wenn Sie ein benutzerdefiniertes Produkt AMI für Amazon erstellen:

- Die Amazon EMR 7.x-Serie basiert auf Amazon Linux 2023. Für diese EMR Amazon-Versionen müssen Sie benutzerdefinierte Images verwenden, die auf Amazon Linux 2023 basierenAMIs. Informationen zur Suche nach einem benutzerdefinierten AMI Basismodell finden Sie unter Ein Linux findenAMI.
- Für EMR Amazon-Versionen unter 7.x wird Amazon Linux 2023 AMIs nicht unterstützt.
- Amazon EMR 5.30.0 und höher sowie die Amazon EMR 6.x-Serie basieren auf Amazon Linux 2. Für diese EMR Amazon-Versionen müssen Sie benutzerdefinierte Images verwenden, die auf Amazon Linux 2 basierenAMIs. Informationen zur Suche nach einem benutzerdefinierten AMI Basismodell finden Sie unter Linux findenAMI.
- Für EMR Amazon-Versionen unter 5.30.0 und 6.x wird Amazon Linux 2 AMIs nicht unterstützt.

- Sie müssen ein 64-Bit-Amazon-Linux verwendenAMI. Eine 32-Bit-Version AMI wird nicht unterstützt.
- Amazon Linux AMIs mit mehreren EBS Amazon-Volumes wird nicht unterstützt.
- Basieren Sie Ihre Anpassung auf das neueste EBS unterstützte Amazon Linux AMI. Eine Liste von Amazon Linux AMIs und entsprechendem AMI IDs finden Sie unter Amazon Linux AMI.
- Kopieren Sie keinen Snapshot einer vorhandenen EMR Amazon-Instance, um eine benutzerdefinierte zu erstellenAMI. Das verursacht Fehler.
- Nur der HVM Virtualisierungstyp und die mit Amazon kompatiblen Instances EMR werden unterstützt. Achten Sie darauf, das HVM Image und einen mit Amazon kompatiblen Instance-Typ auszuwählen, EMR während Sie den AMI Anpassungsprozess durchführen. Kompatible Instances und Virtualisierungstypen finden Sie unter Unterstützte Instance-Typen.
- Ihre Servicerolle muss über Startberechtigungen für verfügenAMI, also AMI muss sie entweder öffentlich sein, oder Sie müssen der Eigentümer von sein AMI oder sie vom Eigentümer mit Ihnen teilen lassen.
- Das Erstellen von Benutzern AMI mit demselben Namen wie Anwendungen führt zu Fehlern (z. B. `hadoophdfs`,`yarn`, oder`spark`).
- Die Inhalte von `/tmp/var`, und `/emr` (sofern sie auf der vorhanden sindAMI) werden beim Start nach `/mnt/tmp/mnt/var`, `/mnt/emr` bzw. verschoben. Dateien werden beibehalten; bei großen Mengen an Daten kann jedoch der Startup länger als erwartet dauern.
- Wenn Sie ein benutzerdefiniertes Amazon Linux verwenden, das auf einem Amazon Linux AMI mit einem Erstellungsdatum vom 11.08.2018 AMI basiert, kann der Oozie-Server nicht gestartet werden. Wenn Sie Oozie verwenden, erstellen Sie eine benutzerdefinierte Version, die auf einer Amazon AMI Linux-ID mit einem anderen Erstellungsdatum AMI basiert. Sie können den folgenden AWS CLI Befehl verwenden, um eine Liste mit Images IDs für alle HVM Amazon Linux-Versionen AMIs mit einer Version 2018.03 zusammen mit dem Veröffentlichungsdatum zurückzugeben, sodass Sie ein geeignetes Amazon Linux AMI als Basis auswählen können. MyRegion Ersetzen Sie es durch Ihre Regionskennung, z. B. us-west-2.

```
aws ec2 --region MyRegion describe-images --owner amazon --query 'Images[?
Name!=`null`]|[?starts_with(Name, `amzn-ami-hvm-2018.03`) == `true`].
[CreationDate,ImageId,Name]' --output text | sort -rk1
```

- In Fällen, in denen Sie einen Domainnamen verwenden, der nicht VPC dem Standard entspricht AmazonProvidedDNS, sollten Sie die `rotate` Option in der Betriebssystemkonfiguration nicht verwenden. DNS

Weitere Informationen finden Sie AMI im EBSEC2Amazon-Benutzerhandbuch unter Creating an Amazon-gestütztes Linux.

Ändern der Amazon Linux-Version beim Erstellen eines EMR Clusters

Wenn Sie einen Cluster mit Amazon EMR 6.6.0 oder höher starten, verwendet er automatisch die neueste Amazon Linux 2-Version, die für die EMR AMI Amazon-Standardversion validiert wurde. Sie können mit der EMR Amazon-Konsole oder dem eine andere Amazon Linux-Version für Ihren Cluster angeben AWS CLI.

Amazon EMR console

So ändern Sie die Amazon-Linux-Version, wenn Sie einen Cluster über die Konsole erstellen

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und anschließend Create cluster aus.
3. Wählen Sie als EMRVersion emr-6.6.0 oder höher aus.
4. Wählen Sie unter Betriebssystemoptionen die Amazon-Linux-Version und aktivieren Sie das Kontrollkästchen Aktuelle Amazon-Linux-Updates automatisch anwenden.
5. Wählen Sie alle anderen Optionen aus, die für Ihren Cluster gelten.
6. Um Ihren Cluster jetzt zu starten, wählen Sie Cluster erstellen aus.

AWS CLI

Um die Amazon-Linux-Version zu ändern, wenn Sie einen Cluster mit AWS CLI erstellen

- Verwenden Sie den `--os-release-label`-Parameter, um die Amazon-Linux-Version anzugeben, wenn Sie den Befehl `aws emr` create-cluster ausführen.

```
aws emr create-cluster --name "Cluster with Different Amazon Linux Release" \
--os-release-label 2.0.20210312.1 \
--release-label emr-6.6.0 --use-default-roles \
--instance-count 2 --instance-type m5.xlarge
```

Anpassen des EBS Amazon-Root-Geräte-Volumes

EBSStandardeinstellungen für das Root-Volume

Bei Amazon EMR 4.x und höher können Sie die Größe des Root-Volumes angeben, wenn Sie einen Cluster erstellen. Mit EMR Amazon-Versionen 6.15.0 und höher können Sie auch das Root-Volume IOPS und den Durchsatz angeben. Die Attribute gelten nur für das EBS Amazon-Root-Geräte-Volume und gelten für alle Instances im Cluster. Die Attribute gelten nicht für Speicher-Volumes, die Sie separat für jeden Instance-Typ beim Erstellen Ihres Clusters angeben.

- Die Standardgröße des Root-Volumes beträgt 15 GiB in Amazon EMR 6.10.0 und höher. Frühere Versionen haben eine Standardgröße für das Root-Volume von 10 GiB. Sie können dies auf bis zu 100 GiB einstellen.
- Das Standard-Root-Volume IOPS ist 3000. Sie können dies auf bis zu 16 000 einstellen.
- Der Standard-Root-Volume-Durchsatz beträgt 125 MiB/s. Sie können dies auf bis zu 1 000 MiB/s einstellen.

> **Note**
>
> Die Größe des Root-Volumes IOPS darf nicht höher als 1 Volume zu 500 IOPS (1:500) sein, wohingegen das Verhältnis von Root-Volumen IOPS und Durchsatz nicht höher als 1 IOPS zu 0,25 Durchsatz (1:0,25) sein darf.

Weitere Informationen zu Amazon finden Sie EBS unter Amazon EC2 Root Device Volume.

Volumetyp des Root-Geräts mit der Standardeinstellung AMI

Wenn Sie die Standardversion verwendenAMI, wird der Volumetyp des Root-Geräts von der EMR Amazon-Version bestimmt, die Sie verwenden.

- Mit EMR Amazon-Versionen 6.15.0 und höher EMR fügt Amazon General Purpose SSD (gp3) als Volumetyp für das Root-Gerät hinzu.
- Bei EMR Amazon-Versionen unter 6.15.0 EMR fügt Amazon General Purpose SSD (gp2) als Volumetyp für das Root-Gerät hinzu.

Volumetyp des Stammgeräts mit dem benutzerdefinierten AMI

Ein benutzerdefiniertes Gerät AMI kann unterschiedliche Volumetypen für das Root-Gerät haben. Amazon verwendet EMR immer Ihren benutzerdefinierten AMI Volumetyp.

- Mit EMR Amazon-Versionen 6.15.0 und höher können Sie die Größe und den Durchsatz des Stamm-Volumes für Ihr benutzerdefiniertes Volume konfigurierenAMI, vorausgesetzt, diese Attribute gelten für den benutzerdefinierten AMI Volume-Typ. IOPS
- Bei EMR Amazon-Versionen unter 6.15.0 können Sie nur die Größe des Root-Volumes für Ihr benutzerdefiniertes Volume konfigurieren. AMI

Wenn Sie bei der Erstellung Ihres Clusters weder die Größe des Root-Volumes noch den Durchsatz konfigurieren, EMR verwendet Amazon AMI gegebenenfalls die benutzerdefinierten Werte. IOPS Wenn Sie sich entscheiden, diese Werte bei der Erstellung Ihres Clusters zu konfigurieren, EMR verwendet Amazon die von Ihnen angegebenen Werte, sofern die Werte mit dem benutzerdefinierten AMI Root-Volume kompatibel sind und von diesem unterstützt werden. Weitere Informationen finden Sie unter Verwenden Sie ein benutzerdefiniertes AMI.

Preise nach Root-Gerät-Volumegröße

Die Kosten für das EBS Root-Geräte-Volume werden pro Stunde anteilig berechnet, basierend auf den monatlichen EBS Gebühren für diesen Volume-Typ in der Region, in der der Cluster ausgeführt wird. Gleiches gilt für Speicher-Volumes. Die Gebühren gelten für GB, aber da Sie die Größe des Root-Volumes in GiB angeben, sollten Sie dies bei Ihren Kostenschätzungen berücksichtigen (1 GB entspricht 0,931323 GiB).

SSDGP2 und GP3 für allgemeine Zwecke werden unterschiedlich abgerechnet. Verwenden Sie die folgenden Formeln, um die Gebühren für das Volumen der EBS Root-Geräte in Ihrem Cluster zu schätzen:

SSDGP2 für allgemeine Zwecke

Die Kosten für GP2 beinhalten nur die EBS Volumengröße in GB.

```
($EBS size in GB/month) * 0.931323 / 30 / 24 * EMR_EBSRootVolumesizeInGiB *
 InstanceCount
```

Nehmen wir zum Beispiel einen Cluster, der über einen Primärknoten, einen Kernknoten, verfügt und das Amazon AMI Linux-Basisvolume mit dem standardmäßigen 10-GiB-Root-Geräte-

Volume verwendet. Wenn die EBS Kosten in der Region USD 0,10$ pro GB pro Monat betragen, entspricht das ungefähr 0,00129$ pro Instance und pro Stunde und 0,00258$ pro Stunde für den Cluster (0,10$ pro GB/Monat geteilt durch 30 Tage, geteilt durch 24 Stunden, multipliziert mit 10 GB, multipliziert mit 2 Cluster-Instances).

GP3 für allgemeine Zwecke SSD

Die Kosten für gp3 beinhalten die EBS Volumengröße in GB, IOPS über 3000 (3000 IOPS kostenlos) und den Durchsatz über 125 MB/s (125 MB/s kostenlos).

```
($EBS size in GB/month) * 0.931323 / 30 / 24 * EMR_EBSRootVolumesizeInGiB *
 InstanceCount
+
($EBS IOPS/Month)/30/24* (EMR_EBSRootVolumeIops - 3000) * InstanceCount
+
($EBS throughput/Month)/30/24* (EMR_EBSRootVolumeThroughputInMb/s - 125) *
 InstanceCount
```

Nehmen wir zum Beispiel einen Cluster, der über einen Primärknoten, einen Kernknoten, verfügt und das Amazon AMI Linux-Basissystem mit der Standardgröße des Root-Device-Volumes von 15 GiBIOPS, 4000 und 140 Durchsatz verwendet. Wenn die EBS Kosten in der Region 0,10 USD USD/GB/Monat, 0,005 USD/bereitgestellt/Monat über 3000 und 0,040 USD/bereitgestellte IOPS MB/Monat über 125 betragen. Das entspricht ungefähr 0,009293 USD pro Instance und Stunde und 0,018586 USD pro Stunde für den Cluster.

Festlegen benutzerdefinierter Root-Gerät-Volume-Einstellungen

> **Note**
>
> Die Größe des Root-Volumes IOPS darf nicht höher als 1 Volume zu 500 IOPS (1:500) sein, wohingegen das Verhältnis zwischen Root-Volume und Durchsatz nicht höher als 1 zu 0,25 Durchsatz (1:0,25) sein darf. IOPS IOPS

Console

So geben Sie Volumenattribute für EBS Amazon-Root-Geräte von der EMR Amazon-Konsole aus an

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und anschließend Create cluster aus.
3. Wählen Sie EMR Amazon-Version 6.15.0 oder höher aus.
4. Navigieren Sie unter Cluster-Konfiguration zum Bereich EBSRoot-Volume und geben Sie einen Wert für eines der Attribute ein, die Sie konfigurieren möchten.
5. Wählen Sie alle anderen Optionen aus, die für Ihren Cluster gelten.
6. Um Ihren Cluster jetzt zu starten, wählen Sie Cluster erstellen aus.

CLI

Um Volumenattribute für EBS Amazon-Root-Geräte anzugeben mit dem AWS CLI

- Verwenden Sie die Parameter `--ebs-root-volume-size`, `--ebs-root-volume-iops` und `--ebs-root-volume-throughput` des Befehls create-cluster wie im folgenden Beispiel gezeigt.

> **Note**
>
> Linux-Zeilenfortsetzungszeichen (\) sind aus Gründen der Lesbarkeit enthalten. Sie können entfernt oder in Linux-Befehlen verwendet werden. Entfernen Sie sie unter Windows oder ersetzen Sie sie durch ein Caret-Zeichen (^).

```
aws emr create-cluster --release-label emr-6.15.0\
--ebs-root-volume-size 20 \
--ebs-root-volume-iops 3000\
--ebs-root-volume-throughput 135\
--instance-groups InstanceGroupType=MASTER,\
InstanceCount=1,InstanceType=m5.xlarge
 InstanceGroupType=CORE,InstanceCount=2,InstanceType=m5.xlarge
```

Konfigurieren der Cluster-Software

Wenn Sie eine Softwareversion auswählen, EMR verwendet Amazon ein Amazon Machine Image (AMI) mit Amazon Linux, um die Software zu installieren, die Sie beim Start Ihres Clusters auswählen, wie Hadoop, Spark und Hive. Amazon EMR stellt regelmäßig neue Versionen zur Verfügung und fügt neue Funktionen, neue Anwendungen und allgemeine Updates hinzu. Wir empfehlen, dass Sie die neueste Version zum Starten Ihres Clusters verwenden (sofern möglich). Die neueste Version ist die Standardoption beim Starten eines Clusters über die Konsole.

Weitere Informationen zu EMR Amazon-Versionen und den mit jeder Version verfügbaren Softwareversionen finden Sie im EMRAmazon-Versionshandbuch. Weitere Informationen zur Bearbeitung der Standardkonfigurationen von Anwendungen und Software, die auf Ihrem Cluster installiert sind, finden Sie unter Anwendungen konfigurieren im Amazon EMR Release Guide. Einige Versionen der Open-Source-Komponenten des Hadoop- und Spark-Ökosystems, die in EMR Amazon-Versionen enthalten sind, enthalten Patches und Verbesserungen, die im Amazon EMR Release Guide dokumentiert sind.

Zusätzlich zur standardmäßigen Software und den zur Installation auf Ihrem Cluster verfügbaren Anwendungen können Sie mit Bootstrap-Aktionen benutzerdefinierte Software installieren. Bootstrap-Aktionen sind Skripts, die beim Start Ihres Clusters und beim Ausführen neue, bei der Erstellung des Clusters hinzugefügter Knoten in den Instances ausgeführt werden. Bootstrap-Aktionen sind auch nützlich, um AWS CLI Befehle auf jedem Knoten aufzurufen, um Objekte von Amazon S3 auf jeden Knoten in Ihrem Cluster zu kopieren.

> **Note**
>
> Bootstrap-Aktionen werden in EMR Amazon-Version 4.x und höher unterschiedlich verwendet. Weitere Informationen zu diesen Unterschieden zu den EMR AMI Amazon-Versionen 2.x und 3.x finden Sie unter In 4.x eingeführte Unterschiede im EMR Amazon-Versionshandbuch.

Erstellen von Bootstrap-Aktionen zur Installation zusätzlicher Software

Sie können eine Bootstrap-Aktion verwenden, um zusätzliche Software zu installieren oder die Konfiguration von Cluster-Instances anzupassen. Bootstrap-Aktionen sind Skripte, die auf dem Cluster ausgeführt werden, nachdem Amazon die Instance mit dem Amazon Linux Amazon Machine Image (AMI) EMR gestartet hat. Bootstrap-Aktionen werden ausgeführt, bevor Amazon

die Anwendungen EMR installiert, die Sie bei der Erstellung des Clusters angeben, und bevor die Clusterknoten mit der Datenverarbeitung beginnen. Wenn Sie einem aktiven Cluster Knoten hinzufügen, werden die Bootstrap-Aktionen auf diesen Knoten auch auf die gleiche Weise ausgeführt. Sie können benutzerdefinierte Bootstrap-Aktionen erstellen und sie beim Erstellen Ihres Clusters angeben.

Die meisten vordefinierten Bootstrap-Aktionen für die EMR AMI Amazon-Versionen 2.x und 3.x werden in den EMR Amazon-Versionen 4.x nicht unterstützt. Zum Beispiel `configure-Hadoop` und `configure-daemons` werden in EMR Amazon-Version 4.x nicht unterstützt. Stattdessen bietet Amazon EMR Release 4.x diese Funktionalität nativ. Weitere Informationen zur Migration von Bootstrap-Aktionen von den EMR AMI Amazon-Versionen 2.x und 3.x auf EMR Amazon-Version 4.x finden Sie unter Anpassen der Cluster- und Anwendungskonfiguration mit früheren AMI Versionen von Amazon EMR im Amazon-Versionshandbuch. EMR

Bootstrap-Aktionen – Grundlagen

Bootstrap-Aktionen werden standardmäßig als Hadoop-Benutzer ausgeführt. Sie können eine Bootstrap-Aktion mit Root-Berechtigungen ausführen, indem Sie `sudo` verwenden.

Alle EMR Amazon-Verwaltungsoberflächen unterstützen Bootstrap-Aktionen. Sie können bis zu 16 Bootstrap-Aktionen pro Cluster angeben, indem Sie mehrere `bootstrap-actions` Parameter von der Konsole aus angeben, AWS CLI, oder. API

In der EMR Amazon-Konsole können Sie optional eine Bootstrap-Aktion angeben, während Sie einen Cluster erstellen.

Wenn Sie den verwendenCLI, können Sie Verweise auf Bootstrap-Aktionsskripte an Amazon übergeben, EMR indem Sie den `--bootstrap-actions` Parameter hinzufügen, wenn Sie den Cluster mithilfe des `create-cluster` Befehls erstellen.

```
--bootstrap-actions Path="s3://mybucket/filename",Args=[arg1,arg2]
```

Wenn die Bootstrap-Aktion einen Fehlercode ungleich Null zurückgibt, EMR behandelt Amazon dies als Fehler und beendet die Instance. Wenn zu viele Instances ihre Bootstrap-Aktionen nicht erfolgreich ausführen, EMR beendet Amazon den Cluster. Wenn nur einige Instances ausfallen, EMR versucht Amazon, die ausgefallenen Instances neu zuzuweisen und fortzufahren. Verwenden Sie den Cluster-Fehlercode `lastStateChangeReason`, um Fehler zu identifizieren, die durch eine Bootstrap-Aktion verursacht wurden.

Eine bedingte eine Bootstrap-Aktion ausführen

Um Bootstrap-Aktionen nur auf dem Hauptknoten auszuführen, können Sie eine benutzerdefinierte Bootstrap-Aktion mit etwas Logik verwenden, um festzustellen, ob es sich bei dem Knoten um einen Hauptknoten handelt.

```
#!/bin/bash
if grep isMaster /mnt/var/lib/info/instance.json | grep false;
then
    echo "This is not master node, do nothing,exiting"
    exit 0
fi
echo "This is master, continuing to execute script"
# continue with code logic for master node below
```

Die folgende Ausgabe wird von einem Core-Knoten aus gedruckt.

```
This is not master node, do nothing, exiting
```

Die folgende Ausgabe wird vom Hauptknoten aus gedruckt.

```
This is master, continuing to execute script
```

Um diese Logik zu verwenden, laden Sie Ihre Bootstrap-Aktion, einschließlich des obigen Codes, in Ihren Amazon-S3-Bucket hoch. Fügen Sie auf der AWS CLI den `aws emr create-cluster` API Aufruf den `--bootstrap-actions` Parameter hinzu und geben Sie den Speicherort Ihres Bootstrap-Skripts als Wert von an. `Path`

Aktionen beim Herunterfahren

Ein Bootstrap-Aktionsskript kann eine oder mehrere Shutdown-Aktionen durchführen, indem es Skripts in das Verzeichnis `/mnt/var/lib/instance-controller/public/shutdown-actions/` schreibt. Wenn ein Cluster beendet wird, werden alle Skripts in diesem Verzeichnis parallel ausgeführt. Jedes Skript muss innerhalb von 60 Sekunden ausgeführt und abgeschlossen werden.

Es wird nicht garantiert, dass Shutdown-Aktionsskripts ausgeführt werden, wenn der Knoten mit einem Fehler beendet wird.

> **Note**
>
> Wenn Sie EMR Amazon-Versionen 4.0 und höher verwenden, müssen Sie das `/mnt/var/lib/instance-controller/public/shutdown-actions/` Verzeichnis auf dem Master-Knoten manuell erstellen. Es ist standardmäßig zwar nicht vorhanden, nach Erstellung werden die Skripts in diesem Verzeichnis aber trotzdem vor dem Herunterfahren ausgeführt. Weitere Informationen zum Herstellen einer Verbindung mit dem Master-Knoten zum Erstellen von Verzeichnissen finden Sie unter Connect zum Primärknoten her mit SSH.

Benutzerdefinierte Bootstrap-Aktionen verwenden

Sie können ein benutzerdefiniertes Skript erstellen, um eine angepasste Bootstrap-Aktion auszuführen. Jede der EMR Amazon-Schnittstellen kann auf eine benutzerdefinierte Bootstrap-Aktion verweisen.

> **Note**
>
> Für eine optimale Leistung empfehlen wir, benutzerdefinierte Bootstrap-Aktionen, -Skripts und andere Dateien, die Sie mit Amazon verwenden möchten, EMR in einem Amazon S3 S3-Bucket zu speichern, der sich in demselben AWS-Region Cluster befindet.

Inhalt

Benutzerdefinierte Bootstrap-Aktionen hinzufügen

Console

Um mit der Konsole einen Cluster mit einer Bootstrap-Aktion zu erstellen

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und anschließend Create cluster aus.

3. Wählen Sie unter Bootstrap-Aktionen die Option Hinzufügen aus, um einen Namen, einen Skriptspeicherort und optionale Argumente für Ihre Aktion anzugeben. Wählen Sie Bootstrap-Aktion hinzufügen aus.
4. Fügen Sie optional weitere Bootstrap-Aktionen hinzu.
5. Wählen Sie alle anderen Optionen aus, die für Ihren Cluster gelten.
6. Um Ihren Cluster jetzt zu starten, wählen Sie Cluster erstellen aus.

CLI

Um einen Cluster mit einer benutzerdefinierten Bootstrap-Aktion zu erstellen, verwenden Sie AWS CLI

Wenn Sie die Aktion AWS CLI zum Einbeziehen einer Bootstrap-Aktion verwenden, geben Sie `Path` und `Args` als kommagetrennte Liste an. Bei dem folgenden Beispiel wird keine Argumentliste verwendet.

- Um einen Cluster mit einer benutzerdefinierten Bootstrap-Aktion zu starten, geben Sie den folgenden Befehl ein und ersetzen *myKey* mit dem Namen Ihres EC2 key pair. Fügen Sie `--bootstrap-actions` als Parameter ein und geben Sie den Speicherort Ihres Bootstrap-Skripts als Wert von `Path` an.
  - Linux UNIX - und Mac OS X-Benutzer:

```
aws emr create-cluster --name "Test cluster" --release-label emr-4.0.0 \
--use-default-roles --ec2-attributes KeyName=myKey \
--applications Name=Hive Name=Pig \
--instance-count 3 --instance-type m5.xlarge \
--bootstrap-actions Path="s3://elasticmapreduce/bootstrap-actions/download.sh"
```

  - Windows-Nutzer:

```
aws emr create-cluster --name "Test cluster" --release-label emr-4.2.0 --use-
default-roles --ec2-attributes KeyName=myKey --applications Name=Hive Name=Pig
 --instance-count 3 --instance-type m5.xlarge --bootstrap-actions Path="s3://
elasticmapreduce/bootstrap-actions/download.sh"
```

Wenn Sie die Instance-Anzahl ohne den `--instance-groups`-Parameter angeben, wird ein einzelner Primärknoten gestartet. Die verbleibenden Instances werden dabei als Core-Knoten gestartet. Alle Knoten verwenden den im Befehl angegebenen Instance-Typ.

> **Note**
>
> Wenn Sie noch nicht die standardmäßige EMR Amazon-Servicerolle und das EC2 Instanzprofil erstellt haben, geben Sie ein, `aws emr create-default-roles` um sie zu erstellen, bevor Sie den `create-cluster` Unterbefehl eingeben.

Weitere Informationen zur Verwendung von EMR Amazon-Befehlen in der AWS CLI finden Sie unterhttps://docs.aws.amazon.com/cli/latest/reference/emr.

Verwenden einer benutzerdefinierten Bootstrap-Aktion zum Kopieren eines Objekts aus Amazon S3 in jeden Knoten

Sie können mit einer Bootstrap-Aktion Objekte von Amazon S3 in jeden Knoten eines Cluster kopieren, bevor Ihre Anwendungen installiert werden. Der AWS CLI ist auf jedem Knoten eines Clusters installiert, sodass Ihre Bootstrap-Aktion AWS CLI Befehle aufrufen kann.

Das folgende Beispiel zeigt ein einfaches Skript für eine Bootstrap-Aktion, die die Datei `myfile.jar` aus Amazon S3 zum lokalen Ordner `/mnt1/myfolder` auf jedem Cluster-Knoten kopiert. Das Skript wird mit dem Dateinamen `copymyfile.sh` in Amazon S3 mit den folgenden Inhalten gespeichert.

```
#!/bin/bash
aws s3 cp s3://mybucket/myfilefolder/myfile.jar /mnt1/myfolder
```

Wenn Sie den Cluster starten, geben Sie das Skript an. Das folgende AWS CLI Beispiel verdeutlicht dies:

```
aws emr create-cluster --name "Test cluster" --release-label emr-7.2.0 \
--use-default-roles --ec2-attributes KeyName=myKey \
--applications Name=Hive Name=Pig \
--instance-count 3 --instance-type m5.xlarge \
--bootstrap-actions Path="s3://mybucket/myscriptfolder/copymyfile.sh"
```

Cluster-Hardware und Netzwerken konfigurieren

Ein wichtiger Aspekt bei der Erstellung eines EMR Amazon-Clusters ist die Konfiguration von EC2 Amazon-Instances und Netzwerkoptionen. Dieses Kapitel behandelt diese Optionen im Detail und beschreibt entsprechende bewährte Methoden und Richtlinien.

- Knotentypen — EC2 Amazon-Instances in einem EMR Cluster sind in Knotentypen unterteilt. Es gibt drei Knotentypen: Primärknoten, Core-Knoten und Aufgabenknoten. Jeder Knotentyp führt eine Reihe von Rollen aus, die durch die von Ihnen auf dem Cluster installierten verteilten Anwendungen definiert werden. Während eines Hadoop MapReduce - oder Spark-Jobs verarbeiten Komponenten auf Kern- und Taskknoten beispielsweise Daten, übertragen die Ausgabe an Amazon S3 oder HDFS stellen Statusmetadaten zurück an den primären Knoten. Bei einem einzigen Knoten-Cluster werden alle Komponenten auf dem Primärknoten ausgeführt. Weitere Informationen finden Sie unter De Knotentypen verstehen: Primär-, Core- und Aufgabenknoten.
- EC2Instances — Wenn Sie einen Cluster erstellen, treffen Sie Entscheidungen über die EC2 Amazon-Instances, auf denen die einzelnen Knotentypen ausgeführt werden sollen. Der EC2 Instance-Typ bestimmt das Verarbeitungs- und Speicherprofil des Knotens. Die Wahl der EC2 Amazon-Instance für Ihre Knoten ist wichtig, da sie das Leistungsprofil der einzelnen Knotentypen in Ihrem Cluster bestimmt. Weitere Informationen finden Sie unter EC2Amazon-Instances konfigurieren.
- Netzwerk — Sie können Ihren EMR Amazon-Cluster in einem VPC öffentlichen Subnetz, einem privaten Subnetz oder einem gemeinsam genutzten Subnetz starten. Ihre Netzwerkkonfiguration bestimmt, wie Kunden und Services Verbindungen zu Clustern herstellen können, um ihre Arbeit zu erledigen, wie Cluster mit Datenspeichern und anderen AWS -Ressourcen verbunden werden und welche Optionen Sie zur Steuerung des Datenverkehrs auf diesen Verbindungen haben. Weitere Informationen finden Sie unter Netzwerk konfigurieren.
- Instance-Gruppierung — Die Sammlung von EC2 Instances, die jeden Knotentyp hosten, wird entweder als Instance-Flotte oder als einheitliche Instance-Gruppe bezeichnet. Die Konfiguration der Instance-Gruppierung ist eine Auswahl, die Sie beim Erstellen eines Clusters treffen. Diese Auswahl bestimmt, wie Sie Ihrem Cluster Knoten hinzufügen können, während er läuft. Die Konfiguration gilt für alle Knotentypen. Er kann später nicht mehr geändert werden. Weitere Informationen finden Sie unter Einen Cluster mit Instance-Flotten oder einheitlichen Instance-Gruppen erstellen.

> **Note**
>
> Die Konfiguration der Instance-Flotten ist nur in EMR Amazon-Versionen 4.8.0 und höher verfügbar, mit Ausnahme von 5.0.0 und 5.0.3.

De Knotentypen verstehen: Primär-, Core- und Aufgabenknoten

Verwenden Sie diesen Abschnitt, um zu verstehen, wie Amazon jeden dieser Knotentypen EMR verwendet, und als Grundlage für die Cluster-Kapazitätsplanung.

Primärknoten

Der Primärknoten verwaltet die Cluster und führt die Master-Komponenten von verteilten Anwendungen aus. Beispielsweise führt der primäre Knoten den YARN ResourceManager Service aus, um Ressourcen für Anwendungen zu verwalten. Er führt auch den HDFS NameNode Dienst aus, verfolgt den Status der an den Cluster übermittelten Jobs und überwacht den Zustand der Instanzgruppen.

Um den Fortschritt eines Clusters zu überwachen und direkt mit Anwendungen zu interagieren, können Sie SSH als Hadoop-Benutzer eine Verbindung zum primären Knoten herstellen. Weitere Informationen finden Sie unter Connect zum Primärknoten her mit SSH. Durch das Verbinden mit dem Primärknoten erhalten Sie direkten Zugriff auf Verzeichnisse und Dateien, wie z. B. Hadoop-Protokolldateien. Weitere Informationen finden Sie unter Anzeige von -Protokolldateien. Sie können auch Benutzeroberflächen anzeigen, die von den Anwendungen als auf dem Primärknoten ausgeführte Websites veröffentlicht werden. Weitere Informationen finden Sie unter Auf EMR Amazon-Clustern gehostete Weboberflächen anzeigen.

> **Note**
>
> Mit Amazon EMR 5.23.0 und höher können Sie einen Cluster mit drei primären Knoten starten, um die Hochverfügbarkeit von Anwendungen wie YARN Resource Manager, Spark HDFS NameNode, Hive und Ganglia zu unterstützen. Der Primärknoten ist mit diesem Feature keine potenzielle einzelne Fehlerquelle mehr. Wenn einer der Primärknoten ausfällt, wechselt Amazon EMR automatisch zu einem Standby-Primärknoten und ersetzt den ausgefallenen Primärknoten durch einen neuen mit derselben Konfiguration und denselben

Bootstrap-Aktionen. Weitere Informationen finden Sie unter Primärknoten planen und konfigurieren.

Core-Knoten

Core-Knoten werden vom Primärknoten verwaltet. Auf den Kernknoten wird der Data Node-Daemon ausgeführt, um die Datenspeicherung als Teil des Hadoop Distributed File Systems () zu koordinieren. HDFS Außerdem führen sie den TaskTracker-Daemon und andere parallele Rechenaufgaben für Daten aus, die für installierte Anwendungen erforderlich sind. Auf einem Core-Knoten werden beispielsweise YARN NodeManager Daemons, MapReduce Hadoop-Aufgaben und Spark-Executors ausgeführt.

Es gibt nur eine Core-Instance-Gruppe oder Instance-Flotte pro Cluster, aber es können mehrere Knoten auf mehreren EC2 Amazon-Instances in der Instance-Gruppe oder Instance-Flotte laufen. Mit Instanzgruppen können Sie EC2 Amazon-Instances hinzufügen und entfernen, während der Cluster läuft. Sie können auch ein Auto Scaling einrichten, um Instances auf der Grundlage des Werts einer Metrik hinzuzufügen. Weitere Informationen zum Hinzufügen und Entfernen von EC2 Amazon-Instances mit der Instanzgruppen-Konfiguration finden Sie unterClusterskalierung verwenden.

Mit Instance-Flotten können Sie Instances effektiv hinzufügen und entfernen, indem Sie die Zielkapazitäten der Instance-Flotte für On-Demand- und Spot Instances entsprechend anpassen. Weitere Informationen zu den Zielkapazitäten finden Sie unter Instance-Flotten-Optionen.

Warning

Wenn Sie HDFS Daemons von einem laufenden Core-Node entfernen oder Core-Nodes beenden, besteht die Gefahr von Datenverlust. Seien Sie beim Konfigurieren von Core-Knoten für die Verwendung von Spot Instances vorsichtig. Weitere Informationen finden Sie unter Wann sollten Sie Spot Instances verwenden?.

Aufgabenknoten

Sie können Task-Knoten verwenden, um Leistung für parallel Berechnungsaufgaben für Daten hinzuzufügen, z. B. MapReduce Hadoop-Aufgaben und Spark-Executoren. Task-Knoten führen den Data Node-Daemon nicht aus und speichern auch keine Daten darin. HDFS Wie bei Core-Nodes können Sie Task-Knoten zu einem Cluster hinzufügen, indem Sie EC2 Amazon-Instances zu einer

bestehenden einheitlichen Instance-Gruppe hinzufügen oder indem Sie die Zielkapazitäten für eine Task-Instance-Flotte ändern.

Mit der einheitlichen Instance-Gruppenkonfiguration können Sie über bis zu 48 Aufgaben-Instance-Gruppen verfügen. Durch die Möglichkeit, Instance-Gruppen auf diese Weise hinzuzufügen, können Sie EC2 Amazon-Instance-Typen und Preisoptionen wie On-Demand-Instances und Spot-Instances kombinieren. Dadurch haben Sie die Flexibilität, kosteneffizient auf Workload-Anforderungen zu reagieren.

Mit der Instance-Flottenkonfiguration ist die Möglichkeit integriert, Instance-Typen und Kaufoptionen zu kombinieren, sodass nur eine Aufgaben-Instance-Flotte vorhanden ist.

Da Spot-Instances häufig zum Ausführen von Task-Knoten verwendet werden, EMR verfügt Amazon über Standardfunktionen für die Planung von YARN Jobs, sodass laufende Jobs nicht fehlschlagen, wenn Task-Knoten, die auf Spot-Instances ausgeführt werden, beendet werden. Amazon ermöglicht EMR dies, indem es die Ausführung von Anwendungsmasterprozessen nur auf Kernknoten zulässt. Der Anwendungsmasterprozess steuert die Ausführung von Aufträgen und muss während der gesamten Laufzeit des Auftrags aktiv bleiben.

EMRAmazon-Version 5.19.0 und höher verwendet die integrierte YARNNode Labels-Funktion, um dies zu erreichen. (Frühere Versionen verwendeten einen Code-Patch). Eigenschaften in den Klassifizierungen `yarn-site` und in der `capacity-scheduler` Konfiguration sind standardmäßig so konfiguriert, dass der YARN Capacity-Scheduler und der Fair-Scheduler die Vorteile von Node-Labels nutzen. Amazon kennzeichnet Kernknoten EMR automatisch mit dem `CORE` Label und legt Eigenschaften fest, sodass Anwendungsmaster nur für Knoten mit dem CORE Label geplant werden. Durch manuelles Ändern verwandter Eigenschaften in den Konfigurationsklassifizierungen von Yarn-Site und Capacity-Scheduler oder direkt in den zugehörigen XML Dateien könnte diese Funktion beeinträchtigt oder verändert werden.

Ab der Amazon EMR 6.x-Release-Serie ist die Funktion YARN Node Labels standardmäßig deaktiviert. Die Anwendungs-Primär-Prozesse können standardmäßig sowohl auf Core- als auch auf Aufgabenknoten ausgeführt werden. Sie können die Funktion „YARNNode Labels“ aktivieren, indem Sie die folgenden Eigenschaften konfigurieren:

- `yarn.node-labels.enabled: true`
- `yarn.node-labels.am.default-node-label-expression: 'CORE'`

Ab der Amazon EMR 7.x-Release-Serie EMR weist Amazon Instances YARN Node-Labels nach ihrem Markttyp zu, z. B. On-Demand oder Spot. Sie können Node-Labels aktivieren und

Anwendungsprozesse auf ON_ beschränken, DEMAND indem Sie die folgenden Eigenschaften konfigurieren:

```
yarn.node-labels.enabled: true
yarn.node-labels.am.default-node-label-expression: 'ON_DEMAND'
```

Wenn Sie Amazon EMR 7.0 oder höher verwenden, können Sie den Anwendungsprozess mit der folgenden Konfiguration auf Knoten mit dem `CODE` Label beschränken:

```
yarn.node-labels.enabled: true
yarn.node-labels.am.default-node-label-expression: 'CORE'
```

Wenn Ihr Cluster für EMR Amazon-Versionen 7.2 und höher verwaltete Skalierung mit Knotenbezeichnungen verwendet, versucht AmazonEMR, den Cluster unabhängig vom Anwendungsprozess und der Nachfrage der Executoren zu skalieren.

Wenn Sie beispielsweise EMR Amazon-Versionen 7.2 oder höher verwenden und den Anwendungsprozess auf `ON_DEMAND` Knoten beschränken, skaliert Managed Scaling die `ON_DEMAND` Knoten nach oben, wenn die Nachfrage nach Anwendungsprozessen steigt. Ähnlich verhält es sich, wenn Sie den Anwendungsprozess auf `CORE` Knoten beschränken, bei verwalteter Skalierung die `CORE` Knoten hochskaliert, wenn die Nachfrage nach Anwendungsprozessen steigt.

Informationen zu spezifischen Eigenschaften finden Sie unter EMRAmazon-Einstellungen zur Vermeidung von Auftragsausfällen aufgrund der Kündigung der Spot-Instance des Task-Knotens.

EC2Amazon-Instances konfigurieren

EC2Instances gibt es in verschiedenen Konfigurationen, die als Instance-Typen bezeichnet werden. Instance-Typen haben unterschiedliche CPU Eingabe-/Ausgabe- und Speicherkapazitäten. Zusätzlich zum Instance-Typ können Sie verschiedene Kaufoptionen für EC2 Amazon-Instances wählen. Sie können verschiedene Instance-Typen und Kaufoptionen innerhalb von einheitlichen Instance-Gruppen oder Instance-Flotten angeben. Weitere Informationen finden Sie unter Einen Cluster mit Instance-Flotten oder einheitlichen Instance-Gruppen erstellen. Hinweise zur Auswahl von Instance-Typen und Kaufoptionen für Ihre Anwendung finden Sie unter Bewährte Methoden für die Konfiguration des Clusters.

> **Important**
>
> Wenn Sie mithilfe von einen Instance-Typ auswählen AWS Management Console, entspricht die Anzahl von v, die für jeden Instance-Typ CPU angezeigt wird, der Anzahl der YARN virtuellen Kerne für diesen Instance-Typ, nicht der Anzahl von EC2 vCPUs für diesen Instance-Typ. Weitere Informationen zur Anzahl der vCPUs einzelnen Instance-Typen finden Sie unter EC2Amazon-Instance-Typen.

Themen

Unterstützte Instance-Typen

In diesem Abschnitt werden die Instance-Typen beschrieben, die Amazon EMR unterstützt, geordnet nach AWS-Region. Weitere Informationen zu Instance-Typen finden Sie unter EC2Amazon-Instances und Amazon AMI Linux-Instance-Typmatrix.

Nicht alle Instance-Typen sind in allen Regionen verfügbar. Die Instance-Verfügbarkeit hängt von der Verfügbarkeit und der Nachfrage in der angegebenen Region und Availability Zone ab. Die Availability Zone einer Instance wird durch das Subnetz bestimmt, das Sie zum Starten Ihres Clusters verwenden.

Überlegungen

Beachten Sie Folgendes, wenn Sie Instance-Typen für Ihren EMR Amazon-Cluster auswählen.

> **Important**
>
> Wenn Sie mithilfe von einen Instance-Typ auswählen AWS Management Console, entspricht die Anzahl von v, die für jeden Instance-Typ CPU angezeigt wird, der Anzahl der YARN virtuellen Kerne für diesen Instance-Typ, nicht der Anzahl von EC2 vCPUs für diesen Instance-Typ. Weitere Informationen zur Anzahl der vCPUs einzelnen Instance-Typen finden Sie unter EC2Amazon-Instance-Typen.

- Wenn Sie einen Cluster mit einem Instance-Typ erstellen, der in der angegebenen Region und Verfügbarkeitszone nicht verfügbar ist, schlägt die Bereitstellung Ihres Clusters möglicherweise fehl oder die Bereitstellung bleibt hängen. Informationen zur Instance-Verfügbarkeit finden Sie auf der EMRAmazon-Preisseite oder in den Unterstützte Instanztypen von AWS-Region Tabellen auf dieser Seite.
- Ab der EMR Amazon-Release-Version 5.13.0 verwenden alle Instances HVM Virtualisierung und EBS -gestützten Speicher für Root-Volumes. Bei Verwendung von EMR Amazon-Release-Versionen vor 5.13.0 verwenden PVM einige Instances der vorherigen Generation Virtualisierung. Weitere Informationen finden Sie unter AMILinux-Virtualisierungstypen.
- Aufgrund mangelnder Hardwareunterstützung und fehlender Standardeinstellungen, die zu einer Unterauslastung von Arbeitsspeicher und Kernen führen können, empfehlen wir Ihnen nicht, die Instance-Typen,,,,,,, zu verwenden,,,,,,,`c7a`,,,,,,,,,,,`c7i`,,`m7i`,,`m7i-flex`,,`r7a`,,`r7i`,,`r7iz`,,`i4i.12xlarge`,,,,, verwenden, `i4i.24xlarge` wenn Sie EMR Amazon-Versionen unter 5.36.1 und 6.10.0 ausführen. Wenn Sie diese Instance-Typen in diesen Versionen ausführen, kann es zu Leistungseinbußen kommen und Sie werden nicht die erwarteten Vorteile neuerer Instance-Typen sehen, wie z. B. vs. `c7i` `c6i` Für eine optimale Ressourcennutzung und Leistung bei diesen Leistungstypen sollten Sie 5.36.1 und höher oder 6.10.0 und höher ausführen, um deren Funktionen zu maximieren.
- Einige Instance-Typen unterstützen Enhanced Networking. Weitere Informationen finden Sie unter Enhanced Networking in Linux.
- NVIDIAund CUDA Treiber werden standardmäßig auf GPU Instance-Typen installiert.

Unterstützte Instanztypen von AWS-Region

In den folgenden Tabellen sind die EC2 Amazon-Instance-Typen aufgeführt, die Amazon EMR unterstützt, geordnet nach AWS-Region. In den Tabellen sind auch die frühesten EMR Amazon-Versionen der Serien 5.x, 6.x und 7.x aufgeführt, die jeden Instance-Typ unterstützen.

USA Ost (Nord-Virginia) – us-east-1

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5dn.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5zn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.3xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.6xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6g.xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.2xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.4xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.8xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.12xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6g.16xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6idn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6idn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.32xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.2xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.4xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.8xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5a.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5a.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5ad.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.24xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5d.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5n.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.9xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.18xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7a.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.32xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7g.xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.2xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.4xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7g.8xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.12xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.16xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7gn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
Beschleunigte Datenverarbeitung	g3.4xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	g3.8xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3.16xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3s.xlarge	emr-5.19.0, emr-6.0.0, emr-7.0.0
	g4dn.xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.2xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.4xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.8xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.12xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.16xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g5.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	g5.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.48xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g6.xlarge	emr-7.2.0
	g6.2xlarge	emr-7.2.0
	g6.4xlarge	emr-7.2.0
	g6.8xlarge	emr-7.2.0
	g6.12xlarge	emr-7.2.0
	g6.16xlarge	emr-7.2.0
	g6.24xlarge	emr-7.2.0
	g6.48xlarge	emr-7.2.0
	gr6.4xlarge	emr-7.2.0
	gr6.8xlarge	emr-7.2.0
	p2.xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	p2.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p2.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.2xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p5.48xlarge	emr-6.14.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5b.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.24xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5dn.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6idn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6idn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7a.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.32xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.32xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x1.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1e.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x2gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	z1d.xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.3xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.6xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.12xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
Speicheroptimiert	d3.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	d3en.6xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	h1.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	h1.4xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	h1.8xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	h1.16xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	i3.xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.2xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.4xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.8xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.16xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3en.xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.2xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.3xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.6xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.12xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.24xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i4g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	im4gn.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	is4gen.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

USA Ost (Ohio) – us-east-2

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5dn.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5dn.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5zn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.3xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.6xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6g.xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.2xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.4xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.8xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.12xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.16xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6idn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6idn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7a.2xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	m7a.4xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	m7a.8xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	m7a.12xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	m7a.16xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	m7a.24xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	m7a.32xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	m7a.48xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	m7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	m7i.2xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	m7i.4xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	m7i.8xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7i.12xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	m7i.16xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	m7i.24xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	m7i.48xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	m7i-flex.xlarge	emr-4.6.0, emr-5.0.1, emr-6.0.0, emr-7.0.0
	m7i-flex.2xlarge	emr-4.6.0, emr-5.0.1, emr-6.0.0, emr-7.0.0
	m7i-flex.4xlarge	emr-4.6.0, emr-5.0.1, emr-6.0.0, emr-7.0.0
	m7i-flex.8xlarge	emr-4.6.0, emr-5.0.1, emr-6.0.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5a.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5ad.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5ad.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.24xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5n.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.9xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.18xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7a.xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	c7a.2xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	c7a.4xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	c7a.8xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7a.12xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	c7a.16xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	c7a.24xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	c7a.32xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	c7a.48xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	c7g.xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.2xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.4xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.8xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.12xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.16xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7i.xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7i.2xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	c7i.4xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	c7i.8xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	c7i.12xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	c7i.16xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	c7i.24xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	c7i.48xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
Beschleunigte Datenverarbeitung	g3.4xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3.8xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3.16xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3s.xlarge	emr-5.19.0, emr-6.0.0, emr-7.0.0
	g4dn.xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	g4dn.2xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.4xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.8xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.12xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.16xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g5.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	g5.48xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g6.xlarge	emr-7.2.0
	g6.2xlarge	emr-7.2.0
	g6.4xlarge	emr-7.2.0
	g6.8xlarge	emr-7.2.0
	g6.12xlarge	emr-7.2.0
	g6.16xlarge	emr-7.2.0
	g6.24xlarge	emr-7.2.0
	g6.48xlarge	emr-7.2.0
	gr6.4xlarge	emr-7.2.0
	gr6.8xlarge	emr-7.2.0
	p2.xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p2.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p2.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.2xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	p3.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p5.48xlarge	emr-6.14.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5b.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5b.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.24xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5dn.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6idn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7a.xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	r7a.2xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	r7a.4xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	r7a.8xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7a.12xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	r7a.16xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	r7a.24xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	r7a.32xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	r7a.48xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	r7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	r7i.2xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	r7i.4xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	r7i.8xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	r7i.12xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	r7i.16xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	r7i.24xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7i.48xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	r7iz.xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	r7iz.2xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	r7iz.4xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	r7iz.8xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	r7iz.12xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	r7iz.16xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	r7iz.32xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	x1.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1e.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x1e.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	z1d.xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.3xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.6xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	z1d.12xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
Speicheroptimiert	d3.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	h1.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	h1.4xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	h1.8xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	h1.16xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	i3.xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.2xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.4xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3.8xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.16xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3en.xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.2xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.3xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.6xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.12xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.24xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i4g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i4g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.24xlarge	emr-5.0.1, emr-6.0.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	im4gn.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	im4gn.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

USA West (Nordkalifornien) – us-west-1

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5zn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.3xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.6xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6g.xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.2xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.4xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.8xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.12xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.16xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6idn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6idn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7i-flex.2xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.4xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.8xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5a.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5a.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5n.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.9xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.18xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7g.xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.2xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.4xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.8xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.12xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.16xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
Beschleunigte Datenverarbeitung	g3.4xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3.8xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3.16xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g4dn.xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	g4dn.2xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.4xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.8xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.12xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.16xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x2iedn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	z1d.xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.3xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.6xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.12xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
Speicheroptimiert	i3.xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.2xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.4xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.8xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3.16xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3en.xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.2xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.3xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.6xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.12xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.24xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

USA West (Oregon) – us-west-2

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5dn.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5dn.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5zn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.3xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.6xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6g.xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.2xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.4xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.8xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.12xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.16xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6idn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6idn.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7a.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.32xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.2xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.4xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.8xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5a.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5ad.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.24xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5n.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5n.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.9xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.18xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7a.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7a.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.32xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7g.xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.2xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.4xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.8xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.12xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.16xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
Beschleunigte Datenverarbeitung	g3.4xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3.8xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3.16xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3s.xlarge	emr-5.19.0, emr-6.0.0, emr-7.0.0
	g4dn.xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.2xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	g4dn.4xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.8xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.12xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.16xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g5.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.48xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	g6.xlarge	emr-7.2.0
	g6.2xlarge	emr-7.2.0
	g6.4xlarge	emr-7.2.0
	g6.8xlarge	emr-7.2.0
	g6.12xlarge	emr-7.2.0
	g6.16xlarge	emr-7.2.0
	g6.24xlarge	emr-7.2.0
	g6.48xlarge	emr-7.2.0
	gr6.4xlarge	emr-7.2.0
	gr6.8xlarge	emr-7.2.0
	p2.xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p2.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p2.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.2xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	p5.48xlarge	emr-6.14.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5b.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5b.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.24xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5dn.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6idn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7a.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7a.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.32xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7iz.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.32xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	x1.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1e.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x1e.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x2iedn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	z1d.xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.3xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.6xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.12xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Speicheroptimiert	d3.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.6xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	h1.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	h1.4xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	h1.8xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	h1.16xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	i3.xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.2xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.4xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.8xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.16xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3en.xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.2xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.3xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.6xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.12xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3en.24xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i4g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i4i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	im4gn.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

AWS GovCloud (US-West) - -1 us-gov-west

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5dn.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m6g.xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.2xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.4xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.8xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.12xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6g.16xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6idn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6idn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.48xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i-flex.xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7i-flex.2xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.4xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.8xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5a.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5a.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5n.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.9xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.18xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
Beschleunigte Datenverarbeitung	g3.4xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	g3.8xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3.16xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g4dn.xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.2xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.4xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.8xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.12xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.16xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	p2.xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p2.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p2.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.2xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	p3.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5dn.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5dn.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6idn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.48xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x1.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1e.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x2iedn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
Speicheroptimiert	d3.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	i3.xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.2xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3.4xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.8xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.16xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3en.xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.2xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.3xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.6xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.12xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.24xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

AWS GovCloud (US-Ost) - -1 us-gov-east

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	m5.2xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	m5.4xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	m5.8xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5.12xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	m5.16xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	m5.24xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	m5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5d.xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	m5d.2xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5d.4xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	m5d.8xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	m5d.12xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	m5d.16xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	m5d.24xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	m5dn.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m6g.xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.2xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.4xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.8xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.12xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6g.16xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m7i.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.48xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i-flex.xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7i-flex.2xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	m7i-flex.4xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	m7i-flex.8xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	c5.2xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	c5.4xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	c5.9xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	c5.12xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	c5.18xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	c5.24xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	c5a.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5a.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5d.xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	c5d.2xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	c5d.4xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	c5d.9xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	c5d.18xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	c5n.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5n.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.9xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.18xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7i.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7i.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7i.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7i.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7i.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7i.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7i.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7i.48xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
Beschleunigte Datenverarbeitung	g4dn.xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.2xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.4xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	g4dn.8xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.12xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.16xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	r5.2xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	r5.4xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	r5.8xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	r5.12xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	r5.16xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	r5.24xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	r5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5d.xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	r5d.2xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	r5d.4xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	r5d.8xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	r5d.12xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	r5d.16xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	r5d.24xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5dn.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r7i.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7i.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.48xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1e.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x1e.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Speicheroptimiert	i3.xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	i3.2xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	i3.4xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	i3.8xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	i3.16xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	i3en.xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.2xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.3xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.6xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.12xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.24xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.24xlarge	emr-5.16.0, emr-6.0.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Afrika (Kapstadt) – af-south-1

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5.2xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5.4xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5.8xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5.12xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5.16xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5.24xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5d.xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5d.2xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5d.4xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5d.8xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5d.12xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5d.16xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5d.24xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6g.xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.2xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.4xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.8xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.12xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.16xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5.2xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5.4xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5.9xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5.12xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5.18xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5.24xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5a.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5ad.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5ad.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.24xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5d.xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5d.2xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5d.4xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5d.9xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5d.12xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5d.18xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5d.24xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5n.xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5n.2xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5n.4xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5n.9xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5n.18xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
Beschleunigte Datenverarbeitung	g4dn.xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.2xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.4xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.8xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.12xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.16xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5.2xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5.4xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5.8xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5.12xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5.16xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5.24xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5d.xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5d.2xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5d.4xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5d.8xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5d.12xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5d.16xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5d.24xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5dn.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5dn.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	x1.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1e.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
Speicheroptimiert	i3.xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	i3.2xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	i3.4xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3.8xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	i3.16xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	i3en.xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	i3en.2xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	i3en.3xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	i3en.6xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	i3en.12xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	i3en.24xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i4i.12xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.24xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Asien-Pazifik (Hongkong) – ap-east-1

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5d.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5d.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5d.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5d.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5d.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5d.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5d.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m6g.xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.2xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.4xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6g.8xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.12xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.16xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5.9xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5.18xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5a.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5d.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5d.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5d.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5d.9xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5d.18xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5n.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.9xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.18xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6gn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6gn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7g.xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.2xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.4xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.8xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.12xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.16xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Beschleunigte Datenverarbeitung	g4dn.xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.2xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.4xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.8xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.12xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.16xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5d.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5d.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5d.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5d.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5d.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5d.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5d.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x1.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
Speicheroptimiert	i3.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	i3.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	i3.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	i3.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	i3.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	i3en.xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.2xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.3xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.6xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.12xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.24xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Asien-Pazifik (Jakarta) – ap-southeast-3

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5.2xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	m5.4xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	m5.8xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	m5.12xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	m5.16xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	m5.24xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	m5d.xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	m5d.2xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	m5d.4xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	m5d.8xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	m5d.12xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	m5d.16xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5d.24xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	m6g.xlarge	emr-5.30.2, emr-6.1.1, emr-7.0.0
	m6g.2xlarge	emr-5.30.2, emr-6.1.1, emr-7.0.0
	m6g.4xlarge	emr-5.30.2, emr-6.1.1, emr-7.0.0
	m6g.8xlarge	emr-5.30.2, emr-6.1.1, emr-7.0.0
	m6g.12xlarge	emr-5.30.2, emr-6.1.1, emr-7.0.0
	m6g.16xlarge	emr-5.30.2, emr-6.1.1, emr-7.0.0
	m6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	c5.2xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	c5.4xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5.9xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	c5.12xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	c5.18xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	c5.24xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	c5d.xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	c5d.2xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	c5d.4xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	c5d.9xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	c5d.12xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	c5d.18xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	c5d.24xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	c5n.xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5n.2xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	c5n.4xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	c5n.9xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	c5n.18xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	c6g.xlarge	emr-5.31.1, emr-6.1.1, emr-7.0.0
	c6g.2xlarge	emr-5.31.1, emr-6.1.1, emr-7.0.0
	c6g.4xlarge	emr-5.31.1, emr-6.1.1, emr-7.0.0
	c6g.8xlarge	emr-5.31.1, emr-6.1.1, emr-7.0.0
	c6g.12xlarge	emr-5.31.1, emr-6.1.1, emr-7.0.0
	c6g.16xlarge	emr-5.31.1, emr-6.1.1, emr-7.0.0
	c6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
Beschleunigte Datenverarbeitung	g5.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	g5.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.48xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	r5.2xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	r5.4xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	r5.8xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	r5.12xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	r5.16xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	r5.24xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	r5d.xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	r5d.2xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	r5d.4xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5d.8xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	r5d.12xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	r5d.16xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	r5d.24xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	r6g.xlarge	emr-5.31.1, emr-6.1.1, emr-7.0.0
	r6g.2xlarge	emr-5.31.1, emr-6.1.1, emr-7.0.0
	r6g.4xlarge	emr-5.31.1, emr-6.1.1, emr-7.0.0
	r6g.8xlarge	emr-5.31.1, emr-6.1.1, emr-7.0.0
	r6g.12xlarge	emr-5.31.1, emr-6.1.1, emr-7.0.0
	r6g.16xlarge	emr-5.31.1, emr-6.1.1, emr-7.0.0
	r6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r7i.xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	r7i.2xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	r7i.4xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	r7i.8xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	r7i.12xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	r7i.16xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	r7i.24xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	r7i.48xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
Speicheroptimiert	i3.xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	i3.2xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3.4xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	i3.8xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	i3.16xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	i3en.xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	i3en.2xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	i3en.3xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	i3en.6xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	i3en.12xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	i3en.24xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.24xlarge	emr-5.30.2, emr-6.0.1, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Asien-Pazifik (Mumbai) – ap-south-1

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6g.xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.2xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.4xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6g.8xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.12xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.16xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6idn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.xlarge	emr-4.6.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.2xlarge	emr-4.6.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.4xlarge	emr-4.6.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.8xlarge	emr-4.6.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5a.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5d.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5n.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.9xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.18xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7g.xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.2xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.4xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.8xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.12xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.16xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
Beschleunigte Datenverarbeitung	g4dn.xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.2xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	g4dn.4xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.8xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.12xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.16xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g5.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.48xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	p2.xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p2.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p2.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	x1.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x1.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1e.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x2iedn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	z1d.xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.3xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.6xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.12xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
Speicheroptimiert	d3.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	d3.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	i3.xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.2xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.4xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.8xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.16xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3en.xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.2xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.3xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.6xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.12xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3en.24xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	is4gen.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	is4gen.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Asien-Pazifik (Hyderabad) – ap-south-2

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5d.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5d.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5d.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5d.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5d.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5d.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5d.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6g.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6g.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6g.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6g.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6g.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6g.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6gd.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6gd.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6gd.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6gd.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6gd.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6gd.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6i.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6i.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6i.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6i.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6i.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6i.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6i.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6i.32xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5.9xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5.18xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5d.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5d.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5d.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5d.9xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5d.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5d.18xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5d.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6g.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6g.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6g.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6g.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6g.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6g.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6i.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6i.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6i.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6i.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6i.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6i.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6i.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6i.32xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7g.xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.2xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.4xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.8xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.12xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.16xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5d.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5d.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5d.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5d.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5d.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5d.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5d.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6g.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6g.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6g.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6g.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6g.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6g.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6i.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6i.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6i.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6i.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6i.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6i.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6i.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6i.32xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x2iedn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
Speicheroptimiert	i3.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3en.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3en.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3en.3xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3en.6xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3en.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3en.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i4i.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Asien-Pazifik (Osaka) – ap-northeast-3

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m6g.xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.2xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.4xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.8xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.12xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6g.16xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5n.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.9xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.18xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6gn.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Beschleunigte Datenverarbeitung	g4dn.xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.2xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.4xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.8xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.12xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.16xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	x1.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1e.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
Speicheroptimiert	i3.xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	i3.2xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3.4xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	i3.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	i3.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	i3en.xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.2xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.3xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.6xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.12xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.24xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.24xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Asien-Pazifik (Seoul) – ap-northeast-2

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5zn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.3xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.6xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6g.xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.2xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.4xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.8xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.12xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.16xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7i-flex.2xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.4xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.8xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5a.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5a.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5n.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.9xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.18xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7g.xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7g.2xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.4xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.8xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.12xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.16xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
Beschleunigte Datenverarbeitung	g3.4xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3.8xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3.16xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3s.xlarge	emr-5.19.0, emr-6.0.0, emr-7.0.0
	g4dn.xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.2xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.4xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.8xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.12xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.16xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g5.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	g5.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.48xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	p2.xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p2.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p2.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.2xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	p3.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5b.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5b.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.24xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5dn.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	x1.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1e.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x1e.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x2iedn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	z1d.xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.3xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.6xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.12xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
Speicheroptimiert	i3.xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.2xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.4xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.8xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.16xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3en.xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.2xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.3xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.6xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.12xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.24xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i4i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Asien-Pazifik (Singapur) – ap-southeast-1

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5dn.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5dn.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5zn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5zn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.3xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.6xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6g.xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.2xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.4xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.8xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.12xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.16xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6idn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6idn.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.2xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.4xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.8xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5a.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5ad.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5ad.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.24xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5n.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.9xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.18xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7g.xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.2xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.4xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7g.8xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.12xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.16xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
Beschleunigte Datenverarbeitung	g3.4xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3.8xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3.16xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g4dn.xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.2xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.4xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.8xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	g4dn.12xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.16xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	p2.xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p2.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p2.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.2xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5b.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.24xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5dn.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5dn.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6idn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6idn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	x1.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x1e.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x2iedn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	z1d.xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.3xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.6xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.12xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
Speicheroptimiert	d3.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	d3.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.6xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	i3.xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.2xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.4xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.8xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.16xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3en.xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.2xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.3xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.6xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.12xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.24xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i4g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	im4gn.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	is4gen.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Asien-Pazifik (Sydney) – ap-southeast-2

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5zn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5zn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.3xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.6xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6g.xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.2xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.4xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.8xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.12xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.16xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6idn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6idn.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.2xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.4xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.8xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5a.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5ad.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5ad.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.24xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5n.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.9xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.18xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7g.xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.2xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.4xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7g.8xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.12xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.16xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
Beschleunigte Datenverarbeitung	g3.4xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3.8xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3.16xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3s.xlarge	emr-5.19.0, emr-6.0.0, emr-7.0.0
	g4dn.xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.2xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.4xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	g4dn.8xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.12xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.16xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g5.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.48xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	p2.xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	p2.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p2.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.2xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5b.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.24xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5dn.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	x1.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1e.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	z1d.xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.3xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	z1d.6xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.12xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
Speicheroptimiert	d3.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	i3.xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.2xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.4xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.8xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.16xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3en.xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3en.2xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.3xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.6xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.12xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.24xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	im4gn.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Asien-Pazifik (Tokio) – ap-northeast-1

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5dn.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5zn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.3xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.6xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6g.xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.2xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.4xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6g.8xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.12xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.16xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6idn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7a.32xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.2xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7i-flex.4xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.8xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5a.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5a.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5n.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5n.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.9xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.18xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7a.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7a.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.32xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7g.xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.2xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.4xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.8xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.12xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.16xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
Beschleunigte Datenverarbeitung	g3.4xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3.8xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3.16xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3s.xlarge	emr-5.19.0, emr-6.0.0, emr-7.0.0
	g4dn.xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.2xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	g4dn.4xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.8xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.12xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.16xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g5.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.48xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	p2.xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p2.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p2.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.2xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5b.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.24xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5dn.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6idn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6idn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7a.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.32xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.32xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x1.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1e.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x2iedn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	z1d.xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.3xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.6xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.12xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
Speicheroptimiert	d3.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	d3.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.6xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	i3.xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.2xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.4xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3.8xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.16xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3en.xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.2xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.3xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.6xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.12xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.24xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i4i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	im4gn.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	is4gen.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Kanada (Zentral) – ca-central-1

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6g.xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.2xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.4xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.8xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.12xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.16xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6idn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6idn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.xlarge	emr-5.0.2, emr-6.0.0, emr-7.0.0
	m7i.2xlarge	emr-5.0.2, emr-6.0.0, emr-7.0.0
	m7i.4xlarge	emr-5.0.2, emr-6.0.0, emr-7.0.0
	m7i.8xlarge	emr-5.0.2, emr-6.0.0, emr-7.0.0
	m7i.12xlarge	emr-5.0.2, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7i.16xlarge	emr-5.0.2, emr-6.0.0, emr-7.0.0
	m7i.24xlarge	emr-5.0.2, emr-6.0.0, emr-7.0.0
	m7i.48xlarge	emr-5.0.2, emr-6.0.0, emr-7.0.0
	m7i-flex.xlarge	emr-4.8.2, emr-5.0.2, emr-6.0.0, emr-7.0.0
	m7i-flex.2xlarge	emr-4.8.2, emr-5.0.2, emr-6.0.0, emr-7.0.0
	m7i-flex.4xlarge	emr-4.8.2, emr-5.0.2, emr-6.0.0, emr-7.0.0
	m7i-flex.8xlarge	emr-4.8.2, emr-5.0.2, emr-6.0.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5a.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5d.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5n.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.9xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.18xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7g.xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.2xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.4xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.8xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.12xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.16xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7i.xlarge	emr-5.0.2, emr-6.0.0, emr-7.0.0
	c7i.2xlarge	emr-5.0.2, emr-6.0.0, emr-7.0.0
	c7i.4xlarge	emr-5.0.2, emr-6.0.0, emr-7.0.0
	c7i.8xlarge	emr-5.0.2, emr-6.0.0, emr-7.0.0
	c7i.12xlarge	emr-5.0.2, emr-6.0.0, emr-7.0.0
	c7i.16xlarge	emr-5.0.2, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7i.24xlarge	emr-5.0.2, emr-6.0.0, emr-7.0.0
	c7i.48xlarge	emr-5.0.2, emr-6.0.0, emr-7.0.0
Beschleunigte Datenverarbeitung	g3.4xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3.8xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3.16xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g4dn.xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.2xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.4xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.8xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.12xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.16xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g5.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	g5.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.48xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	p3.2xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5b.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5b.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.24xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.xlarge	emr-5.0.2, emr-6.0.0, emr-7.0.0
	r7i.2xlarge	emr-5.0.2, emr-6.0.0, emr-7.0.0
	r7i.4xlarge	emr-5.0.2, emr-6.0.0, emr-7.0.0
	r7i.8xlarge	emr-5.0.2, emr-6.0.0, emr-7.0.0
	r7i.12xlarge	emr-5.0.2, emr-6.0.0, emr-7.0.0
	r7i.16xlarge	emr-5.0.2, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7i.24xlarge	emr-5.0.2, emr-6.0.0, emr-7.0.0
	r7i.48xlarge	emr-5.0.2, emr-6.0.0, emr-7.0.0
	x1.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1e.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
Speicheroptimiert	d3.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3.xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.2xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.4xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.8xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.16xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3en.xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.2xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.3xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.6xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.12xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.24xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i4g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i4g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.0.2, emr-6.0.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.24xlarge	emr-5.0.2, emr-6.0.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	im4gn.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Kanada West (Calgary) – ca-west-1

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m5.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m5.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m5.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m5.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m5.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m5.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m5d.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m5d.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m5d.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m5d.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5d.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m5d.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m5d.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m6g.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m6g.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m6g.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m6g.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m6g.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m6g.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m6gd.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m6gd.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m6gd.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6gd.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m6gd.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m6gd.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m6i.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m6i.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m6i.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m6i.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m6i.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m6i.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m6i.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m6i.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m6id.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6id.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m6id.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m6id.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m6id.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m6id.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m6id.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	m6id.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c5.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c5.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c5.9xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c5.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5.18xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c5.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c6g.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c6g.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c6g.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c6g.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c6g.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c6g.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c6gn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c6gn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c6gn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c6gn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6gn.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c6gn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c6i.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c6i.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c6i.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c6i.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c6i.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c6i.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c6i.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c6i.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c6id.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c6id.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6id.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c6id.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c6id.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c6id.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c6id.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	c6id.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	r5.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	r5.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	r5.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	r5.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	r5.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	r6g.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	r6g.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	r6g.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	r6g.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	r6g.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	r6g.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	r6i.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	r6i.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	r6i.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	r6i.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	r6i.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6i.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	r6i.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	r6i.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	r6id.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	r6id.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	r6id.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	r6id.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	r6id.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	r6id.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	r6id.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	r6id.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
Speicheroptimiert	i3en.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3en.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	i3en.3xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	i3en.6xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	i3en.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	i3en.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	i4i.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	i4i.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

China (Ningxia) – cn-northwest-1

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m6g.xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.2xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6g.4xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.8xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.12xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.16xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5a.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5d.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7g.xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7g.2xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.4xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.8xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.12xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.16xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
Beschleunigte Datenverarbeitung	g4dn.xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.2xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.4xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.8xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.12xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.16xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g5.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	g5.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.48xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	p3.2xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	z1d.xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.3xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	z1d.6xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.12xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
Speicheroptimiert	i3.xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.2xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.4xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.8xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.16xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3en.xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.2xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.3xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.6xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.12xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3en.24xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.5.3, emr-6.0.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.24xlarge	emr-5.5.3, emr-6.0.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

China (Peking) – cn-north-1

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m6g.xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.2xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6g.4xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.8xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.12xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.16xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5a.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6gn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6gn.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c7g.xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7g.2xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.4xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.8xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.12xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.16xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
Beschleunigte Datenverarbeitung	g3.4xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3.8xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3.16xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3s.xlarge	emr-5.19.0, emr-6.0.0, emr-7.0.0
	g4dn.xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.2xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.4xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	g4dn.8xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.12xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.16xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g5.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.48xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	p2.xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	p2.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p2.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.2xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x2iedn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
Speicheroptimiert	i3.xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.2xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.4xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.8xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.16xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3en.xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.2xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.3xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3en.6xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.12xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.24xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Europa (Frankfurt) – eu-central-1

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5dn.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5zn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.3xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.6xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6g.xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.2xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.4xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6g.8xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.12xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.16xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6idn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7a.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.48xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.2xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7i-flex.4xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.8xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5a.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5a.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5ad.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.24xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5n.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.9xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.18xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6gn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7a.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.32xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7g.xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7g.2xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.4xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.8xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.12xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.16xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
Beschleunigte Datenverarbeitung	g3.4xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3.8xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3.16xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3s.xlarge	emr-5.19.0, emr-6.0.0, emr-7.0.0
	g4dn.xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	g4dn.2xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.4xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.8xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.12xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.16xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g5.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	g5.48xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	p2.xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p2.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p2.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.2xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5b.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.24xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5dn.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5dn.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6idn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6idn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7a.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.32xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7iz.32xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	x1.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1e.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x2iedn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	z1d.xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.3xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.6xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.12xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Speicheroptimiert	d3.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.6xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	i3.xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.2xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3.4xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.8xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.16xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3en.xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.2xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.3xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.6xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.12xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.24xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	im4gn.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	is4gen.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Europa (Zürich) – eu-central-2

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5d.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5d.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5d.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5d.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5d.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5d.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5d.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6g.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6g.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6g.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6g.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6g.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6g.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6gd.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6gd.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6gd.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6gd.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6gd.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6gd.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6i.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6i.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6i.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6i.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6i.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6i.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6i.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6i.32xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5.9xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5.18xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5d.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5d.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5d.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5d.9xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5d.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5d.18xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5d.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6g.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6g.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6g.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6g.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6g.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6g.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6gd.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6gd.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6gd.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6gd.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6gd.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6gd.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5d.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5d.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5d.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5d.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5d.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5d.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5d.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6g.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6g.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6g.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6g.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6g.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6g.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6gd.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6gd.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6gd.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6gd.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6gd.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6gd.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6i.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6i.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6i.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6i.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6i.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6i.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6i.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6i.32xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
Speicheroptimiert	d3.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	d3.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	d3.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	d3.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3en.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3en.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3en.3xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3en.6xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3en.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3en.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Europa (Irland) – eu-west-1

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5dn.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5dn.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	m5zn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.3xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.6xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6g.xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.2xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.4xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6g.8xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.12xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.16xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6idn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7a.32xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7a.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.2xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7i-flex.4xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.8xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5a.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5a.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5ad.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.24xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5n.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.9xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.18xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6gn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7a.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.32xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7a.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7g.xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7g.2xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.4xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.8xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.12xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.16xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7gn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
Beschleunigte Datenverarbeitung	g3.4xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3.8xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3.16xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3s.xlarge	emr-5.19.0, emr-6.0.0, emr-7.0.0
	g4dn.xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.2xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.4xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.8xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.12xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.16xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g5.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	g5.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.48xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	p2.xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p2.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p2.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.2xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	p3.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5b.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5b.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.24xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5dn.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6idn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7a.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7a.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.32xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7a.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7iz.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7iz.32xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	x1.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1e.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x1e.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x2iedn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	z1d.xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.3xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.6xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.12xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Speicheroptimiert	d3.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.6xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3en.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	h1.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	h1.4xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	h1.8xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	h1.16xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	i3.xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.2xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.4xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.8xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.16xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3en.xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.2xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.3xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.6xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.12xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3en.24xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i4g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i4i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	im4gn.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Europa (London) – eu-west-2

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6g.xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.2xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.4xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.8xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.12xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.16xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7i.xlarge	emr-5.0.3, emr-6.0.0, emr-7.0.0
	m7i.2xlarge	emr-5.0.3, emr-6.0.0, emr-7.0.0
	m7i.4xlarge	emr-5.0.3, emr-6.0.0, emr-7.0.0
	m7i.8xlarge	emr-5.0.3, emr-6.0.0, emr-7.0.0
	m7i.12xlarge	emr-5.0.3, emr-6.0.0, emr-7.0.0
	m7i.16xlarge	emr-5.0.3, emr-6.0.0, emr-7.0.0
	m7i.24xlarge	emr-5.0.3, emr-6.0.0, emr-7.0.0
	m7i.48xlarge	emr-5.0.3, emr-6.0.0, emr-7.0.0
	m7i-flex.xlarge	emr-4.8.2, emr-5.0.3, emr-6.0.0, emr-7.0.0
	m7i-flex.2xlarge	emr-4.8.2, emr-5.0.3, emr-6.0.0, emr-7.0.0
	m7i-flex.4xlarge	emr-4.8.2, emr-5.0.3, emr-6.0.0, emr-7.0.0
	m7i-flex.8xlarge	emr-4.8.2, emr-5.0.3, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5a.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5a.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5n.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5n.9xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.18xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6gn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7g.xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.2xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.4xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.8xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.12xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.16xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7i.xlarge	emr-5.0.3, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7i.2xlarge	emr-5.0.3, emr-6.0.0, emr-7.0.0
	c7i.4xlarge	emr-5.0.3, emr-6.0.0, emr-7.0.0
	c7i.8xlarge	emr-5.0.3, emr-6.0.0, emr-7.0.0
	c7i.12xlarge	emr-5.0.3, emr-6.0.0, emr-7.0.0
	c7i.16xlarge	emr-5.0.3, emr-6.0.0, emr-7.0.0
	c7i.24xlarge	emr-5.0.3, emr-6.0.0, emr-7.0.0
	c7i.48xlarge	emr-5.0.3, emr-6.0.0, emr-7.0.0
Beschleunigte Datenverarbeitung	g3.4xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3.8xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3.16xlarge	emr-5.18.0, emr-6.0.0, emr-7.0.0
	g3s.xlarge	emr-5.19.0, emr-6.0.0, emr-7.0.0
	g4dn.xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	g4dn.2xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.4xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.8xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.12xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.16xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g5.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	g5.48xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	p3.2xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.8xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
	p3.16xlarge	emr-5.10.0, emr-6.0.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5b.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.24xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	r7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.xlarge	emr-5.0.3, emr-6.0.0, emr-7.0.0
	r7i.2xlarge	emr-5.0.3, emr-6.0.0, emr-7.0.0
	r7i.4xlarge	emr-5.0.3, emr-6.0.0, emr-7.0.0
	r7i.8xlarge	emr-5.0.3, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7i.12xlarge	emr-5.0.3, emr-6.0.0, emr-7.0.0
	r7i.16xlarge	emr-5.0.3, emr-6.0.0, emr-7.0.0
	r7i.24xlarge	emr-5.0.3, emr-6.0.0, emr-7.0.0
	r7i.48xlarge	emr-5.0.3, emr-6.0.0, emr-7.0.0
	x1.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x2iedn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	z1d.xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.3xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.6xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	z1d.12xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
Speicheroptimiert	d3.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	d3.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	i3.xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.2xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.4xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.8xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.16xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3en.xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.2xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.3xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.6xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.12xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.24xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.0.3, emr-6.0.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.24xlarge	emr-5.0.3, emr-6.0.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	im4gn.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	im4gn.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Europa (Mailand) – eu-south-1

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5.2xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5.4xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5.8xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5.12xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5.16xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5.24xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5a.xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5a.2xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5a.4xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5a.8xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5a.12xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5a.16xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5a.24xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5d.xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5d.2xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5d.4xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5d.8xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5d.12xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5d.16xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m5d.24xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	m6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6g.xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.2xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.4xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.8xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.12xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.16xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5.2xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5.4xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5.9xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5.12xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5.18xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5.24xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5a.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5ad.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.24xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5d.xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5d.2xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5d.4xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5d.9xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5d.12xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5d.18xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5d.24xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5n.xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5n.2xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5n.4xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5n.9xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c5n.18xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	c6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6gn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7g.xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7g.2xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.4xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.8xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.12xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.16xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
Beschleunigte Datenverarbeitung	g4dn.xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.2xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.4xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.8xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.12xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.16xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5.2xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5.4xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5.8xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5.12xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5.16xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5.24xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5a.xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5a.2xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5a.4xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5a.8xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5a.12xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5a.16xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5a.24xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5b.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.24xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5d.xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5d.2xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5d.4xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5d.8xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5d.12xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5d.16xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5d.24xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	r5dn.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
Speicheroptimiert	i3.xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3.2xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	i3.4xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	i3.8xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	i3.16xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	i3en.xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	i3en.2xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	i3en.3xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	i3en.6xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	i3en.12xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	i3en.24xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.24xlarge	emr-5.29.0, emr-6.0.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Europa (Spanien) – eu-south-2

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5d.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5d.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5d.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5d.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5d.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5d.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5d.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6g.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6g.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6g.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6g.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6g.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6g.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6gd.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6gd.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6gd.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6gd.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6gd.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6gd.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6idn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6idn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.48xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7i.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m7i.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m7i.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m7i.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m7i.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m7i.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m7i.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m7i.48xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m7i-flex.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m7i-flex.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m7i-flex.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m7i-flex.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5.9xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5.18xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5d.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5d.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5d.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5d.9xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5d.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5d.18xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5d.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6g.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6g.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6g.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6g.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6g.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6g.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7a.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c7a.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c7a.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c7a.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c7a.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c7a.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c7a.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c7a.32xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7a.48xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c7g.xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.2xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.4xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.8xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.12xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.16xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7i.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c7i.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c7i.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c7i.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c7i.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c7i.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c7i.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c7i.48xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5d.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5d.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5d.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5d.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5d.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5d.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5d.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6g.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6g.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6g.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6g.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6g.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6g.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r7a.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r7a.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r7a.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r7a.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r7a.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r7a.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r7a.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7a.32xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r7a.48xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r7i.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r7i.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r7i.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r7i.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r7i.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r7i.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r7i.48xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
Speicheroptimiert	i3.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3en.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3en.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3en.3xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3en.6xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3en.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3en.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Europa (Paris) – eu-west-3

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m6g.xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.2xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.4xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.8xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.12xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.16xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.xlarge	emr-5.5.3, emr-6.0.0, emr-7.0.0
	m7i.2xlarge	emr-5.5.3, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7i.4xlarge	emr-5.5.3, emr-6.0.0, emr-7.0.0
	m7i.8xlarge	emr-5.5.3, emr-6.0.0, emr-7.0.0
	m7i.12xlarge	emr-5.5.3, emr-6.0.0, emr-7.0.0
	m7i.16xlarge	emr-5.5.3, emr-6.0.0, emr-7.0.0
	m7i.24xlarge	emr-5.5.3, emr-6.0.0, emr-7.0.0
	m7i.48xlarge	emr-5.5.3, emr-6.0.0, emr-7.0.0
	m7i-flex.xlarge	emr-4.9.2, emr-5.5.3, emr-6.0.0, emr-7.0.0
	m7i-flex.2xlarge	emr-4.9.2, emr-5.5.3, emr-6.0.0, emr-7.0.0
	m7i-flex.4xlarge	emr-4.9.2, emr-5.5.3, emr-6.0.0, emr-7.0.0
	m7i-flex.8xlarge	emr-4.9.2, emr-5.5.3, emr-6.0.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5a.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5n.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.9xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.18xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6gn.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7i.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7i.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7i.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7i.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7i.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7i.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7i.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7i.48xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
Beschleunigte Datenverarbeitung	g4dn.xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.2xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.4xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.8xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.12xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.16xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5dn.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7i.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.48xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
Speicheroptimiert	d3.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	d3.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3.xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.2xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.4xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.8xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.16xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3en.xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.2xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.3xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.6xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.12xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.24xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.5.3, emr-6.0.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.24xlarge	emr-5.5.3, emr-6.0.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	im4gn.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	im4gn.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	is4gen.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	is4gen.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Europa (Stockholm) – eu-north-1

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	m5.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	m5.4xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	m5.8xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	m5.12xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5.16xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	m5.24xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	m5d.xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	m5d.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	m5d.4xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	m5d.8xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	m5d.12xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	m5d.16xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	m5d.24xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	m6g.xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.2xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.4xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6g.8xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.12xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.16xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6idn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6idn.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6idn.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7a.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7a.48xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	m7i.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	m7i.4xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	m7i.8xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	m7i.12xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	m7i.16xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7i.24xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	m7i.48xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	m7i-flex.xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	m7i-flex.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	m7i-flex.4xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	m7i-flex.8xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c5.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c5.4xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c5.9xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c5.12xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c5.18xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5.24xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c5a.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5d.xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c5d.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c5d.4xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c5d.9xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5d.12xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c5d.18xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c5d.24xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c5n.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.9xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.18xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6gn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7a.xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c7a.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c7a.4xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c7a.8xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c7a.12xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c7a.16xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7a.24xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c7a.32xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c7a.48xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c7g.xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.2xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.4xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.8xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.12xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.16xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7i.xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c7i.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c7i.4xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c7i.8xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c7i.12xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c7i.16xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c7i.24xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	c7i.48xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
Beschleunigte Datenverarbeitung	g4dn.xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	g4dn.2xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.4xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.8xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.12xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.16xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g5.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	g5.48xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	p5.48xlarge	emr-6.14.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r5.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r5.4xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r5.8xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r5.12xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r5.16xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r5.24xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r5b.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5b.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.24xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5d.xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r5d.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r5d.4xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r5d.8xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r5d.12xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r5d.16xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r5d.24xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r5dn.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5dn.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5dn.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6idn.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6idn.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6idn.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7a.xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r7a.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r7a.4xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r7a.8xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r7a.12xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r7a.16xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r7a.24xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r7a.32xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r7a.48xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r7i.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7i.4xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r7i.8xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r7i.12xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r7i.16xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r7i.24xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	r7i.48xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x2iedn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
Speicheroptimiert	i3.xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	i3.2xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	i3.4xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	i3.8xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	i3.16xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	i3en.xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.2xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.3xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3en.6xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.12xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.24xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.24xlarge	emr-5.17.0, emr-6.0.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Naher Osten (Bahrain) – me-south-1

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	m5.2xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	m5.4xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	m5.8xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	m5.12xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	m5.16xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	m5.24xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	m5d.xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	m5d.2xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	m5d.4xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	m5d.8xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5d.12xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	m5d.16xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	m5d.24xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	m6g.xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.2xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.4xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.8xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.12xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.16xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	c5.2xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	c5.4xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	c5.9xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	c5.12xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	c5.18xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	c5.24xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5a.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5ad.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.24xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5d.xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	c5d.2xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	c5d.4xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	c5d.9xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	c5d.18xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	c5n.xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	c5n.2xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	c5n.4xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	c5n.9xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	c5n.18xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6gn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
Beschleunigte Datenverarbeitung	g4dn.xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.2xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.4xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.8xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.12xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.16xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	r5.2xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5.4xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	r5.8xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	r5.12xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	r5.16xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	r5.24xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	r5d.xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	r5d.2xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	r5d.4xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	r5d.8xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	r5d.12xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	r5d.16xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	r5d.24xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
Speicheroptimiert	i3.xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	i3.2xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	i3.4xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	i3.8xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	i3.16xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	i3en.xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.2xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.3xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.6xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.12xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3en.24xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.24xlarge	emr-5.24.0, emr-6.0.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Naher Osten (UAE) - me-central-1

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5d.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5d.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5d.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5d.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5d.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5d.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m5d.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6g.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6g.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6g.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6g.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6g.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6g.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6gd.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6gd.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6gd.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6gd.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6gd.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6gd.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6i.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6i.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6i.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6i.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6i.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6i.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6i.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	m6i.32xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5.9xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5.18xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5d.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5d.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5d.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5d.9xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5d.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c5d.18xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5d.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6g.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6g.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6g.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6g.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6g.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6g.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
Beschleunigte Datenverarbeitung	g5.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.48xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5d.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5d.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5d.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5d.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5d.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r5d.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5d.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6g.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6g.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6g.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6g.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6g.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6g.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6i.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6i.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6i.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6i.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6i.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6i.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6i.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	r6i.32xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
Speicheroptimiert	i3.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3.4xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3.8xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3.16xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3en.xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3en.2xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3en.3xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3en.6xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3en.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i3en.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.24xlarge	emr-5.36.0, emr-6.7.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Südamerika (São Paulo) – sa-east-1

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
Allgemeine Zwecke	m5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	m5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	m5zn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.3xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m5zn.6xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m5zn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6g.xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.2xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6g.4xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.8xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.12xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6g.16xlarge	emr-5.30.0, emr-6.1.0, emr-7.0.0
	m6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	m6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	m6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	m7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7gd.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7gd.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	m7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.2xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	m7i-flex.4xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
	m7i-flex.8xlarge	emr-4.2.0, emr-5.0.0, emr-6.0.0, emr-7.0.0
Für Datenverarbeitung optimiert	c5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5a.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5a.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5a.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c5ad.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5ad.24xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.9xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.18xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	c5n.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.9xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c5n.18xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	c6a.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6a.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6a.48xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	c6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6gn.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6gn.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	c6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	c6id.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6id.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.12xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.24xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6id.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	c6in.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c6in.24xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c6in.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	c7g.xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.2xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.4xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.8xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.12xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7g.16xlarge	emr-5.36.1, emr-6.7.0, emr-7.0.0
	c7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	c7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	c7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
Beschleunigte Datenverarbeitung	g4dn.xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.2xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.4xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.8xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.12xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g4dn.16xlarge	emr-5.30.0, emr-6.0.0, emr-7.0.0
	g5.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	g5.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.12xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	g5.48xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
RAM-optimiert	r5.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5a.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.8xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.16xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5a.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.2xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.4xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5ad.12xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5ad.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5ad.24xlarge	emr-5.20.0, emr-6.0.0, emr-7.0.0
	r5b.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5b.24xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r5d.xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.2xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r5d.4xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.8xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.12xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.16xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5d.24xlarge	emr-5.13.0, emr-6.0.0, emr-7.0.0
	r5n.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r5n.24xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6g.xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.2xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.4xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.8xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.12xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6g.16xlarge	emr-5.31.0, emr-6.1.0, emr-7.0.0
	r6gd.xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.2xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.4xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.8xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.12xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0
	r6gd.16xlarge	emr-5.33.0, emr-6.3.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r6i.xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.2xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.4xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.8xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.12xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.16xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.24xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r6i.32xlarge	emr-5.35.0, emr-6.6.0, emr-7.0.0
	r7g.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	r7g.12xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7g.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	r7i.xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.2xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.4xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.8xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.16xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	r7i.48xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	x1.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x1.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x1e.xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.2xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.4xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.8xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.16xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x1e.32xlarge	emr-5.36.1, emr-6.10.0, emr-7.0.0
	x2idn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2idn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.2xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.4xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	x2iedn.8xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.16xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.24xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
	x2iedn.32xlarge	emr-5.36.1, emr-6.9.0, emr-7.0.0
Speicheroptimiert	i3.xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.2xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.4xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.8xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3.16xlarge	emr-5.9.0, emr-6.0.0, emr-7.0.0
	i3en.xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.2xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.3xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0

Instance-Klasse	Instance-Typ	Unterstützte EMR Mindestversion von Amazon (5.x, 6.x, 7.x)
	i3en.6xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.12xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i3en.24xlarge	emr-5.25.0, emr-6.0.0, emr-7.0.0
	i4i.xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.2xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.4xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.8xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.12xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	i4i.16xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0
	i4i.24xlarge	emr-5.0.0, emr-6.0.0, emr-7.0.0
	i4i.32xlarge	emr-5.36.1, emr-6.8.0, emr-7.0.0

Instances der vorherigen Generation

Amazon EMR unterstützt Instances der vorherigen Generation, um Anwendungen zu unterstützen, die für diese Instances optimiert sind und noch nicht aktualisiert wurden. Weitere Informationen zu diesen Instance-Typen und Upgrade-Pfaden finden Sie unter Instances der vorherigen Generation.

Instance-Klasse	Instance-Typen
General Purpose	m1.small[1] | m1.medium[1] | m1.large[1] | m1.xlarge[1] | m3.xlarge[1] | m3.2xlarge[1] | m4.large | m4.xlarge | m4.2xlarge | m4.4xlarge | m4.10xlarge | m4.16xlarge
Compute Optimized	c1.medium[1] [2] | c1.xlarge[1] | c3.xlarge[1] | c3.2xlarge[1] | c3.4xlarge[1] | c3.8xlarge[1] | c4.large | c4.xlarge | c4.2xlarge | c4.4xlarge | c4.8xlarge
Memory Optimized	m2.xlarge[1] | m2.2xlarge[1] | m2.4xlarge[1] | r3.xlarge | r3.2xlarge | r3.4xlarge | r3.8xlarge | r4.xlarge | r4.2xlarge | r4.4xlarge | r4.8xlarge | r4.16xlarge
Storage Optimized	d2.xlarge | d2.2xlarge | d2.4xlarge | d2.8xlarge | i2.xlarge | i2.2xlarge | i2.4xlarge | i2.8xlarge

[1] Verwendet PVM Virtualisierung AMI mit EMR Amazon-Release-Versionen vor 5.13.0. Weitere Informationen finden Sie unter AMILinux-Virtualisierungstypen.

[2] Nicht unterstützt in Version 5.15.0.

Instance-Kaufoptionen

Wenn Sie einen Cluster einrichten, wählen Sie eine Kaufoption für EC2 Amazon-Instances. Sie können On-Demand-Instances, Spot Instances oder beides auswählen. Die Preise variieren basierend auf dem Instance-Typ und der Region. Der EMR Amazon-Preis wird zusätzlich zum EC2 Amazon-Preis (der Preis für die zugrunde liegenden Server) und zum EBS Amazon-Preis (wenn EBS Amazon-Volumen angehängt werden) berechnet. Aktuelle Preise finden Sie unter EMRAmazon-Preise.

Ihre Wahl zur Verwendung von Instance-Gruppen oder Instance-Flotten in Ihrem Cluster bestimmt, wie Sie die Instance-Kaufoptionen ändern können, während der Cluster ausgeführt wird. Wenn Sie

einheitliche Instance-Gruppen wählen, können Sie die Kaufoption für eine Instance-Gruppe nur angeben, wenn Sie sie erstellen. Der Instance-Typ und die Kaufoption gelten für alle EC2 Amazon-Instances in jeder Instance-Gruppe. Bei der Wahl von Instance-Flotten können Sie die Kaufoptionen ändern, nachdem eine Instance-Flotte erstellt wurde. Sie können die Kaufoptionen kombinieren, um eine festgelegte Zielkapazität zu erfüllen. Weitere Informationen zu diesen Konfigurationen finden Sie unter Einen Cluster mit Instance-Flotten oder einheitlichen Instance-Gruppen erstellen.

On-Demand Instances

Bei On-Demand-Instances zahlen Sie für die Rechenkapazität nach Sekunde. Optional können Sie für diese On-Demand-Instances Reserved Instance- oder Dedicated Instance-Kaufoptionen verwenden. Bei Reserved Instances leisten Sie eine einmalige Zahlung für eine Instance, um Kapazität zu reservieren. Dedicated Instances sind auf Host-Hardwareebene physisch von Instances isoliert, die zu anderen AWS Konten gehören. Weitere Informationen zu Kaufoptionen finden Sie unter Instance-Kaufoptionen im EC2Amazon-Benutzerhandbuch.

Verwenden von Reserved Instances

Um Reserved Instances in Amazon zu verwendenEMR, verwenden Sie Amazon, EC2 um die Reserved Instance zu kaufen, und geben die Reservierungsparameter an, einschließlich des Umfangs der Reservierung, der entweder für eine Region oder eine Availability Zone gilt. Weitere Informationen finden Sie unter Amazon EC2 Reserved Instances und Buying Reserved Instances im EC2Amazon-Benutzerhandbuch. Wenn nach dem Kauf einer Reserved Instance alle der folgenden Bedingungen erfüllt sind, EMR verwendet Amazon die Reserved Instance, wenn ein Cluster gestartet wird:

- Eine On-Demand-Instance ist in der Cluster-Konfiguration angegeben, die mit der Reserved-Instance-Spezifikation übereinstimmt.
- Der Cluster wird im Rahmen der Instance-Reservierung (Availability Zone oder Region) gestartet.
- Die Reserved Instance-Kapazität ist noch verfügbar.

Angenommen, Sie kaufen eine Reserved Instance `m5.xlarge` mit der gewünschten Instance-Reservierung für die Region USA Ost. Anschließend starten Sie einen EMR Amazon-Cluster in den USA Ost, der zwei `m5.xlarge` Instances verwendet. Die erste Instance wird nach dem Tarif für Reserved Instances abgerechnet und die andere nach dem On-Demand-Tarif. Die Reserved Instance-Kapazität wird verwendet, bevor die On-Demand-Instances erstellt werden.

Verwenden von Dedicated Instances

Um Dedicated Instances zu verwenden, kaufen Sie Dedicated Instances über Amazon EC2 und erstellen dann eine VPC mit dem Attribut Dedicated Tenancy. Innerhalb von Amazon EMR geben Sie dann an, dass darin ein Cluster gestartet werden sollVPC. Alle On-Demand-Instances im Cluster, die der Dedicated Instance-Spezifikation entsprechen, verwenden verfügbare Dedicated Instances beim Start des Clusters.

> **Note**
>
> Amazon EMR unterstützt die Einstellung des `dedicated` Attributs für einzelne Instances nicht.

Spot-Instances

Spot-Instances bei Amazon EMR bieten Ihnen die Möglichkeit, EC2 Amazon-Instance-Kapazität zu geringeren Kosten als beim Kauf auf Abruf zu erwerben. Der Nachteil der Verwendung von Spot Instances besteht darin, dass Instances möglicherweise beendet werden, wenn die Spot-Kapazität für den von Ihnen ausgeführten Instance-Typ nicht mehr verfügbar ist. Weitere Informationen dazu, wann Sie Spot Instances für Ihre Anwendung verwenden sollten, finden Sie unter Wann sollten Sie Spot Instances verwenden?.

Wenn Amazon EC2 über ungenutzte Kapazitäten verfügt, bietet Amazon EC2 Instances zu reduzierten Kosten an, die als Spot-Preis bezeichnet werden. Dieser Preis schwankt abhängig von Verfügbarkeit und Bedarf und wird nach Region und Availability Zone festgelegt. Wenn Sie sich für Spot-Instances entscheiden, geben Sie den maximalen Spot-Preis an, den Sie für jeden EC2 Instance-Typ zu zahlen bereit sind. Wenn der Spot-Preis in der Availability Zone des Clusters unter dem für diesen Instance-Typ angegebenen maximalen Spot-Preis liegt, werden Instances gestartet. Während die Instances ausgeführt werden, wird Ihnen der aktuelle Spot-Preis nicht Ihr maximaler Spot-Preis in Rechnung gestellt.

> **Note**
>
> Spot-Instances mit definierter Laufzeit (auch Spot-Blöcke genannt) stehen Neukunden ab dem 1. Juli 2021 nicht mehr zur Verfügung. Für Kunden, die diese Funktion bereits genutzt haben, werden wir Spot-Instances mit einer definierten Laufzeit bis zum 31. Dezember 2022 weiterhin unterstützen.

Aktuelle Preise finden Sie unter Preise für Amazon EC2 Spot-Instances. Weitere Informationen finden Sie unter Spot-Instances im EC2Amazon-Benutzerhandbuch. Wenn Sie einen Cluster erstellen und konfigurieren, geben Sie Netzwerkoptionen an, die letztendlich die Availability Zone bestimmen, in der Ihr Cluster gestartet wird. Weitere Informationen finden Sie unter Netzwerk konfigurieren.

> **Tip**
>
> Sie können den aktuellen Spot-Preis in der Konsole anzeigen, indem Sie mit dem Mauszeiger auf die QuickInfo für Informationen neben der Kaufoption Spot (Spot) zeigen, wenn Sie einen Cluster mittels Advanced Options (Erweiterte Optionen) erstellen. Die Preise für jede Availability Zone in der ausgewählten Region werden angezeigt. Die grünen Zeilen enthalten die niedrigsten Preise. Aufgrund der Spot-Preisschwankungen zwischen Availability Zones kann es sein, dass durch Auswählen der Availability Zone mit dem niedrigsten Anfangspreis möglicherweise nicht der niedrigste Preis für die Nutzungsdauer des Clusters erzielt wird. Um optimale Ergebnisse zu erzielen, sehen Sie sich den Availability Zone-Preisverlauf an, bevor Sie sich entscheiden. Weitere Informationen finden Sie unter Preisverlauf für Spot-Instances im EC2Amazon-Benutzerhandbuch.

Die Spot-Instance-Optionen hängen davon ab, ob Sie einheitliche Instance-Gruppen oder Instance-Flotten in der Cluster-Konfiguration verwenden.

Spot-Instances in einheitlichen Instance-Gruppen

Wenn Sie Spot-Instances in einer einheitlichen Instance-Gruppe verwenden, muss es sich bei allen Instances in der Instance-Gruppe um Spot-Instances handeln. Sie geben ein einzelnes Subnetz eine oder Availability Zone für den Cluster an. Für jede Instance-Gruppe legen Sie eine einzelne Spot Instance und einen maximalen Spot-Preis fest. Die Spot Instances des entsprechenden Typs werden gestartet, wenn der Spot-Preis in der Region und Availability Zone des Clusters unter dem maximalen Spot-Preis liegt. Instances werden beendet, wenn der Spot-Preis Ihren maximalen Spot-Preis übersteigt. Sie legen den maximalen Spot-Preis nur beim Konfigurieren einer Instance-Gruppe fest. Er kann später nicht mehr geändert werden. Weitere Informationen finden Sie unter Einen Cluster mit Instance-Flotten oder einheitlichen Instance-Gruppen erstellen.

Spot-Instances in Instance-Flotten

Wenn Sie die Instance-Flottenkonfiguration verwenden, erhalten Sie durch zusätzliche Optionen mehr Kontrolle darüber, wie Spot-Instances gestartet und beendet werden. Grundsätzlich verwenden Instance-Flotten zum Starten von Instances eine andere Methode als einheitliche Instance-Gruppen.

Hierbei legen Sie eine Zielkapazität für Spot-Instances (und On-Demand-Instances) und bis zu fünf Instance-Typen fest. Sie können auch eine gewichtete Kapazität für jeden Instance-Typ angeben oder die V CPU (YARNvcores) des Instance-Typs als gewichtete Kapazität verwenden. Die gewichtete Kapazität wird im Rahmen der Zielkapazität berücksichtigt, wenn eine Instance dieses Typs bereitgestellt wird. Amazon EMR stellt Instances mit beiden Kaufoptionen bereit, bis die Zielkapazität für jedes Ziel erreicht ist. Darüber hinaus können Sie eine Reihe von Availability Zones definieren, aus denen Amazon EMR beim Starten von Instances wählen kann. Sie stellen außerdem zusätzliche Spot-Optionen für jede Flotte bereit, einschließlich eines Bereitstellungs-Timeouts. Weitere Informationen finden Sie unter Instance-Flotten konfigurieren.

Instance-Speicher

Übersicht

Der Instance-Speicher und der EBS Amazon-Volume-Speicher werden für HDFS Daten und für Puffer, Caches, Scratch-Daten und andere temporäre Inhalte verwendet, die von einigen Anwendungen möglicherweise in das lokale Dateisystem „verschüttet“ werden.

Amazon EBS funktioniert innerhalb von Amazon anders EMR als bei regulären EC2 Amazon-Instances. EBSAmazon-Volumes, die an EMR Amazon-Clustern angehängt sind, sind kurzlebig: Die Volumes werden gelöscht, wenn Cluster und Instances beendet werden (z. B. beim Verkleinern von Instance-Gruppen), sodass Sie nicht erwarten sollten, dass Daten bestehen bleiben. Obwohl die Daten kurzlebig sind, ist es möglich, dass sie repliziert werden, je nach Anzahl und Spezialisierung der Knoten im HDFS Cluster. Wenn Sie EBS Amazon-Speichervolumes hinzufügen, werden diese als zusätzliche Volumes bereitgestellt. Sie sind nicht Teil des Startvolumes. YARNist so konfiguriert, dass alle zusätzlichen Volumes verwendet werden, aber Sie sind dafür verantwortlich, die zusätzlichen Volumes als lokalen Speicher zuzuweisen (z. B. für lokale Protokolldateien).

Überlegungen

Beachten Sie die folgenden zusätzlichen Überlegungen, wenn Sie Amazon EBS mit EMR Clustern verwenden:

- Sie können einen Snapshot eines EBS Amazon-Volumes nicht erstellen und es dann innerhalb von Amazon wiederherstellenEMR. Um wiederverwendbare benutzerdefinierte Konfigurationen zu erstellen, verwenden Sie eine benutzerdefinierte AMI (verfügbar in EMR Amazon-Version 5.7.0 und höher). Weitere Informationen finden Sie unter Verwenden Sie ein benutzerdefiniertes AMI.
- Ein verschlüsseltes EBS Amazon-Root-Geräte-Volume wird nur unterstützt, wenn ein benutzerdefiniertes Volume verwendet wirdAMI. Weitere Informationen finden Sie unter

Benutzerdefiniertes Volume AMI mit einem verschlüsselten EBS Amazon-Root-Geräte-Volume erstellen.

- Wenn Sie Tags mithilfe von Amazon anwenden EMRAPI, werden diese Operationen auf EBS Volumes angewendet.
- Es gilt eine Beschränkung von 25 Volumes pro Instance.
- Die EBS Amazon-Volumes auf den Kernknoten dürfen nicht weniger als 5 GB groß sein.
- Amazon EBS hat ein festes Limit von 2.500 EBS Volumen pro Instance-Startanfrage. Dieses Limit gilt auch für Amazon EMR auf EC2 Clustern. Wir empfehlen, dass Sie Cluster mit der Gesamtzahl der EBS Volumes innerhalb dieses Limits starten und den Cluster dann manuell oder mit Amazon EMR Managed Scaling nach Bedarf hochskalieren. Weitere Informationen zum EBS Volumenlimit finden Sie unter Servicekontingenten.

EBSAmazon-Standardspeicher für Instances

Für EC2 Instances, die EBS nur über Speicher verfügen, EMR weist Amazon den Instances EBS Amazon-GP2- oder GP3-Speichervolumes zu. Wenn Sie einen Cluster mit EMR Amazon-Versionen 5.22.0 und höher erstellen, erhöht sich die Standardmenge an EBS Amazon-Speicher im Verhältnis zur Größe der Instance.

Wir teilen jeden erhöhten Speicherplatz auf mehrere Volumes auf. Dies führt zu einer höheren IOPS Leistung und damit zu einer höheren Leistung für einige standardisierte Workloads. Wenn Sie eine andere EBS Amazon-Instance-Speicherkonfiguration verwenden möchten, können Sie dies angeben, wenn Sie einen EMR Cluster erstellen oder Knoten zu einem vorhandenen Cluster hinzufügen. Sie können Amazon EBS GP2- oder GP3-Volumes als Root-Volumes verwenden und GP2- oder GP3-Volumes als zusätzliche Volumes hinzufügen. Weitere Informationen finden Sie unter Angabe zusätzlicher EBS Speichervolumes.

Die folgende Tabelle zeigt die Standardanzahl von Amazon EBS GP2-Speicher-Volumes, Größen und Gesamtgrößen pro Instance-Typ. Hinweise zu gp2-Volumes im Vergleich zu gp3-Volumes finden Sie unter Vergleich der EBS Amazon-Volumetypen gp2 und gp3.

Standard-Amazon EBS GP2-Speichervolumen und -größe nach Instance-Typ für Amazon EMR 5.22.0 und höher

Instance-Größe	Anzahl der Volumes	Volume-Größe (GiB)	Gesamtgröße (GB)
*.large	1	32	32

Instance-Größe	Anzahl der Volumes	Volume-Größe (GiB)	Gesamtgröße (GB)
*.xlarge	2	32	64
*.2xlarge	4	32	128
*.4xlarge	4	64	256
*.8xlarge	4	128	512
9xlarge	4	144	576
10xlarge	4	160	640
12xlarge	4	192	768
*.16xlarge	4	256	1024
18xlarge	4	288	1 152
24xlarge	4	384	1536

EBSStandard-Amazon-Root-Volume für Instances

Mit EMR Amazon-Versionen 6.15 und höher fügt Amazon EMR automatisch ein Amazon EBS General Purpose SSD (gp3) als Root-Gerät hinzu, um die Leistung AMIs zu verbessern. In früheren Versionen EMR fügt Amazon EBS General Purpose SSD (gp2) als Root-Gerät hinzu.

	6.15 und höher	6.14 und niedriger
Root-Volume-Standardtyp		
Standardgröße		
Standard IOPS		
Standarddurchsatz		

Informationen zum Anpassen des Volumes des EBS Amazon-Root-Geräts finden Sie unterAngabe zusätzlicher EBS Speichervolumes.

Angabe zusätzlicher EBS Speichervolumes

Wenn Sie Instance-Typen in Amazon konfigurierenEMR, können Sie zusätzliche EBS Volumes angeben, um Kapazität hinzuzufügen, die über den Instance-Speicher (falls vorhanden) und das EBS Standard-Volume hinausgeht. Amazon EBS bietet die folgenden Volumetypen an: General Purpose (SSD), Provisioned IOPS (SSD), Throughput Optimized (HDD), Cold (HDD) und Magnetic. Diese unterscheiden sich bei den Leistungsmerkmalen und im Preis, sodass Sie Ihren Speicher den Analyse- und Business-Anforderungen Ihrer Anwendungen entsprechend anpassen können. Beispielsweise benötigen einige Anwendungen den Überlauf auf Datenträger, während andere im Speicher oder unter Verwendung Amazon S3 sicher arbeiten können.

Sie können EBS Amazon-Volumes nur beim Start des Clusters und wenn Sie eine zusätzliche Task-Knoten-Instance-Gruppe hinzufügen, an Instances anhängen. Wenn eine Instance in einem EMR Amazon-Cluster ausfällt, werden sowohl die Instance als auch die angehängten EBS Amazon-Volumes durch neue Volumes ersetzt. Wenn Sie also ein EBS Amazon-Volume manuell trennen, EMR behandelt Amazon dies als Fehler und ersetzt sowohl den Instance-Speicher (falls zutreffend) als auch die Volume Stores.

Amazon erlaubt Ihnen EMR nicht, Ihren Volume-Typ für einen vorhandenen EMR Cluster von gp2 auf gp3 zu ändern. Um gp3 für Ihre Workloads zu verwenden, starten Sie einen neuen Cluster. EMR Darüber hinaus empfehlen wir nicht, den Durchsatz und IOPS auf einem Cluster zu aktualisieren, der verwendet wird oder der bereitgestellt wird, da Amazon den Durchsatz und die IOPS Werte, die Sie beim Clusterstart angeben, für jede neue Instance EMR verwendet, die beim Cluster-Scale-up hinzugefügt wird. Weitere Informationen erhalten Sie unter Vergleich der EBS Amazon-Volumetypen gp2 und gp3 und Auswahl IOPS und Durchsatz bei der Migration zu gp3.

> **Important**
>
> Um ein GP3-Volume mit Ihrem EMR Cluster zu verwenden, müssen Sie einen neuen Cluster starten.

Vergleich der EBS Amazon-Volumetypen gp2 und gp3

Hier finden Sie einen Vergleich der Kosten zwischen den gp2- und gp3-Volumes in der Region USA Ost (Nord-Virginia). Die aktuellsten Informationen finden Sie auf der Produktseite von Amazon EBS General Purpose Volumes und auf der EBSAmazon-Preisseite.

Volume-Typ	gp3	gp2
Volume-Größe	1 GiB–16 TiB	1 GiB–16 TiB
Standard/Baseline IOPS	3000	3 IOPS /GiB (mindestens 100IOPS) bis maximal IOPS 16.000. Volumen, die kleiner als 1 TiB sind, können auch auf bis zu 3.000 IOPS TiB hochgehen.
Max. /Lautstärke IOPS	16,000	16,000
Standard-/Baseline-Durchsatz	125 MiB/s	Die Durchsatzgrenze liegt zwischen 128 MiB/s und 250 MiB/s, abhängig von der Volume-Größe.
Max. Durchsatz pro Volume	1 000 MiB/s	250 MiB/s
Preis	0,08$ pro GiB-Monat 3.000$ IOPS kostenlos und 0,005$ pro bereitgestellter Monat über 3.000; 125 MIB/s kostenlos und 0,04$ bereitgestellte IOPS MIB/s-Monat über 125 MIB/s	0,10 USD/GiB-Monat

Auswahl IOPS und Durchsatz bei der Migration zu gp3

Bei der Bereitstellung eines GP2-Volumes müssen Sie die Größe des Volumes ermitteln, um die Proportionen und den Durchsatz zu ermitteln. IOPS Mit gp3 müssen Sie kein größeres Volume bereitstellen, um eine höhere Leistung zu erzielen. Sie können die gewünschte Größe und Leistung

je nach Anwendungsanforderungen wählen. Durch die Auswahl der richtigen Größe und der richtigen Leistungsparameter (IOPS, Durchsatz) können Sie maximale Kostensenkungen erzielen, ohne die Leistung zu beeinträchtigen.

Die folgende Tabelle hilft Ihnen bei der Auswahl der gp3-Konfigurationsoptionen:

Volume-Größe	IOPS	Durchsatz
1–170 GiB	3000	125 MiB/s
170–334 GiB	3000	125 MiB/s, wenn der gewählte EC2 Instance-Typ 125 MiB/s oder weniger unterstützt, verwenden Sie je nach Nutzung mehr, maximal 250 MiB/s*.
334–1 000 GiB	3000	125 MiB/s, wenn der gewählte EC2 Instance-Typ 125 MiB/s oder weniger unterstützt, je nach Nutzung höher verwenden, maximal 250 MiB/s*.
Über 1 000 GiB	Passen Sie gp2 IOPS (Größe in GiB x 3) oder Max an, abhängig IOPS vom aktuellen GP2-Volumen	125 MiB/s, wenn der gewählte EC2 Instance-Typ 125 MiB/s oder weniger unterstützt, je nach Nutzung höher verwenden, maximal 250 MiB/s*.

*Gp3 kann einen Durchsatz von bis zu 1 000 MiB/s bieten. Da gp2 einen maximalen Durchsatz von 250 MiB/s bietet, müssen Sie diese Grenze möglicherweise nicht überschreiten, wenn Sie gp3 verwenden.

Netzwerk konfigurieren

Die meisten Cluster werden mithilfe von Amazon Virtual Private Cloud (AmazonVPC) in ein virtuelles Netzwerk gestartet. A VPC ist ein isoliertes virtuelles Netzwerk innerhalb Ihres AWS Kontos AWS , das logisch isoliert ist. Sie können Aspekte wie private IP-Adressbereiche, Subnetze, Routing-Tabellen und Netzwerk-Gateways konfigurieren. Weitere Informationen finden Sie im VPCAmazon-Benutzerhandbuch.

VPCbietet die folgenden Funktionen:

- Verarbeitung sensibler Daten

  Das Starten eines Clusters in einem VPC ähnelt dem Starten des Clusters in einem privaten Netzwerk mit zusätzlichen Tools wie Routing-Tabellen und NetzwerkACLs, um zu definieren, wer Zugriff auf das Netzwerk hat. Wenn Sie sensible Daten in Ihrem Cluster verarbeiten, möchten Sie möglicherweise die zusätzliche Zugriffskontrolle nutzen, die das Starten Ihres Clusters in einem VPC bietet. Außerdem können Sie Ihre Ressourcen in einem privaten Subnetz starten, in dem keine dieser Ressourcen über eine direkte Internetverbindung verfügt.
- Zugreifen auf Ressourcen in einem internen Netzwerk

  Wenn sich Ihre Datenquelle in einem privaten Netzwerk befindet, kann es unpraktisch oder unerwünscht sein, diese Daten AWS für den Import in Amazon hochzuladenEMR, entweder aufgrund der zu übertragenden Datenmenge oder wegen der Vertraulichkeit der Daten. Stattdessen können Sie den Cluster in einem starten VPC und Ihr Rechenzentrum VPC über eine VPN Verbindung mit Ihrem verbinden, sodass der Cluster auf Ressourcen in Ihrem internen Netzwerk zugreifen kann. Wenn Sie beispielsweise eine Oracle-Datenbank in Ihrem Rechenzentrum haben, VPN ermöglicht das Starten Ihres Clusters in einem VPC mit diesem Netzwerk verbundenen Netzwerk, dass der Cluster auf die Oracle-Datenbank zugreifen kann.

Öffentliche und private Subnetze

Sie können EMR Amazon-Cluster sowohl in öffentlichen als auch in privaten VPC Subnetzen starten. Das bedeutet, dass Sie für den Betrieb eines EMR Amazon-Clusters keine Internetverbindung benötigen. Möglicherweise müssen Sie jedoch die Netzwerkadressübersetzung (NAT) und VPN Gateways konfigurieren, um auf Dienste oder Ressourcen zuzugreifen, die sich außerhalb des befindenVPC, z. B. in einem Unternehmensintranet oder an Endpunkten für öffentliche AWS Dienste wie. AWS Key Management Service

> **Important**
>
> Amazon unterstützt EMR nur das Starten von Clustern in privaten Subnetzen in Version 4.2 und höher.

Weitere Informationen zu Amazon VPC finden Sie im VPCAmazon-Benutzerhandbuch.

Themen

- VPCAmazon-Optionen
- Richten Sie einVPC, um Cluster zu hosten
- Starten Sie Cluster in einem VPC
- Amazon-S3-Mindestrichtlinie für private Subnetze
- Weitere Ressourcen, um mehr über Folgendes zu erfahren VPCs

VPCAmazon-Optionen

Wenn Sie einen EMR Amazon-Cluster in einem startenVPC, können Sie ihn entweder in einem öffentlichen, privaten oder gemeinsam genutzten Subnetz starten. Es gibt geringe, aber erwähnenswerte Unterschiede in Bezug auf die Konfiguration, je nachdem, welchen Subnetztyp Sie für ein Cluster auswählen.

Öffentliche Subnetze

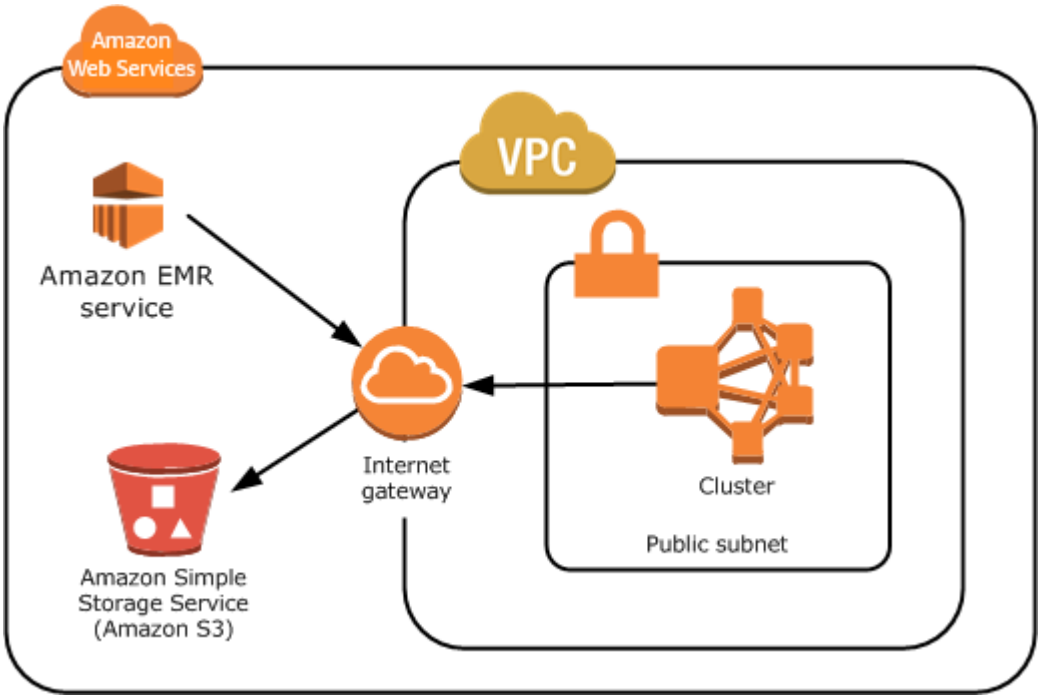
EMRCluster in einem öffentlichen Subnetz benötigen ein verbundenes Internet-Gateway. Dies liegt daran, dass EMR Amazon-Cluster auf AWS Services und Amazon zugreifen müssenEMR. Wenn ein Service, wie Amazon S3, die Möglichkeit bietet, einen VPC Endpunkt zu erstellen, können Sie über den Endpunkt auf diese Dienste zugreifen, anstatt über ein Internet-Gateway auf einen öffentlichen Endpunkt zuzugreifen. Darüber hinaus EMR kann Amazon nicht mit Clustern in öffentlichen Subnetzen über ein Network Address Translation (NAT) -Gerät kommunizieren. Zu diesem Zweck ist ein Internet-Gateway erforderlich, aber Sie können in komplexeren Szenarien immer noch eine NAT Instance oder ein Gateway für anderen Datenverkehr verwenden.

Alle Instances in einem Cluster stellen entweder über einen VPC Endpunkt oder ein Internet-Gateway eine Verbindung zu Amazon S3 her. Andere AWS Dienste, die derzeit keine VPC Endgeräte unterstützen, verwenden nur ein Internet-Gateway.

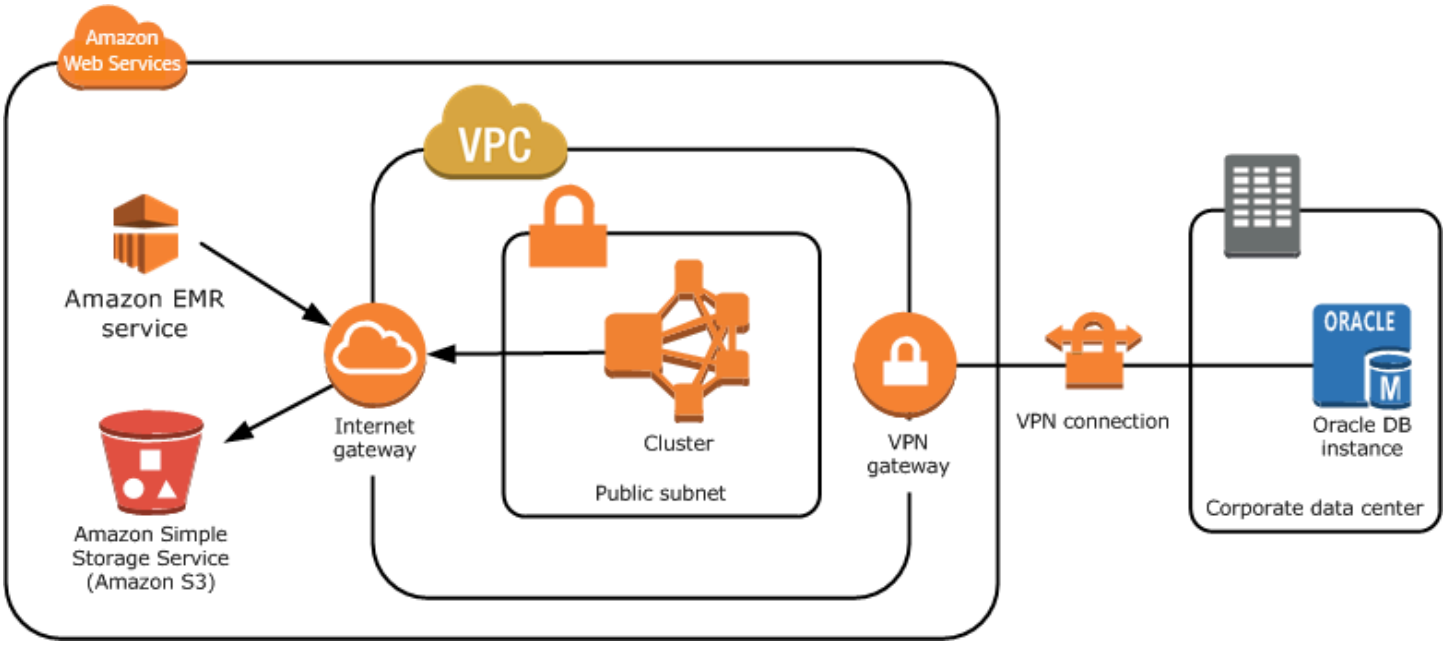
Wenn Sie über zusätzliche AWS Ressourcen verfügen, die Sie nicht mit dem Internet-Gateway verbinden möchten, können Sie diese Komponenten in einem privaten Subnetz starten, das Sie in Ihrem erstellen. VPC

Cluster in einem öffentlichen Subnetz verwenden zwei Sicherheitsgruppen: eine für den Primärknoten und eine für Core- und Aufgabenknoten. Weitere Informationen finden Sie unter Steuerung des Netzwerkverkehrs mit Sicherheitsgruppen.

Das folgende Diagramm zeigt, wie ein EMR Amazon-Cluster in einem VPC öffentlichen Subnetz ausgeführt wird. Der Cluster kann über das Internet-Gateway eine Verbindung zu anderen AWS Ressourcen wie Amazon S3 S3-Buckets herstellen.



Das folgende Diagramm zeigt, wie Sie einen VPC so einrichten, dass ein Cluster in der auf Ressourcen in Ihrem eigenen Netzwerk zugreifen VPC kann, z. B. auf eine Oracle-Datenbank.



Private Subnetze

Mit einem privaten Subnetz können Sie AWS Ressourcen starten, ohne dass das Subnetz über ein angeschlossenes Internet-Gateway verfügen muss. Amazon EMR unterstützt das Starten von Clustern in privaten Subnetzen mit den Release-Versionen 4.2.0 oder höher.

> **Note**
>
> Wenn Sie einen EMR Amazon-Cluster in einem privaten Subnetz einrichten, empfehlen wir, dass Sie auch VPCEndpunkte für Amazon S3 einrichten. Wenn sich Ihr EMR Cluster in einem privaten Subnetz ohne VPC Endpunkte für Amazon S3 befindet, fallen zusätzliche NAT Gateway-Gebühren an, die mit dem S3-Verkehr verbunden sind, da der Verkehr zwischen Ihrem EMR Cluster und S3 nicht innerhalb Ihres bleibt. VPC

Private Subnetze unterscheiden sich von öffentlichen Subnetzen in folgenden Punkten:

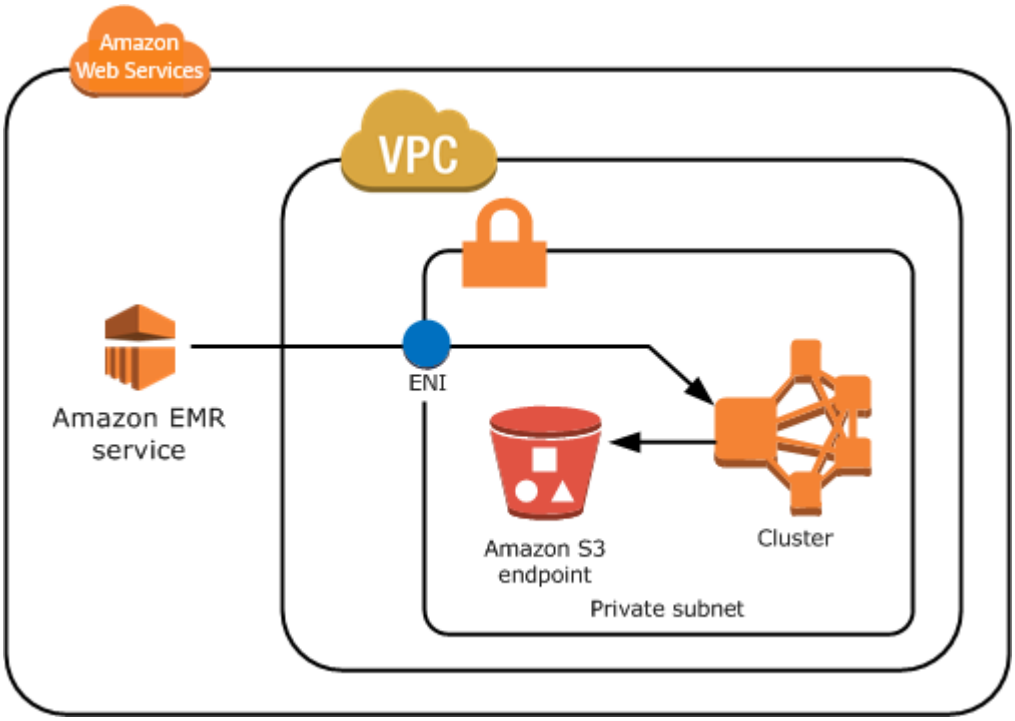
- Um auf AWS Dienste zuzugreifen, die keinen VPC Endpunkt bereitstellen, müssen Sie dennoch eine NAT Instance oder ein Internet-Gateway verwenden.
- Sie müssen mindestens eine Route zum Amazon EMR Service Logs-Bucket und zum Amazon Linux-Repository in Amazon S3 angeben. Weitere Informationen finden Sie unter Amazon-S3-Mindestrichtlinie für private Subnetze

- Wenn Sie EMRFS Funktionen verwenden, benötigen Sie einen Amazon S3 VPC S3-Endpunkt und eine Route von Ihrem privaten Subnetz zu DynamoDB.
- Das Debuggen funktioniert nur, wenn Sie eine Route von Ihrem privaten Subnetz zu einem öffentlichen SQS Amazon-Endpunkt angeben.
- Das Erstellen einer privaten Subnetzkonfiguration mit einer NAT Instance oder einem Gateway in einem öffentlichen Subnetz wird nur mit der unterstützt. AWS Management Console Der einfachste Weg, NAT Instances und Amazon VPC S3-Endpunkte für EMR Amazon-Cluster hinzuzufügen und zu konfigurieren, besteht darin, die Seite VPCSubnetzliste in der EMR Amazon-Konsole zu verwenden. Informationen zur Konfiguration von NAT Gateways finden Sie unter NATGateways im VPCAmazon-Benutzerhandbuch.
- Sie können ein Subnetz mit einem vorhandenen EMR Amazon-Cluster nicht von öffentlich zu privat oder umgekehrt ändern. Um einen EMR Amazon-Cluster in einem privaten Subnetz zu finden, muss der Cluster in diesem privaten Subnetz gestartet werden.

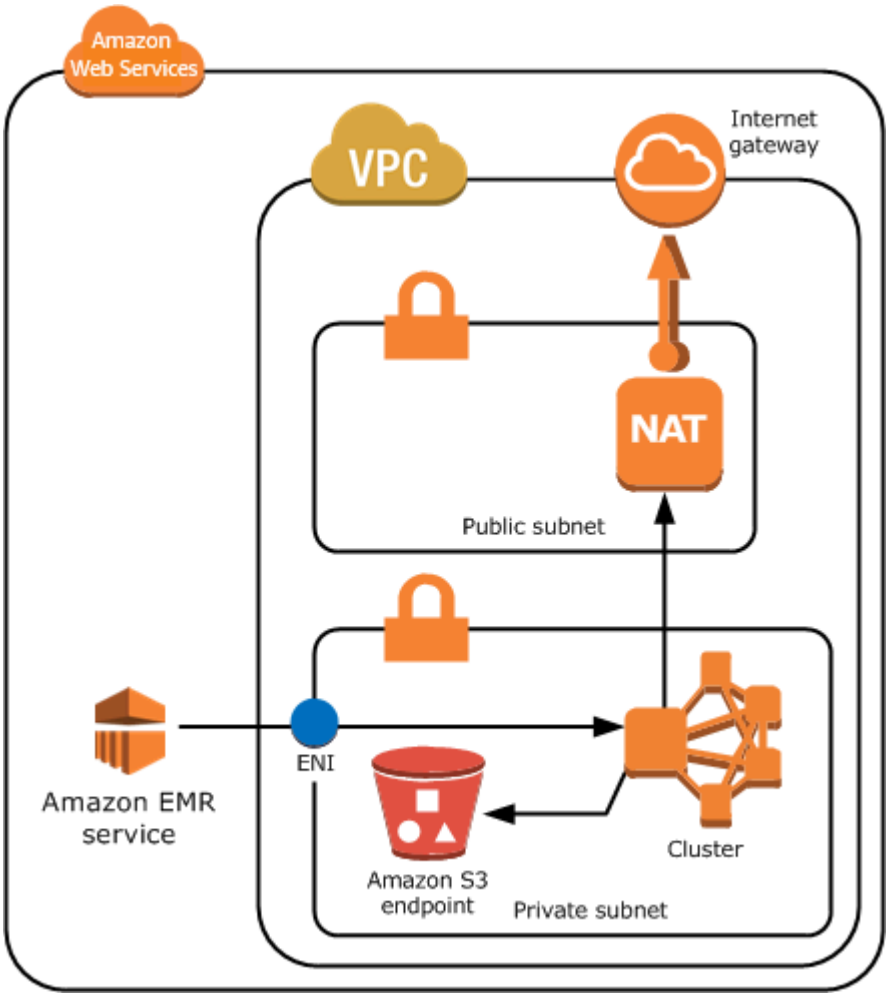
Amazon EMR erstellt und verwendet verschiedene Standardsicherheitsgruppen für die Cluster in einem privaten Subnetz: ElasticMapReduce -Master-Private, ElasticMapReduce -Slave-Private und -. ElasticMapReduce ServiceAccess Weitere Informationen finden Sie unter Steuerung des Netzwerkverkehrs mit Sicherheitsgruppen.

Für eine vollständige Liste Ihres NACLs Clusters wählen Sie Sicherheitsgruppen für Primär und Sicherheitsgruppen für Core & Task auf der Seite Cluster-Details der EMR Amazon-Konsole.

Die folgende Abbildung zeigt, wie ein EMR Amazon-Cluster in einem privaten Subnetz konfiguriert ist. Die einzige Kommunikation außerhalb des Subnetzes erfolgt mit AmazonEMR.



Die folgende Abbildung zeigt eine Beispielkonfiguration für einen EMR Amazon-Cluster in einem privaten Subnetz, das mit einer NAT Instance verbunden ist, die sich in einem öffentlichen Subnetz befindet.



Gemeinsam genutzte Subnetze

VPCDurch die gemeinsame Nutzung können Kunden Subnetze mit anderen AWS Konten innerhalb derselben AWS Organisation gemeinsam nutzen. Sie können EMR Amazon-Cluster sowohl in öffentlichen, gemeinsam genutzten als auch in privaten, gemeinsam genutzten Subnetzen starten. Beachten Sie dabei die folgenden Einschränkungen.

Der Subnetzbesitzer muss ein Subnetz mit Ihnen teilen, bevor Sie dort einen EMR Amazon-Cluster starten können. Die Freigabe für gemeinsame Subnetze kann jedoch zu einem späteren Zeitpunkt wieder aufgehoben werden. Weitere Informationen finden Sie unter Arbeiten mit Shared. VPCs Wenn ein Cluster in einem gemeinsam genutzten Subnetz gestartet wird und dieses gemeinsame Subnetz dann nicht mehr gemeinsam genutzt wird, können Sie je nach Zustand des EMR Amazon-Clusters bestimmte Verhaltensweisen beobachten, wenn das Subnetz nicht gemeinsam genutzt wird.

- Die gemeinsame Nutzung des Subnetzes wurde aufgehoben, bevor der Cluster erfolgreich gestartet wurde — Wenn der Eigentümer die gemeinsame Nutzung des Amazon VPC -

oder Subnetzes beendet, während der Teilnehmer einen Cluster startet, kann der Cluster möglicherweise nicht gestartet oder teilweise initialisiert werden, ohne dass alle angeforderten Instances bereitgestellt werden.

- Nach erfolgreichem Start des Clusters wird die gemeinsame Nutzung des Subnetzes aufgehoben — Wenn der Eigentümer ein Subnetz oder Amazon nicht mehr VPC mit dem Teilnehmer teilt, kann die Größe der Cluster des Teilnehmers nicht geändert werden, um neue Instances hinzuzufügen oder fehlerhafte Instances zu ersetzen.

Wenn Sie einen EMR Amazon-Cluster starten, werden mehrere Sicherheitsgruppen erstellt. In einem gemeinsamen Subnetz steuert der Subnetzteilnehmer diese Sicherheitsgruppen. Der Subnetzbesitzer kann diese Sicherheitsgruppen zwar anzeigen, jedoch keine Aktionen bei diesen durchführen. Wenn der Subnetzbesitzer die Sicherheitsgruppe entfernen oder ändern möchte, muss der Teilnehmer, der die Sicherheitsgruppe erstellt hat, die Aktion durchführen.

Steuern Sie VPC die Berechtigungen mit IAM

Standardmäßig können alle -Benutzer sämtliche Subnetze für das Konto sehen einen Cluster in einem Subnetz starten.

Wenn Sie einen Cluster in einem startenVPC, können Sie AWS Identity and Access Management (IAM) verwenden, um den Zugriff auf Cluster zu kontrollieren und Aktionen mithilfe von Richtlinien einzuschränken, genau wie bei Clustern, die in Amazon EC2 Classic gestartet werden. Weitere Informationen zu IAM finden Sie im IAMBenutzerhandbuch.

Sie können damit auch steuernIAM, wer Subnetze erstellen und verwalten darf. Sie können beispielsweise eine IAM Rolle zur Verwaltung von Subnetzen und eine zweite Rolle erstellen, die Cluster starten, aber keine VPC Amazon-Einstellungen ändern kann. Weitere Informationen zur Verwaltung von Richtlinien und Aktionen bei Amazon EC2 und Amazon VPC finden Sie unter IAMRichtlinien für Amazon EC2 im EC2Amazon-Benutzerhandbuch.

Richten Sie einVPC, um Cluster zu hosten

Bevor Sie Cluster in einem starten könnenVPC, müssen Sie ein VPC und ein Subnetz erstellen. Für öffentliche Subnetze müssen Sie ein Internet-Gateway erstellen und es dem Subnetz hinzufügen. In den folgenden Anweisungen wird beschrieben, wie Sie einen Amazon-Cluster erstellen, der VPC in der Lage ist, EMR Amazon-Cluster zu hosten.

Um ein VPC With-Subnetz für einen EMR Amazon-Cluster zu erstellen

1. Öffnen Sie die VPC Amazon-Konsole unter https://console.aws.amazon.com/vpc/.
2. Wählen Sie oben rechts auf der Seite die AWS-Regionfür Sie VPC aus.
3. Wählen Sie Erstellen VPC.
4. Wählen Sie auf der VPCEinstellungsseite VPC und mehr.
5. Aktivieren Sie unter Automatische Generierung von Namenstags die Option Automatisch generieren und geben Sie einen Namen für Ihren ein. VPC Auf diese Weise können Sie das VPC UND-Subnetz in der VPC Amazon-Konsole identifizieren, nachdem Sie sie erstellt haben.
6. Geben Sie im IPv4CIDRBlockfeld einen privaten IP-Adressraum für Sie ein, VPC um eine korrekte DNS Hostnamenauflösung sicherzustellen. Andernfalls kann es zu Ausfällen des EMR Amazon-Clusters kommen. Dieser Raum umfasst die folgenden IP-Adressbereiche:

   - 10.0.0.0 – 10.255.255.255
   - 172.16.0.0 – 172.31.255.255
   - 192.168.0.0 – 192.168.255.255
7. Wählen Sie unter Anzahl der Availability Zones (AZs) die Anzahl der Availability Zones aus, in denen Sie Ihre Subnetze starten möchten.
8. Wählen Sie unter Anzahl der öffentlichen Subnetze ein einzelnes öffentliches Subnetz aus, das Sie Ihrem hinzufügen möchten. VPC Wenn die vom Cluster verwendeten Daten im Internet verfügbar sind (z. B. in Amazon S3 oder AmazonRDS), müssen Sie nur ein öffentliches Subnetz verwenden und müssen kein privates Subnetz hinzufügen.
9. Wählen Sie unter Anzahl der privaten Subnetze die Anzahl der privaten Subnetze aus, die Sie Ihrem hinzufügen möchten. VPC Wählen Sie eine oder mehrere aus, wenn die Daten für Ihre Anwendung in Ihrem eigenen Netzwerk gespeichert sind (z. B. in einer Oracle-Datenbank). Für ein VPC in einem privaten Subnetz müssen alle EC2 Amazon-Instances mindestens eine Route zu Amazon EMR über die elastic network interface haben. In der Konsole wird diese automatisch konfiguriert.
10. Wählen Sie unter NATGateways optional das Hinzufügen NAT von Gateways aus. Sie sind nur erforderlich, wenn Sie über private Subnetze verfügen, die mit dem Internet kommunizieren müssen.
11. Wählen Sie unter VPCEndpunkte optional aus, ob Sie Ihren Subnetzen Endpunkte für Amazon S3 hinzufügen möchten.
12. Stellen Sie sicher, dass DNSHostnamen aktivieren und Auflösung aktivieren aktiviert DNS sind. Weitere Informationen finden Sie unter Verwenden DNS mit Ihrem VPC.

13. Wählen Sie Erstellen VPC.
14. Ein Statusfenster zeigt den Fortschritt an. Wenn die Arbeit abgeschlossen ist, wählen Sie Ansicht, VPC um zu Ihrer VPCs Seite zu navigieren, auf der Ihre Standardseite VPC und die, VPC die Sie gerade erstellt haben, angezeigt werden. Da VPC es sich bei der von Ihnen erstellten Datei nicht um eine Standarddatei handeltVPC, wird in der VPC Spalte Standard der Wert Nein angezeigt.
15. Wenn Sie Ihren VPC mit einem DNS Eintrag verknüpfen möchten, der keinen Domainnamen enthält, navigieren Sie zu DHCPOptionssätzen, wählen Sie Optionssatz erstellen DHCP aus und lassen Sie einen Domainnamen weg. Nachdem Sie Ihren Optionssatz erstellt haben, navigieren Sie zu Ihrem neuenVPC, wählen Sie im Menü Aktionen die Option DHCPOptionssatz bearbeiten und wählen Sie den neuen Optionssatz aus. Sie können den Domainnamen nicht mit der Konsole bearbeiten, nachdem der DNS Optionssatz erstellt wurde.

    Es hat sich bei Hadoop und verwandten Anwendungen bewährt, die Auflösung des vollqualifizierten Domänennamens (FQDN) für Knoten sicherzustellen. Um eine korrekte DNS Auflösung zu gewährleisten, konfigurieren Sie eineVPC, die einen DHCP Optionssatz enthält, dessen Parameter auf die folgenden Werte gesetzt sind:

    - domain-name = **`ec2.internal`**

      Verwenden Sie **`ec2.internal`**, wenn Ihre Region USA Ost (Nord-Virginia) ist. Verwenden Sie für andere Regionen *`region-name`***`.compute.internal`**. Verwenden Sie für Beispiele in `us-west-2`**`us-west-2.compute.internal`**. Verwenden **`us-gov-west-1.compute.internal`** Sie für die Region AWS GovCloud (US-West).
    - domain-name-servers = **`AmazonProvidedDNS`**

    Weitere Informationen finden Sie unter DHCPOptionssätze im VPCAmazon-Benutzerhandbuch.
16. Gehen Sie nach der VPC Erstellung auf die Seite Subnetze und notieren Sie sich die Subnetz-ID eines der Subnetze Ihres neuen. VPC Sie verwenden diese Informationen, wenn Sie den EMR Amazon-Cluster im startenVPC.

Starten Sie Cluster in einem VPC

Nachdem Sie ein Subnetz eingerichtet haben, das für das Hosten von EMR Amazon-Clustern konfiguriert ist, starten Sie den Cluster in diesem Subnetz, indem Sie bei der Erstellung des Clusters die zugehörige Subnetz-ID angeben.

> **Note**
>
> Amazon EMR unterstützt private Subnetze in den Release-Versionen 4.2 und höher.

Wenn der Cluster gestartet wird, EMR fügt Amazon Sicherheitsgruppen hinzu, je nachdem, ob der Cluster in VPC privaten oder öffentlichen Subnetzen gestartet wird. Alle Sicherheitsgruppen erlauben den Eingang über Port 8443, um mit dem EMR Amazon-Service zu kommunizieren, aber die IP-Adressbereiche variieren für öffentliche und private Subnetze. Amazon EMR verwaltet all diese Sicherheitsgruppen und muss dem AWS Bereich im Laufe der Zeit möglicherweise weitere IP-Adressen hinzufügen. Weitere Informationen finden Sie unter Steuerung des Netzwerkverkehrs mit Sicherheitsgruppen.

Um den Cluster auf einem zu verwaltenVPC, EMR verbindet Amazon ein Netzwerkgerät mit dem primären Knoten und verwaltet es über dieses Gerät. Sie können dieses Gerät mithilfe der EC2 API Amazon-Aktion anzeigen `DescribeInstances`. Wenn Sie dieses Gerät ändern, fällt der Cluster möglicherweise aus.

Console

Um mit der Konsole einen Cluster in VPC einem zu starten

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und anschließend Create cluster aus.
3. Gehen Sie unter Netzwerk zum Feld Virtuelle private Cloud (VPC). Geben Sie den Namen Ihres ein VPC oder wählen Sie Durchsuchen, um Ihren auszuwählenVPC. Wählen Sie alternativ Create, VPC um einen zu erstellenVPC, den Sie für Ihren Cluster verwenden können.
4. Wählen Sie alle anderen Optionen aus, die für Ihren Cluster gelten.
5. Um Ihren Cluster jetzt zu starten, wählen Sie Cluster erstellen aus.

AWS CLI

Um einen Cluster in einem zu starten, VPC verwenden Sie AWS CLI

> **Note**
>
> Das AWS CLI bietet keine Möglichkeit, eine NAT Instanz automatisch zu erstellen und sie mit Ihrem privaten Subnetz zu verbinden. Um jedoch einen S3-Endpunkt in Ihrem Subnetz zu erstellen, können Sie die VPC CLI Amazon-Befehle verwenden. Verwenden Sie die Konsole, um NAT Instances zu erstellen und Cluster in einem privaten Subnetz zu starten.

Nachdem Ihr konfiguriert VPC ist, können Sie EMR Amazon-Cluster darin starten, indem Sie den `create-cluster` Unterbefehl mit dem `--ec2-attributes` Parameter verwenden. Verwenden Sie den `--ec2-attributes` Parameter, um das VPC Subnetz für Ihren Cluster anzugeben.

- Um einen Cluster in einem bestimmten Subnetz zu erstellen, geben Sie den folgenden Befehl ein: replace *myKey* mit dem Namen Ihres EC2 Amazon-Schlüsselpaars und ersetzen *77XXXX03* mit Ihrer Subnetz-ID.

  ```
  aws emr create-cluster --name "Test cluster" --release-label emr-4.2.0 --
  applications Name=Hadoop Name=Hive Name=Pig --use-default-roles --ec2-attributes
   KeyName=myKey,SubnetId=subnet-77XXXX03 --instance-type m5.xlarge --instance-
  count 3
  ```

  Wenn Sie die Instance-Anzahl ohne den `--instance-groups`-Parameter angeben, wird ein einzelner Primärknoten gestartet. Die verbleibenden Instances werden dabei als Core-Knoten gestartet. Alle Knoten verwenden den im Befehl angegebenen Instance-Typ.

  > **Note**
  >
  > Wenn Sie noch nicht die standardmäßige EMR Amazon-Servicerolle und das EC2 Instanzprofil erstellt haben, geben Sie ein, `aws emr create-default-roles` um sie zu erstellen, bevor Sie den `create-cluster` Unterbefehl eingeben.

Sicherstellen der verfügbaren IP-Adressen für einen EMR Cluster auf EC2

Um sicherzustellen, dass beim Start ein Subnetz mit ausreichend freien IP-Adressen verfügbar ist, überprüft die EC2 Subnetzauswahl die IP-Verfügbarkeit. Bei der Erstellung wird ein Subnetz mit der erforderlichen Anzahl an IP-Adressen verwendet, um Kern-, Primär- und Taskknoten nach Bedarf zu starten, auch wenn bei der ersten Erstellung nur Kernknoten für den Cluster erstellt werden. EMRüberprüft die Anzahl der IP-Adressen, die für den Start von Primär- und Taskknoten während der Erstellung erforderlich sind, und berechnet separat die Anzahl der IP-Adressen, die zum Starten von Kernknoten benötigt werden. Die Mindestanzahl der erforderlichen Primär- und Task-Instances oder Knoten wird automatisch von Amazon festgelegtEMR.

> **Important**
>
> Wenn in den Subnetzen nicht VPC genug verfügbar ist, IPs um wichtige Knoten aufzunehmen, wird ein Fehler zurückgegeben und der Cluster wird nicht erstellt.

In den meisten Bereitstellungsfällen gibt es einen Zeitunterschied zwischen den einzelnen Starts von Kern-, Primär- und Taskknoten. Darüber hinaus ist es möglich, dass sich mehrere Cluster ein Subnetz teilen. In diesen Fällen kann die Verfügbarkeit von IP-Adressen schwanken, sodass beispielsweise nachfolgende Task-Node-Starts durch verfügbare IP-Adressen eingeschränkt werden können.

Amazon-S3-Mindestrichtlinie für private Subnetze

Für private Subnetze müssen Sie Amazon mindestens die Möglichkeit bieten, auf Amazon EMR Linux-Repositorys zuzugreifen. Diese private Subnetzrichtlinie ist Teil der VPC Endpunktrichtlinien für den Zugriff auf Amazon S3. Mit Amazon EMR 5.25.0 oder höher müssen Sie Amazon Zugriff auf den System-Bucket gewähren, der Spark-Ereignisprotokolle sammelt, EMR um mit einem Klick Zugriff auf den persistenten Spark-Verlaufsserver zu ermöglichen. Wenn Sie die Protokollierung aktivieren, geben Sie PUT Berechtigungen für einen Bucket ein. `aws157-logs-*` Weitere Informationen finden Sie unter Zugriff auf den persistenten Spark History Server mit nur einem Klick.

Es ist dem Benutzer überlassen, den Businessanforderungen entsprechende Richtlinieneinschränkungen festzulegen. Sie können beispielsweise die Region `packages.us-east-1.amazonaws.com` angeben, um einen mehrdeutigen Amazon-S3-Bucket-Namen zu vermeiden. Die folgende Beispielrichtlinie bietet Berechtigungen für den Zugriff auf Amazon Linux-Repositorys und den EMR Amazon-System-Bucket zum Sammeln von Spark-Ereignisprotokollen. Ersetzen *`MyRegion`* zum Beispiel mit der Region, in der sich Ihre Log-Buckets befinden. `us-east-1`

Weitere Informationen zur Verwendung von IAM Richtlinien mit VPC Amazon-Endpunkten finden Sie unter Endpunktrichtlinien für Amazon S3.

```
{
   "Version": "2008-10-17",
   "Statement": [
       {
           "Sid": "AmazonLinuxAMIRepositoryAccess",
           "Effect": "Allow",
           "Principal": "*",
           "Action": "s3:GetObject",
           "Resource": [
            "arn:aws:s3:::packages.MyRegion.amazonaws.com/*",
            "arn:aws:s3:::repo.MyRegion.amazonaws.com/*",
            "arn:aws:s3:::repo.MyRegion.emr.amazonaws.com/*"
           ]
       },
       {
           "Sid": "EnableApplicationHistory",
           "Effect": "Allow",
           "Principal": "*",
           "Action": [
               "s3:Put*",
               "s3:Get*",
               "s3:Create*",
               "s3:Abort*",
               "s3:List*"
           ],
           "Resource": [
            "arn:aws:s3:::prod.MyRegion.appinfo.src/*"
           ]
       }
   ]
}
```

Die folgende Beispielrichtlinie stellt die Berechtigungen bereit, die für den Zugriff auf Amazon-Linux-2-Repositorys erforderlich sind. Amazon Linux 2 AMI ist die Standardeinstellung.

```
{
   "Statement": [
       {
           "Sid": "AmazonLinux2AMIRepositoryAccess",
           "Effect": "Allow",
```

```
            "Principal": "*",
            "Action": "s3:GetObject",
            "Resource": [
             "arn:aws:s3:::amazonlinux.MyRegion.amazonaws.com/*",
             "arn:aws:s3:::amazonlinux-2-repos-MyRegion/*"
            ]
        }
    ]
}
```

Weitere Ressourcen, um mehr über Folgendes zu erfahren VPCs

In den folgenden Themen erfahren Sie mehr über VPCs Subnetze.

- Private Subnetze in einem VPC
  - Szenario 2: VPC mit öffentlichen und privaten Subnetzen () NAT
  - NATInstanzen
  - Hochverfügbarkeit für VPC NAT Amazon-Instances: Ein Beispiel
- Öffentliche Subnetze in einem VPC
  - Szenario 1: VPC mit einem einzigen öffentlichen Subnetz
- Allgemeine Informationen VPC
  - VPCAmazon-Benutzerhandbuch
  - VPCPeering
  - Verwenden Sie Elastic Network Interfaces mit Ihrem VPC
  - Stellen Sie eine sichere Verbindung zu Linux-Instances her, die privat laufen VPC

Einen Cluster mit Instance-Flotten oder einheitlichen Instance-Gruppen erstellen

Wenn Sie einen Cluster erstellen und die Konfiguration der Primär-, Core- und Aufgabenknoten angeben, stehen Ihnen zwei Konfigurationsoptionen zur Verfügung. Sie können Instance-Flotten oder einheitliche Instance-Gruppen verwenden. Die Konfigurationsoption, die Sie auswählen, gilt für alle Knoten und für die gesamte Nutzungsdauer des Clusters. Instance-Flotten und Instance-Gruppen können nicht gleichzeitig in einem Cluster vorhanden sein. Die Konfiguration der Instance-Flotten ist in EMR Amazon-Version 4.8.0 und höher verfügbar, mit Ausnahme der 5.0.x-Versionen.

Sie können die EMR Amazon-Konsole, die oder Amazon verwenden AWS CLI, EMR API um Cluster mit einer der beiden Konfigurationen zu erstellen. Wenn Sie den Befehl `create-cluster` in der

AWS CLI verwenden, erstellen Sie den Cluster mit den `--instance-fleets`-Parametern, um den Cluster mittels Instance-Flotten zu erstellen, oder mit den `--instance-groups`-Parametern, um den Cluster mittels einheitlicher Instance-Gruppen zu erstellen.

Das Gleiche gilt für die Nutzung des Amazon EMRAPI. Sie geben mit der Konfiguration `InstanceGroups` eine Reihe von `InstanceGroupConfig`-Objekten oder mit der Konfiguration `InstanceFleets` eine Reihe von `InstanceFleetConfig`-Objekten an.

In der neuen EMR Amazon-Konsole können Sie wählen, ob Sie bei der Erstellung eines Clusters entweder Instance-Gruppen oder Instance-Flotten verwenden möchten, und Sie haben die Möglichkeit, Spot-Instances mit jeder zu verwenden. Wenn Sie mit der alten EMR Amazon-Konsole bei der Erstellung Ihres Clusters die Standardeinstellungen für Schnelloptionen verwenden, EMR wendet Amazon die Konfiguration einheitlicher Instance-Gruppen auf den Cluster an und verwendet On-Demand-Instances. Um Spot-Instances mit einheitlichen Instance-Gruppen zu verwenden oder Instance-Flotten und anderen Anpassungen zu konfigurieren, wählen Sie Advanced Options (Erweiterte Optionen) aus.

Instance-Flotten

Die Konfiguration der Instance-Flotten bietet die größte Vielfalt an Bereitstellungsoptionen für Amazon-Instances. EC2 Jeder Knotentyp verfügt über eine einzelne Instance-Flotte und die Aufgaben-Instance-Flotte ist optional. Sie können bis zu fünf EC2 Instance-Typen pro Flotte oder 30 EC2 Instance-Typen pro Flotte angeben, wenn Sie einen Cluster mithilfe von AWS CLI oder Amazon EMR API und einer Zuweisungsstrategie für On-Demand- und Spot-Instances erstellen. Für die Core- und Aufgaben-Instance-Flotten weisen Sie eine Zielkapazität für On-Demand-Instances und eine zweite für Spot Instances zu. Amazon EMR wählt eine beliebige Mischung der angegebenen Instance-Typen, um die Zielkapazitäten zu erfüllen, und stellt sowohl On-Demand-Instances als auch Spot-Instances bereit.

Für den primären Knotentyp EMR wählt Amazon einen einzelnen Instance-Typ aus Ihrer Instance-Liste aus, und Sie geben an, ob er als On-Demand- oder Spot-Instance bereitgestellt wird. Instance-Flotten bieten auch zusätzliche Optionen für Spot Instance- und On-Demand-Käufe. Zu den Spot-Instance-Optionen gehören ein Timeout, das festlegt, welche Maßnahme ergriffen werden soll, wenn Spot-Kapazität nicht bereitgestellt werden kann, und eine bevorzugte Zuweisungsstrategie (kapazitätsoptimiert) für den Start von Spot Instance-Flotten. On-Demand-Instance-Flotten können auch mit der Option der Zuweisungsstrategie (niedrigster Preis) gestartet werden. Wenn Sie eine Servicerolle verwenden, die nicht die EMR Standard-Servicerolle ist, oder eine EMR verwaltete Richtlinie in Ihrer Servicerolle verwenden, müssen Sie der benutzerdefinierten Cluster-Servicerolle

zusätzliche Berechtigungen hinzufügen, um die Option für die Zuweisungsstrategie zu aktivieren. Weitere Informationen finden Sie unter Servicerolle für Amazon EMR (EMRRolle).

Weitere Information zum Konfigurieren von Instance-Flotten finden Sie unter Instance-Flotten konfigurieren.

Einheitliche Instance-Gruppen

Einheitliche Instance-Gruppen bieten eine einfachere Einrichtung als Instance-Flotten. Jeder EMR Amazon-Cluster kann bis zu 50 Instance-Gruppen umfassen: eine primäre Instance-Gruppe, die eine EC2 Amazon-Instance enthält, eine Core-Instance-Gruppe, die eine oder mehrere EC2 Instances enthält, und bis zu 48 optionale Task-Instance-Gruppen. Jede Kern- und Task-Instance-Gruppe kann eine beliebige Anzahl von EC2 Amazon-Instances enthalten. Sie können jede Instance-Gruppe skalieren, indem Sie EC2 Amazon-Instances manuell hinzufügen und entfernen, oder Sie können die automatische Skalierung einrichten. Weitere Informationen über das Hinzufügen und Entfernen von Instances finden Sie unter Clusterskalierung verwenden.

Weitere Informationen zum Konfigurieren von einheitlichen Instance-Gruppen finden Sie unter Einheitliche Instance-Gruppen konfigurieren.

Arbeiten mit Instance-Flotten und Instance-Gruppen

Themen

- Instance-Flotten konfigurieren
- Kapazitätsreservierungen mit Instance-Flotten verwenden
- Einheitliche Instance-Gruppen konfigurieren
- Bewährte Methoden für Instance- und Availability Zone-Flexibilität
- Bewährte Methoden für die Konfiguration des Clusters

Instance-Flotten konfigurieren

> **Note**
>
> Die Konfiguration der Instance-Flotten ist nur in EMR Amazon-Versionen 4.8.0 und höher verfügbar, mit Ausnahme von 5.0.0 und 5.0.3.

Mit der Instance-Flottenkonfiguration für EMR Amazon-Cluster können Sie eine Vielzahl von Bereitstellungsoptionen für EC2 Amazon-Instances auswählen und eine flexible und elastische Ressourcenstrategie für jeden Knotentyp in Ihrem Cluster entwickeln.

In einer Instance-Flottenkonfiguration geben Sie eine Zielkapazität für On-Demand-Instances und Spot Instances innerhalb jeder Flotte an. Wenn der Cluster gestartet wird, stellt Amazon EMR Instances bereit, bis die Ziele erfüllt sind. Wenn Amazon aufgrund einer Preiserhöhung oder eines Instance-Fehlers eine Spot-Instance in einem laufenden Cluster EC2 zurückfordert, EMR versucht Amazon, die Instance durch einen der von Ihnen angegebenen Instance-Typen zu ersetzen. Dies erleichtert die Wiedererlangung der Kapazität während eines Anstiegs der Spot-Preise.

Sie können maximal fünf EC2 Amazon-Instance-Typen pro Flotte angeben, die Amazon EMR bei der Erfüllung der Ziele verwendet, oder maximal 30 EC2 Amazon-Instance-Typen pro Flotte, wenn Sie einen Cluster mit AWS CLI oder Amazon EMR API und einer Zuweisungsstrategie für On-Demand- und Spot-Instances erstellen.

Sie können auch mehrere Subnetze für verschiedene Availability Zones auswählen. Wenn Amazon den Cluster EMR startet, durchsucht es diese Subnetze nach den von Ihnen angegebenen Instances und Kaufoptionen. Wenn Amazon ein AWS großes Ereignis in einer oder mehreren Availability Zones EMR feststellt, versucht Amazon EMR automatisch, den Verkehr von den betroffenen Availability Zones wegzuleiten und versucht, neue Cluster zu starten, die Sie entsprechend Ihrer Auswahl in alternativen Availability Zones erstellen. Beachten Sie, dass die Auswahl der Cluster-Availability-Zone nur bei der Cluster-Erstellung erfolgt. Bestehende Clusterknoten werden bei einem Ausfall der Availability Zone nicht automatisch in einer neuen Availability Zone neu gestartet.

Überlegungen zur Arbeit mit Instance-Flotten

Beachten Sie die folgenden Punkte, wenn Sie Instance-Flotten mit Amazon EMR verwenden.

- Sie können eine Instance-Flotte haben, und zwar nur eine pro Knotentyp (Primär, Core, Aufgabe). Sie können bis zu fünf EC2 Amazon-Instance-Typen für jede Flotte auf der angeben AWS Management Console (oder maximal 30 Typen pro Instance-Flotte, wenn Sie einen Cluster mit AWS CLI oder Amazon EMR API und an erstellenZuweisungsstrategie für Flotten).
- Amazon EMR wählt einen oder alle der angegebenen EC2 Amazon-Instance-Typen aus, um sowohl Spot- als auch On-Demand-Kaufoptionen bereitzustellen.
- Legen Sie Zielkapazitäten für Spot- und On-Demand-Instances für die Core- und Aufgaben-Flotte fest. Verwenden Sie v CPU oder eine generische Einheit, die jeder EC2 Amazon-Instance zugewiesen ist, die auf die Ziele angerechnet wird. Amazon EMR stellt Instances bereit, bis jede Zielkapazität vollständig erfüllt ist. Für die Primär-Flotte ist das Ziel immer auf 1 gesetzt.

- Sie können ein Subnetz (Availability Zone) oder einen Bereich auswählen. Wenn Sie sich für einen Bereich entscheiden, stellt Amazon EMR die Kapazität in der Availability Zone bereit, die am besten zu Ihnen passt.
- Hinweise zum Angeben der Zielkapazität für Spot-Instances:
  - Bestimmen Sie für jeden Instance-Typ einen maximalen Spot-Preis. Amazon stellt EMR Spot-Instances bereit, wenn der Spot-Preis unter dem maximalen Spot-Preis liegt. Sie zahlen den Spot-Preis, nicht unbedingt den maximalen Spot-Preis.
  - Für jede Flotte definieren Sie einen Timeout-Zeitraum für die Bereitstellung von Spot-Instances. Wenn Amazon keine Spot-Kapazität bereitstellen EMR kann, können Sie den Cluster beenden oder stattdessen zur Bereitstellung von On-Demand-Kapazität wechseln. Dies gilt nur für die Bereitstellung von Clustern, nicht für deren Größenänderung. Wenn der Timeout-Zeitraum während der Größenänderung des Clusters endet, werden Spot-Anfragen, die nicht bereitgestellt wurden, für ungültig erklärt, ohne dass sie auf On-Demand-Kapazität übertragen werden.
- Für jede Flotte können Sie eine der folgenden Zuweisungsstrategien für Ihre Spot-Instances angeben: preiskapazitätsoptimiert, kapazitätsoptimiert capacity-optimized-prioritized, kostengünstigster Preis oder diversifiziert über alle Pools hinweg.
- Für jede Flotte können Sie die folgenden Zuweisungsstrategien für Ihre On-Demand-Instances anwenden: die Strategie mit dem niedrigsten Preis oder die Strategie mit Priorität.
- Für jede Flotte mit On-Demand-Instances können Sie wählen, ob Sie Optionen zur Kapazitätsreservierung anwenden möchten.
- Wenn Sie eine Zuweisungsstrategie für Instance-Flotten verwenden, sollten Sie bei der Auswahl von Subnetzen für Ihren EMR Cluster die folgenden Überlegungen beachten:
  - Wenn Amazon EMR einen Cluster mit einer Task-Flotte bereitstellt, filtert es Subnetze heraus, denen genügend verfügbare IP-Adressen fehlen, um alle Instances des angeforderten EMR Clusters bereitzustellen. Dazu gehören IP-Adressen, die für die Primär-, Kern- und Task-Instance-Flotten beim Cluster-Start erforderlich sind. Amazon nutzt EMR dann seine Zuweisungsstrategie, um den Instance-Pool auf der Grundlage des Instance-Typs und der verbleibenden Subnetze mit ausreichend IP-Adressen zu bestimmen, um den Cluster zu starten.
  - Wenn Amazon aufgrund unzureichender verfügbarer IP-Adressen EMR nicht den gesamten Cluster starten kann, versucht Amazon, Subnetze mit ausreichend freien IP-Adressen zu identifizieren, um die wesentlichen (Kern- und primären) Instance-Flotten zu starten. In solchen Szenarien wird Ihre Task-Instance-Flotte in einen angehaltenen Zustand versetzt, anstatt den Cluster mit einem Fehler zu beenden.

- Wenn keines der angegebenen Subnetze genügend IP-Adressen für die Bereitstellung der wichtigsten Core- und primären Instance-Flotten enthält, schlägt der Clusterstart mit einem _ fehl. VALIDATION ERROR Dadurch wird ein Ereignis mit einem CRITICALschwerwiegenden Clusterabbruch ausgelöst, das Sie darüber informiert, dass der Cluster nicht gestartet werden kann. Um dieses Problem zu vermeiden, empfehlen wir, die Anzahl der IP-Adressen in Ihren Subnetzen zu erhöhen.
- Wenn Sie On-Demand-Instances starten, können Sie offene oder gezielte Kapazitätsreservierungen für Primär-, Kern- und Task-Knoten in Ihren Konten verwenden. Bei On-Demand-Instances mit einer Zuweisungsstrategie für Instance-Flotten ist die Kapazität möglicherweise nicht ausreichend. Wir empfehlen, dass Sie mehrere Instance-Typen angeben, um zu diversifizieren und das Risiko einer unzureichenden Kapazität zu verringern. Weitere Informationen finden Sie unter the section called “Kapazitätsreservierungen mit Instance-Flotten verwenden”.

Instance-Flotten-Optionen

Verwenden Sie die folgenden Richtlinien, um Instance-Flotten-Optionen zu verstehen.

Themen

Festlegen von Zielkapazitäten

Geben Sie die Zielkapazitäten für die Core- und Aufgaben-Flotte an. Wenn Sie dies tun, bestimmt dies die Anzahl der On-Demand-Instances und Spot-Instances, die Amazon EMR bereitstellt. Wenn Sie eine Instance angeben, können Sie entscheiden, wie viel jede Instance beim Ziel mit eingerechnet wird. Wenn eine On-Demand-Instance bereitgestellt wird, wird sie beim On-Demand-Ziel mit eingerechnet. Dies gilt auch für Spot-Instances. Im Gegensatz zu Core- und Aufgaben-Flotten besteht die Primär-Flotte immer aus einer Instance. Daher ist die Zielkapazität für diese Flotte immer auf 1 gesetzt.

Wenn Sie die Konsole verwenden, werden standardmäßig die vom EC2 Amazon-Instance-Typ als Anzahl für die Zielkapazitäten verwendet. vCPUs Sie können dies in Generische Einheiten ändern

und dann die Anzahl für jeden EC2 Instance-Typ angeben. Wenn Sie den verwenden AWS CLI, weisen Sie jedem Instanztyp manuell generische Einheiten zu.

> **Important**
>
> Wenn Sie mithilfe von einen Instanztyp auswählen AWS Management Console, entspricht die Anzahl von v, die für jeden Instanztyp CPU angezeigt wird, der Anzahl der YARN virtuellen Kerne für diesen Instanztyp, nicht der Anzahl von EC2 vCPUs für diesen Instanztyp. Weitere Informationen zur Anzahl der vCPUs einzelnen Instance-Typen finden Sie unter EC2Amazon-Instance-Typen.

Für jede Flotte geben Sie bis zu fünf EC2 Amazon-Instance-Typen an. Wenn Sie einen verwenden Zuweisungsstrategie für Flotten und mit dem AWS CLI oder Amazon einen Cluster erstellen EMRAPI, können Sie bis zu 30 EC2 Instance-Typen pro Instance-Flotte angeben. Amazon EMR wählt eine beliebige Kombination dieser EC2 Instance-Typen, um Ihre Zielkapazitäten zu erfüllen. Da Amazon die Zielkapazität vollständig füllen EMR möchte, kann es zu einer Überschreitung kommen. Wenn es beispielsweise zwei nicht ausgefüllte Einheiten gibt und Amazon nur eine Instance mit einer Anzahl von fünf Einheiten bereitstellen EMR kann, wird die Instance trotzdem bereitgestellt, was bedeutet, dass die Zielkapazität um drei Einheiten überschritten wird.

Wenn Sie die Zielkapazität reduzieren, um die Größe eines laufenden Clusters zu ändern, EMR versucht Amazon, Anwendungsaufgaben abzuschließen und beendet Instances, um das neue Ziel zu erreichen. Weitere Informationen finden Sie unter Beendigung bei Aufgaben-Abschluss.

Start-Optionen

Für Spot Instances können Sie einen maximalen Spot-Preis für jeden Instance-Typ in einer Flotte angeben. Sie können den Preis entweder als Prozentsatz des On-Demand-Preises oder als einen bestimmten Betrag in US-Dollar festlegen. Amazon stellt EMR Spot-Instances bereit, wenn der aktuelle Spot-Preis in einer Availability Zone unter Ihrem maximalen Spot-Preis liegt. Sie zahlen den Spot-Preis, nicht unbedingt den maximalen Spot-Preis.

> **Note**
>
> Spot-Instances mit definierter Laufzeit (auch Spot-Blöcke genannt) stehen Neukunden ab dem 1. Juli 2021 nicht mehr zur Verfügung. Für Kunden, die diese Funktion bereits genutzt

haben, werden wir Spot-Instances mit einer definierten Laufzeit bis zum 31. Dezember 2022 weiterhin unterstützen.

In Amazon EMR 5.12.1 und höher verfügbar, haben Sie die Möglichkeit, Spot- und On-Demand-Instance-Flotten mit optimierter Kapazitätszuweisung zu starten. Diese Option für die Zuweisungsstrategie kann in der alten Version AWS Management Console oder mithilfe von festgelegt werden. API `RunJobFlow` Beachten Sie, dass Sie die Zuweisungsstrategie in der neuen-Konsole nicht anpassen können. Für die Verwendung der Option „Zuweisungsstrategie“ sind zusätzliche Berechtigungen für Servicerollen erforderlich. Wenn Sie die standardmäßige EMR Amazon-Servicerolle und die verwaltete Richtlinie (`EMR_DefaultRole`und`AmazonEMRServicePolicy_v2`) für den Cluster verwenden, sind die Berechtigungen für die Option Zuweisungsstrategie bereits enthalten. Wenn Sie nicht die standardmäßige EMR Amazon-Servicerolle und die verwaltete Richtlinie verwenden, müssen Sie sie hinzufügen, um diese Option nutzen zu können. Siehe Servicerolle für Amazon EMR (EMRRolle).

Weitere Informationen zu Spot-Instances finden Sie unter Spot-Instances im EC2 Amazon-Benutzerhandbuch. Weitere Informationen zu On-Demand-Instances finden Sie unter On-Demand-Instances im EC2 Amazon-Benutzerhandbuch.

Wenn Sie On-Demand-Instance-Flotten mit der Zuweisungsstrategie zum niedrigsten Preis starten möchten, haben Sie die Möglichkeit, Kapazitätsreservierungen zu verwenden. Optionen zur Kapazitätsreservierung können über Amazon festgelegt werden EMR API`RunJobFlow`. Für Kapazitätsreservierungen sind zusätzliche Berechtigungen für Servicerollen erforderlich, die Sie hinzufügen müssen, um diese Optionen nutzen zu können. Siehe Zuweisungsstrategie-Berechtigungen. Beachten Sie, dass Sie Kapazitätsreservierungen in der neuen Konsole nicht anpassen können.

Optionen für mehrere Subnetze (Availability Zones)

Wenn Sie Instance-Flotten verwenden, können Sie mehrere EC2 Amazon-Subnetze innerhalb einer angebenVPC, die jeweils einer anderen Availability Zone entsprechen. Wenn Sie EC2 -Classic verwenden, geben Sie Availability Zones explizit an. Amazon EMR identifiziert die beste Availability Zone zum Starten von Instances gemäß Ihren Flottenspezifikationen. Instances werden immer nur in einer Availability Zone bereitgestellt. Sie können private Subnetze oder öffentliche Subnetze auswählen, aber Sie können beide nicht mischen, und die von Ihnen angegebenen Subnetze müssen sich innerhalb derselben befinden. VPC

Hauptknoten-Konfiguration

Da die Primär-Instance-Flotte ist nur eine einzelne Instance ist, unterscheidet sich ihre Konfiguration etwas von Core- und Task-Instance-Flotten. Wählen Sie entweder On-Demand oder Spot für die Primär-Instance-Flotte aus, da sie nur aus einer Instance besteht. Wenn Sie die Konsole verwenden, um die Instance-Flotte zu erstellen, wird die Zielkapazität für die Kaufoption, die Sie auswählen, auf "1" festgelegt. Wenn Sie die verwenden AWS CLI, legen Sie je nach Bedarf immer entweder `TargetSpotCapacity` oder `TargetOnDemandCapacity` auf 1 fest. Sie können weiterhin bis zu fünf Instance-Typen für die primäre Instance-Flotte wählen (oder maximal 30, wenn Sie die Zuweisungsstrategie-Option für On-Demand- oder Spot Instances verwenden). Im Gegensatz zu Core- und Task-Instance-Flotten, bei denen Amazon EMR möglicherweise mehrere Instances unterschiedlichen Typs bereitstellt, EMR wählt Amazon jedoch einen einzigen Instance-Typ für die Bereitstellung für die primäre Instance-Flotte aus.

Zuweisungsstrategie für Flotten

Mit den EMR Amazon-Versionen 5.12.1 und höher können Sie die Option für die Zuweisungsstrategie mit On-Demand-Instances und Spot-Instances für jeden Clusterknoten verwenden. Wenn Sie einen Cluster mithilfe der AWS CLI Amazon- oder EMR API EMR Amazon-Konsole mit einer Zuweisungsstrategie erstellen, können Sie bis zu 30 EC2 Amazon-Instance-Typen pro Flotte angeben. Mit der standardmäßigen Konfiguration der EMR Amazon-Cluster-Instance-Flotte können Sie bis zu 5 Instance-Typen pro Flotte verwenden. Wir empfehlen Ihnen, die Option für die Zuweisungsstrategie zu verwenden, um eine schnellere Cluster-Bereitstellung, eine genauere Spot-Instance-Zuweisung und weniger Spot Instance-Unterbrechungen zu erzielen.

Themen

- Zuweisungsstrategie mit On-Demand-Instances
- Zuweisungsstrategie mit Spot Instances
- Zuweisungsstrategie-Berechtigungen
- Erforderliche IAM Berechtigungen für eine Zuweisungsstrategie

Zuweisungsstrategie mit On-Demand-Instances

Die folgenden Zuweisungsstrategien sind für Ihre On-Demand-Instances verfügbar:

`lowest-price`(Standard)

Bei der Zuweisungsstrategie mit dem niedrigsten Preis werden On-Demand-Instances aus dem Pool mit dem niedrigsten Preis gestartet, der über verfügbare Kapazität verfügt. Wenn der Pool mit dem niedrigsten Preis keine verfügbare Kapazität hat, stammen die On-Demand-Instances aus dem Pool mit dem nächstniedrigsten Preis und verfügbarer Kapazität.

`prioritized`

Mit der priorisierten Zuweisungsstrategie können Sie für jeden Instance-Typ Ihrer Instance-Flotte einen Prioritätswert angeben. Amazon EMR startet Ihre On-Demand-Instances mit der höchsten Priorität. Wenn Sie diese Strategie verwenden, müssen Sie die Priorität für mindestens einen Instance-Typ konfigurieren. Wenn Sie den Prioritätswert für einen Instance-Typ nicht konfigurieren, EMR weist Amazon diesem Instance-Typ die niedrigste Priorität zu. Jede Instance-Flotte (primär, Core oder Task) in einem Cluster kann für einen bestimmten Instance-Typ einen anderen Prioritätswert haben.

> **Note**
>
> Wenn Sie die capacity-optimized-prioritizedSpot-Zuweisungsstrategie verwenden, EMR wendet Amazon bei der Festlegung von Prioritäten dieselben Prioritäten sowohl auf Ihre On-Demand-Instances als auch auf Spot-Instances an.

Zuweisungsstrategie mit Spot Instances

Für Spot Instances können Sie aus einer der folgenden Zuweisungsstrategien wählen:

`price-capacity-optimized` (empfohlen)

Bei der preis-kapazitätsoptimierten Zuweisungsstrategie werden Spot Instances aus den Spot Instance-Pools gestartet, die über die höchste verfügbare Kapazität und den niedrigsten Preis für die Anzahl der zu startenden Instances verfügen. Aus diesem Grund bietet die Strategie mit optimierter Preis- und Kapazitätsoptimierung in der Regel eine höhere Wahrscheinlichkeit, Spot-Kapazität zu erhalten, und führt zu niedrigeren Unterbrechungsraten. Dies ist die Standardstrategie für EMR Amazon-Versionen 6.10.0 und höher.

`capacity-optimized`

Die kapazitätsoptimierte Zuweisungsstrategie startet Spot Instances in den am meisten verfügbaren Pools mit der geringsten Wahrscheinlichkeit einer kurzfristigen Unterbrechung. Dies

ist eine gute Option für Workloads, bei denen Unterbrechungen aufgrund von Neustarts von Aufgaben höhere Kosten verursachen. Dies ist die Standardstrategie für EMR Amazon-Versionen 6.9.0 und niedriger.

`capacity-optimized-prioritized`

Mit der capacity-optimized-prioritized Zuweisungsstrategie können Sie für jeden Instance-Typ in Ihrer Instance-Flotte einen Prioritätswert angeben. Amazon EMR optimiert zunächst die Kapazität, berücksichtigt jedoch die Prioritäten des Instance-Typs nach bestem Wissen und Gewissen, z. B. wenn die Priorität die Fähigkeit der Flotte, optimale Kapazität bereitzustellen, nicht wesentlich beeinträchtigt. Wir empfehlen diese Option, wenn Sie Workloads haben, die möglichst wenig unterbrochen werden müssen und dennoch bestimmte Instance-Typen benötigt werden. Wenn Sie diese Strategie verwenden, müssen Sie die Priorität für mindestens einen Instance-Typ konfigurieren. Wenn Sie für keinen Instance-Typ eine Priorität konfigurieren, EMR weist Amazon diesem Instance-Typ den niedrigsten Prioritätswert zu. Jede Instance-Flotte (primär, Core oder Task) in einem Cluster kann für einen bestimmten Instance-Typ einen anderen Prioritätswert haben.

> **Note**
>
> Wenn Sie die priorisierte On-Demand-Zuweisungsstrategie verwenden, EMR wendet Amazon bei der Festlegung von Prioritäten den gleichen Prioritätswert sowohl auf Ihre On-Demand-Instances als auch auf Ihre Spot-Instances an.

`diversified`

Mit der diversifizierten Zuweisungsstrategie EC2 verteilt Amazon Spot-Instances auf alle Spot-Kapazitätspools.

`lowest-price`

Bei der preisgünstigsten Zuweisungsstrategie werden Spot Instances aus dem preisgünstigsten Pool mit verfügbarer Kapazität gestartet. Wenn der günstigste Pool keine verfügbare Kapazität aufweist, kommen die Spot Instances aus dem nächstgünstigsten Pool mit verfügbarer Kapazität. Wenn die Kapazität eines Pools knapp wird, bevor er die von Ihnen angeforderte Kapazität erfüllt, greift die EC2 Amazon-Flotte auf den Pool mit dem nächstniedrigsten Preis zurück, um Ihre Anfrage weiter zu bearbeiten. Damit die gewünschte Kapazität auf jeden Fall erreicht wird, erhalten Sie möglicherweise Spot-Instances aus mehreren Pools. Da diese Strategie nur

den Instance-Preis und nicht die Kapazitätsverfügbarkeit berücksichtigt, kann es zu hohen Unterbrechungsraten kommen.

Zuweisungsstrategie-Berechtigungen

Für die Option „Zuweisungsstrategie“ sind mehrere IAM Berechtigungen erforderlich, die automatisch in der standardmäßigen EMR Amazon-Servicerolle und der von Amazon EMR verwalteten Richtlinie (`EMR_DefaultRoleundAmazonEMRServicePolicy_v2`) enthalten sind. Wenn Sie eine benutzerdefinierte Servicerolle oder eine verwaltete Richtlinie für Ihren Cluster verwenden, müssen Sie diese Berechtigungen hinzufügen, bevor Sie den Cluster erstellen. Weitere Informationen finden Sie unter Zuweisungsstrategie-Berechtigungen.

Optionale On-Demand-Kapazitätsreservierungen (ODCRs) sind verfügbar, wenn Sie die Option für die On-Demand-Zuweisungsstrategie verwenden. Mit den Optionen zur Kapazitätsreservierung können Sie angeben, ob reservierte Kapazität zuerst für EMR Amazon-Cluster verwendet werden soll. Auf diese Weise können Sie sicherstellen, dass Ihre kritischen Workloads die Kapazität nutzen, die Sie bereits mit Open oder Targeted ODCRs reserviert haben. Bei unkritischen Workloads können Sie in den Einstellungen für die Kapazitätsreservierung angeben, ob reservierte Kapazität verbraucht werden soll.

Kapazitätsreservierungen können nur von Instances verwendet werden, die ihren Attributen (Instance-Typ, Plattform und Availability Zone) entsprechen. Standardmäßig werden offene Kapazitätsreservierungen automatisch von Amazon verwendet, EMR wenn On-Demand-Instances bereitgestellt werden, die den Instance-Attributen entsprechen. Wenn Sie keine laufenden Instances haben, die den Attributen der Kapazitätsreservierungen entsprechen, bleiben diese ungenutzt, bis Sie eine Instance starten, die ihren Attributen entspricht. Wenn Sie beim Starten Ihres Clusters keine Kapazitätsreservierungen verwenden möchten, müssen Sie in den Startoptionen die Einstellung „Kapazitätsreservierung“ auf Keine setzen.

Sie können jedoch auch eine Kapazitätsreservierung für bestimmte Workloads festlegen. Auf diese Weise können Sie explizit steuern, welche Instances in der reservierten Kapazität ausgeführt werden dürfen. Weitere Informationen über On-Demand-Kapazitätsreservierungen finden Sie unter Kapazitätsreservierungen mit Instance-Flotten verwenden.

Erforderliche IAM Berechtigungen für eine Zuweisungsstrategie

Ihre Servicerolle für Amazon EMR (EMRRolle) benötigen zusätzliche Berechtigungen, um einen Cluster zu erstellen, der die Zuweisungsstrategieoption für On-Demand-Instance-Flotten oder Spot-Instance-Flotten verwendet.

Wir nehmen diese Berechtigungen automatisch in die standardmäßige EMR Amazon-Servicerolle EMR_DefaultRoleund die von Amazon EMR verwaltete Richtlinie auf AmazonEMRServicePolicy_v2.

Wenn Sie eine benutzerdefinierte Servicerolle oder eine verwaltete Richtlinie für Ihren Cluster verwenden, müssen Sie die folgenden Berechtigungen hinzufügen:

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Effect": "Allow",
   "Action": [
    "ec2:DeleteLaunchTemplate",
    "ec2:CreateLaunchTemplate",
    "ec2:DescribeLaunchTemplates",
    "ec2:CreateLaunchTemplateVersion",
    "ec2:CreateFleet"
   ],
   "Resource": "*"
  }
}
```

Die folgenden Berechtigungen für Servicerollen sind erforderlich, um einen Cluster zu erstellen, der offene oder gezielte Kapazitätsreservierungen verwendet. Sie müssen diese Berechtigungen zusätzlich zu den Berechtigungen angeben, die für die Verwendung der Zuweisungsstrategie-Option erforderlich sind.

Example Richtliniendokument für Kapazitätsreservierungen für Servicerollen

Um offene Kapazitätsreservierungen verwenden zu können, müssen Sie die folgenden zusätzlichen Berechtigungen angeben.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeCapacityReservations",
                "ec2:DescribeLaunchTemplateVersions",
```

```
                "ec2:DeleteLaunchTemplateVersions"
            ],
            "Resource": "*"
        }
    ]
}
```

Example

Um gezielte Kapazitätsreservierungen verwenden zu können, müssen Sie die folgenden zusätzlichen Berechtigungen angeben.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeCapacityReservations",
                "ec2:DescribeLaunchTemplateVersions",
                "ec2:DeleteLaunchTemplateVersions",
                "resource-groups:ListGroupResources"
            ],
            "Resource": "*"
        }
    ]
}
```

Instance-Flotten für Ihren Cluster konfigurieren

Console

Um mit der Konsole einen Cluster mit Instance-Flotten zu erstellen

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und anschließend Create cluster aus.
3. Wählen Sie unter Clusterkonfiguration die Option Instance-Flotten aus.
4. Wählen Sie für jede Knotengruppe die Option Instance-Typ hinzufügen und wählen Sie bis zu 5 Instance-Typen für Primär- und Core-Instance-Flotten und bis zu fünfzehn Instance-Typen

für Aufgaben-Instance-Flotten aus. Amazon EMR kann beim Start des Clusters eine beliebige Mischung dieser Instance-Typen bereitstellen.

5. Wählen Sie unter jedem Knotengruppentyp das Drop-Down-Menü Aktionen neben jeder Instance aus, um diese Einstellungen zu ändern:

   EBSVolumen hinzufügen

   Geben Sie EBS Volumes an, die an den Instance-Typ angehängt werden sollen, nachdem EMR Amazon ihn bereitgestellt hat.

   Gewichtete Kapazität bearbeiten

   Ändern Sie diesen Wert für die Core-Knotengruppe auf eine beliebige Anzahl von Einheiten, die Ihren Anwendungen entspricht. Die Anzahl von YARN vCores für jeden Flotteninstance-Typ wird als standardmäßige gewichtete Kapazitätseinheiten verwendet. Sie können die gewichtete Kapazität für den Primärknoten nicht bearbeiten.

   Den maximalen Spot-Preis bearbeiten

   Geben Sie für jeden Instance-Typ in einer Flotte einen maximalen Spot-Preis an. Sie können den Preis entweder als Prozentsatz des On-Demand-Preises oder als einen bestimmten Betrag in US-Dollar festlegen. Wenn der aktuelle Spot-Preis in einer Availability Zone unter Ihrem maximalen Spot-Preis liegt, stellt Amazon EMR Spot-Instances bereit. Sie zahlen den Spot-Preis, nicht unbedingt den maximalen Spot-Preis.

6. Um optional Sicherheitsgruppen für Ihre Knoten hinzuzufügen, erweitern Sie EC2Sicherheitsgruppen (Firewall) im Bereich Netzwerk und wählen Sie Ihre Sicherheitsgruppe für jeden Knotentyp aus.
7. Aktivieren Sie optional das Kontrollkästchen neben Zuweisungsstrategie anwenden, wenn Sie die Option Zuweisungsstrategie verwenden möchten, und wählen Sie die Zuweisungsstrategie aus, die Sie für die Spot Instances angeben möchten. Sie sollten diese Option nicht auswählen, wenn Ihre EMR Amazon-Servicerolle nicht über die erforderlichen Berechtigungen verfügt. Weitere Informationen finden Sie unter Zuweisungsstrategie für Flotten.
8. Wählen Sie alle anderen Optionen aus, die für Ihren Cluster gelten.
9. Um Ihren Cluster jetzt zu starten, wählen Sie Cluster erstellen aus.

AWS CLI

Um einen Cluster mit Instance-Flotten mit den zu erstellen und zu starten AWS CLI, folgen Sie diesen Richtlinien:

- Zum Erstellen und Starten eines Clusters mit Instance-Flotten verwenden Sie den Befehl `create-cluster` zusammen mit `--instance-fleet`-Parametern.
- Um mehr Konfigurationsdetails der Instance-Flotten in einem Cluster zu erhalten, verwenden Sie den Befehl `list-instance-fleets`.
- AMIsUm einem Cluster, den Sie erstellen, mehrere benutzerdefinierte Amazon Linux-Benutzer hinzuzufügen, verwenden Sie die `CustomAmiId` Option für jede `InstanceType` Spezifikation. Sie können Instance-Flottenknoten mit mehreren Instance-Typen und mehreren benutzerdefinierten Instance-Typen konfigurierenAMIs, um Ihren Anforderungen zu entsprechen. Siehe Beispiele: Erstellen eines Clusters mit der Instance-Flotten-Konfiguration.
- Wenn Sie die Zielkapazität für eine Instance-Flotte ändern möchten, verwenden Sie den Befehl `modify-instance-fleet`.
- Zum Hinzufügen einer Aufgaben-Instance-Flotte zu einem Cluster, dem noch keine Flotte zugewiesen wurde, verwenden Sie den Befehl `add-instance-fleet`.
- Mithilfe des CustomAmiId Arguments mit dem add-instance-fleet Befehl AMIs können der Task-Instance-Flotte mehrere benutzerdefinierte hinzugefügt werden. Siehe Beispiele: Erstellen eines Clusters mit der Instance-Flotten-Konfiguration.
- Um die Option für die Zuweisungsstrategie bei der Erstellung einer Instance-Flotte zu verwenden, aktualisieren Sie die Servicerolle so, dass sie das Beispielrichtliniendokument im folgenden Abschnitt enthält.
- Um die Optionen für Kapazitätsreservierungen bei der Erstellung einer Instance-Flotte mit On-Demand-Zuweisungsstrategie zu verwenden, aktualisieren Sie die Servicerolle so, dass sie das Beispielrichtliniendokument im folgenden Abschnitt enthält.
- Die Instance-Flotten sind automatisch in der EMR Standard-Servicerolle und der von Amazon EMR verwalteten Richtlinie (`EMR_DefaultRole`und`AmazonEMRServicePolicy_v2`) enthalten. Wenn Sie eine benutzerdefinierte Servicerolle oder eine vom Kunden verwaltete Richtlinie für Ihren Cluster verwenden, müssen Sie die neuen Berechtigungen für die Zuweisungsstrategie im folgenden Abschnitt hinzufügen.

Beispiele: Erstellen eines Clusters mit der Instance-Flotten-Konfiguration

Die folgenden Beispiele zeigen `create-cluster`-Befehle mit einer Vielzahl von Optionen, die Sie kombinieren können.

> **Note**
>
> Wenn Sie noch nicht die standardmäßige EMR Amazon-Servicerolle und das EC2 Instanzprofil erstellt haben, erstellen `aws emr create-default-roles` Sie diese zunächst, bevor Sie den `create-cluster` Befehl verwenden.

Example Beispiel: Primärinstanz auf Abruf, On-Demand-Core mit Einzelinstanztyp, Standard VPC

```
aws emr create-cluster --release-label emr-5.3.1 --service-role EMR_DefaultRole \
  --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \
  --instance-fleets \

 InstanceFleetType=MASTER,TargetOnDemandCapacity=1,InstanceTypeConfigs=['{InstanceType=m5.xlarg
 \

 InstanceFleetType=CORE,TargetOnDemandCapacity=1,InstanceTypeConfigs=['{InstanceType=m5.xlarge}
```

Example Beispiel: Spot-Primär, Spot-Core mit Einzelinstanztyp, Standard VPC

```
aws emr create-cluster --release-label emr-5.3.1 --service-role EMR_DefaultRole \
  --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \
  --instance-fleets \
    InstanceFleetType=MASTER,TargetSpotCapacity=1,\
InstanceTypeConfigs=['{InstanceType=m5.xlarge,BidPrice=0.5}'] \
    InstanceFleetType=CORE,TargetSpotCapacity=1,\
InstanceTypeConfigs=['{InstanceType=m5.xlarge,BidPrice=0.5}']
```

Example Beispiel: Primär auf Abruf, gemischter Kern mit Einzelinstanztyp, einzelnes EC2 Subnetz

```
aws emr create-cluster --release-label emr-5.3.1 --service-role EMR_DefaultRole \
  --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole,SubnetIds=['subnet-ab12345c'] \
  --instance-fleets \
    InstanceFleetType=MASTER,TargetOnDemandCapacity=1,\
InstanceTypeConfigs=['{InstanceType=m5.xlarge}'] \
```

```
    InstanceFleetType=CORE,TargetOnDemandCapacity=2,TargetSpotCapacity=6,\
InstanceTypeConfigs=['{InstanceType=m5.xlarge,BidPrice=0.5,WeightedCapacity=2}']
```

Example Beispiel: Primär auf Abruf, Spot-Core mit mehreren gewichteten Instance-Typen, Timeout für Spot, Subnetzbereich EC2

```
aws emr create-cluster --release-label emr-5.3.1 --service-role EMR_DefaultRole \
  --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole,SubnetIds=['subnet-
ab12345c','subnet-de67890f'] \
  --instance-fleets \
    InstanceFleetType=MASTER,TargetOnDemandCapacity=1,\
InstanceTypeConfigs=['{InstanceType=m5.xlarge}'] \
    InstanceFleetType=CORE,TargetSpotCapacity=11,\
InstanceTypeConfigs=['{InstanceType=m5.xlarge,BidPrice=0.5,WeightedCapacity=3}',\
'{InstanceType=m4.2xlarge,BidPrice=0.9,WeightedCapacity=5}'],\
LaunchSpecifications={SpotSpecification='{TimeoutDurationMinutes=120,TimeoutAction=SWITCH_TO_ON
```

Example Beispiel: Primäre On-Demand-Instanz, gemischter Kern- und Task-Instance-Typ mit mehreren gewichteten Instance-Typen, Timeout für Core-Spot-Instances, Subnetzbereich EC2

```
aws emr create-cluster --release-label emr-5.3.1 --service-role EMR_DefaultRole \
  --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole,SubnetIds=['subnet-
ab12345c','subnet-de67890f'] \
  --instance-fleets \

 InstanceFleetType=MASTER,TargetOnDemandCapacity=1,InstanceTypeConfigs=['{InstanceType=m5.xlarg
 \
    InstanceFleetType=CORE,TargetOnDemandCapacity=8,TargetSpotCapacity=6,\
InstanceTypeConfigs=['{InstanceType=m5.xlarge,BidPrice=0.5,WeightedCapacity=3}',\
'{InstanceType=m4.2xlarge,BidPrice=0.9,WeightedCapacity=5}'],\
LaunchSpecifications={SpotSpecification='{TimeoutDurationMinutes=120,TimeoutAction=SWITCH_TO_ON
 \
    InstanceFleetType=TASK,TargetOnDemandCapacity=3,TargetSpotCapacity=3,\
InstanceTypeConfigs=['{InstanceType=m5.xlarge,BidPrice=0.5,WeightedCapacity=3}']
```

Example Beispiel: Spot Primary, kein Core oder Task, EBS Amazon-Konfiguration, Standard VPC

```
aws emr create-cluster --release-label Amazon EMR 5.3.1 --service-role EMR_DefaultRole
 \
  --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \
  --instance-fleets \
    InstanceFleetType=MASTER,TargetSpotCapacity=1,\
```

```
LaunchSpecifications={SpotSpecification='{TimeoutDurationMinutes=60,TimeoutAction=TERMINATE_CLU
\
InstanceTypeConfigs=['{InstanceType=m5.xlarge,BidPrice=0.5,\
EbsConfiguration={EbsOptimized=true,EbsBlockDeviceConfigs=[{VolumeSpecification={VolumeType=gp2
\
SizeIn GB=100}},{VolumeSpecification={VolumeType=io1,SizeInGB=100,Iop
 s=100},VolumesPerInstance=4}]}}']
```

Example Beispiel: Mehrere benutzerdefinierte Instance-TypenAMIs, primärer On-Demand-Instance-Typ, On-Demand-Core

```
aws emr create-cluster --release-label Amazon EMR 5.3.1 --service-role EMR_DefaultRole
 \
  --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \
  --instance-fleets \
    InstanceFleetType=MASTER,TargetOnDemandCapacity=1,\
InstanceTypeConfigs=['{InstanceType=m5.xlarge,CustomAmiId=ami-123456},
{InstanceType=m6g.xlarge, CustomAmiId=ami-234567}'] \
    InstanceFleetType=CORE,TargetOnDemandCapacity=1,\
InstanceTypeConfigs=['{InstanceType=m5.xlarge,CustomAmiId=ami-123456},
{InstanceType=m6g.xlarge, CustomAmiId=ami-234567}']
```

Example Beispiel: Fügen Sie einem laufenden Cluster mit mehreren Instanztypen und mehreren benutzerdefinierten Instanztypen einen Task-Knoten hinzu AMIs

```
aws emr add-instance-fleet --cluster-id j-123456 --release-label Amazon EMR 5.3.1 \
  --service-role EMR_DefaultRole \
  --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \
  --instance-fleet \
    InstanceFleetType=Task,TargetSpotCapacity=1,\
InstanceTypeConfigs=['{InstanceType=m5.xlarge,CustomAmiId=ami-123456}',\
'{InstanceType=m6g.xlarge,CustomAmiId=ami-234567}']
```

Example Beispiel: Verwenden Sie eine JSON Konfigurationsdatei

Sie können Instance-Flottenparameter in einer JSON Datei konfigurieren und dann auf die JSON Datei als einzigen Parameter für Instance-Flotten verweisen. Der folgende Befehl verweist beispielsweise auf eine JSON Konfigurationsdatei: *my-fleet-config.json*

```
aws emr create-cluster --release-label emr-5.30.0 --service-role EMR_DefaultRole \
--ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \
```

```
--instance-fleets file://my-fleet-config.json
```

Das Tool *my-fleet-config.json* file gibt Primär-, Kern- und Taskinstanzflotten an, wie im folgenden Beispiel gezeigt. Die Core-Instance-Flotte verwendet einen maximalen Spot-Preis (`BidPrice`) als Prozentsatz von On-Demand, während die Task- und Primärinstance-Flotten einen maximalen Spot-Preis (BidPriceAsPercentageofOnDemandPrice) als Zeichenfolge in verwenden. USD

```
[
    {
        "Name": "Masterfleet",
        "InstanceFleetType": "MASTER",
        "TargetSpotCapacity": 1,
        "LaunchSpecifications": {
            "SpotSpecification": {
                "TimeoutDurationMinutes": 120,
                "TimeoutAction": "SWITCH_TO_ON_DEMAND"
            }
        },
        "InstanceTypeConfigs": [
            {
                "InstanceType": "m5.xlarge",
                "BidPrice": "0.89"
            }
        ]
    },
    {
        "Name": "Corefleet",
        "InstanceFleetType": "CORE",
        "TargetSpotCapacity": 1,
        "TargetOnDemandCapacity": 1,
        "LaunchSpecifications": {
          "OnDemandSpecification": {
            "AllocationStrategy": "lowest-price",
            "CapacityReservationOptions":
            {
                "UsageStrategy": "use-capacity-reservations-first",
                "CapacityReservationResourceGroupArn": "String"
            }
        },
            "SpotSpecification": {
                "AllocationStrategy": "capacity-optimized",
                "TimeoutDurationMinutes": 120,
```

```
                "TimeoutAction": "TERMINATE_CLUSTER"
            }
        },
        "InstanceTypeConfigs": [
            {
                "InstanceType": "m5.xlarge",
                "BidPriceAsPercentageOfOnDemandPrice": 100
            }
        ]
    },
    {
        "Name": "Taskfleet",
        "InstanceFleetType": "TASK",
        "TargetSpotCapacity": 1,
        "LaunchSpecifications": {
          "OnDemandSpecification": {
            "AllocationStrategy": "lowest-price",
            "CapacityReservationOptions":
            {
                "CapacityReservationPreference": "none"
            }
        },
            "SpotSpecification": {
                "TimeoutDurationMinutes": 120,
                "TimeoutAction": "TERMINATE_CLUSTER"
            }
        },
        "InstanceTypeConfigs": [
            {
                "InstanceType": "m5.xlarge",
                "BidPrice": "0.89"
            }
        ]
    }
]
```

Zielkapazitäten für eine Instance-Flotte ändern

Verwenden Sie den Befehl `modify-instance-fleet`, um neue Zielkapazitäten für eine Instance-Flotte anzugeben. Sie müssen die Cluster-ID und die Instance-Flotten-ID angeben. Verwenden Sie den `list-instance-fleets` Befehl, um die Instance-Flotte IDs abzurufen.

```
aws emr modify-instance-fleet --cluster-id <cluster-id> \
```

```
  --instance-fleet \
    InstanceFleetId='<instance-fleet-id>',TargetOnDemandCapacity=1,TargetSpotCapacity=1
```

Eine Aufgaben-Instance-Flotte zu einem Cluster hinzufügen

Wenn ein Cluster nur über Primär- und Core-Instance-Flotten verfügt, können Sie den Befehl `add-instance-fleet` verwenden, um eine Aufgaben-Instance-Flotte hinzuzufügen. Sie können nur diesen Befehl verwenden, um Aufgaben-Instance-Flotten hinzuzufügen.

```
aws emr add-instance-fleet --cluster-id <cluster-id>
  --instance-fleet \
    InstanceFleetType=TASK,TargetSpotCapacity=1,\
LaunchSpecifications={SpotSpecification='{TimeoutDurationMinutes=20,TimeoutAction=TERMINATE_CLU
\
InstanceTypeConfigs=['{InstanceType=m5.xlarge,BidPrice=0.5}']
```

Konfigurationsdetails der Instance-Flotten in einem Cluster abrufen

Verwenden Sie den Befehl `list-instance-fleets`, um Konfigurationsdetails der Instance-Flotten in einem Cluster abzurufen. Der Befehl erfordert die Eingabe einer Cluster-ID. Das folgende Beispiel zeigt den Befehl und die Ausgabe für einen Cluster mit einer Primär-Aufgaben-Instance-Gruppe und einer Core-Aufgaben-Instance-Gruppe. Die vollständige Antwortsyntax finden Sie ListInstanceFleetsin der EMRAPIAmazon-Referenz.

```
list-instance-fleets --cluster-id <cluster-id>
```

```
{
    "InstanceFleets": [
        {
            "Status": {
                "Timeline": {
                    "ReadyDateTime": 1488759094.637,
                    "CreationDateTime": 1488758719.817
                },
                "State": "RUNNING",
                "StateChangeReason": {
                    "Message": ""
                }
            },
            "ProvisionedSpotCapacity": 6,
```

```
            "Name": "CORE",
            "InstanceFleetType": "CORE",
            "LaunchSpecifications": {
                "SpotSpecification": {
                    "TimeoutDurationMinutes": 60,
                    "TimeoutAction": "TERMINATE_CLUSTER"
                }
            },
            "ProvisionedOnDemandCapacity": 2,
            "InstanceTypeSpecifications": [
                {
                    "BidPrice": "0.5",
                    "InstanceType": "m5.xlarge",
                    "WeightedCapacity": 2
                }
            ],
            "Id": "if-1ABC2DEFGHIJ3"
        },
        {
            "Status": {
                "Timeline": {
                    "ReadyDateTime": 1488759058.598,
                    "CreationDateTime": 1488758719.811
                },
                "State": "RUNNING",
                "StateChangeReason": {
                    "Message": ""
                }
            },
            "ProvisionedSpotCapacity": 0,
            "Name": "MASTER",
            "InstanceFleetType": "MASTER",
            "ProvisionedOnDemandCapacity": 1,
            "InstanceTypeSpecifications": [
                {
                    "BidPriceAsPercentageOfOnDemandPrice": 100.0,
                    "InstanceType": "m5.xlarge",
                    "WeightedCapacity": 1
                }
            ],
           "Id": "if-2ABC4DEFGHIJ4"
        }
    ]
}
```

Kapazitätsreservierungen mit Instance-Flotten verwenden

Um On-Demand-Instance-Flotten mit Optionen für Kapazitätsreservierungen zu starten, fügen Sie zusätzliche Servicerollenberechtigungen hinzu, die für die Nutzung von Kapazitätsreservierungsoptionen erforderlich sind. Da Optionen zur Kapazitätsreservierung zusammen mit der On-Demand-Zuweisungsstrategie verwendet werden müssen, müssen Sie auch die für die Zuweisungsstrategie erforderlichen Berechtigungen in Ihre Servicerolle und verwaltete Richtlinie aufnehmen. Weitere Informationen finden Sie unter Zuweisungsstrategie-Berechtigungen.

Amazon EMR unterstützt sowohl offene als auch gezielte Kapazitätsreservierungen. Die folgenden Themen zeigen Konfigurationen von Instance-Flotten, die Sie zusammen mit der `RunJobFlow`-Aktion oder dem `create-cluster`-Befehl verwenden können, um Instance-Flotten mithilfe von On-Demand-Kapazitätsreservierungen zu starten.

Offene Kapazitätsreservierungen nach bestmöglichem Bemühen verwenden

Wenn die On-Demand-Instances des Clusters den in Ihrem Konto verfügbaren Attributen der offenen Kapazitätsreservierungen (Instance-Typ, Plattform, Tenancy und Availability Zone) entsprechen, werden die Kapazitätsreservierungen automatisch angewendet. Es kann jedoch nicht garantiert werden, dass Ihre Kapazitätsreservierungen genutzt werden. Für die Bereitstellung des Clusters EMR bewertet Amazon alle in der Startanfrage angegebenen Instance-Pools und verwendet den Pool mit dem niedrigsten Preis, der über ausreichende Kapazität verfügt, um alle angeforderten Core-Knoten zu starten. Verfügbare offene Kapazitätsreservierungen, die dem Instance-Pool entsprechen, werden automatisch angewendet. Wenn verfügbare offene Kapazitätsreservierungen nicht mit dem Instance-Pool übereinstimmen, bleiben sie ungenutzt.

Sobald die Core-Knoten bereitgestellt sind, wird die Availability Zone ausgewählt und repariert. Amazon EMR stellt Task-Nodes in der ausgewählten Availability Zone in Instance-Pools bereit, wobei mit den günstigsten zuerst begonnen wird, bis alle Task-Knoten bereitgestellt sind. Verfügbare offene Kapazitätsreservierungen, die den Instance-Pools entsprechen, werden automatisch angewendet.

Im Folgenden finden Sie Anwendungsfälle der EMR Amazon-Kapazitätszuweisungslogik für die Nutzung offener Kapazitätsreservierungen nach bestem Wissen.

Beispiel 1: Der Instance-Pool mit dem niedrigsten Preis in der Startanfrage verfügt über verfügbare offene Kapazitätsreservierungen

In diesem Fall EMR führt Amazon mit On-Demand-Instances Kapazität im Instance-Pool mit dem niedrigsten Preis ein. Ihre verfügbaren offenen Kapazitätsreservierungen in diesem Instance-Pool werden automatisch verwendet.

On-Demand-Strategie	Niedrigster Preis		
Angeforderte Kapazität	100		
Instance-Typ	c5.xlarge	m5.xlarge	r5.xlarge
Verfügbare offene Kapazitätsreservierung	150	100	100
Preis auf Abruf	$	$$	$$$
Bereitgestellte Instances	100	-	-
Offene Kapazitätsreservierung verwendet	100	-	-
Verfügbare offene Kapazitätsreservierung	50	100	100

Nachdem die Instance-Flotte gestartet wurde, können Sie `describe-capacity-reservations` ausführen, um zu sehen, wie viele ungenutzte Kapazitätsreservierungen verbleiben.

Beispiel 2: Für den Instance-Pool mit dem niedrigsten Preis in der Startanfrage sind keine offenen Kapazitätsreservierungen verfügbar

In diesem Fall EMR führt Amazon mit On-Demand-Instances Kapazität im Instance-Pool mit dem niedrigsten Preis ein. Ihre offenen Kapazitätsreservierungen bleiben jedoch ungenutzt.

On-Demand-Strategie	Niedrigster Preis

Angeforderte Kapazität	100		
Instance-Typ	c5.xlarge	m5.xlarge	r5.xlarge
Verfügbare offene Kapazitätsreservierung	-	-	100
Preis auf Abruf	$	$$	$$$
Bereitgestellte Instances	100	-	-
Offene Kapazitätsreservierung verwendet	-	-	-
Verfügbare offene Kapazitätsreservierung	-	-	100

Konfigurieren Sie Instance-Flotten so, dass offene Kapazitätsreservierungen nach bestem Wissen und Gewissen verwendet werden

Wenn Sie die `RunJobFlow`-Aktion verwenden, um einen auf Instance-Flotten basierenden Cluster zu erstellen, legen Sie für die On-Demand-Zuweisungsstrategie die Optionen `lowest-price` und `CapacityReservationPreference` für Kapazitätsreservierungen auf `open` fest. Wenn Sie dieses Feld leer lassen, EMR setzt Amazon alternativ die Kapazitätsreservierungspräferenz der On-Demand-Instance standardmäßig auf`open`.

```
"LaunchSpecifications":
    {"OnDemandSpecification": {
        "AllocationStrategy": "lowest-price",
        "CapacityReservationOptions":
         {
            "CapacityReservationPreference": "open"
         }
        }
```

```
    }
```

Sie können Amazon auch verwenden, EMR CLI um mithilfe von offenen Kapazitätsreservierungen einen auf Instance-Flotten basierenden Cluster zu erstellen.

```
aws emr create-cluster \
 --name 'open-ODCR-cluster' \
 --release-label emr-5.30.0 \
 --service-role EMR_DefaultRole \
 --ec2-attributes SubnetId=subnet-22XXXX01,InstanceProfile=EMR_EC2_DefaultRole \
 --instance-fleets
 InstanceFleetType=MASTER,TargetOnDemandCapacity=1,InstanceTypeConfigs=['{InstanceType=c4.xlarg
 \

 InstanceFleetType=CORE,TargetOnDemandCapacity=100,InstanceTypeConfigs=['{InstanceType=c5.xlarg
{InstanceType=m5.xlarge},{InstanceType=r5.xlarge}'],\
   LaunchSpecifications={OnDemandSpecification='{AllocationStrategy=lowest-
price,CapacityReservationOptions={CapacityReservationPreference=open}}'}
```

Wobei gilt,

- `open-ODCR-cluster` wird durch den Namen des Clusters ersetzt, der offenen Kapazitätsreservierungen verwendet.
- `subnet-22XXXX01` wird durch die Subnetz-ID ersetzt.

Zuerst offene Kapazitätsreservierungen verwenden

Sie können sich dafür entscheiden, bei der Bereitstellung eines Amazon-Clusters die Zuweisungsstrategie mit dem niedrigsten Preis außer Kraft zu setzen und zuerst verfügbare offene Kapazitätsreservierungen zu verwenden. EMR In diesem Fall EMR bewertet Amazon alle Instance-Pools mit Kapazitätsreservierungen, die in der Startanfrage angegeben wurden, und verwendet den Pool mit dem niedrigsten Preis, der über ausreichend Kapazität verfügt, um alle angeforderten Core-Knoten zu starten. Wenn keiner der Instance-Pools mit Kapazitätsreservierungen über ausreichend Kapazität für die angeforderten Kernknoten verfügt, EMR greift Amazon auf den im vorherigen Thema beschriebenen Best-Effort-Fall zurück. Das heißt, Amazon EMR bewertet alle in der Startanfrage angegebenen Instance-Pools neu und verwendet den Pool mit dem niedrigsten Preis, der über ausreichende Kapazität verfügt, um alle angeforderten Core-Knoten zu starten. Verfügbare offene Kapazitätsreservierungen, die dem Instance-Pool entsprechen, werden automatisch angewendet.

Wenn verfügbare offene Kapazitätsreservierungen nicht mit dem Instance-Pool übereinstimmen, bleiben sie ungenutzt.

Sobald die Core-Knoten bereitgestellt sind, wird die Availability Zone ausgewählt und repariert. Amazon EMR stellt Task-Nodes in der ausgewählten Availability Zone in Instance-Pools mit Kapazitätsreservierungen bereit, wobei mit den günstigsten zuerst begonnen wird, bis alle Task-Knoten bereitgestellt sind. Amazon EMR verwendet zuerst die verfügbaren offenen Kapazitätsreservierungen, die für jeden Instance-Pool in der ausgewählten Availability Zone verfügbar sind, und verwendet nur bei Bedarf die niedrigste Preisstrategie, um alle verbleibenden Task-Knoten bereitzustellen.

Im Folgenden finden Sie Anwendungsfälle der EMR Amazon-Kapazitätszuweisungslogik, bei der zuerst offene Kapazitätsreservierungen verwendet werden.

Beispiel 1: Der Instance-Pool mit verfügbaren offenen Kapazitätsreservierungen in der Startanfrage verfügt über ausreichend Kapazität für Core-Knoten

In diesem Fall EMR führt Amazon unabhängig vom Preis des Instance-Pools Kapazität im Instance-Pool mit verfügbaren offenen Kapazitätsreservierungen ein. Daher werden Ihre offenen Kapazitätsreservierungen wann immer möglich genutzt, bis alle Core-Knoten bereitgestellt sind.

On-Demand-Strategie	Niedrigster Preis		
Angeforderte Kapazität	100		
Nutzungsstrategie	use-capacity-reservations-first		
Instance-Typ	c5.xlarge	m5.xlarge	r5.xlarge
Verfügbare offene Kapazitätsreservierung	-	-	150
Preis auf Abruf	$	$$	$$$
Bereitgestellte Instances	-	-	100

Offene Kapazitätsreservierung verwendet	-	-	100
Verfügbare offene Kapazitätsreservierung	-	-	50

Beispiel 2: Der Instance-Pool mit verfügbaren Reservierungen für offene Kapazitäten in der Startanfrage verfügt nicht über genügend Kapazität für Core-Knoten

In diesem Fall EMR greift Amazon darauf zurück, Kernknoten mit einer Niedrigpreisstrategie auf den Markt zu bringen, wobei Kapazitätsreservierungen nach besten Kräften genutzt werden.

On-Demand-Strategie	Niedrigster Preis		
Angeforderte Kapazität	100		
Nutzungsstrategie	use-capacity-reservations-first		
Instance-Typ	c5.xlarge	m5.xlarge	r5.xlarge
Verfügbare offene Kapazitätsreservierung	10	50	50
Preis auf Abruf	$	$$	$$$
Bereitgestellte Instances	100	-	-
Offene Kapazitätsreservierung verwendet	10	-	-

Verfügbare Reservierungen für offene Kapazitäten	-	50	50

Nachdem die Instance-Flotte gestartet wurde, können Sie `describe-capacity-reservations` ausführen, um zu sehen, wie viele ungenutzte Kapazitätsreservierungen verbleiben.

Konfigurieren Sie Instance-Flotten so, dass sie zuerst offene Kapazitätsreservierungen verwenden

Wenn Sie die `RunJobFlow`-Aktion verwenden, um einen auf Instance-Flotten basierenden Cluster zu erstellen, legen Sie für die On-Demand-Zuweisungsstrategie die Optionen `lowest-price` und `UsageStrategy` für `CapacityReservationOptions` auf `use-capacity-reservations-first` fest.

```
"LaunchSpecifications":
    {"OnDemandSpecification": {
        "AllocationStrategy": "lowest-price",
        "CapacityReservationOptions":
         {
            "UsageStrategy": "use-capacity-reservations-first"
         }
       }
    }
```

Sie können Amazon auch verwenden, EMR CLI um einen auf Instance-Flotten basierenden Cluster zu erstellen, indem Sie zunächst Kapazitätsreservierungen verwenden.

```
aws emr create-cluster \
  --name 'use-CR-first-cluster' \
  --release-label emr-5.30.0 \
  --service-role EMR_DefaultRole \
  --ec2-attributes SubnetId=subnet-22XXXX01,InstanceProfile=EMR_EC2_DefaultRole \
  --instance-fleets \

 InstanceFleetType=MASTER,TargetOnDemandCapacity=1,InstanceTypeConfigs=['{InstanceType=c4.xlarg
 \

 InstanceFleetType=CORE,TargetOnDemandCapacity=100,InstanceTypeConfigs=['{InstanceType=c5.xlarg
{InstanceType=m5.xlarge},{InstanceType=r5.xlarge}'],\
```

```
LaunchSpecifications={OnDemandSpecification='{AllocationStrategy=lowest-
price,CapacityReservationOptions={UsageStrategy=use-capacity-reservations-first}}'}
```

Wobei gilt,

- `use-CR-first-cluster` wird durch den Namen des Clusters ersetzt, der offenen Kapazitätsreservierungen verwendet.
- `subnet-22XXXX01` wird durch die Subnetz-ID ersetzt.

Zuerst gezielte Kapazitätsreservierungen verwenden

Wenn Sie einen EMR Amazon-Cluster bereitstellen, können Sie sich dafür entscheiden, die Zuweisungsstrategie mit dem niedrigsten Preis außer Kraft zu setzen und zuerst die verfügbaren gezielten Kapazitätsreservierungen zu verwenden. In diesem Fall EMR bewertet Amazon alle Instance-Pools mit gezielten Kapazitätsreservierungen, die in der Startanfrage angegeben sind, und wählt den Pool mit dem niedrigsten Preis aus, der über ausreichend Kapazität verfügt, um alle angeforderten Core-Knoten zu starten. Wenn keiner der Instance-Pools mit gezielten Kapazitätsreservierungen über ausreichend Kapazität für Kernknoten verfügt, EMR greift Amazon auf den zuvor beschriebenen Best-Effort-Fall zurück. Das heißt, Amazon EMR bewertet alle in der Startanfrage angegebenen Instance-Pools neu und wählt den Pool mit dem niedrigsten Preis aus, der über ausreichende Kapazität verfügt, um alle angeforderten Core-Knoten zu starten. Verfügbare offene Kapazitätsreservierungen, die dem Instance-Pool entsprechen, werden automatisch übernommen. Gezielte Kapazitätsreservierungen bleiben jedoch ungenutzt.

Sobald die Core-Knoten bereitgestellt sind, wird die Availability Zone ausgewählt und repariert. Amazon EMR stellt Task-Knoten in Instance-Pools mit gezielten Kapazitätsreservierungen bereit, beginnend mit den günstigsten zuerst, in der ausgewählten Availability Zone, bis alle Task-Knoten bereitgestellt sind. Amazon EMR versucht zunächst, die verfügbaren gezielten Kapazitätsreservierungen zu verwenden, die für jeden Instance-Pool in der ausgewählten Availability Zone verfügbar sind. Nur bei Bedarf EMR verwendet Amazon dann die Strategie mit dem niedrigsten Preis, um alle verbleibenden Task-Knoten bereitzustellen.

Im Folgenden finden Sie Anwendungsfälle der EMR Amazon-Kapazitätszuweisungslogik, bei der zunächst gezielte Kapazitätsreservierungen verwendet werden.

Beispiel 1: Der Instance-Pool mit verfügbaren gezielten Kapazitätsreservierungen in der Startanfrage verfügt über ausreichend Kapazität für Core-Knoten

In diesem Fall EMR führt Amazon unabhängig vom Preis des Instance-Pools Kapazität im Instance-Pool mit verfügbaren gezielten Kapazitätsreservierungen ein. Daher werden Ihre gezielten Kapazitätsreservierungen wann immer möglich genutzt, bis alle Core-Knoten bereitgestellt sind.

On-Demand-Strategie	Niedrigster Preis		
Nutzungsstrategie	use-capacity-reservations-first		
Angeforderte Kapazität	100		
Instance-Typ	c5.xlarge	m5.xlarge	r5.xlarge
Verfügbare gezielte Kapazitätsreservierungen	-	-	150
Preis auf Abruf	$	$$	$$$
Bereitgestellte Instances	-	-	100
Gezielte Kapazitätsreservierung genutzt	-	-	100
Verfügbare gezielte Kapazitätsreservierungen	-	-	50

Example Beispiel 2: Der Instance-Pool mit verfügbaren gezielten Kapazitätsreservierungen in der Startanfrage verfügt nicht über ausreichende Kapazität für Core-Knoten

On-Demand-Strategie	Niedrigster Preis
Angeforderte Kapazität	100
Nutzungsstrategie	use-capacity-reservations-first

Instance-Typ	c5.xlarge	m5.xlarge	r5.xlarge
Verfügbare gezielte Kapazitätsreservierungen	10	50	50
Preis auf Abruf	$	$$	$$$
Bereitgestellte Instances	100	-	-
Gezielte Kapazitätsreservierungen verwendet	10	-	-
Verfügbare gezielte Kapazitätsreservierungen	-	50	50

Nachdem die Instance-Flotte gestartet wurde, können Sie `describe-capacity-reservations` ausführen, um zu sehen, wie viele ungenutzte Kapazitätsreservierungen verbleiben.

Instance-Flotten so konfigurieren, dass sie zuerst gezielte Kapazitätsreservierungen verwenden

Wenn Sie die `RunJobFlow`-Aktion verwenden, um einen auf Instance-Flotten basierenden Cluster zu erstellen, legen Sie für die On-Demand-Zuweisungsstrategie die Optionen `lowest-price` und `UsageStrategy` für `CapacityReservationOptions` auf `use-capacity-reservations-first` und `CapacityReservationResourceGroupArn` für `CapacityReservationOptions` auf `<your resource group ARN>` fest. Weitere Informationen finden Sie unter Arbeiten mit Kapazitätsreservierungen im EC2Amazon-Benutzerhandbuch.

```
"LaunchSpecifications":
    {"OnDemandSpecification": {
        "AllocationStrategy": "lowest-price",
        "CapacityReservationOptions":
         {
            "UsageStrategy": "use-capacity-reservations-first",
            "CapacityReservationResourceGroupArn": "arn:aws:resource-groups:sa-
east-1:123456789012:group/MyCRGroup"
         }
```

```
        }
    }
```

Wo `arn:aws:resource-groups:sa-east-1:123456789012:group/MyCRGroup` wird durch Ihre Ressourcengruppe ersetztARN.

Sie können Amazon auch verwenden, EMR CLI um mithilfe gezielter Kapazitätsreservierungen einen auf Instance-Flotten basierenden Cluster zu erstellen.

```
aws emr create-cluster \
  --name 'targeted-CR-cluster' \
  --release-label emr-5.30.0 \
  --service-role EMR_DefaultRole \
  --ec2-attributes SubnetId=subnet-22XXXX01,InstanceProfile=EMR_EC2_DefaultRole \
  --instance-fleets
 InstanceFleetType=MASTER,TargetOnDemandCapacity=1,InstanceTypeConfigs=['{InstanceType=c4.xlarg
 \
    InstanceFleetType=CORE,TargetOnDemandCapacity=100,\
InstanceTypeConfigs=['{InstanceType=c5.xlarge},{InstanceType=m5.xlarge},
{InstanceType=r5.xlarge}'],\
LaunchSpecifications={OnDemandSpecification='{AllocationStrategy=lowest-
price,CapacityReservationOptions={UsageStrategy=use-capacity-reservations-
first,CapacityReservationResourceGroupArn=arn:aws:resource-groups:sa-
east-1:123456789012:group/MyCRGroup}}'}
```

Wobei gilt,

- `targeted-CR-cluster` wird mithilfe von gezielten Kapazitätsreservierungen durch den Namen Ihres Clusters ersetzt.
- `subnet-22XXXX01` wird durch die Subnetz-ID ersetzt.
- `arn:aws:resource-groups:sa-east-1:123456789012:group/MyCRGroup`wird durch Ihre Ressourcengruppe ersetzt. ARN

Vermeiden Sie es, verfügbare offene Kapazitätsreservierungen zu verwenden

Example

Wenn Sie vermeiden möchten, dass Ihre offenen Kapazitätsreservierungen beim Start eines EMR Amazon-Clusters unerwartet in Anspruch genommen werden, legen Sie die On-Demand-Zuweisungsstrategie auf `lowest-price` und `CapacityReservationPreference`

für `CapacityReservationOptions` festnone. Andernfalls EMR setzt Amazon die Kapazitätsreservierungspräferenz der On-Demand-Instance standardmäßig auf `open` und versucht, verfügbare offene Kapazitätsreservierungen nach bestem Wissen zu verwenden.

```
"LaunchSpecifications":
    {"OnDemandSpecification": {
        "AllocationStrategy": "lowest-price",
        "CapacityReservationOptions":
         {
            "CapacityReservationPreference": "none"
         }
       }
    }
```

Sie können Amazon auch verwenden, EMR CLI um einen auf Instance-Flotten basierenden Cluster zu erstellen, ohne offene Kapazitätsreservierungen zu verwenden.

```
aws emr create-cluster \
  --name 'none-CR-cluster' \
  --release-label emr-5.30.0 \
  --service-role EMR_DefaultRole \
  --ec2-attributes SubnetId=subnet-22XXXX01,InstanceProfile=EMR_EC2_DefaultRole \
  --instance-fleets \

 InstanceFleetType=MASTER,TargetOnDemandCapacity=1,InstanceTypeConfigs=['{InstanceType=c4.xlarg
 \

 InstanceFleetType=CORE,TargetOnDemandCapacity=100,InstanceTypeConfigs=['{InstanceType=c5.xlarg
{InstanceType=m5.xlarge},{InstanceType=r5.xlarge}'],\
LaunchSpecifications={OnDemandSpecification='{AllocationStrategy=lowest-
price,CapacityReservationOptions={CapacityReservationPreference=none}}'}
```

Wobei gilt,

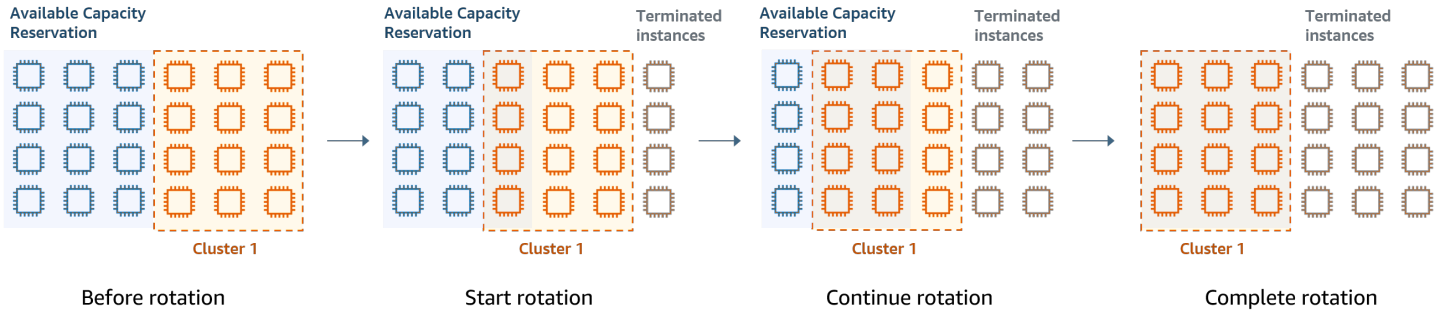
- `none-CR-cluster` wird durch den Namen Ihres Clusters ersetzt, der keine offenen Kapazitätsreservierungen verwendet.
- `subnet-22XXXX01` wird durch die Subnetz-ID ersetzt.

Szenarien für die Verwendung von Kapazitätsreservierungen

In den folgenden Szenarien können Sie von der Verwendung von Kapazitätsreservierungen profitieren.

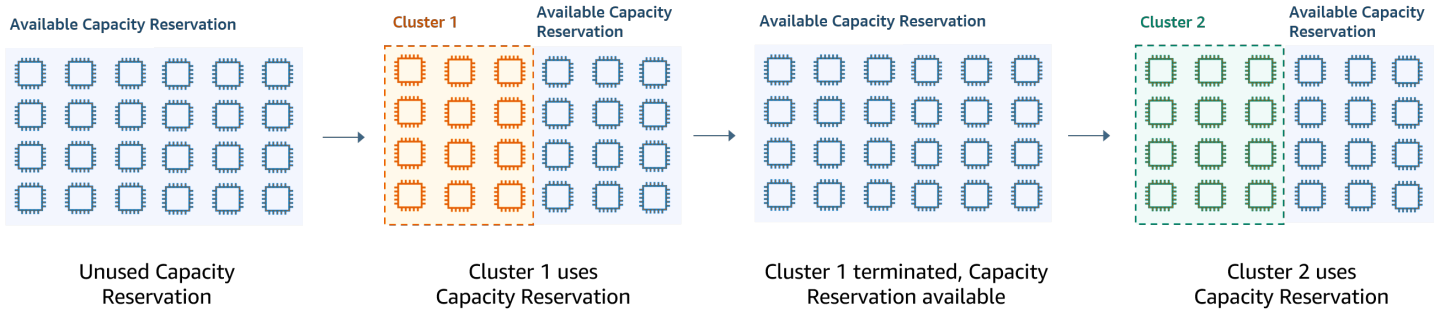
Szenario 1: Rotation eines Clusters mit langer Laufzeit mithilfe von Kapazitätsreservierungen

Wenn Sie einen Cluster mit langer Laufzeit rotieren, stellen Sie möglicherweise strenge Anforderungen an die Instance-Typen und Availability Zones für die neuen Instances, die Sie bereitstellen. Mit Kapazitätsreservierungen können Sie die Kapazitätssicherung verwenden, um die Cluster-Rotation ohne Unterbrechungen abzuschließen.



Szenario 2: Bereitstellung aufeinanderfolgender kurzlebiger Cluster mithilfe von Kapazitätsreservierungen

Sie können Kapazitätsreservierungen auch verwenden, um eine Gruppe aufeinanderfolgender, kurzlebiger Cluster für einzelne Workloads bereitzustellen, sodass, wenn Sie einen Cluster beenden, der nächste Cluster die Kapazitätsreservierungen nutzen kann. Sie können gezielte Kapazitätsreservierungen verwenden, um sicherzustellen, dass nur die vorgesehenen Cluster die Kapazitätsreservierungen nutzen.



Einheitliche Instance-Gruppen konfigurieren

Mit der Instance-Gruppenkonfiguration besteht jeder Knotentyp (Master-, Core- oder Aufgabenknoten) aus demselben Instance-Typ und derselben Kaufoption für Instances: On-Demand

oder Spot. Sie geben diese Einstellungen beim Erstellen einer Instance-Gruppe an. Sie können später nicht mehr geändert werden. Sie können Core- und Aufgaben-Instance-Gruppen jedoch Instances desselben Typs und derselben Kaufoption hinzufügen. Außerdem können Sie Instances entfernen.

Wenn die On-Demand-Instances des Clusters den in Ihrem Konto verfügbaren Attributen der offenen Kapazitätsreservierungen (Instance-Typ, Plattform, Tenancy und Availability Zone) entsprechen, werden die Kapazitätsreservierungen automatisch angewendet. Sie können offene Kapazitätsreservierungen für Primär-, Kern- und Aufgabenknoten verwenden. Sie können jedoch keine gezielten Kapazitätsreservierungen verwenden oder verhindern, dass Instances offene Kapazitätsreservierungen mit übereinstimmenden Attributen starten, wenn Sie Cluster mithilfe von Instance-Gruppen bereitstellen. Wenn Sie gezielte Kapazitätsreservierungen verwenden oder verhindern möchten, dass Instances aufgrund offener Kapazitätsreservierungen starten, verwenden Sie stattdessen Instance-Flotten. Weitere Informationen finden Sie unter Kapazitätsreservierungen mit Instance-Flotten verwenden.

Zum Hinzufügen verschiedener Instance-Typen nach dem Erstellen eines Clusters können Sie zusätzliche Aufgaben-Instance-Gruppen hinzufügen. Sie können verschiedene Instance-Typen und Kaufoptionen für jede Instance-Gruppe auswählen. Weitere Informationen finden Sie unter Clusterskalierung verwenden.

Beim Starten von Instances ist die Kapazitätsreservierungspräferenz der On-Demand-Instance standardmäßig auf open gesetzt, wodurch sie in jeder offenen Kapazitätsreservierung ausgeführt werden kann, die über passende Attribute (Instance-Typ, Plattform, Verfügbarkeitszone) verfügt. Weitere Informationen über On-Demand-Kapazitätsreservierungen finden Sie unter Kapazitätsreservierungen mit Instance-Flotten verwenden.

In diesem Abschnitt wird das Erstellen eines Clusters mit einheitlichen Instance-Gruppen beschrieben. Weitere Informationen zum Ändern einer vorhandenen Instance-Gruppe durch Hinzufügen oder Entfernen von Instances manuell oder automatisch mit Auto Scaling finden Sie unter Verwalten von Clustern.

Die Konsole zum Konfigurieren einheitlicher Instance-Gruppen verwenden

Console

So erstellen Sie einen Cluster mit Instance-Gruppen mithilfe der neuen Konsole

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.

2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und anschließend Create cluster aus.
3. Wählen Sie unter Clusterkonfiguration die Option Instance-Gruppen aus.
4. Unter Knotengruppen gibt es einen Abschnitt für jeden Knotengruppentyp. Aktivieren Sie für die Primärknotengruppe das Kontrollkästchen Mehrere Primärknoten verwenden, wenn Sie drei Primärknoten haben möchten. Aktivieren Sie das Kontrollkästchen Spot-Kaufoption verwenden, wenn Sie Spot-Kauf verwenden möchten.
5. Wählen Sie für die Primär- und Core-Knotengruppen die Option Instance-Typ hinzufügen und wählen Sie bis zu 5 Instance-Typen aus. Wählen Sie für die Aufgabengruppe Instance-Typ hinzufügen und wählen Sie bis zu fünfzehn Instance-Typen aus. Amazon EMR kann beim Start des Clusters eine beliebige Mischung dieser Instance-Typen bereitstellen.
6. Wählen Sie unter jedem Knotengruppentyp das Drop-Down-Menü Aktionen neben jeder Instance aus, um diese Einstellungen zu ändern:

   EBSVolumen hinzufügen

   Geben Sie EBS Volumes an, die an den Instance-Typ angehängt werden sollen, nachdem EMR Amazon ihn bereitgestellt hat.

   Den maximalen Spot-Preis bearbeiten

   Geben Sie für jeden Instance-Typ in einer Flotte einen maximalen Spot-Preis an. Sie können den Preis entweder als Prozentsatz des On-Demand-Preises oder als einen bestimmten Betrag in US-Dollar festlegen. Wenn der aktuelle Spot-Preis in einer Availability Zone unter Ihrem maximalen Spot-Preis liegt, stellt Amazon EMR Spot-Instances bereit. Sie zahlen den Spot-Preis, nicht unbedingt den maximalen Spot-Preis.
7. Erweitern Sie optional die Knotenkonfiguration, um eine JSON Konfiguration einzugeben oder JSON aus Amazon S3 zu laden.
8. Wählen Sie alle anderen Optionen aus, die für Ihren Cluster gelten.
9. Um Ihren Cluster jetzt zu starten, wählen Sie Cluster erstellen aus.

Verwenden Sie den AWS CLI , um einen Cluster mit einheitlichen Instanzgruppen zu erstellen

Um die Instance-Gruppenkonfiguration für einen Cluster mithilfe der AWS CLI anzugeben, verwenden Sie den Befehl `create-cluster` zusammen mit dem Parameter `--instance-groups`. Amazon EMR geht von der On-Demand-Instance-Option aus, sofern Sie das `BidPrice` Argument nicht für eine Instance-Gruppe angeben. Beispiele der Befehle `create-cluster`, mit denen einheitliche

Instance-Gruppen mit On-Demand-Instances gestartet werden, und eine Vielzahl von Cluster-Optionen sehen Sie, wenn Sie `aws emr create-cluster help` in der Befehlszeile eingeben oder den Abschnitt create-cluster in der AWS CLI -Befehlsreferenz lesen.

Sie können die verwenden AWS CLI , um einheitliche Instance-Gruppen in einem Cluster zu erstellen, die Spot-Instances verwenden. Der angebotene Spot-Preis hängt der von Availability Zone ab. Wenn Sie CLI oder verwendenAPI, können Sie die Availability Zone entweder mit dem `AvailabilityZone` Argument (wenn Sie ein EC2 -classic-Netzwerk verwenden) oder mit dem `SubnetID` Argument des `--ec2-attributes` Parameters angeben. Die ausgewählte Availability Zone oder das Subnetz gilt für den Cluster und wird daher für alle Instance-Gruppen verwendet. Wenn Sie nicht explizit eine Availability Zone oder ein Subnetz angeben, EMR wählt Amazon beim Start des Clusters die Availability Zone mit dem niedrigsten Spot-Preis aus.

Das folgende Beispiel zeigt einen Befehl `create-cluster`, mit dem Primär-, Core- und zwei Aufgaben-Instance-Gruppen erstellt werden, die alle Spot Instances verwenden. Ersetzen *myKey* mit dem Namen Ihres EC2 Amazon-Schlüsselpaars.

> **Note**
>
> Linux-Zeilenfortsetzungszeichen (\) sind aus Gründen der Lesbarkeit enthalten. Sie können entfernt oder in Linux-Befehlen verwendet werden. Entfernen Sie sie unter Windows oder ersetzen Sie sie durch ein Caret-Zeichen (^).

```
aws emr create-cluster --name "MySpotCluster" \
  --release-label emr-7.2.0 \
  --use-default-roles \
  --ec2-attributes KeyName=myKey \
  --instance-groups \
    InstanceGroupType=MASTER,InstanceType=m5.xlarge,InstanceCount=1,BidPrice=0.25 \
    InstanceGroupType=CORE,InstanceType=m5.xlarge,InstanceCount=2,BidPrice=0.03 \
    InstanceGroupType=TASK,InstanceType=m5.xlarge,InstanceCount=4,BidPrice=0.03 \
    InstanceGroupType=TASK,InstanceType=m5.xlarge,InstanceCount=2,BidPrice=0.04
```

Mithilfe von können Sie einheitliche Instance-Gruppen-Cluster erstellen, die AMI für jeden Instance-Typ in der Instance-Gruppe einen eindeutigen benutzerdefinierten Wert angeben. CLI Auf diese Weise können Sie verschiedene Instance-Architekturen in derselben Instance-Gruppe verwenden. Jeder Instanztyp muss einen benutzerdefinierten Instanztyp AMI mit einer passenden Architektur verwenden. Sie würden beispielsweise einen m5.xlarge-Instance-Typ mit einer benutzerdefinierten

x86_64-Architektur und einen m6g.xlarge-Instance-Typ mit einer entsprechenden benutzerdefinierten AMI () -Architektur konfigurieren. `AWS  AARCH64` ARM AMI

Das folgende Beispiel zeigt einen einheitlichen Instanzgruppen-Cluster, der aus zwei Instanztypen mit jeweils eigenen benutzerdefinierten Instanztypen erstellt wurde. AMI Beachten Sie, dass die benutzerdefinierten AMIs Werte nur auf Instanztypebene angegeben werden, nicht auf Clusterebene. Dadurch sollen Konflikte zwischen dem Instanztyp AMIs und einem AMI auf Clusterebene vermieden werden, die dazu führen würden, dass der Clusterstart fehlschlägt.

```
aws emr create-cluster
  --release-label emr-5.30.0 \
  --service-role EMR_DefaultRole \
  --ec2-attributes SubnetId=subnet-22XXXX01,InstanceProfile=EMR_EC2_DefaultRole \
  --instance-groups \

 InstanceGroupType=MASTER,InstanceType=m5.xlarge,InstanceCount=1,CustomAmiId=ami-123456
 \

 InstanceGroupType=CORE,InstanceType=m6g.xlarge,InstanceCount=1,CustomAmiId=ami-234567
```

Sie können einer InstanzgruppeAMIs, die Sie einem laufenden Cluster hinzufügen, mehrere benutzerdefinierte Instanzen hinzufügen. Das `CustomAmiId`-Argument kann zusammen mit dem `add-instance-groups`-Befehl verwendet werden, wie im folgenden Beispiel gezeigt.

```
aws emr add-instance-groups --cluster-id j-123456 \
  --instance-groups \

 InstanceGroupType=Task,InstanceType=m5.xlarge,InstanceCount=1,CustomAmiId=ami-123456
```

Verwenden Sie JavaSDK, um eine Instanzgruppe zu erstellen

Instanziieren Sie ein Objekt `InstanceGroupConfig`, das die Konfiguration einer Instance-Gruppe für einen Cluster angibt. Um Spot-Instances zu verwenden, legen Sie die Eigenschaften `withBidPrice` und `withMarket` für das Objekt `InstanceGroupConfig` fest. Der folgende Code zeigt, wie Primär-, Core- und Aufgaben-Instance-Gruppen definiert werden, die Spot Instances ausführen.

```
InstanceGroupConfig instanceGroupConfigMaster = new InstanceGroupConfig()
 .withInstanceCount(1)
 .withInstanceRole("MASTER")
```

```
  .withInstanceType("m4.large")
  .withMarket("SPOT")
  .withBidPrice("0.25");

InstanceGroupConfig instanceGroupConfigCore = new InstanceGroupConfig()
  .withInstanceCount(4)
  .withInstanceRole("CORE")
  .withInstanceType("m4.large")
  .withMarket("SPOT")
  .withBidPrice("0.03");

InstanceGroupConfig instanceGroupConfigTask = new InstanceGroupConfig()
  .withInstanceCount(2)
  .withInstanceRole("TASK")
  .withInstanceType("m4.large")
  .withMarket("SPOT")
  .withBidPrice("0.10");
```

Bewährte Methoden für Instance- und Availability Zone-Flexibilität

Jeder AWS-Region hat mehrere isolierte Standorte, die als Availability Zones bezeichnet werden. Beim Starten einer Instance können Sie optional eine Availability Zone (AZ) oder AWS-Region in der Region angeben, die Sie verwenden. Die Flexibilität der Availability Zone ist die Verteilung von Instanzen auf mehrereAZs. Wenn eine Instance ausfällt, können Sie Ihre Anwendung so gestalten, dass eine Instance in einer anderen AZ Anfragen verarbeiten kann. Weitere Informationen zu Availability Zones finden Sie in der Dokumentation zu Regionen und Zonen im EC2Amazon-Benutzerhandbuch.

Instance-Fexibilität ist die Verwendung mehrerer Instance-Typen zur Erfüllung der Kapazitätsanforderungen. Wenn Sie Flexibilität bei Instances zum Ausdruck bringen, können Sie die Gesamtkapazität für alle Instance-Größen, Familien und Generationen nutzen. Im Vergleich zu einem Cluster, der einen einzigen Instance-Typ verwendet, verbessert sich die Wahrscheinlichkeit, die erforderliche Menge an Rechenkapazität zu finden und zuzuweisen.

Die Flexibilität von Instances und Availability Zones reduziert Fehler bei unzureichender Kapazität (ICE) und Spot-Unterbrechungen im Vergleich zu einem Cluster mit einem einzigen Instance-Typ oder AZ. Verwenden Sie die hier beschriebenen bewährten Methoden, um zu bestimmen, welche Instances Sie diversifizieren sollten, nachdem Sie die ursprüngliche Instance-Familie und Größe kennen. Dieser Ansatz maximiert die Verfügbarkeit von EC2 Amazon-Kapazitätspools bei minimaler Leistung und Kostenabweichung.

Flexibilität in Bezug auf Availability Zones

Wir empfehlen Ihnen, alle Availability Zones für die Verwendung in Ihrer Virtual Private Cloud (VPC) zu konfigurieren und sie für Ihren EMR Cluster auszuwählen. Cluster dürfen nur in einer Availability Zone existieren, aber mit EMR Amazon-Instance-Flotten können Sie mehrere Subnetze für verschiedene Availability Zones auswählen. Wenn Amazon den Cluster EMR startet, durchsucht es diese Subnetze nach den von Ihnen angegebenen Instances und Kaufoptionen. Wenn Sie einen EMR Cluster für mehrere Subnetze bereitstellen, kann Ihr Cluster im Vergleich zu Clustern in einem einzelnen Subnetz auf einen größeren EC2 Amazon-Kapazitätspool zugreifen.

Wenn Sie eine bestimmte Anzahl von Availability Zones für die Verwendung in Ihrer Virtual Private Cloud (VPC) für Ihren EMR Cluster priorisieren müssen, können Sie die Spot Placement Score-Funktion bei Amazon EC2 nutzen. Mit der Bewertung der Spot-Platzierung geben Sie die Rechenanforderungen für Ihre Spot-Instances an und geben dann die zehn besten AWS-Regionen oder Availability Zones EC2 zurück, die auf einer Skala von 1 bis 10 bewertet wurden. Eine Punktzahl von 10 zeigt an, dass Ihre Spot-Anforderung sehr wahrscheinlich erfolgreich sein wird. Eine Punktzahl von 1 zeigt an, dass Ihre Spot-Anforderung sehr wahrscheinlich nicht erfolgreich sein wird. Weitere Informationen zur Verwendung von Spot Placement Scoring finden Sie unter Spot Placement Score im EC2Amazon-Benutzerhandbuch.

Flexibel sein bei Instance-Typen

Instance-Fexibilität ist die Verwendung mehrerer Instance-Typen zur Erfüllung der Kapazitätsanforderungen. Die Instance-Flexibilität kommt sowohl der Nutzung von Amazon EC2 Spot als auch der On-Demand-Instance zugute. Dank der Instance-Flexibilität von Spot-Instances kann Amazon EC2 Instances mithilfe von Echtzeit-Kapazitätsdaten aus tieferen Kapazitätspools starten. Außerdem wird vorhergesagt, welche Instances am verfügbarsten sind. Dies bietet weniger Unterbrechungen und kann die Gesamtkosten eines Workloads reduzieren. Mit On-Demand-Instances reduziert die Instance-Flexibilität Fehler bei unzureichender Kapazität (ICE), wenn die Gesamtkapazität über eine größere Anzahl von Instance-Pools bereitgestellt wird.

Für Instanzgruppen-Cluster können Sie bis zu 50 EC2 Instance-Typen angeben. Für Instanzflotten mit Zuweisungsstrategie können Sie bis zu 30 EC2 Instanztypen für jede Primär-, Kern- und Taskknotengruppe angeben. Eine breitere Palette von Instances verbessert die Vorteile der Instance-Flexibilität.

Ausdrücken der Instance-Flexibilität

Beachten Sie die folgenden bewährten Methoden, um die Instance-Flexibilität für Ihre Anwendung zum Ausdruck zu bringen.

Themen

Die Instance-Familie und -größe ermitteln

Amazon EMR unterstützt mehrere Instance-Typen für unterschiedliche Anwendungsfälle. Diese Instance-Typen sind in der Unterstützte Instance-Typen-Dokumentation aufgeführt. Jeder Instance-Typ gehört zu einer Instance-Familie, die beschreibt, für welche Anwendung der Typ optimiert ist.

Bei neuen Workloads sollten Sie einen Vergleich mit Instance-Typen aus der Allzweckfamilie durchführen, z. B. `m5` oder `c5`. Überwachen Sie anschließend das Betriebssystem und die YARN Metriken von Ganglia und ermitteln Amazon CloudWatch Sie Systemengpässe bei Spitzenlast. Zu den Engpässen gehören Speicher CPU -, Speicher- und I/O-Operationen. Nachdem Sie die Engpässe identifiziert haben, wählen Sie rechenoptimiert, arbeitsspeicheroptimiert, speicheroptimiert oder eine andere geeignete Instance-Familie für Ihre Instance-Typen. Weitere Informationen finden Sie auf der Seite Ermitteln Sie die richtige Infrastruktur für Ihre Spark-Workloads im EMR Amazon-Best-Practices-Leitfaden unter GitHub.

Identifizieren Sie als Nächstes den kleinsten YARN Container oder Spark-Executor, den Ihre Anwendung benötigt. Dies ist die kleinste Instance-Größe, die zum Container passt, und die minimale Instance-Größe für den Cluster. Verwenden Sie diese Metrik, um Instances zu ermitteln, mit denen Sie weiter diversifizieren können. Eine kleinere Instance ermöglicht mehr Instance-Flexibilität.

Für maximale Instance-Flexibilität sollten Sie so viele Instances wie möglich nutzen. Wir empfehlen Ihnen, mit Instances zu diversifizieren, die ähnliche Hardwarespezifikationen haben. Dadurch wird der Zugriff auf EC2 Kapazitätspools bei minimalen Kosten- und Leistungsschwankungen maximiert. Diversifizieren Sie zwischen verschiedenen Größen. Priorisieren Sie dazu zuerst AWS Graviton und frühere Generationen. Als allgemeine Regel gilt: Versuchen Sie, für jeden Workload flexibel über mindestens 15 Instance-Typen hinweg zu sein. Wir empfehlen, mit allgemeinen, rechenoptimierten oder arbeitsspeicheroptimierten Instances zu beginnen. Diese Instance-Typen bieten die größte Flexibilität.

Zusätzliche Instances hinzufügen

Fügen Sie für eine maximale Vielfalt zusätzliche Instance-Typen hinzu. Priorisieren Sie zuerst die Instance-Größe, Graviton und Generierungsflexibilität. Dies ermöglicht den Zugriff auf zusätzliche EC2 Kapazitätspools mit ähnlichen Kosten- und Leistungsprofilen. Wenn Sie aufgrund von

Unterbrechungen ICE oder punktuellen Unterbrechungen mehr Flexibilität benötigen, sollten Sie die Flexibilität von Varianten und Produktreihen in Betracht ziehen. Jeder Ansatz hat Kompromisse, die von Ihrem Anwendungsfall und Ihren Anforderungen abhängen.

- Größenflexibilität – Diversifizieren Sie zunächst mit Instances unterschiedlicher Größe innerhalb derselben Produktfamilie. Instances innerhalb derselben Familie bieten dieselben Kosten und dieselbe Leistung, können aber auf jedem Host eine unterschiedliche Anzahl von Containern starten. Wenn die Mindestgröße des Executors, die Sie benötigen, 2 V CPU und 8 GB Arbeitsspeicher beträgt, beträgt die Mindestgröße der Instanz. `m5.xlarge` Geben Sie aus Gründen der Größenflexibilität `m5.xlarge`, `m5.2xlarge`, `m5.4xlarge`, `m5.8xlarge`, `m5.12xlarge`, `m5.16xlarge` und `m5.24xlarge` an.
- Graviton-Flexibilität – Neben der Größe können Sie mit Graviton-Instances auch eine größere Vielfalt an Optionen erzielen. Graviton-Instances werden von AWS Graviton2-Prozessoren angetrieben, die das beste Preis-Leistungs-Verhältnis für Cloud-Workloads bei Amazon bieten. EC2 Mit der minimalen Instance-Größe von `m5.xlarge` können Sie beispielsweise `m6g.xlarge`, `m6g.2xlarge`, `m6g.4xlarge`, `m6g.8xlarge` und `m6g.16xlarge` für die Graviton-Flexibilität einschließen.
- Flexibilität bei der Generierung – Ähnlich wie Graviton und Größenflexibilität haben auch Instances der Familien früherer Generationen dieselben Hardwarespezifikationen. Dies führt zu einem ähnlichen Kosten- und Leistungsprofil mit einer Erhöhung des insgesamt zugänglichen EC2 Amazon-Pools. Für Flexibilität bei der Generierung schließen Sie `m4.xlarge`, `m4.2xlarge`, `m4.10xlarge` und `m4.16xlarge` ein.
- Familien- und Variantenflexibilität
  - Kapazität – Um die Kapazität zu optimieren, empfehlen wir Instance-Flexibilität für alle Instance-Familien. Gängige Instances aus verschiedenen Instance-Familien verfügen über tiefere Instance-Pools, die bei der Erfüllung der Kapazitätsanforderungen helfen können. Instances aus verschiedenen Familien haben jedoch unterschiedliche Verhältnisse zwischen V CPU und Speicher. Dies führt zu einer Unterauslastung, wenn der erwartete Anwendungscontainer für eine andere Instance dimensioniert ist. Schließen Sie beispielsweise mit `m5.xlarge` für Datenverarbeitung optimierte Instances wie `c5` oder arbeitsspeicheroptimierte Instances wie `r5` ein, um die Flexibilität der Instance-Familie zu gewährleisten.
  - Kosten – Zur Kostenoptimierung empfehlen wir die variantenübergreifende Instance-Flexibilität. Diese Instanzen haben dasselbe Speicher- und CPU V-Verhältnis wie die ursprüngliche Instanz. Der Nachteil bei der Variantenflexibilität besteht darin, dass diese Instances kleinere Kapazitätspools haben, was zu begrenzter zusätzlicher Kapazität oder höheren Spot-Unterbrechungen führen kann. Fügen Sie `m5.xlarge` beispielsweise AMD basierte Instances

(m5a), SSD based instances () oder netzwerkoptimierte Instances (m5d) hinzu, um die Flexibilität von Instanzvarianten zu gewährleisten. m5n

Bewährte Methoden für die Konfiguration des Clusters

Verwenden Sie die Anleitungen in diesem Abschnitt, um die Instance-Typen, Kaufoptionen und die bereitzustellende Speichermenge für jeden Knotentyp in einem EMR Cluster zu ermitteln.

Welchen Instance-Typ sollten Sie verwenden?

Es gibt mehrere Möglichkeiten, EC2 Amazon-Instances zu einem Cluster hinzuzufügen. Welche Methode Sie wählen sollten, hängt davon ab, ob Sie die Instance-Gruppen-Konfiguration oder die Instance-Flotten-Konfiguration für den Cluster verwenden.

- Instance-Gruppen
  - Fügen Sie vorhandenen Core- und Task-Instance-Gruppen manuell Instances desselben Typs hinzu.
  - Fügen Sie manuell eine Task-Instance-Gruppe hinzu, die einen anderen Instance-Typ verwenden kann.
  - Richten Sie die automatische Skalierung in Amazon EMR für eine Instance-Gruppe ein und fügen Sie Instances automatisch hinzu und entfernen Sie sie basierend auf dem Wert einer von Ihnen angegebenen CloudWatch Amazon-Metrik. Weitere Informationen finden Sie unter Clusterskalierung verwenden.
- Instance-Flotten
  - Fügen Sie eine einzelne Task-Instance-Flotte hinzu.
  - Ändern Sie die Zielkapazität für On-Demand- und Spot-Instances für vorhandene Core- und Task-Instance-Flotten. Weitere Informationen finden Sie unter Instance-Flotten konfigurieren.

Eine Möglichkeit zum Planen der Instances Ihres Clusters ist die Ausführung eines Test-Clusters mit einem repräsentativen Beispielsatz von Daten und die Überwachung der Auslastung der Knoten im Cluster. Weitere Informationen finden Sie unter Einen Cluster anzeigen und überwachen. Eine andere Möglichkeit besteht in der Berechnung der Kapazität der Instances, die Sie erwägen, und im Vergleichen dieses Werts mit der Größe Ihrer Daten.

Im Allgemeinen benötigt der primäre Knotentyp, der Aufgaben zuweist, keine EC2 Instance mit viel Rechenleistung; EC2 Amazon-Instances für den Core-Knotentyp, die Aufgaben verarbeiten und Daten speichernHDFS, benötigen sowohl Rechenleistung als auch Speicherkapazität;

EC2 Amazon-Instances für den Task-Knotentyp, die keine Daten speichern, benötigen nur Rechenleistung. Richtlinien zu verfügbaren EC2 Amazon-Instances und deren Konfiguration finden Sie unterEC2Amazon-Instances konfigurieren.

Die folgenden Richtlinien gelten für die meisten EMR Amazon-Cluster.

- Es gibt ein CPU V-Limit für die Gesamtzahl der EC2 On-Demand-Amazon-Instances, die Sie auf einem AWS Konto pro ausführen AWS-Region. Weitere Informationen zum CPU V-Limit und dazu, wie Sie eine Limiterhöhung für Ihr Konto beantragen können, finden Sie unter On-Demand-Instances im EC2Amazon-Benutzerhandbuch für Linux-Instances.
- Der Primärknoten stellt keine großen Datenverarbeitungsanforderungen. Für Cluster mit einer großen Anzahl von Knoten oder für Cluster mit Anwendungen, die speziell auf dem primären Knoten (JupyterHub, Hue usw.) bereitgestellt werden, ist möglicherweise ein größerer primärer Knoten erforderlich, der zur Verbesserung der Cluster-Leistung beitragen kann. Erwägen Sie beispielsweise, eine m5.xlarge-Instance für kleine Cluster (50 oder weniger Knoten) zu verwenden und für größere Cluster auf einen größeren Instance-Typ umzusteigen.
- Die benötigte Rechenleistung der Core- und Aufgabenknoten hängt von der Art der Verarbeitung ab, die Ihre Anwendung durchführt. Viele Jobs können auf Allzweck-Instance-Typen ausgeführt werden, die eine ausgewogene Leistung in Bezug auf Festplattenspeicher und Eingabe/Ausgabe bieten. CPU Rechenintensive Cluster können von der Ausführung auf CPU High-Instances profitieren, die proportional mehr als haben. CPU RAM Datenbank- und Arbeitsspeicher-Caching-Anwendungen können von der Ausführung auf High-Memory-Instances profitieren. Netzwerkintensive und CPU intensive Anwendungen wie Parsing und maschinelles Lernen können von der Ausführung auf Cluster-Recheninstanzen profitieren, die proportional hohe Ressourcen und eine höhere Netzwerkleistung bereitstellen. NLP CPU
- Wenn einzelne Phasen Ihres Clusters unterschiedliche Kapazitätserfordernisse haben, können Sie mit einer geringen Anzahl von Core-Knoten beginnen und die Anzahl von Aufgabenknoten den wechselnden Anforderungen der Auftragsverlaufskapazität entsprechend erhöhen oder verringern.
- Die Menge der Daten, die Sie verarbeiten können, hängt von der Kapazität Ihrer Core-Knoten und der Datenmenge als Eingabe, während der Verarbeitung, und als Ausgabe ab. Die Eingabe-, intermediären und Ausgabedatensätze befinden sich während der Verarbeitung alle auf dem Cluster.

Wann sollten Sie Spot Instances verwenden?

Wenn Sie einen Cluster in Amazon starten, können Sie wählenEMR, ob Sie Primär-, Kern- oder Task-Instances auf Spot-Instances starten möchten. Da jeder Typ von Instance-Gruppe eine andere

Rolle im Cluster hat, hat das Starten der einzelnen Knotentypen auf Spot-Instances bestimmte Auswirkungen. Sie können eine Instance-Kaufoption nicht ändern, während der Cluster ausgeführt wird. Um On-Demand-Instances in Spot Instances oder umgekehrt zu ändern, müssen Sie im Fall von Primär- und Core-Knoten den Cluster beenden und einen neuen Cluster starten. Im Fall von Aufgabenknoten können Sie eine neue Aufgaben-Instance-Gruppe oder -Flotte starten und die alte entfernen.

Themen

EMRAmazon-Einstellungen zur Vermeidung von Auftragsausfällen aufgrund der Kündigung der Spot-Instance des Task-Knotens

Da Spot-Instances häufig zum Ausführen von Task-Knoten verwendet werden, EMR verfügt Amazon über Standardfunktionen für die Planung von YARN Jobs, sodass laufende Jobs nicht fehlschlagen, wenn Task-Knoten, die auf Spot-Instances ausgeführt werden, beendet werden. Amazon ermöglicht EMR dies, indem es die Ausführung von Anwendungsmasterprozessen nur auf Kernknoten zulässt. Der Anwendungsmasterprozess steuert die Ausführung von Aufträgen und muss während der gesamten Laufzeit des Auftrags aktiv bleiben.

EMRAmazon-Version 5.19.0 und höher verwendet die integrierte YARNNode Labels-Funktion, um dies zu erreichen. (Frühere Versionen verwendeten einen Code-Patch). Eigenschaften in den Klassifizierungen `yarn-site` und in der `capacity-scheduler` Konfiguration sind standardmäßig so konfiguriert, dass der YARN Capacity-Scheduler und der Fair-Scheduler die Vorteile von Node-Labels nutzen. Amazon kennzeichnet Kernknoten EMR automatisch mit dem `CORE` Label und legt Eigenschaften fest, sodass Anwendungsmaster nur für Knoten mit dem CORE Label geplant werden. Durch manuelles Ändern verwandter Eigenschaften in den Konfigurationsklassifizierungen von Yarn-Site und Capacity-Scheduler oder direkt in den zugehörigen XML Dateien könnte diese Funktion beeinträchtigt oder verändert werden.

Amazon EMR konfiguriert standardmäßig die folgenden Eigenschaften und Werte. Seien Sie vorsichtig, wenn Sie diese Eigenschaften konfigurieren.

> **Note**
>
> Ab der Amazon EMR 6.x-Release-Serie ist die Funktion YARN Node Labels standardmäßig deaktiviert. Die Anwendungs-Primär-Prozesse können standardmäßig sowohl auf Core- als auch auf Aufgabenknoten ausgeführt werden. Sie können die Funktion „YARNNode Labels“ aktivieren, indem Sie die folgenden Eigenschaften konfigurieren:
>
> - `yarn.node-labels.enabled: true`
> - `yarn.node-labels.am.default-node-label-expression: 'CORE'`

- yarn-site (yarn-site.xml) auf allen Knoten
  - `yarn.node-labels.enabled: true`
  - `yarn.node-labels.am.default-node-label-expression: 'CORE'`
  - `yarn.node-labels.fs-store.root-dir: '/apps/yarn/nodelabels'`
  - `yarn.node-labels.configuration-type: 'distributed'`
- yarn-site (yarn-site.xml) auf Primär- und Core-Knoten
  - `yarn.nodemanager.node-labels.provider: 'config'`
  - `yarn.nodemanager.node-labels.provider.configured-node-partition: 'CORE'`
- capacity-scheduler (capacity-scheduler.xml) auf allen Knoten
  - `yarn.scheduler.capacity.root.accessible-node-labels: '*'`
  - `yarn.scheduler.capacity.root.accessible-node-labels.CORE.capacity: 100`
  - `yarn.scheduler.capacity.root.default.accessible-node-labels: '*'`
  - `yarn.scheduler.capacity.root.default.accessible-node-labels.CORE.capacity: 100`

Primärknoten auf einer Spot Instance

Der Primärknoten kontrolliert und steuert den Cluster. Wenn er beendet wird, wird der Cluster beendet. Daher sollten Sie den Primärknoten nur als Spot Instance starten, wenn Sie einen Cluster ausführen, dessen plötzliche Beendigung akzeptabel ist. Dies kann der Fall sein, wenn Sie eine neue Anwendung testen, wenn Sie einen Cluster vorliegen haben, der Daten in regelmäßigen Abständen in einem externen Speicher wie Amazon S3 ablegt oder wenn Sie einen Cluster ausführen, bei dem die Kosten eine wichtigere Rollen spielen als der Abschluss des Clusters.

Wenn Sie die Primär-Instance-Gruppe als Spot Instance starten, wird der Cluster erst gestartet, wenn die Spot-Instance-Anforderung erfüllt ist. Diese Tatsache muss bei der Auswahl des maximalen Spot-Preises berücksichtigt werden.

Sie können einen Spot-Instance-Primärknoten nur beim Starten des Clusters hinzufügen. Primärknoten können einem aktuell ausgeführten Cluster weder hinzugefügt noch daraus entfernt werden.

Normalerweise führen Sie den Primärknoten nur als Spot Instance aus, wenn Sie den gesamten Cluster (alle Instance-Gruppen) als Spot Instances ausführen.

Core-Knoten auf Spot Instances

Kernknoten verarbeiten Daten und speichern Informationen mithilfe vonHDFS. Das Beenden einer Core-Instance birgt das Risiko eines Datenverlusts. Aus diesem Grund sollten Sie Kernknoten nur dann auf Spot-Instances ausführen, wenn ein teilweiser HDFS Datenverlust tolerierbar ist.

Wenn Sie die Core-Instance-Gruppe als Spot-Instances starten, EMR wartet Amazon, bis alle angeforderten Core-Instances bereitgestellt werden können, bevor die Instance-Gruppe gestartet wird. Mit anderen Worten, wenn Sie sechs EC2 Amazon-Instances anfordern und nur fünf zu oder unter Ihrem maximalen Spot-Preis verfügbar sind, wird die Instance-Gruppe nicht gestartet. Amazon wartet EMR weiterhin, bis alle sechs EC2 Amazon-Instances verfügbar sind oder bis Sie den Cluster beenden. Sie können die Anzahl der Spot-Instances in einer Core-Instance-Gruppe ändern, um einem ausgeführten Cluster Kapazitäten hinzuzufügen. Weitere Informationen zum Arbeiten mit Instance-Gruppen und zur Art, wie Spot-Instances mit Instance-Flotten funktionieren, finden Sie unter the section called “Instance-Flotten oder Instance-Gruppen konfigurieren”.

Aufgabenknoten auf Spot Instances

Die Taskknoten verarbeiten Daten, speichern aber keine persistenten DatenHDFS. Wenn sie beendet werden da der Spot-Preis über Ihren maximalen Spot-Preis geklettert ist, gehen keine Daten verloren und die Auswirkung auf Ihrem Cluster ist minimal.

Wenn Sie eine oder mehrere Task-Instance-Gruppen als Spot-Instances starten, stellt EMR Amazon so viele Task-Knoten wie möglich bereit, wobei Ihr maximaler Spot-Preis verwendet wird. Das heißt, wenn Sie eine Task-Instance-Gruppe mit sechs Knoten anfordern und nur fünf Spot-Instances zu oder unter Ihrem maximalen Spot-Preis verfügbar sind, EMR startet Amazon die Instance-Gruppe mit fünf Knoten und fügt die sechste nach Möglichkeit später hinzu.

Das Starten von Aufgaben-Instance-Gruppen als Spot-Instances stellt eine strategische Möglichkeit dar, die Kapazität Ihres Clusters zu erweitern und gleichzeitig die Kosten zu minimieren. Wenn Sie

Ihre Primär- und Kern-Instance-Gruppen als On-Demand-Instances starten, ist ihre Kapazität für die Ausführung des Clusters garantiert. Sie können Ihren Instance-Gruppen nach Bedarf Task-Instances hinzufügen, um ein Spitzenaufkommen an Datenverkehr zu verarbeiten oder die Datenverarbeitung zu beschleunigen.

Sie können Task-Knoten mithilfe der Konsole, AWS CLI, oder hinzufügen oder entfernenAPI. Sie können auch zusätzliche Aufgabengruppen hinzufügen, können aber diese nach dem Erstellen nicht mehr entfernen.

Instance-Konfigurationen für Anwendungsszenarien

Die folgende Tabelle stellt eine kurze Referenz für Knotentyp-Kaufoptionen und -Konfigurationen dar, die für bestimmte Anwendungsszenarien in der Regel geeignet sind. Klicken Sie auf den Link, um weitere Informationen zu den einzelnen Szenariotypen anzuzeigen.

Anwendungsszenario	Kaufoption für Primärknoten	Kaufoption für Core-Knoten	Kaufoption für Aufgabenknoten
Langläufer-Cluster und Data Warehouses	On-Demand	On-Demand oder Instance-Flottenkombination	Spot- oder Instance-Flottenkombination
Kostengesteuerte Workloads	Spot-Instances	Spot-Instances	Spot-Instances
Datenkritische Workloads	On-Demand	On-Demand	Spot- oder Instance-Flottenkombination
Testen von Anwendungen	Spot-Instances	Spot-Instances	Spot-Instances

Es gibt mehrere Szenarien, in denen Spot-Instances für den Betrieb eines EMR Amazon-Clusters nützlich sind.

Langläufer-Cluster und Data Warehouses

Wenn Sie einen persistenten EMR Amazon-Cluster mit vorhersehbaren Schwankungen der Rechenkapazität betreiben, wie z. B. ein Data Warehouse, können Sie mit Spot-Instances Spitzennachfrage zu geringeren Kosten bewältigen. Sie können Ihre Primär- und Core-Instance-

Gruppen als On-Demand starten, um die normale Kapazität zu bewältigen, und die Aufgaben-Instance-Gruppe als Spot Instances für Ihre maximale Workload-Anforderungen starten.

Kostengesteuerte Workloads

Wenn Sie kurzlebige Cluster ausführen, für die niedrige Kosten wichtiger sind als die Zeit bis zum Abschluss des Vorgangs, der Verlust von Teilarbeiten akzeptabel ist, können Sie den gesamten Cluster (Primär-, Core- und Aufgaben-Instance-Gruppen) als Spot Instances ausführen, um von den größten Kosteneinsparungen zu profitieren.

Datenkritische Workloads

Wenn Sie einen Cluster ausführen, für den niedrige Kosten wichtiger sind als die Zeit bis zum Abschluss des Vorgangs, der Verlust von Teilarbeiten jedoch nicht akzeptabel ist, können Sie die Primär- und Core-Instance-Gruppen als On-Demand-Instances starten und durch eine oder mehrere Aufgaben-Instance-Gruppen der Spot Instances ergänzen. Wenn Sie die primären Instance-Gruppen und die Kern-Instance-Gruppen als On-Demand-Instances ausführen, wird sichergestellt, dass Ihre Daten dauerhaft gespeichert werden HDFS und dass der Cluster vor einer Kündigung aufgrund von Schwankungen auf dem Spotmarkt geschützt ist. Gleichzeitig werden Kosteneinsparungen erzielt, die sich aus der Ausführung der Task-Instance-Gruppen als Spot-Instances ergeben.

Testen von Anwendungen

Wenn Sie eine neue Anwendung testen, um sie für den Start in einer Produktionsumgebung vorzubereiten, können Sie den gesamten Cluster (Primär-, Core- und Aufgaben-Instance-Gruppen) als Spot Instances ausführen, um die Kosten der Tests zu senken.

Berechnung der erforderlichen HDFS Kapazität eines Clusters

Die Menge an HDFS Speicherplatz, die Ihrem Cluster zur Verfügung steht, hängt von den folgenden Faktoren ab:

- Die Anzahl der EC2 Amazon-Instances, die für Kernknoten verwendet werden.
- Die Kapazität des EC2 Amazon-Instance-Speichers für den verwendeten Instance-Typ. Weitere Informationen zu Instance-Speicher-Volumes finden Sie unter Amazon Amazon EC2 Instance Store im EC2Amazon-Benutzerhandbuch.
- Anzahl und Größe der EBS Amazon-Volumes, die an Kernknoten angeschlossen sind.
- Ein Replikationsfaktor, der berücksichtigt, wie jeder Datenblock gespeichert wird, um eine RAID ähnliche Redundanz HDFS zu erreichen. Standardmäßig beträgt der Replikationsfaktor 3 für einen

Cluster mit 10 oder mehr Core-Knoten, 2 für einen Cluster mit 4 bis 9 Core-Knoten und 1 für einen Cluster mit maximal 3 Knoten.

Um die HDFS Kapazität eines Clusters zu berechnen, fügen Sie für jeden Kernknoten die Volumenkapazität des Instance-Speichers zur EBS Amazon-Speicherkapazität hinzu (falls verwendet). Multiplizieren Sie das Ergebnis mit der Anzahl der Core-Knoten und dividieren Sie dann die Summe durch den Replikationsfaktor basierend auf der Anzahl der Core-Knoten. Beispielsweise stehen für einen Cluster mit 10 Core-Knoten des Typs i2.xlarge, die über 800 GB Instance-Speicher ohne angehängte EBS Amazon-Volumes verfügen, insgesamt etwa 2.666 GB zur Verfügung HDFS (10 Knoten x 800 GB — 3 Replikationsfaktor).

Wenn der berechnete HDFS Kapazitätswert kleiner als Ihre Daten ist, können Sie die HDFS Speichermenge auf folgende Weise erhöhen:

- Erstellen eines Clusters mit zusätzlichen EBS Amazon-Volumes oder Hinzufügen von Instance-Gruppen mit angehängten EBS Amazon-Volumes zu einem vorhandenen Cluster
- Hinzufügen weiterer Core-Knoten
- Auswahl eines EC2 Amazon-Instance-Typs mit größerer Speicherkapazität
- Verwenden der Datenkomprimierung
- Ändern der Hadoop-Konfigurationseinstellungen zum Verringern des Replikationsfaktors

Die Reduzierung des Replikationsfaktors sollte mit Vorsicht verwendet werden, da dadurch die Redundanz der HDFS Daten und die Fähigkeit des Clusters, sich nach verlorenen oder beschädigten HDFS Blöcken wiederherzustellen, verringert werden.

Konfigurieren der Cluster-Protokollierung und des Debuggings

Bei der Planung Ihres Clusters müssen Sie sich unter anderem für die verfügbare Debugging-Unterstützung entscheiden. Wenn Sie Ihre Datenverarbeitungsanwendung erstmals entwickeln, empfehlen wir Ihnen, die Anwendung auf einem Cluster zu testen, indem Sie eine kleine, aber repräsentative Untermenge Ihrer Daten verarbeiten. Wenn Sie dies tun, möchten Sie wahrscheinlich alle Debugging-Tools nutzen, die Amazon EMR anbietet, z. B. die Archivierung von Protokolldateien in Amazon S3.

Wenn Sie die Entwicklung Ihrer Anwendung abgeschlossen haben und die Datenverarbeitung in die Produktionsumgebung wechselt, können Sie das Debuggen verringern. Auf diese Weise können

Sie die Kosten für die Speicherung von Protokolldateiarchiven in Amazon S3 einsparen und die Verarbeitungslast für den Cluster reduzieren, da dieser den Zustand nicht mehr zu Amazon S3 schreiben muss. Der Nachteil ist, das Ihnen bei Problemen weniger Tools zur Verfügung stehen, um das Problem zu untersuchen.

Standardmäßige Protokolldateien

Standardmäßig schreibt jeder Cluster Protokolldateien auf dem Primärknoten. Die Dateien werden in das `/mnt/var/log/`-Verzeichnis geschrieben. Sie können auf sie zugreifen, indem SieSSH, wie unter beschrieben, eine Verbindung zum Primärknoten herstellen. Connect zum Primärknoten her mit SSH Amazon EMR sammelt bestimmte System- und Anwendungsprotokolle, die von EMR Amazon-Daemons und anderen EMR Amazon-Prozessen generiert wurden, um einen effektiven Servicebetrieb sicherzustellen.

> **Note**
>
> Wenn Sie Amazon EMR Version 6.8.0 oder früher verwenden, werden Protokolldateien während der Clusterbeendigung in Amazon S3 gespeichert, sodass Sie nicht mehr auf die Protokolldateien zugreifen können, wenn der primäre Knoten beendet wird. Amazon EMR veröffentlicht 6.9.0 und höher und archiviert Protokolle während der Cluster-Scale-Down in Amazon S3, sodass die auf dem Cluster generierten Protokolldateien auch nach dem Beenden des Knotens bestehen bleiben.

Sie müssen nicht alles aktivieren, um die Protokolldateien auf dem Primärknoten schreiben zu lassen. Dies ist das Standardverhalten von Amazon EMR und Hadoop.

Ein Cluster generiert mehrere Arten von Protokolldateien. Diese umfassen unter anderem:

- Schrittprotokolle — Diese Protokolle werden vom EMR Amazon-Service generiert und enthalten Informationen über den Cluster und die Ergebnisse der einzelnen Schritte. Die Protokolldateien werden im `/mnt/var/log/hadoop/steps/`-Verzeichnis auf dem Primärknoten gespeichert. Jeder Schritt protokolliert seine Ergebnisse in einem separaten, nummerierten Unterverzeichnis: `/mnt/var/log/hadoop/steps/s-stepId1/` für den ersten Schritt, `/mnt/var/log/hadoop/steps/s-stepId2/` für den zweiten Schritt, und so weiter. Die 13-stelligen Schrittkennungen (z. B. stepId 1, stepId 2) sind für einen Cluster eindeutig.
- Hadoop- und YARN Komponentenprotokolle — Die Protokolle für Komponenten, die YARN sowohl Apache als auch zugeordnet sind MapReduce, befinden sich beispielsweise in separaten Ordnern

in. `/mnt/var/log` Die Speicherorte der Protokolldateien für die Hadoop-Komponenten unter `/mnt/var/log` lauten folgendermaßen: hadoop-hdfs, hadoop-mapreduce, hadoop-httpfs und hadoop-yarn. Das hadoop-state-pusher Verzeichnis ist für die Ausgabe des Hadoop-State-Pusher-Prozesses vorgesehen.

- Bootstrap-Aktion-Protokolle – Wenn Ihr Auftrag Bootstrap-Aktionen verwendet, werden die Ergebnisse dieser Aktionen protokolliert. Die Protokolldateien werden in /mnt/var/log/bootstrap-actions/ auf dem Primärknoten gespeichert. Jede Bootstrap-Aktion protokolliert ihre Ergebnisse in einem separaten, nummerierten Unterverzeichnis: `/mnt/var/log/bootstrap-actions/1/` für die erste Bootstrap-Aktion, `/mnt/var/log/bootstrap-actions/2/` für die zweite, und so weiter.
- Instanzstatusprotokolle — Diese Protokolle enthalten Informationen über den Speicherstatus und die CPU Garbage-Collector-Threads des Knotens. Die Protokolldateien werden in `/mnt/var/log/instance-state/` auf dem Primärknoten gespeichert.

Archivieren von Protokolldateien in Amazon S3

> **Note**
>
> Sie können mit dem `yarn logs`-Dienstprogramm derzeit keine Protokollzusammenführung in Amazon S3 durchführen.

Amazon EMR veröffentlicht 6.9.0 und höher und archiviert Protokolle während der Cluster-Scale-Down in Amazon S3, sodass die auf dem Cluster generierten Protokolldateien auch nach dem Beenden des Knotens bestehen bleiben. Dieses Verhalten wird automatisch aktiviert, sodass Sie nichts unternehmen müssen, um es zu aktivieren. Für EMR Amazon-Versionen 6.8.0 und früher können Sie einen Cluster so konfigurieren, dass die auf dem primären Knoten gespeicherten Protokolldateien regelmäßig in Amazon S3 archiviert werden. Auf diese Weise wird sichergestellt, dass die Protokolldateien verfügbar sind, nachdem der Cluster beendet wird (unabhängig davon, ob dieser normal heruntergefahren wurde oder ob ein Fehler aufgetreten ist). Amazon EMR archiviert die Protokolldateien in Intervallen von 5 Minuten auf Amazon S3.

Um die Protokolldateien in Amazon S3 für EMR Amazon-Versionen 6.8.0 und früher zu archivieren, müssen Sie diese Funktion aktivieren, wenn Sie den Cluster starten. Sie können dies mit der KonsoleCLI, dem oder dem API tun. Die Protokollierung ist bei über die Konsole gestarteten

Clustern standardmäßig aktiviert. Für Cluster, die mit dem CLI oder gestartet wurdenAPI, muss die Protokollierung bei Amazon S3 manuell aktiviert werden.

Console

Wie Sie Protokolldateien auf Amazon S3 mit der neuen Konsole archivieren

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und anschließend Create cluster aus.
3. Aktivieren Sie unter Cluster-Protokolle das Kontrollkästchen Cluster-spezifische Protokolle in Amazon S3 veröffentlichen.
4. Geben Sie im Feld Speicherort von Amazon S3 einen Amazon-S3-Pfad zum Speichern Ihrer Protokolle ein. Wenn Sie den Namen eines Ordners eingeben, der nicht im Bucket vorhanden ist, wird er von Amazon S3 erstellt.

   Wenn Sie diesen Wert festlegen, EMR kopiert Amazon die Protokolldateien von den EC2 Instances im Cluster nach Amazon S3. Dadurch wird verhindert, dass die Protokolldateien verloren gehen, wenn der Cluster endet und die Instances, die den Cluster hosten, EC2 beendet werden. Diese Protokolle sind bei der Fehlerbehebung hilfreich. Weitere Informationen finden Sie unter Protokolldateien anzeigen.
5. Aktivieren Sie optional das Kontrollkästchen Clusterspezifische Protokolle verschlüsseln. Wählen Sie dann einen AWS KMS Schlüssel aus der Liste aus, geben Sie einen Schlüssel einARN, oder erstellen Sie einen neuen Schlüssel. Diese Option ist nur mit EMR Amazon-Version 5.30.0 und höher verfügbar, mit Ausnahme von Version 6.0.0. Um diese Option zu verwenden, fügen Sie Berechtigungen AWS KMS für Ihr EC2 Instance-Profil und Ihre EMR Amazon-Rolle hinzu. Weitere Informationen finden Sie unter Um in Amazon S3 gespeicherte Protokolldateien mit einem vom AWS KMS Kunden verwalteten Schlüssel zu verschlüsseln.
6. Wählen Sie alle anderen Optionen aus, die für Ihren Cluster gelten.
7. Um Ihren Cluster jetzt zu starten, wählen Sie Cluster erstellen aus.

CLI

Um Protokolldateien auf Amazon S3 zu archivieren mit dem AWS CLI

Um Protokolldateien mit dem in Amazon S3 zu archivieren AWS CLI, geben Sie den `create-cluster` Befehl ein und geben Sie den Amazon S3 S3-Protokollpfad mithilfe des `--log-uri` Parameters an.

1. Um Dateien in Amazon S3 zu protokollieren, geben Sie den folgenden Befehl ein und ersetzen Sie *myKey* mit dem Namen Ihres EC2 key pair.

```
aws emr create-cluster --name "Test cluster" --release-label emr-7.2.0 --log-
uri s3://DOC-EXAMPLE-BUCKET/logs --applications Name=Hadoop Name=Hive Name=Pig
 --use-default-roles --ec2-attributes KeyName=myKey --instance-type m5.xlarge --
instance-count 3
```

2. Wenn Sie die Instance-Anzahl ohne den `--instance-groups`-Parameter angeben, wird ein einzelner Primärknoten gestartet. Die verbleibenden Instances werden dabei als Core-Knoten gestartet. Alle Knoten verwenden den im Befehl angegebenen Instance-Typ.

> **Note**
>
> Wenn Sie noch nicht die standardmäßige EMR Amazon-Servicerolle und das EC2 Instanzprofil erstellt haben, geben Sie ein, `aws emr create-default-roles` um sie zu erstellen, bevor Sie den `create-cluster` Unterbefehl eingeben.

Um in Amazon S3 gespeicherte Protokolldateien mit einem vom AWS KMS Kunden verwalteten Schlüssel zu verschlüsseln

Mit EMR Amazon-Version 5.30.0 und höher (außer Amazon EMR 6.0.0) können Sie in Amazon S3 gespeicherte Protokolldateien mit einem AWS KMS vom Kunden verwalteten Schlüssel verschlüsseln. Um diese Option über die Konsole zu aktivieren, führen Sie die Schritte unter Archivieren von Protokolldateien in Amazon S3 aus. Ihr EC2 Amazon-Instance-Profil und Ihre EMR Amazon-Rolle müssen die folgenden Voraussetzungen erfüllen:

- Das für Ihren Cluster verwendete EC2 Amazon-Instance-Profil muss über eine Nutzungsberechtigung verfügen`kms:GenerateDataKey`.

- Die für Ihren Cluster verwendete EMR Amazon-Rolle muss über eine Nutzungsberechtigung verfügen`kms:DescribeKey`.
- Das EC2 Amazon-Instance-Profil und die EMR Amazon-Rolle müssen der Liste der Hauptbenutzer für den angegebenen vom AWS KMS Kunden verwalteten Schlüssel hinzugefügt werden, wie die folgenden Schritte zeigen:
  1. Öffnen Sie die Konsole AWS Key Management Service (AWS KMS) unter https://console.aws.amazon.com/kms.
  2. Um die AWS Region zu ändern, verwenden Sie die Regionsauswahl in der oberen rechten Ecke der Seite.
  3. Wählen Sie den Alias des KMS Schlüssels aus, den Sie ändern möchten.
  4. Wählen Sie auf der Seite mit den Schlüsseldetails unter Key Users (Schlüsselbenutzer(die Option Add (Hinzufügen) aus.
  5. Wählen Sie im Dialogfeld „Schlüsselbenutzer hinzufügen“ Ihr EC2 Amazon-Instance-Profil und Ihre EMR Amazon-Rolle aus.
  6. Wählen Sie Hinzufügen aus.

Weitere Informationen finden Sie unter Von Amazon EMR verwendete IAM Servicerollen und Verwenden von Schlüsselrichtlinien im AWS Key Management Service Developer Guide.

So aggregieren Sie Protokolle in Amazon S3 über die AWS CLI

> **Note**
>
> Sie können mit dem `yarn logs`-Dienstprogramm derzeit keine Protokollzusammenführung durchführen. Sie können die durch dieses Verfahren unterstützte Aggregation nutzen.

Bei der Protokollaggregation (Hadoop 2.x) werden Protokolle für eine bestimmte Anwendung aus allen Containern in einer einzigen Datei zusammengestellt. Um die Protokollaggregation für Amazon S3 mithilfe von zu aktivieren AWS CLI, verwenden Sie beim Clusterstart eine Bootstrap-Aktion, um die Protokollaggregation zu aktivieren und den Bucket zum Speichern der Protokolle anzugeben.

- Um die Protokollaggregation zu aktivieren, erstellen Sie die folgende Konfigurationsdatei mit dem Namen `myConfig.json`, die Folgendes enthält:

```
[
```

```
  {
    "Classification": "yarn-site",
    "Properties": {
      "yarn.log-aggregation-enable": "true",
      "yarn.log-aggregation.retain-seconds": "-1",
      "yarn.nodemanager.remote-app-log-dir": "s3:\/\/DOC-EXAMPLE-BUCKET\/logs"
    }
  }
]
```

Geben Sie den folgenden Befehl ein und ersetzen Sie *myKey* mit dem Namen Ihres EC2 key pair. Sie können zusätzlich jeden der roten Texte durch Ihre eigenen Konfigurationen ersetzen.

```
aws emr create-cluster --name "Test cluster" \
--release-label emr-7.2.0 \
--applications Name=Hadoop \
--use-default-roles \
--ec2-attributes KeyName=myKey \
--instance-type m5.xlarge \
--instance-count 3 \
--configurations file://./myConfig.json
```

Wenn Sie die Instance-Anzahl ohne den `--instance-groups`-Parameter angeben, wird ein einzelner Primärknoten gestartet. Die verbleibenden Instances werden dabei als Core-Knoten gestartet. Alle Knoten verwenden den im Befehl angegebenen Instance-Typ.

> **Note**
>
> Wenn Sie die standardmäßige EMR Servicerolle und das EC2 Instanzprofil noch nicht erstellt haben, führen Sie zunächst den Befehl aus, `aws emr create-default-roles` um sie zu erstellen, bevor Sie den `create-cluster` Unterbefehl ausführen.

Weitere Informationen zur Verwendung von EMR Amazon-Befehlen finden Sie in der AWS CLIAWS CLI Befehlsreferenz.

Protokollspeicherorte

Die folgende Liste enthält alle Protokolltypen und ihre Speicherorte in Amazon S3. Sie können diese zur Behebung von EMR Amazon-Problemen verwenden.

Schrittprotokolle

```
s3://DOC-EXAMPLE-LOG-BUCKET/<cluster-id>/steps/<step-id>/
```

Anwendungsprotokolle

```
s3://DOC-EXAMPLE-LOG-BUCKET/<cluster-id>/containers/
```

Dieser Speicherort umfasst Container `stderr` und `stdout`, `directory.info`, `prelaunch.out` und `launch_container.sh`-Protokolle.

Resource-Manager-Protokolle

```
s3://DOC-EXAMPLE-LOG-BUCKET/<cluster-id>/node/<leader-instance-id>/
applications/hadoop-yarn/
```

Hadoop HDFS

```
s3://DOC-EXAMPLE-LOG-BUCKET/<cluster-id>/node/<all-instance-id>/
applications/hadoop-hdfs/
```

Dieser Speicherort umfasst NameNode DataNode, und YARN TimelineServer Protokolle.

Knoten-Manager-Protokolle

```
s3://DOC-EXAMPLE-LOG-BUCKET/<cluster-id>/node/<all-instance-id>/
applications/hadoop-yarn/
```

Instance-Statusprotokolle

```
s3://DOC-EXAMPLE-LOG-BUCKET/<cluster-id>/node/<all-instance-id>/daemons/
instance-state/
```

EMRAmazon-Bereitstellungsprotokolle

```
s3://DOC-EXAMPLE-LOG-BUCKET/<cluster-id>/node/<leader-instance-id>/
provision-node/*
```

Hive-Protokolle

```
s3://DOC-EXAMPLE-LOG-BUCKET/<cluster-id>/node/<leader-instance-id>/
applications/hive/*
```

- Um Hive-Protokolle in Ihrem Cluster zu finden, entfernen Sie das Sternchen (*) und fügen Sie `/var/log/hive/` an den obigen Link an.
- Um HiveServer zwei Protokolle zu finden, entfernen Sie das Sternchen (*) und fügen Sie es `var/log/hive/hiveserver2.log` an den obigen Link an.

- Um CLI Hive-Protokolle zu finden, entfernen Sie das Sternchen (*) und fügen `/var/log/hive/user/hadoop/hive.log` Sie es an den obigen Link an.
- Um Hive-Metastore-Server-Protokolle zu finden, entfernen Sie das Sternchen (*) und fügen Sie `/var/log/hive/user/hive/hive.log` an den obigen Link an.

Wenn Ihr Fehler im Primär- oder Aufgabenknoten Ihrer Tez-Anwendung auftritt, stellen Sie die Protokolle des entsprechenden Hadoop-Containers bereit.

Tag-Cluster

Es kann praktisch sein, Ihre AWS Ressourcen auf unterschiedliche Weise zu kategorisieren, z. B. nach Zweck, Eigentümer oder Umgebung. Sie können dies in Amazon erreichen, EMR indem Sie Ihren EMR Amazon-Clustern mithilfe von Tags benutzerdefinierte Metadaten zuweisen. Ein Tag besteht aus einem Schlüssel und einem Wert, die Sie beide selbst definieren können. Für Amazon ist der Cluster die RessourcenebeneEMR, die Sie taggen können. Sie können beispielsweise eine Gruppe von Tags für die Cluster Ihres Kontos definieren, mit deren Hilfe Sie den Besitzer der einzelnen Cluster verfolgen oder eine Produktions-Cluster von einem Test-Cluster unterscheiden können. Wir empfehlen das Erstellen eines einheitlichen Satzes von Tags, um Anforderungen Ihres Unternehmens zu erfüllen.

Wenn Sie einem EMR Amazon-Cluster ein Tag hinzufügen, wird das Tag auch an jede aktive EC2 Amazon-Instance weitergegeben, die dem Cluster zugeordnet ist. Wenn Sie ein Tag aus einem EMR Amazon-Cluster entfernen, wird dieses Tag auch von jeder zugehörigen aktiven EC2 Amazon-Instance entfernt.

> **Important**
>
> Verwenden Sie die EMR Amazon-Konsole oder CLI zur Verwaltung von Tags auf EC2 Amazon-Instances, die Teil eines Clusters sind, anstelle der EC2 Amazon-Konsole oderCLI, weil Änderungen, die Sie in Amazon vornehmen, EC2 nicht mit dem EMR Amazon-Tagging-System synchronisiert werden.

Sie können eine EC2 Amazon-Instance identifizieren, die Teil eines EMR Amazon-Clusters ist, indem Sie nach den folgenden System-Tags suchen. In diesem Beispiel: *`CORE`* ist der Wert für die Instance-Gruppenrolle und *`j-12345678`* ist ein Beispiel für einen Job-Flow-Identifikationswert (Cluster):

- aws:elasticmapreduce: = instance-group-role*`CORE`*

- aws:elasticmapreduce: job-flow-id =*j-12345678*

> **Note**
>
> Amazon EMR und Amazon EC2 interpretieren Ihre Tags als eine Zeichenfolge ohne semantische Bedeutung.

Sie können mit Tags arbeiten, indem Sie die AWS Management Console CLI, und die API verwenden.

Sie können Tags hinzufügen, wenn Sie einen neuen EMR Amazon-Cluster erstellen, und Sie können Tags zu einem laufenden EMR Amazon-Cluster hinzufügen, bearbeiten oder entfernen. Das Bearbeiten eines Tags ist ein Konzept, das für die EMR Amazon-Konsole gilt. Wenn Sie jedoch das CLI und verwendenAPI, um ein Tag zu bearbeiten, entfernen Sie das alte Tag und fügen ein neues hinzu. Sie können Tag-Schlüssel und Werte bearbeiten und Tags jederzeit aus einer Ressource entfernen, während der Cluster ausgeführt wird. Sie können Tags jedoch nicht hinzufügen, bearbeiten oder aus einem beendeten Cluster oder beendeten Instances entfernen, die zuvor einem Cluster zugeordnet waren, der noch aktiv ist. Darüber hinaus können Sie den Wert eines Tags zwar auf eine leere Zeichenfolge, jedoch nicht auf Null festlegen.

Wenn Sie AWS Identity and Access Management (IAM) mit Ihren EC2 Amazon-Instances für ressourcenbasierte Berechtigungen nach Tag verwenden, werden Ihre IAM Richtlinien auf Tags angewendet, die Amazon an die Amazon-Instances eines Clusters EMR weitergibt. EC2 Damit EMR Amazon-Tags an Ihre EC2 Amazon-Instances weitergegeben werden können, EC2 muss Ihre IAM Richtlinie für Amazon Berechtigungen zum Aufrufen von Amazon EC2 CreateTags und DeleteTags APIs zulassen. Weitergaben von Tags können sich auch auf die ressourcenbasierten Berechtigungen EC2 Ihres Amazon auswirken. An Amazon weitergegebene Tags EC2 können in Ihrer IAM Richtlinie als Bedingungen gelesen werden, genau wie andere EC2 Amazon-Tags. Denken Sie beim Hinzufügen von Tags zu Ihren EMR Amazon-Clustern an Ihre IAM Richtlinien, um zu verhindern, dass Benutzer falsche Berechtigungen für einen Cluster haben. Um Probleme zu vermeiden, stellen Sie sicher, dass Ihre IAM Richtlinien keine Bedingungen für Tags enthalten, die Sie auch in Ihren EMR Amazon-Clustern verwenden möchten. Weitere Informationen finden Sie unter Steuern des Zugriffs auf EC2 Amazon-Ressourcen.

Tag-Einschränkungen

Die folgenden grundlegenden Einschränkungen gelten für Tags (Markierungen):

- Einschränkungen, die für EC2 Amazon-Ressourcen gelten, gelten EMR auch für Amazon. Weitere Informationen finden Sie unter https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/Using_Tags.html#tag-restrictions.
- Verwenden Sie das `aws:` Präfix nicht in Tag-Namen und -Werten, da es für die AWS Verwendung reserviert ist. Sie können darüber hinaus keine Tag-Namen oder Werte mit diesem Präfix bearbeiten oder löschen.
- Sie können Tags in einem beendeten Cluster nicht ändern oder bearbeiten.
- Ein Tag-Wert kann eine leere Zeichenfolge, aber nicht null sein. Darüber hinaus kann ein Tag-Schlüssel keine leere Zeichenfolge sein.
- Schlüssel und Werte können alphabetische Zeichen in jeder Sprache, numerische Zeichen, Leerzeichen, unsichtbare Trennzeichen und die folgenden Symbole sein: _ . : / = + - @.

Weitere Informationen zum Taggen mit dem AWS Management Console finden Sie unter Arbeiten mit Tags in der Konsole im EC2Amazon-Benutzerhandbuch. Weitere Informationen zum Taggen über Amazon EC2API oder die Befehlszeile finden Sie unter APIund im CLI Überblick im EC2Amazon-Benutzerhandbuch.

Markieren von Ressourcen für die Fakturierung

Sie können Tags verwenden, um Ihre AWS Rechnung so zu organisieren, dass sie Ihre eigene Kostenstruktur widerspiegeln. Melden Sie sich dazu an, um Ihre AWS Kontorechnung mit den Tag-Schlüsselwerten zu erhalten. Anschließend können Sie Ihre Abrechnungsdaten nach Tag-Schlüsselwerten organisieren, um die Kosten kombinierter Ressourcen zu ermitteln. Obwohl Amazon EMR und Amazon unterschiedliche Abrechnungen EC2 haben, werden die Tags auf jedem Cluster auch auf jeder zugehörigen Instance platziert, sodass Sie Tags verwenden können, um zugehörige Amazon EMR - und EC2 Amazon-Kosten miteinander zu verknüpfen.

Beispielsweise können Sie mehrere Ressourcen mit einem bestimmten Anwendungsnamen markieren und dann Ihre Fakturierungsinformationen so organisieren, dass Sie die Gesamtkosten dieser Anwendung über mehrere Services hinweg sehen können. Weitere Informationen finden Sie unter Kostenzuordnung und Tagging im AWS Billing -Benutzerhandbuch.

Hinzufügen von Tags zu einem Cluster

Sie können dem Cluster auch Tags hinzufügen, wenn Sie ihn erstellen.

Console

So fügen Sie Tags hinzu, wenn Sie einen Cluster mit der neuen Konsole erstellen

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und anschließend Create cluster aus.
3. Wählen Sie unter Tags die Option Neuen Tag hinzufügen aus. Geben Sie im Feld Schlüssel ein Tag an. Geben Sie optional ein Tag im Feld Wert an.
4. Wählen Sie alle anderen Optionen aus, die für Ihren Cluster gelten.
5. Um Ihren Cluster jetzt zu starten, wählen Sie Cluster erstellen aus.

AWS CLI

Um Tags hinzuzufügen, wenn Sie einen Cluster mit dem AWS CLI

Im folgenden Beispiel wird gezeigt, wie ein Tag einem neuen Cluster über die AWS CLI hinzugefügt wird. Zum Hinzufügen von Tags beim Erstellen eines Clusters geben Sie den Unterbefehl `create-cluster` mit dem Parameter `--tags` ein.

- Um ein Tag mit dem Namen hinzuzufügen *`costCenter`* mit Schlüsselwert *`marketing`* wenn Sie einen Cluster erstellen, geben Sie den folgenden Befehl ein und ersetzen Sie *`myKey`* mit dem Namen Ihres EC2 key pair.

  ```
  aws emr create-cluster --name "Test cluster" --release-label emr-4.0.0 --
  applications Name=Hadoop Name=Hive Name=Pig --tags "costCenter=marketing" --
  use-default-roles --ec2-attributes KeyName=myKey --instance-type m5.xlarge --
  instance-count 3
  ```

  Wenn Sie die Instance-Anzahl ohne den Parameter `--instance-groups` angeben, wird ein einzelner Master-Knoten gestartet. Die verbleibenden Instances werden dabei als Core-Knoten gestartet. Alle Knoten verwenden den im Befehl angegebenen Instance-Typ.

> **Note**
>
> Wenn Sie die EMR Standard-Servicerolle und das EC2 Instanzprofil noch nicht erstellt haben, geben Sie aws ein, `emr create-default-roles` um sie zu erstellen, bevor Sie den `create-cluster` Unterbefehl eingeben.

Weitere Informationen zur Verwendung von EMR Amazon-Befehlen in der AWS CLI finden Sie unterhttps://docs.aws.amazon.com/cli/latest/reference/emr.

Sie können Tags auch einem vorhandenen Cluster hinzufügen.

Console

So fügen Sie Tags zu einem vorhandenen Cluster über die neue Konsole hinzu

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und dann den Cluster aus, den Sie aktualisieren möchten.
3. Wählen Sie auf der Cluster-Detailseite auf der Registerkarte Tags die Option Tags verwalten aus. Geben Sie im Feld Schlüssel ein Tag an. Geben Sie optional ein Tag im Feld Wert an.
4. Wählen Sie Änderungen speichern aus. Die Registerkarte Tags wird mit der neuen Anzahl von Tags aktualisiert, die Sie in Ihrem Cluster haben. Wenn Sie jetzt beispielsweise zwei Tags haben, lautet die Bezeichnung Ihres Tabs Tags (2).

AWS CLI

Um einem laufenden Cluster Tags hinzuzufügen, verwenden Sie den AWS CLI

- Geben Sie den Unterbefehl `add-tags` mit dem Parameter `--tag` ein, um der Cluster-ID Tags zuzuweisen. Sie können die Cluster-ID mithilfe der Konsole oder des Befehls `list-clusters` finden. Der Unterbefehl `add-tags` akzeptiert derzeit nur einen Ressourcenbezeichner.

Um beispielsweise einem laufenden Cluster zwei Tags hinzuzufügen, eines mit einem Schlüssel namens *costCenter* mit einem Wert von *marketing* und ein anderer namens *other* mit einem Wert von *accounting*, geben Sie den folgenden Befehl ein und ersetzen Sie *j-KT4XXXXXXXX1NM* mit Ihrer Cluster-ID.

```
aws emr add-tags --resource-id j-KT4XXXXXXXX1NM --tag "costCenter=marketing" --
tag "other=accounting"
```

Beachten Sie, dass beim Hinzufügen von Tags mit dem AWS CLI keine Ausgabe des Befehls erfolgt. Weitere Informationen zur Verwendung von EMR Amazon-Befehlen in der AWS CLI finden Sie unterhttps://docs.aws.amazon.com/cli/latest/reference/emr.

Tags in einem Cluster anzeigen

Wenn Sie alle Tags anzeigen möchten, die mit einem Cluster verknüpft sind, können Sie diese in der Konsole oder AWS CLI ansehen.

Console

So zeigen Sie Tags in einem Cluster mit der neuen Konsole an

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und dann den Cluster aus, den Sie aktualisieren möchten.
3. Um alle Ihre Tags anzuzeigen, wählen Sie auf der Cluster-Detailseite die Registerkarte Tags aus.

AWS CLI

Um Tags auf einem Cluster mit dem anzuzeigen AWS CLI

Um die Tags auf einem Cluster mithilfe von anzuzeigen AWS CLI, geben Sie den `describe-cluster` Unterbefehl mit dem `--query` Parameter ein.

- Um die Tags eines Clusters anzuzeigen, geben Sie den folgenden Befehl ein und ersetzen Sie *j-KT4XXXXXXXX1NM* mit Ihrer Cluster-ID.

```
aws emr describe-cluster --cluster-id j-KT4XXXXXX1NM --query Cluster.Tags
```

Die Ausgabe enthält alle Tag-Informationen über den Cluster ähnlich wie diese:

```
Value: accounting     Value: marketing
Key: other            Key: costCenter
```

Weitere Informationen zur Verwendung von EMR Amazon-Befehlen in der AWS CLI finden Sie unterhttps://docs.aws.amazon.com/cli/latest/reference/emr.

Tags aus einem Cluster entfernen

Wenn Sie ein Tag nicht mehr benötigen, können Sie es aus dem Cluster entfernen.

Console

So entfernen Sie Tags in einem Cluster mit der neuen Konsole

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und dann den Cluster aus, den Sie aktualisieren möchten.
3. Wählen Sie auf der Cluster-Detailseite auf der Registerkarte Tags die Option Tags verwalten aus.
4. Wählen Sie Entfernen für jedes Schlüssel-Wert-Paar, das Sie entfernen möchten.
5. Wählen Sie Änderungen speichern.

AWS CLI

Um Tags auf einem Cluster mit dem zu entfernen AWS CLI

Geben Sie den `remove-tags`-Unterbefehl mit dem `--tag-keys`-Parameter ein. Beim Entfernen eines Tags ist nur der Schlüsselname erforderlich.

- Um ein Tag aus einem Cluster zu entfernen, geben Sie den folgenden Befehl ein und ersetzen *j-KT4XXXXXXXX1NM* mit Ihrer Cluster-ID.

```
aws emr remove-tags --resource-id j-KT4XXXXXX1NM --tag-keys "costCenter"
```

> **Note**
>
> Sie können derzeit nicht mehrere Tags mit einem einzigen Befehl entfernen.

Weitere Informationen zur Verwendung von EMR Amazon-Befehlen in der AWS CLI finden Sie unterhttps://docs.aws.amazon.com/cli/latest/reference/emr.

Treiber und Drittanbieter-Anwendungsintegration

Sie können mehrere beliebte Big-Data-Anwendungen auf Amazon EMR gegen Nebenkosten ausführen. Das bedeutet, dass Sie eine geringe zusätzliche Gebühr pro Stunde für die Drittanbieter-Anwendung zahlen, während der Cluster ausgeführt wird. So können Sie die Anwendung nutzen, ohne eine Jahreslizenz erwerben zu müssen. In den folgenden Abschnitten werden einige der Tools beschrieben, die Sie mit verwenden können. EMR

Themen

Verwenden Sie Business Intelligence-Tools mit Amazon EMR

Sie können beliebte Business Intelligence-Tools wie Microsoft Excel, MicroStrategy, und Tableau mit Amazon verwendenQlikView, um Ihre Daten EMR zu untersuchen und zu visualisieren. Für viele dieser Tools ist ein Treiber ODBC (Open Database Connectivity) oder JDBC (Java Database Connectivity) erforderlich. Informationen zum Herunterladen und Installieren der neuesten Treiber finden Sie unter http://awssupportdatasvcs.com/bootstrap-actions/Simba/latest/.

Ältere Versionen von Treibern finden Sie unter http://awssupportdatasvcs.com/bootstrap-actions/Simba/.

Sicherheit bei Amazon EMR

Sicherheit und Compliance sind eine Verantwortung, mit der Sie sich teilen AWS. Dieses Modell der geteilten Verantwortung kann Ihnen helfen, Ihre betriebliche Belastung zu verringern, da AWS die Komponenten vom Host-Betriebssystem und der Virtualisierungsebene bis hin zur physischen Sicherheit der Einrichtungen, in denen EMR Cluster betrieben werden, betrieben, verwaltet und kontrolliert werden. Sie übernehmen die Verantwortung für die Verwaltung und Aktualisierung der EMR Amazon-Cluster sowie die Konfiguration der Anwendungssoftware und die AWS bereitgestellten Sicherheitskontrollen. Diese Differenzierung der Verantwortung wird allgemein als Sicherheit der Cloud und Sicherheit in der Cloud bezeichnet.

- Sicherheit der Cloud — AWS ist verantwortlich für den Schutz der Infrastruktur, AWS -Services in der sie ausgeführt wird AWS. AWS bietet Ihnen auch Dienste, die Sie sicher nutzen können. Auditoren von Drittanbietern testen und überprüfen die Effektivität unserer Sicherheitsmaßnahmen im Rahmen der AWS -Compliance-Programme regelmäßig. Weitere Informationen zu den Compliance-Programmen, die für Amazon geltenEMR, finden Sie AWS -Services unter Umfang nach Compliance-Programmen.
- Sicherheit in der Cloud — Sie sind auch dafür verantwortlich, alle erforderlichen Sicherheitskonfigurations- und Verwaltungsaufgaben zur Sicherung eines EMR Amazon-Clusters durchzuführen. Kunden, die einen EMR Amazon-Cluster bereitstellen, sind für die Verwaltung der auf den Instances installierten Anwendungssoftware und die Konfiguration der AWS bereitgestellten Funktionen wie Sicherheitsgruppen, Verschlüsselung und Zugriffskontrolle gemäß Ihren Anforderungen, geltenden Gesetzen und Vorschriften verantwortlich.

Diese Dokumentation hilft Ihnen zu verstehen, wie Sie das Modell der gemeinsamen Verantwortung bei der Nutzung von Amazon anwenden könnenEMR. Die Themen in diesem Kapitel zeigen Ihnen, wie Sie Amazon konfigurieren EMR und andere verwenden AWS -Services , um Ihre Sicherheits- und Compliance-Ziele zu erreichen.

Netzwerk- und Infrastruktursicherheit

Als verwalteter Service EMR ist Amazon durch die AWS globalen Netzwerksicherheitsverfahren geschützt, die im Whitepaper Amazon Web Services: Sicherheitsprozesse im Überblick beschrieben sind. AWS Die Dienste zum Schutz von Netzwerken und Infrastrukturen bieten Ihnen differenzierten Schutz sowohl auf Host- als auch auf Netzwerkebene. Support EMR AWS -Services und

Anwendungsfunktionen von Amazon, die Ihren Netzwerkschutz- und Compliance-Anforderungen entsprechen.

- EC2Amazon-Sicherheitsgruppen fungieren als virtuelle Firewall für EMR Amazon-Cluster-Instances und begrenzen den eingehenden und ausgehenden Netzwerkverkehr. Weitere Informationen finden Sie unter Steuern des Netzwerkverkehrs mit Sicherheitsgruppen.
- Amazon EMR Block Public Access (BPA) verhindert, dass Sie einen Cluster in einem öffentlichen Subnetz starten, wenn der Cluster über eine Sicherheitskonfiguration verfügt, die eingehenden Datenverkehr von öffentlichen IP-Adressen an einem Port zulässt. Weitere Informationen finden Sie unter Verwenden von Amazon, um den öffentlichen Zugriff zu EMR blockieren.
- Secure Shell (SSH) bietet Benutzern eine sichere Möglichkeit, eine Verbindung zur Befehlszeile auf Cluster-Instances herzustellen. Sie können SSH damit auch Weboberflächen anzeigen, die Anwendungen auf dem Master-Knoten eines Clusters hosten. Weitere Informationen finden Sie unter Verwenden eines EC2 key pair für SSH Anmeldeinformationen und Connect zu einem Cluster herstellen.

Updates für das standardmäßige Amazon Linux AMI für Amazon EMR

> **Important**
>
> EMRCluster, auf denen Amazon Linux oder Amazon Linux 2 Amazon Machine Images (AMIs) ausgeführt werden, verwenden das Standardverhalten von Amazon Linux und laden wichtige und kritische Kernel-Updates, die einen Neustart erfordern, nicht automatisch herunter und installieren sie. Dies ist dasselbe Verhalten wie bei anderen EC2 Amazon-Instances, auf denen das standardmäßige Amazon Linux ausgeführt wirdAMI. Wenn neue Amazon Linux-Softwareupdates, die einen Neustart erfordern (wie Kernel und CUDA Updates)NVIDIA, verfügbar werden, nachdem eine EMR Amazon-Version verfügbar wird, laden EMR Cluster-Instances, die standardmäßig ausgeführt werden, diese Updates AMI nicht automatisch herunter und installieren sie. Um Kernel-Updates zu erhalten, können Sie Ihr Amazon so anpassen EMR AMI, dass es das neueste Amazon Linux verwendet AMI.

Abhängig von der Sicherheit Ihrer Anwendung und der Dauer der Ausführung eines Clusters können Sie wählen, ob Sie Ihr Cluster regelmäßig neu starten, um Sicherheitsupdates anzuwenden, oder ob Sie eine Bootstrap-Aktion zum Anpassen von Paketinstallation und Updates erstellen. Sie können

außerdem Sicherheitsupdates erst testen und dann auf ausgeführten Cluster-Instances installieren. Weitere Informationen finden Sie unter Verwenden des standardmäßigen Amazon Linux AMI für Amazon EMR. Beachten Sie, dass Ihre Netzwerkkonfiguration Linux-Repositorys in Amazon S3 zulassen HTTP und HTTPS zu diesen gelangen muss, da andernfalls Sicherheitsupdates nicht erfolgreich sein werden.

AWS Identity and Access Management mit Amazon EMR

AWS Identity and Access Management (IAM) ist ein AWS Service, der einem Administrator hilft, den Zugriff auf AWS Ressourcen sicher zu kontrollieren. IAMAdministratoren kontrollieren, wer authentifiziert (angemeldet) und autorisiert werden kann (über Berechtigungen verfügt), um EMR Amazon-Ressourcen zu nutzen. IAMZu den Identitäten gehören Benutzer, Gruppen und Rollen. Eine IAM Rolle ähnelt einem IAM Benutzer, ist jedoch keiner bestimmten Person zugeordnet und soll von jedem Benutzer übernommen werden können, der Berechtigungen benötigt. Weitere Informationen finden Sie unter AWS Identity and Access Management Für Amazon EMR. Amazon EMR verwendet mehrere IAM Rollen, um Sie bei der Implementierung von Zugriffskontrollen für EMR Amazon-Cluster zu unterstützen. IAMist ein AWS Service, den Sie ohne zusätzliche Kosten nutzen können.

- IAMRolle für Amazon EMR (EMRRolle) — steuert, wie der EMR Amazon-Service in Ihrem Namen AWS -Services auf andere zugreifen kann, z. B. die Bereitstellung von EC2 Amazon-Instances beim Start des EMR Amazon-Clusters. Weitere Informationen finden Sie unter IAM Servicerollen für EMR Amazon-Berechtigungen AWS -Services und Ressourcen konfigurieren.
- IAMRolle für EC2 Cluster-Instances (EC2Instance-Profil) — eine Rolle, die jeder EC2 Instance im EMR Amazon-Cluster zugewiesen wird, wenn die Instance gestartet wird. Anwendungsprozesse, die auf dem Cluster ausgeführt werden, verwenden diese Rolle, um mit anderen zu interagieren AWS -Services, z. B. mit Amazon S3. Weitere Informationen finden Sie unter IAMRolle für EC2 Cluster-Instances.
- IAMRolle für Anwendungen (Runtime-Rolle) — eine IAM Rolle, die Sie angeben können, wenn Sie einen Job oder eine Anfrage an einen EMR Amazon-Cluster senden. Der Job oder die Abfrage, die Sie an Ihren EMR Amazon-Cluster senden, verwendet die Runtime-Rolle, um auf AWS Ressourcen wie Objekte in Amazon S3 zuzugreifen. Sie können Runtime-Rollen bei Amazon EMR für Spark- und Hive-Jobs angeben. Mithilfe von Runtime-Rollen können Sie Jobs, die auf demselben Cluster ausgeführt werden, mithilfe verschiedener IAM Rollen isolieren. Weitere Informationen finden Sie unter IAMRolle als Runtime-Rolle bei Amazon verwenden EMR.

Personalidentitäten beziehen sich auf Benutzer, in denen Workloads erstellt oder ausgeführt werden. AWS Amazon EMR bietet folgenden Support für Personalidentitäten:

- AWS IAMIdentity Center (Idc) wird AWS -Service für die Verwaltung des Benutzerzugriffs auf AWS Ressourcen empfohlen. Es handelt sich um einen zentralen Ort, an dem Sie Ihren Mitarbeitern Identitäten zuweisen und so konsistenten Zugriff auf mehrere AWS Konten und Anwendungen haben können. Amazon EMR unterstützt die Identitäten von Mitarbeitern durch vertrauenswürdige Weitergabe von Identitäten. Mit der Funktion zur Weitergabe vertrauenswürdiger Identitäten kann sich ein Benutzer bei der Anwendung anmelden, und diese Anwendung kann die Identität des Benutzers an andere AWS -Services weitergeben, um den Zugriff auf Daten oder Ressourcen zu autorisieren. Weitere Informationen finden Sie unter Unterstützung für AWS IAMIdentity Center bei Amazon aktivierenEMR.

  Das Lightweight Directory Access Protocol (LDAP) ist ein offenes, herstellerneutrales, branchenübliches Anwendungsprotokoll für den Zugriff auf und die Verwaltung von Informationen über Benutzer, Systeme, Dienste und Anwendungen über das Netzwerk. LDAPwird häufig für die Benutzerauthentifizierung gegenüber Unternehmensidentitätsservern wie Active Directory (AD) und Open verwendet. LDAP Durch die Aktivierung LDAP mit EMR Clustern ermöglichen Sie es Benutzern, ihre vorhandenen Anmeldeinformationen für die Authentifizierung und den Zugriff auf Cluster zu verwenden. Weitere Informationen finden Sie unter Support für LDAP mit Amazon aktivieren EMR.

  Kerberos ist ein Netzwerkauthentifizierungsprotokoll, das entwickelt wurde, um mithilfe von Secret-Key-Kryptografie eine starke Authentifizierung für Client-/Serveranwendungen zu ermöglichen. Wenn Sie Kerberos verwenden, EMR konfiguriert Amazon Kerberos für die Anwendungen, Komponenten und Subsysteme, die es auf dem Cluster installiert, sodass sie sich gegenseitig authentifizieren. Um auf einen Cluster mit konfiguriertem Kerberos zuzugreifen, muss ein Kerberos-Prinzipal im Kerberos-Domänencontroller () vorhanden sein. KDC Weitere Informationen finden Sie unter Unterstützung für Kerberos bei Amazon aktivieren. EMR

Cluster mit einem Mandanten und mehreren Mandanten

Ein Cluster ist standardmäßig für einen einzelnen Mandanten mit dem EC2 Instanzprofil als Identität konfiguriert. IAM In einem Single-Tenant-Cluster hat jeder Job vollen und vollständigen Zugriff auf den Cluster, und der Zugriff auf alle AWS -Services Ressourcen erfolgt auf der Grundlage des EC2 Instanzprofils. In einem Multi-Tenant-Cluster sind die Mandanten voneinander isoliert und die Mandanten haben keinen vollen und vollständigen Zugriff auf die Cluster und EC2 Instanzen des

Clusters. Bei Clustern mit mehreren Mandanten werden entweder die Runtime-Rollen oder die Belegschaft identifiziert. In einem Multi-Tenant-Cluster können Sie auch die Unterstützung für eine differenzierte Zugriffskontrolle (FGAC) über AWS Lake Formation oder Apache Ranger aktivieren. Bei einem Cluster, der über Runtime-Rollen verfügt oder der Zugriff auf das EC2 Instanzprofil FGAC aktiviert ist, ist der Zugriff über iptables ebenfalls deaktiviert.

> **Important**
>
> Jeder Benutzer, der Zugriff auf einen Single-Tenant-Cluster hat, kann jede Software auf dem Linux-Betriebssystem (OS) installieren, von Amazon installierte Softwarekomponenten ändern oder entfernen EMR und sich auf die EC2 Instances auswirken, die Teil des Clusters sind. Wenn Sie sicherstellen möchten, dass Benutzer keine Konfigurationen eines EMR Amazon-Clusters installieren oder ändern können, empfehlen wir Ihnen, Multi-Tenancy für den Cluster zu aktivieren. Sie können Multi-Tenancy auf einem Cluster aktivieren, indem Sie die Unterstützung für Runtime Role, AWS IAM Identity Center, Kerberos oder aktivieren. LDAP

Datenschutz

Mit können Sie Ihre Daten kontrollieren AWS, indem Sie Tools verwenden AWS -Services , mit denen Sie bestimmen, wie die Daten gesichert sind und wer Zugriff darauf hat. Mit Diensten wie AWS Identity and Access Management (IAM) können Sie den Zugriff auf AWS -Services und Ressourcen sicher verwalten. AWS CloudTrail ermöglicht Erkennung und Prüfung. Amazon EMR macht es Ihnen leicht, ruhende Daten in Amazon S3 zu verschlüsseln, indem Sie Schlüssel verwenden, die entweder von Ihnen verwaltet werden AWS oder vollständig von Ihnen verwaltet werden. Amazon unterstützt EMR auch die Aktivierung der Verschlüsselung für Daten während der Übertragung. Weitere Informationen finden Sie unter Verschlüsseln von Daten im Ruhezustand und bei der Übertragung.

Datenzugriffskontrolle

Mit der Datenzugriffskontrolle können Sie steuern, auf welche Daten eine IAM Identität oder eine Mitarbeiteridentität zugreifen kann. Amazon EMR unterstützt die folgenden Zugriffskontrollen:

- IAMidentitätsbasierte Richtlinien — verwalten Sie Berechtigungen für IAM Rollen, die Sie bei Amazon verwenden. EMR IAMRichtlinien können mit Tagging kombiniert werden, um den Zugriff auf Basis zu kontrollieren. cluster-by-cluster Weitere Informationen finden Sie unter AWS Identity and Access Management Für Amazon EMR.

- AWS Lake Formationzentralisiert die Rechteverwaltung Ihrer Daten und erleichtert die gemeinsame Nutzung innerhalb Ihrer Organisation und extern. Sie können Lake Formation verwenden, um einen detaillierten Zugriff auf Spaltenebene auf Datenbanken und Tabellen im Glue-Datenkatalog zu ermöglichen. AWS Weitere Informationen finden Sie unter AWS Lake Formation Mit Amazon verwendenEMR.
- Amazon S3 S3-Zugriff gewährt Zuordnungsidentitäten, Identitäten in Verzeichnissen wie Active Directory oder AWS Identity and Access Management (IAM) -Prinzipalen Datensätzen in S3 zuzuordnen. Darüber hinaus ermöglicht der S3-Zugriff die Protokollierung der Identität des Endbenutzers und der Anwendung, die für den Zugriff auf S3-Daten verwendet wird. AWS CloudTrail Weitere Informationen finden Sie unter Verwenden von Amazon S3 S3-Zugriffsberechtigungen mit Amazon EMR.
- Apache Ranger ist ein Framework zur Aktivierung, Überwachung und Verwaltung einer umfassenden Datensicherheit auf der gesamten Hadoop-Plattform. Amazon EMR unterstützt eine auf Apache Ranger basierende, feinkörnige Zugriffskontrolle für Apache Hive Metastore und Amazon S3. Weitere Informationen finden Sie unter Integrieren von Apache Ranger mit Amazon EMR.

Sicherheitskonfigurationen zum Einrichten der Cluster-Sicherheit verwenden

Sie können EMR Amazon-Sicherheitskonfigurationen verwenden, um Datenverschlüsselung, Kerberos-Authentifizierung und Amazon S3-Autorisierung für Ihre Cluster EMRFS zu konfigurieren. Zunächst erstellen Sie eine Sicherheitskonfiguration. Anschließend kann die Sicherheitskonfiguration bei der Erstellung von wiederholt verwendet werden.

Sie können die AWS Management Console, die AWS Command Line Interface (AWS CLI) oder die verwenden, um AWS SDKs Sicherheitskonfigurationen zu erstellen. Sie können auch eine AWS CloudFormation Vorlage verwenden, um eine Sicherheitskonfiguration zu erstellen. Weitere Informationen finden Sie im AWS CloudFormation Benutzerhandbuch und in der Vorlagenreferenz für AWS::EMR:: SecurityConfiguration.

Themen

- Eine Sicherheitskonfiguration erstellen
- Angabe einer Sicherheitskonfiguration für einen Cluster

Eine Sicherheitskonfiguration erstellen

Dieses Thema behandelt allgemeine Verfahren zum Erstellen einer Sicherheitskonfiguration mit der EMR Amazon-Konsole und dem AWS CLI, gefolgt von einer Referenz zu den Parametern, die Verschlüsselung, Authentifizierung und IAM Rollen für umfassenEMRFS. Weitere Informationen zu diesen Funktionen finden Sie in den folgenden Themen:

- Verschlüsseln von Daten im Ruhezustand und im Transit
- Verwenden Sie Kerberos für die Authentifizierung bei Amazon EMR
- IAMRollen für EMRFS Anfragen an Amazon S3 konfigurieren

So erstellen Sie eine Sicherheitskonfiguration mithilfe der Konsole

1. Öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie im Navigationsbereich Security Configurations (Sicherheitskonfigurationen), Create security configuration (Sicherheitskonfiguration erstellen) aus.
3. Geben Sie in Name (Name) einen Namen für die Sicherheitskonfiguration ein.
4. Wählen Sie Optionen für Verschlüsselung, und Authentifizierung aus wie in den folgenden Abschnitten beschrieben. Wählen Sie anschließend Erstellen aus.

Um eine Sicherheitskonfiguration mit dem zu erstellen AWS CLI

- Verwenden Sie den Befehl `create-security-configuration` wie im folgenden Beispiel gezeigt.
  - Wählen Sie in der &Snowconsole; Ihren Auftrag aus der Tabelle. *`SecConfigName`*, geben Sie den Namen der Sicherheitskonfiguration an. Dies ist der Name, den Sie angeben, wenn Sie einen Cluster erstellen, der diese Sicherheitskonfiguration verwendet.
  - Geben Sie für *`SecConfigDef`* eine JSON Inline-Struktur oder den Pfad zu einer lokalen JSON Datei an, z. *`file://MySecConfig.json`* B. Die JSON Parameter definieren Optionen für Verschlüsselung, IAMRollen für EMRFS den Zugriff auf Amazon S3 und Authentifizierung, wie in den folgenden Abschnitten beschrieben.

```
aws emr create-security-configuration --name "SecConfigName" --security-
configuration SecConfigDef
```

Datenverschlüsselung konfigurieren

Bevor Sie die Verschlüsselung in einer Sicherheitskonfiguration konfigurieren, erstellen Sie die Schlüssel und Zertifikate, die für die Verschlüsselung verwendet werden. Weitere Informationen erhalten Sie unter Bereitstellung von Schlüsseln für die Verschlüsselung ruhender Daten bei Amazon EMR und Bereitstellung von Zertifikaten für die Verschlüsselung von Daten während der Übertragung mit EMR Amazon-Verschlüsselung.

Beim Erstellen einer Sicherheits-Konfiguration legen Sie zwei Verschlüsselungsoptionen fest: Verschlüsselung von Daten während der Übertragung und im Ruhezustand. Zu den Optionen für die Verschlüsselung von Daten im Ruhezustand gehören sowohl Amazon S3 mit EMRFS Verschlüsselung als auch Verschlüsselung auf lokalen Festplatten. Verschlüsselungsoptionen bei der Übertragung aktivieren die Open-Source-Verschlüsselungsfunktionen für bestimmte Anwendungen, die Transport Layer Security () unterstützen. TLS Die Optionen für die Verschlüsselung während der Übertragung und im Ruhezustand können gemeinsam oder einzeln aktiviert werden. Weitere Informationen finden Sie unter Verschlüsseln von Daten im Ruhezustand und im Transit.

> **Note**
>
> Bei der Nutzung AWS KMS fallen Gebühren für die Speicherung und Verwendung von Verschlüsselungsschlüsseln an. Weitere Informationen finden Sie unter AWS KMS - Preisgestaltung.

Angeben von Verschlüsselungsoptionen mit der Konsole

Wählen Sie Optionen unter Encryption (Verschlüsselung) entsprechend den folgenden Anleitungen aus.

- Wählen Sie Optionen unter At rest encryption (Verschlüsselung im Ruhezustand) aus, um innerhalb des Dateisystems gespeicherte Daten zu verschlüsseln.

  Sie können Daten in Amazon S3, auf lokalen Datenträgern oder in beiden Speichern verschlüsseln.
- Wählen Sie unter S3-Datenverschlüsselung für den Verschlüsselungsmodus einen Wert aus, um festzulegen, mit EMRFS welcher Methode Amazon S3-Daten EMR verschlüsselt werden.

  Der nächste Schritt hängt von dem von Ihnen gewählten Verschlüsselungsmodus ab:

  - SSE-S3

Angaben zur serverseitigen Verschlüsselung mit von Amazon S3 verwalteten Schlüsseln. Sie müssen nicht mehr tun, da Amazon S3 die Handhabung der Schlüssel für Sie übernimmt.

- SSE- KMS oder CSE - KMS

  Gibt serverseitige Verschlüsselung mit AWS KMS verwalteten Schlüsseln (SSE-KMS) oder clientseitige Verschlüsselung mit AWS KMS-verwalteten Schlüsseln (-) an. CSE KMS Wählen Sie für AWS KMS key einen Schlüssel aus. Der Schlüssel muss in derselben Region wie Ihr Cluster existieren. EMR Schlüsselanforderungen finden Sie unter AWS KMS keys Für die Verschlüsselung verwenden.

- CSE-Benutzerdefiniert

  Gibt die clientseitige Verschlüsselung unter Verwendung eines benutzerdefinierten clientseitigen Stammschlüssels (-custom) an. CSE Geben Sie für das S3-Objekt den Speicherort Ihrer benutzerdefinierten JAR Key-Provider-Datei in Amazon S3 ARN oder Amazon S3 ein. Geben Sie dann für Key Provider Class den vollständigen Klassennamen einer Klasse ein, die in Ihrer Anwendung deklariert ist, die die Schnittstelle implementiert. EncryptionMaterialsProvider

- Wählen Sie unter Local disk encryption (Lokale Laufwerksverschlüsselung) einen Wert für Key provider type (Schlüsselanbietertyp) aus.

  - AWS KMS key

    Wählen Sie diese Option, um eine AWS KMS key anzugeben. Wählen Sie für AWS KMS key einen Schlüssel aus. Der Schlüssel muss in derselben Region wie Ihr EMR Cluster existieren. Weitere Informationen zu den Anforderungen für Schlüssel finden Sie unter AWS KMS keys Für die Verschlüsselung verwenden.

    EBSVerschlüsselung

    Wenn Sie dies AWS KMS als Ihren Schlüsselanbieter angeben, können Sie die EBS Verschlüsselung aktivieren, um EBS Root-Geräte und Speichervolumes zu verschlüsseln. Um diese Option zu aktivieren, müssen Sie der EMR Amazon-Servicerolle die von `EMR_DefaultRole` Ihnen angegebenen Berechtigungen zur Verwendung der von AWS KMS key Ihnen angegebenen Rechte erteilen. Weitere Informationen zu den Anforderungen für Schlüssel finden Sie unter Aktivierung der EBS Verschlüsselung durch Bereitstellung zusätzlicher Berechtigungen für KMS Schlüssel.

  - Custom (Benutzerdefiniert)

Wählen Sie diese Option aus, um einen benutzerdefinierten Schlüsselanbieter festzulegen. Geben Sie für das S3-Objekt den Speicherort Ihrer benutzerdefinierten JAR Key-Provider-Datei in Amazon S3 ARN oder Amazon S3 ein. Geben Sie für Key Provider Class den vollständigen Klassennamen einer Klasse ein, die in Ihrer Anwendung deklariert ist, die die EncryptionMaterialsProvider Schnittstelle implementiert. Der Klassenname, den Sie hier angeben, muss sich von dem für CSE -Custom angegebenen Klassennamen unterscheiden.

- Wählen Sie Verschlüsselung bei der Übertragung, um die TLS Open-Source-Verschlüsselungsfunktionen für Daten während der Übertragung zu aktivieren. Wählen Sie anhand der folgenden Anleitungen einen Certificate provider type (Zertifikatanbietertyp) aus:
  - PEM

    Wählen Sie diese Option, um PEM Dateien zu verwenden, die Sie in einer ZIP-Datei bereitstellen. In der ZIP-Datei sind zwei Artefakte erforderlich: privateKey .pem und certificateChain .pem. Eine dritte Datei, trustedCertificates .pem, ist optional. Details dazu finden Sie unter Bereitstellung von Zertifikaten für die Verschlüsselung von Daten während der Übertragung mit EMR Amazon-Verschlüsselung. Geben Sie für das S3-Objekt den Speicherort des Zip-Felds in Amazon S3 ARN oder Amazon S3 an.
  - Custom (Benutzerdefiniert)

    Wählen Sie diese Option, um einen benutzerdefinierten Zertifikatsanbieter anzugeben, und geben Sie dann für S3-Objekt den Speicherort Ihrer benutzerdefinierten JAR Zertifikatsanbieterdatei in Amazon S3 ARN oder Amazon S3 ein. Geben Sie für Key Provider Class den vollständigen Klassennamen einer Klasse ein, die in Ihrer Anwendung deklariert ist, die die TLSArtifactsProvider Schnittstelle implementiert.

Angeben von Verschlüsselungsoptionen mit dem AWS CLI

In den folgenden Abschnitten werden Beispielszenarien zur Veranschaulichung von Wohlgeformten --security-configuration JSON für verschiedene Konfigurationen und Schlüsselanbieter verwendet, gefolgt von einer Referenz für die JSON Parameter und die entsprechenden Werte.

Beispiel der Datenverschlüsselungsoptionen während der Übertragung

Das nachstehende Beispiel veranschaulicht das folgende Szenario:

- Die Verschlüsselung von Daten während der Übertragung ist aktiviert und die Verschlüsselung von Daten im Ruhezustand ist deaktiviert.

- Eine ZIP-Datei mit Zertifikaten in Amazon S3 wird als Schlüsselanbieter verwendet (die Zertifikatanforderungen finden Sie unter Bereitstellung von Zertifikaten für die Verschlüsselung von Daten während der Übertragung mit EMR Amazon-Verschlüsselung).

```
aws emr create-security-configuration --name "MySecConfig" --security-configuration '{
 "EncryptionConfiguration": {
  "EnableInTransitEncryption": true,
  "EnableAtRestEncryption": false,
  "InTransitEncryptionConfiguration": {
   "TLSCertificateConfiguration": {
    "CertificateProviderType": "PEM",
    "S3Object": "s3://MyConfigStore/artifacts/MyCerts.zip"
   }
  }
 }
}'
```

Das nachstehende Beispiel veranschaulicht das folgende Szenario:

- Die Verschlüsselung von Daten während der Übertragung ist aktiviert und die Verschlüsselung von Daten im Ruhezustand ist deaktiviert.
- Ein benutzerdefinierter Schlüsselanbieter wird verwendet (die Zertifikatanforderungen finden Sie unter Bereitstellung von Zertifikaten für die Verschlüsselung von Daten während der Übertragung mit EMR Amazon-Verschlüsselung).

```
aws emr create-security-configuration --name "MySecConfig" --security-configuration '{
 "EncryptionConfiguration": {
  "EnableInTransitEncryption": true,
  "EnableAtRestEncryption": false,
  "InTransitEncryptionConfiguration": {
   "TLSCertificateConfiguration": {
    "CertificateProviderType": "Custom",
    "S3Object": "s3://MyConfig/artifacts/MyCerts.jar",
    "CertificateProviderClass": "com.mycompany.MyCertProvider"
   }
  }
  }
```

```
}'
```

Beispiel der Datenverschlüsselungsoptionen im Ruhezustand

Das nachstehende Beispiel veranschaulicht das folgende Szenario:

- Die Verschlüsselung von Daten während der Übertragung ist deaktiviert und die Verschlüsselung von Daten im Ruhezustand ist aktiviert.
- SSE-S3 wird für die Amazon S3 S3-Verschlüsselung verwendet.
- Die lokale Festplattenverschlüsselung wird AWS KMS als Schlüsselanbieter verwendet.

```
aws emr create-security-configuration --name "MySecConfig" --security-configuration '{
 "EncryptionConfiguration": {
  "EnableInTransitEncryption": false,
  "EnableAtRestEncryption": true,
  "AtRestEncryptionConfiguration": {
   "S3EncryptionConfiguration": {
    "EncryptionMode": "SSE-S3"
   },
   "LocalDiskEncryptionConfiguration": {
    "EncryptionKeyProviderType": "AwsKms",
    "AwsKmsKey": "arn:aws:kms:us-
east-1:123456789012:key/12345678-1234-1234-1234-123456789012"
   }
  }
  }
}'
```

Das nachstehende Beispiel veranschaulicht das folgende Szenario:

- Die Verschlüsselung von Daten während der Übertragung ist aktiviert und verweist auf eine Zip-Datei mit PEM Zertifikaten in Amazon S3 unter Verwendung von. ARN
- SSE- KMS wird für die Amazon S3 S3-Verschlüsselung verwendet.
- Die lokale Festplattenverschlüsselung wird AWS KMS als Schlüsselanbieter verwendet.

```
aws emr create-security-configuration --name "MySecConfig" --security-configuration '{
 "EncryptionConfiguration": {
  "EnableInTransitEncryption": true,
```

```
  "EnableAtRestEncryption": true,
  "InTransitEncryptionConfiguration": {
   "TLSCertificateConfiguration": {
    "CertificateProviderType": "PEM",
    "S3Object": "arn:aws:s3:::MyConfigStore/artifacts/MyCerts.zip"
   }
  },
  "AtRestEncryptionConfiguration": {
   "S3EncryptionConfiguration": {
    "EncryptionMode": "SSE-KMS",
    "AwsKmsKey": "arn:aws:kms:us-
east-1:123456789012:key/12345678-1234-1234-1234-123456789012"
   },
   "LocalDiskEncryptionConfiguration": {
    "EncryptionKeyProviderType": "AwsKms",
    "AwsKmsKey": "arn:aws:kms:us-
east-1:123456789012:key/12345678-1234-1234-1234-123456789012"
   }
  }
 }
}'
```

Das nachstehende Beispiel veranschaulicht das folgende Szenario:

- Die Verschlüsselung von Daten während der Übertragung ist aktiviert und verweist auf eine ZIP-Datei mit PEM Zertifikaten in Amazon S3.
- CSE- KMS wird für die Amazon S3 S3-Verschlüsselung verwendet.
- Die lokale Festplattenverschlüsselung verwendet einen benutzerdefinierten Schlüsselanbieter, auf den von its verwiesen wirdARN.

```
aws emr create-security-configuration --name "MySecConfig" --security-configuration '{
 "EncryptionConfiguration": {
  "EnableInTransitEncryption": true,
  "EnableAtRestEncryption": true,
  "InTransitEncryptionConfiguration": {
   "TLSCertificateConfiguration": {
    "CertificateProviderType": "PEM",
    "S3Object": "s3://MyConfigStore/artifacts/MyCerts.zip"
   }
  },
```

```
  "AtRestEncryptionConfiguration": {
   "S3EncryptionConfiguration": {
    "EncryptionMode": "CSE-KMS",
    "AwsKmsKey": "arn:aws:kms:us-
east-1:123456789012:key/12345678-1234-1234-1234-123456789012"
   },
   "LocalDiskEncryptionConfiguration": {
    "EncryptionKeyProviderType": "Custom",
    "S3Object": "arn:aws:s3:::artifacts/MyKeyProvider.jar",
    "EncryptionKeyProviderClass": "com.mycompany.MyKeyProvider"
   }
  }
 }
}'
```

Das nachstehende Beispiel veranschaulicht das folgende Szenario:

- Die Verschlüsselung von Daten während der Übertragung anhand eines benutzerdefinierten Schlüsselanbieters ist aktiviert.
- CSE-Custom wird für Amazon S3 S3-Daten verwendet.
- Die lokale Laufwerksverschlüsselung verwendet einen benutzerdefinierten Schlüsselanbieter.

```
aws emr create-security-configuration --name "MySecConfig" --security-configuration '{
 "EncryptionConfiguration": {
  "EnableInTransitEncryption": "true",
  "EnableAtRestEncryption": "true",
  "InTransitEncryptionConfiguration": {
   "TLSCertificateConfiguration": {
    "CertificateProviderType": "Custom",
    "S3Object": "s3://MyConfig/artifacts/MyCerts.jar",
    "CertificateProviderClass": "com.mycompany.MyCertProvider"
   }
  },
  "AtRestEncryptionConfiguration": {
   "S3EncryptionConfiguration": {
    "EncryptionMode": "CSE-Custom",
    "S3Object": "s3://MyConfig/artifacts/MyCerts.jar",
    "EncryptionKeyProviderClass": "com.mycompany.MyKeyProvider"
    },
   "LocalDiskEncryptionConfiguration": {
```

```
    "EncryptionKeyProviderType": "Custom",
    "S3Object": "s3://MyConfig/artifacts/MyCerts.jar",
    "EncryptionKeyProviderClass": "com.mycompany.MyKeyProvider"
   }
  }
 }
}'
```

Das nachstehende Beispiel veranschaulicht das folgende Szenario:

- Die Verschlüsselung von Daten während der Übertragung ist deaktiviert und die Verschlüsselung von Daten im Ruhezustand ist aktiviert.
- Die Amazon S3 S3-Verschlüsselung ist mit SSE - aktiviertKMS.
- Es werden mehrere AWS KMS Schlüssel verwendet, einer pro S3-Bucket, und Verschlüsselungsausnahmen werden auf diese einzelnen S3-Buckets angewendet.
- Die lokale Laufwerksverschlüsselung ist deaktiviert.

```
aws emr create-security-configuration --name "MySecConfig" --security-configuration '{
   "EncryptionConfiguration": {
     "AtRestEncryptionConfiguration": {
             "S3EncryptionConfiguration": {
           "EncryptionMode": "SSE-KMS",
           "AwsKmsKey": "arn:aws:kms:us-
east-1:123456789012:key/12345678-1234-1234-1234-123456789012",
           "Overrides": [
              {
                "BucketName": "sse-s3-bucket-name",
                "EncryptionMode": "SSE-S3"
              },
              {
                "BucketName": "cse-kms-bucket-name",
                "EncryptionMode": "CSE-KMS",
                "AwsKmsKey": "arn:aws:kms:us-
east-1:123456789012:key/12345678-1234-1234-1234-123456789012"
              },
              {
                "BucketName": "sse-kms-bucket-name",
                "EncryptionMode": "SSE-KMS",
                "AwsKmsKey": "arn:aws:kms:us-
east-1:123456789012:key/12345678-1234-1234-1234-123456789012"
              }
```

```
            ]
            }
          },
     "EnableInTransitEncryption": false,
      "EnableAtRestEncryption": true
  }
}'
```

Das nachstehende Beispiel veranschaulicht das folgende Szenario:

- Die Verschlüsselung von Daten während der Übertragung ist deaktiviert und die Verschlüsselung von Daten im Ruhezustand ist aktiviert.
- Die Amazon S3 S3-Verschlüsselung ist mit SSE -S3 aktiviert und die lokale Festplattenverschlüsselung ist deaktiviert.

```
aws emr create-security-configuration --name "MyS3EncryptionConfig" --security-
configuration '{
    "EncryptionConfiguration": {
        "EnableInTransitEncryption": false,
        "EnableAtRestEncryption": true,
        "AtRestEncryptionConfiguration": {
            "S3EncryptionConfiguration": {
                "EncryptionMode": "SSE-S3"
            }
        }
     }
}'
```

Das nachstehende Beispiel veranschaulicht das folgende Szenario:

- Die Verschlüsselung von Daten während der Übertragung ist deaktiviert und die Verschlüsselung von Daten im Ruhezustand ist aktiviert.
- Die lokale Festplattenverschlüsselung ist AWS KMS als Schlüsselanbieter aktiviert und die Amazon S3 S3-Verschlüsselung ist deaktiviert.

```
aws emr create-security-configuration --name "MyLocalDiskEncryptionConfig" --security-
configuration '{
    "EncryptionConfiguration": {
        "EnableInTransitEncryption": false,
```

```
        "EnableAtRestEncryption": true,
        "AtRestEncryptionConfiguration": {
            "LocalDiskEncryptionConfiguration": {
                "EncryptionKeyProviderType": "AwsKms",
                "AwsKmsKey": "arn:aws:kms:us-
east-1:123456789012:key/12345678-1234-1234-1234-123456789012"
            }
        }
    }
}'
```

Das nachstehende Beispiel veranschaulicht das folgende Szenario:

- Die Verschlüsselung von Daten während der Übertragung ist deaktiviert und die Verschlüsselung von Daten im Ruhezustand ist aktiviert.
- Die lokale Festplattenverschlüsselung ist AWS KMS als Schlüsselanbieter aktiviert und die Amazon S3 S3-Verschlüsselung ist deaktiviert.
- EBSVerschlüsselung ist aktiviert.

```
aws emr create-security-configuration --name "MyLocalDiskEncryptionConfig" --security-
configuration '{
    "EncryptionConfiguration": {
        "EnableInTransitEncryption": false,
        "EnableAtRestEncryption": true,
        "AtRestEncryptionConfiguration": {
            "LocalDiskEncryptionConfiguration": {
                "EnableEbsEncryption": true,
                "EncryptionKeyProviderType": "AwsKms",
                "AwsKmsKey": "arn:aws:kms:us-
east-1:123456789012:key/12345678-1234-1234-1234-123456789012"
            }
        }
    }
}'
```

Das nachstehende Beispiel veranschaulicht das folgende Szenario:

SSE- EMR - WAL wird für die EMR WAL Verschlüsselung verwendet

```
aws emr create-security-configuration --name "MySecConfig" \
    --security-configuration '{
```

```
        "EncryptionConfiguration": {
            "EMRWALEncryptionConfiguration":{ },
            "EnableInTransitEncryption":false, "EnableAtRestEncryption":false
        }
    }'
```

`EnableInTransitEncryption`und könnte `EnableAtRestEncryption` immer noch wahr sein, wenn Sie die entsprechende Verschlüsselung aktivieren möchten.

Das nachstehende Beispiel veranschaulicht das folgende Szenario:

- SSE- KMS - WAL wird für die EMR WAL Verschlüsselung verwendet
- Serverseitige Verschlüsselung wird AWS Key Management Service als Schlüsselanbieter verwendet

```
aws emr create-security-configuration --name "MySecConfig" \
    --security-configuration '{
        "EncryptionConfiguration": {
            "EMRWALEncryptionConfiguration":{
                "AwsKmsKey":"arn:aws:kms:us-
east-1:123456789012:key/12345678-1234-1234-1234-123456789012"
                },
            "EnableInTransitEncryption":false, "EnableAtRestEncryption":false
        }
    }'
```

`EnableInTransitEncryption`und könnte `EnableAtRestEncryption` immer noch wahr sein, wenn Sie die entsprechende Verschlüsselung aktivieren möchten.

JSONReferenz für Verschlüsselungseinstellungen

In der folgenden Tabelle sind die JSON Parameter für die Verschlüsselungseinstellungen aufgeführt und die akzeptablen Werte für jeden Parameter beschrieben.

Parameter	Beschreibung
`"EnableInTransitEncryption" : true | false`	Geben Sie `true` an, um die Verschlüsselung der Daten während der Übertragung zu aktivieren, und `false`, um sie zu deaktivieren. Wenn der Parameter nicht definiert wird, gilt

Parameter	Beschreibung
	`false` und die Verschlüsselung von Daten während der Übertragung ist deaktiviert.
`"EnableAtRestEncryption": true | false`	Geben Sie `true` an, um die Verschlüsselung der Daten im Ruhezustand zu aktivieren, und `false`, um sie zu deaktivieren. Wenn der Parameter nicht definiert wird, gilt `false` und die Verschlüsselung von Daten im Ruhezustand ist deaktiviert.
Parameter für die Verschlüsselung während der Übertragung	
`"InTransitEncryptionConfiguration" :`	Gibt eine Sammlung von Werten für die Verschlüsselung von Daten während der Übertragung an, wenn `EnableInTransitEncryption` `true` ist.
`"CertificateProviderType": "PEM" | "Custom"`	Gibt an, ob `PEM`-Zertifikate verwendet werden sollen, auf die über eine ZIP-Datei oder über einen `Custom` Zertifikatsanbieter verwiesen wird. Wenn angegeben, `S3Object` muss `PEM` es sich um einen Verweis auf den Speicherort einer ZIP-Datei mit den Zertifikaten in Amazon S3 handeln. Wenn Benutzerdefiniert angegeben ist, `S3Object` muss es sich um einen Verweis auf den Speicherort einer JAR Datei in Amazon S3 handeln, gefolgt von einem `CertificateProviderClass` Eintrag.

Parameter	Beschreibung
`"S3Object" : " ZipLocation "| "JarLocation "`	Stellt den Speicherort in Amazon S3 für eine ZIP-Datei bereit, wenn `PEM` angegeben , oder für eine JAR Datei, wenn `Custom` angegeben. Das Format kann ein Pfad (zum Beispiel`s3://MyConfig/artifacts/CertFiles.zip`) oder ein ARN (zum Beispiel,) sein`arn:aws:s3:::Code/MyCertProvider.jar)` . Wenn eine ZIP-Datei ausgewählt wurde, muss sie Dateien enthalten, deren Namen `privateKey.pem` und `certificateChain.pem` sind. Eine Datei mit dem Namen `trustedCertificates.pem` ist optional.
`"CertificateProviderClass" : "MyClassID "`	Nur erforderlich, wenn für angegeben `Custom` ist`CertificateProviderType` . *`MyClassID`* gibt einen vollständigen Klassennamen an, der in der JAR Datei deklariert ist, die die TLSArtifactsProvider Schnittstelle implementiert. Beispiel, `com.mycompany.MyCertProvider` .
Parameter für die Verschlüsselung im Ruhezustand	
`"AtRestEncryptionConfiguration" :`	Gibt eine Sammlung von Werten für die Verschlüsselung im Ruhezustand an, wenn dies der `EnableAtRestEncryption` Fall ist`true`, einschließlich Amazon S3 S3-Verschlüsselung und Verschlüsselung lokaler Festplatten.
Amazon S3 S3-Verschlüsselungsparameter	

Parameter	Beschreibung
`"S3EncryptionConfiguration" :`	Gibt eine Sammlung von Werten an, die für die Amazon S3 S3-Verschlüsselung mit dem Amazon EMR File System (EMRFS) verwendet werden.
`"EncryptionMode" : "SSE-S3" | "SSE-KMS" | "CSE-KMS" | "CSE-Custom"`	Gibt den Typ der zu verwendenden Amazon S3 S3-Verschlüsselung an. Wenn `SSE-S3` angegeben, sind keine weiteren Amazon S3 S3-Verschlüsselungswerte erforderlich. Wenn entweder `SSE-KMS` oder angegeben `CSE-KMS` ist, AWS KMS key ARN muss an als `AwsKmsKey` Wert angegeben werden. Wenn `CSE-Custom` ausgewählt wurde, müssen `S3Object`- und `EncryptionKeyProviderClass` -Werte angegeben werden.
`"AwsKmsKey" : "` *`MyKeyARN`*`"`	Nur erforderlich, wenn entweder `SSE-KMS` oder für angegeben `CSE-KMS` ist`EncryptionMode` . *`MyKeyARN`*muss ein vollständig ARN spezifizierter Schlüssel sein (z. B.`arn:aws:kms:us-east-1:123456789012:key/12345678-1234-1234-1234-123456789012`).
`"S3Object" : "`*`JarLocation`*` "`	Nur erforderlich, wenn für angegeben `CSE-Custom` ist`CertificateProviderType` . *`JarLocation`* gibt den Speicherort in Amazon S3 für eine JAR Datei an. Das Format kann ein Pfad (zum Beispiel`s3://MyConfig/artifacts/MyKeyProvider.jar`) oder ein ARN (zum Beispiel,) sein`arn:aws:s3:::Code/MyKeyProvider.jar)` .

Parameter	Beschreibung
`"EncryptionKeyProviderClass" : "`*`MyS3KeyClassID`*` "`	Nur erforderlich, wenn für angegeben `CSE-Custom` istEncryptionMode .*`MyS3KeyClassID`* gibt den vollständigen Klassennamen einer Klasse an, die in der Anwendung deklariert ist, die die EncryptionMaterialsProvider Schnittstelle implementiert; zum Beispiel*`com.mycompany.MyS3KeyProvider`* .
Parameter für Verschlüsselung auf dem lokalen Datenträger	
`"LocalDiskEncryptionConfiguration"`	Gibt die Schlüsselanbieter und die entsprechenden Werte an, die für die lokale Laufwerksverschlüsselung verwendet werden müssen.
`"EnableEbsEncryption": true | false`	Geben Sie `true` an, ob die EBS Verschlüsselung aktiviert werden soll. EBSDurch die Verschlüsselung werden das EBS Root-Gerät-Volume und die angeschlossenen Speichervolumes verschlüsselt. Um die EBS Verschlüsselung zu verwenden, müssen Sie `AwsKms` als Ihr `EncryptionKeyProviderType` angeben.
`"EncryptionKeyProviderType": "AwsKms" | "Custom"`	Gibt den Schlüsselanbieter an. Wenn `AwsKms` angegeben, ARN muss ein KMS Schlüssel als `AwsKmsKey` Wert angegeben werden. Wenn **`Custom`** ausgewählt wurde, müssen `S3Object`- und `EncryptionKeyProviderClass` -Werte angegeben werden.

Parameter	Beschreibung
`"AwsKmsKey : " MyKeyARN"`	Nur erforderlich, wenn für angegeben `AwsKms` istType. *`MyKeyARN`*muss ein vollständig ARN spezifizierter Schlüssel sein (z. B.`arn:aws:kms:us-east-1:123456789012:key/12345678-1234-1234-1234-456789012123`).
`"S3Object" : "JarLocation "`	Nur erforderlich, wenn für angegeben `CSE-Custom` istCertificateProviderType . *`JarLocation`* gibt den Speicherort in Amazon S3 für eine JAR Datei an. Das Format kann ein Pfad (zum Beispiels`3://MyConfig/artifacts/MyKeyProvider.jar`) oder ein ARN (zum Beispiel,) sein`arn:aws:s3:::Code/MyKeyProvider.jar)` .
`"EncryptionKeyProviderClass" : "MyLocalDiskKeyClassID "`	Nur erforderlich, wenn für angegeben `Custom` istType. *`MyLocalDiskKeyClassID`* gibt den vollständigen Klassennamen einer Klasse an, die in der Anwendung deklariert ist, die die EncryptionMaterialsProvider Schnittstelle implementiert; zum Beispiel*`com.mycompany.MyLocalDiskKeyProvider`* .
EMRWALVerschlüsselungsparameter	
`"EMRWALEncryptionConfiguration"`	Gibt den Wert für die EMR WAL Verschlüsselung an.
`"AwsKmsKey"`	Gibt die CMK Schlüssel-ID Arn an.

Konfiguration der Kerberos-Authentifizierung

Eine Sicherheitskonfiguration mit Kerberos-Einstellungen kann nur von einem Cluster verwendet werden, das mit Kerberos-Attributen erstellt wurde, andernfalls tritt ein Fehler auf. Weitere

Informationen finden Sie unter Verwenden Sie Kerberos für die Authentifizierung bei Amazon EMR. Kerberos ist nur in der EMR Amazon-Release-Version 5.10.0 und höher verfügbar.

Kerberos-Einstellungen unter Verwendung der Konsole angeben

Wählen Sie anhand der folgenden Anleitungen Optionen in Kerberos authentication (Kerberos-Authentifizierung) aus.

Parameter		Beschreibung
Kerberos		Gibt an, dass Kerberos für Cluster aktiviert ist, die diese Sicherheitskonfiguration verwenden. Wenn ein Cluster diese Sicherheitskonfiguration verwendet, müssen für den Cluster auch Kerberos-Einstellungen angegeben sein, andernfalls tritt ein Fehler auf.
Anbieter	Cluster-spezifisch KDC	Gibt an, dass Amazon KDC auf dem primären Knoten eines Clusters, der diese Sicherheitskonfiguration verwendet, eine EMR erstellt. Sie geben den Realm-Namen und das KDC Admin-Passwort an, wenn Sie den Cluster erstellen. Sie können bei Bedarf KDC von anderen Clustern aus darauf verweisen. Erstellen Sie diese Cluster mit einer anderen Sicherheitskonfiguration, geben Sie eine externe KDC Konfiguration an und verwenden Sie den Bereichsnamen und das KDC Administratorkennwort, die Sie für den dedizierten Cluster angeben. KDC
	Extern KDC	Nur mit Amazon EMR 5.20.0 und höher verfügbar. Gibt an, dass Cluster, die diese Sicherheitskonfigu ration verwenden, Kerberos-Prinzipale mithilfe eines Servers außerhalb des Clusters authentifizieren. KDC A KDC wird auf dem Cluster nicht erstellt. Wenn Sie den Cluster erstellen, geben Sie den Bereichsnamen und das KDC Administratorkennwort für den externen Cluster anKDC.

Parameter		Beschreibung
Gültigkeitsdauer des Tickets		Optional. Gibt den Zeitraum an, für den ein von der ausgestelltes Kerberos-Ticket auf Clustern gültig KDC ist, die diese Sicherheitskonfiguration verwenden. Ticket-Gültigkeitsdauern werden aus Sicherhei tsgründen beschränkt. Cluster-Anwendungen und Services verlängern Tickets automatisch, wenn sie ablaufen. Benutzer, die SSH mithilfe von Kerberos-Anmeldeinformationen eine Verbindung zum Cluster herstellen, müssen von der Befehlszeile des primären Knotens `kinit` aus starten, um das Ticket zu verlängern, nachdem ein Ticket abgelaufen ist.
Bereichsübergreifende Vertrauen sstellung		Gibt eine bereichsübergreifende Vertrauensstellung zwischen einem Cluster, der ausschließlich Clustern zugeordnet KDC ist, die diese Sicherheitskonfiguration verwenden, und einem Cluster KDC in einem anderen Kerberos-Bereich an. Prinzipale (in der Regel Benutzer) aus einem anderen Bereich werden gegenüber Clustern authentifiziert, die diese Konfiguration verwenden. Eine zusätzlic he Konfiguration im anderen Kerberos-Bereich ist erforderlich. Weitere Informationen finden Sie unter Tutorial: Konfigurieren einer bereichsübergreifenden Vertrauensstellung mit einer Active-Directory-Domain.
Realitäts übergreif ende Vertrauen seigensch aften	Bereich	Gibt den Kerberos-Bereichsnamen des anderen Bereichs in der Vertrauensstellung an. Gemäß der Konvention sind Kerberos-Bereichsnamen mit dem Domainnamen identisch, jedoch ausschließlich in Großbuchstaben.
	Domain	Gibt den Domain-Namen des anderen Bereichs in der Vertrauensstellung an.

Parameter		Beschreibung
	Admin-Server	Gibt den vollqualifizierten Domänennamen (FQDN) oder die IP-Adresse des Admin-Servers im anderen Bereich der Vertrauensstellung an. Der Admin-Server und der KDC Server laufen in der Regel auf demselben Computer mit demselbenFQDN, kommunizieren aber über unterschiedliche Ports. Falls kein Port angegeben ist, wird Port 749 verwendet , da es dabei um den Kerberos-Standard handelt. Sie können optional einen Port angeben (beispielsweise `domain.example.com :749`).
	KDCServer	Gibt den vollqualifizierten Domänennamen (FQDN) oder die IP-Adresse des KDC Servers im anderen Bereich der Vertrauensstellung an. Der KDC Server und der Admin-Server laufen normalerweise auf demselben Computer mit denselbenFQDN, verwenden jedoch unterschiedliche Ports. Falls kein Port angegeben ist, wird Port 88 verwendet , da es dabei um den Kerberos-Standard handelt. Sie können optional einen Port angeben (beispielsweise `domain.example.com :88`).
Extern KDC		Gibt an, dass externe KDC Cluster vom Cluster verwendet werden.

Parameter			Beschreibung
Externe KDC Eigenschaften	Admin-Server		Gibt den vollqualifizierten Domänennamen (FQDN) oder die IP-Adresse des externen Admin-Servers an. Der Admin-Server und der KDC Server laufen in der Regel auf demselben Computer mit denselben AnschlüssenFQDN, kommunizieren aber über unterschiedliche Ports. Falls kein Port angegeben ist, wird Port 749 verwendet, da es dabei um den Kerberos-Standard handelt. Sie können optional einen Port angeben (beispielsweise `domain.example.com :749`).
	KDCServer		Gibt den vollqualifizierten Domänennamen (FQDN) des externen KDC Servers an. Der KDC Server und der Admin-Server laufen normalerweise auf demselben Computer mit denselbenFQDN, verwenden jedoch unterschiedliche Ports. Falls kein Port angegeben ist, wird Port 88 verwendet, da es dabei um den Kerberos-Standard handelt. Sie können optional einen Port angeben (beispielsweise `domain.example.com :88`).
	Active-Directory-Integration		Gibt an, dass die Kerberos-Prinzipalauthentifizierung in eine Microsoft-Active-Directory-Domain integriert ist.
	Active-Directory-Integrationseigenschaften	Active-Directory-Bereich	Gibt den Kerberos-Bereichsnamen der Active-Directory-Domain an. Gemäß der Konvention sind Kerberos-Bereichsnamen in der Regel identisch mit dem Domainnamen, jedoch ausschließlich in Großbuchstaben.
		Active-Directory-Domain	Gibt den Active-Directory-Domainnamen an.

Parameter			Beschreibung
		Active-Directory-Server	Gibt den vollqualifizierten Domänennamen (FQDN) des Microsoft Active Directory-Domänencontrollers an.

Angeben von Kerberos-Einstellungen mithilfe von AWS CLI

Die folgende Referenztabelle zeigt JSON Parameter für Kerberos-Einstellungen in einer Sicherheitskonfiguration. Beispielkonfigurationen finden Sie unter Beispiele für Konfigurationen.

Parameter	Beschreibung
`"AuthenticationConfiguration": {`	Erforderlich für Kerberos. Gibt an, dass eine Authentifizierungskonfiguration Teil dieser Sicherheitskonfiguration ist.
`"KerberosConfiguration": {`	Erforderlich für Kerberos. Gibt die Kerberos-Konfigurationseigenschaften an.

`"Provider": "ClusterDedicatedKdc",`
–oder–
`"Provider": "ExternalKdc",`
	`ClusterDedicatedKdc` gibt an, dass Amazon KDC auf dem primären Knoten eines Clusters, der diese Sicherheitskonfiguration verwendet, eine EMR erstellt. Sie geben den Realm-Namen und das KDC Admin-Passwort an, wenn Sie den Cluster erstellen. Sie können bei Bedarf KDC von anderen Clustern aus darauf verweisen. Erstellen Sie diese Cluster mit einer anderen Sicherheitskonfiguration, geben Sie eine externe KDC Konfiguration an und verwenden Sie den Bereichsnamen und das KDC Administratorkennw

Parameter	Beschreibung
	ort, die Sie bei der Erstellung des Clusters mit dem dedizierten Cluster angegeben haben. KDC *ExternalKdc* gibt an, dass der Cluster ein externes System verwendet. KDC Amazon erstellt EMR keine KDC auf dem primären Knoten. Ein Cluster, der diese Sicherheitskonfiguration verwendet , muss den Realm-Namen und das KDC Admin-Passwort des externen Clusters angebenKDC.
`"ClusterDedicatedKdcConfiguration": {`	Erforderlich, wenn *ClusterDedicatedKdc* angegeben ist.
`"TicketLifetimeInHours": 24,`	Optional. Gibt den Zeitraum an, für den ein von der ausgestelltes Kerberos-Ticket auf Clustern gültig KDC ist, die diese Sicherheitskonfiguration verwenden. Ticket-Gültigkeitsdauern werden aus Sicherheitsgründen beschränkt. Cluster-Anwendungen und Services verlängern Tickets automatisch, wenn sie ablaufen. Benutzer, die SSH mithilfe von Kerberos-Anmeldeinformationen eine Verbindung zum Cluster herstelle n, müssen von der Befehlszeile des primären Knotens `kinit` aus starten, um das Ticket zu verlängern, nachdem ein Ticket abgelaufen ist.

Parameter	Beschreibung
"CrossRealmTrustConfiguration": {	Gibt eine bereichsübergreifende Vertrauensstellung zwischen einem Cluster, der ausschließlich Clustern zugeordnet KDC ist, die diese Sicherheitskonfiguration verwenden , und einem Cluster KDC in einem anderen Kerberos-Bereich an. Prinzipale (in der Regel Benutzer) aus einem anderen Bereich werden gegenüber Clustern authentifiziert, die diese Konfiguration verwenden . Eine zusätzliche Konfiguration im anderen Kerberos-Bereich ist erforderlich. Weitere Informationen finden Sie unter Tutorial: Konfigurieren einer bereichsübergreifenden Vertrauensstellung mit einer Active-Directory-Domain.
"Realm": "*KDC2.COM*",	Gibt den Kerberos-Bereichsnamen des anderen Bereichs in der Vertrauensstellung an. Gemäß der Konvention sind Kerberos-Bereichsnamen mit dem Domainnamen identisch, jedoch ausschließlich in Großbuchstaben.
"Domain": "*kdc2.com*",	Gibt den Domain-Namen des anderen Bereichs in der Vertrauensstellung an.

Parameter	Beschreibung
"AdminServer": "*kdc.com:749* ",	Gibt den vollqualifizierten Domänennamen (FQDN) oder die IP-Adresse des Admin-Servers im anderen Bereich der Vertrauensstellung an. Der Admin-Server und der KDC Server laufen in der Regel auf demselben Computer mit demselbenFQDN, kommunizieren aber über unterschiedliche Ports. Falls kein Port angegeben ist, wird Port 749 verwendet, da es dabei um den Kerberos-Standard handelt. Sie können optional einen Port angeben (beispielsweise `domain.example.com` *:749*).
"KdcServer": "*kdc.com:88* "	Gibt den vollqualifizierten Domänennamen (FQDN) oder die IP-Adresse des KDC Servers im anderen Bereich der Vertrauensstellung an. Der KDC Server und der Admin-Server laufen normalerweise auf demselben Computer mit denselbenFQDN, verwenden jedoch unterschiedliche Ports. Falls kein Port angegeben ist, wird Port 88 verwendet, da es dabei um den Kerberos-Standard handelt. Sie können optional einen Port angeben (beispielsweise `domain.example.com` *:88*).
}	

Parameter	Beschreibung
}	
"ExternalKdcConfiguration": {	Erforderlich, wenn *ExternalKdc* angegeben ist.
"TicketLifetimeInHours": *24*,	Optional. Gibt den Zeitraum an, für den ein von der ausgestelltes Kerberos-Ticket auf Clustern gültig KDC ist, die diese Sicherheitskonfiguration verwenden. Ticket-Gültigkeitsdauern werden aus Sicherheitsgründen beschränkt. Cluster-Anwendungen und Services verlängern Tickets automatisch, wenn sie ablaufen. Benutzer, die SSH mithilfe von Kerberos-Anmeldeinformationen eine Verbindung zum Cluster herstellen, müssen von der Befehlszeile des primären Knotens `kinit` aus starten, um das Ticket zu verlängern, nachdem ein Ticket abgelaufen ist.
"KdcServerType": "Single",	Gibt an, dass auf einen einzelnen KDC Server verwiesen wird. `Single`ist derzeit der einzige unterstützte Wert.

Parameter				Beschreibung
			"AdminServer": "*kdc.com:749* ",	Gibt den vollqualifizierten Domänennamen (FQDN) oder die IP-Adresse des externen Admin-Servers an. Der Admin-Server und der KDC Server laufen in der Regel auf demselben Computer mit denselben AnschlüssenFQDN, kommunizieren aber über unterschiedliche Ports. Falls kein Port angegeben ist, wird Port 749 verwendet, da es dabei um den Kerberos-Standard handelt. Sie können optional einen Port angeben (beispielsweise `domain.example.com` *`:749`*).
			"KdcServer": "*kdc.com:88* ",	Gibt den vollqualifizierten Domänennamen (FQDN) des externen KDC Servers an. Der KDC Server und der Admin-Server laufen normalerweise auf demselben Computer mit denselbenFQDN, verwenden jedoch unterschiedliche Ports. Falls kein Port angegeben ist, wird Port 88 verwendet, da es dabei um den Kerberos-Standard handelt. Sie können optional einen Port angeben (beispielsweise `domain.example.com` *`:88`*).
			`"AdIntegrationConfiguration": {`	Gibt an, dass die Kerberos-Prinzipal authentifizierung in eine Microsoft-Active-Directory-Domain integriert ist.

Parameter	Beschreibung
"AdRealm": "*AD.DOMAIN.COM* ",	Gibt den Kerberos-Bereichsnamen der Active-Directory-Domain an. Gemäß der Konvention sind Kerberos-Bereichsnamen in der Regel identisch mit dem Domainnamen, jedoch ausschließlich in Großbuchstaben.
"AdDomain": "*ad.domain.com* "	Gibt den Active-Directory-Domainnamen an.
"AdServer": "*ad.domain.com* "	Gibt den vollqualifizierten Domänennamen (FQDN) des Microsoft Active Directory-Domänencontrollers an.
}	
}	
}	
}	

IAMRollen für EMRFS Anfragen an Amazon S3 konfigurieren

IAMRollen für EMRFS ermöglichen es Ihnen, verschiedene Berechtigungen für EMRFS Daten in Amazon S3 zu vergeben. Sie erstellen Zuordnungen, die eine IAM Rolle angeben, die für Berechtigungen verwendet wird, wenn eine Zugriffsanfrage eine von Ihnen angegebene Kennung enthält. Bei der ID kann es sich um einen Hadoop-Benutzer oder eine Hadoop-Rolle oder ein Amazon-S3-Präfix handeln.

Weitere Informationen finden Sie unter IAMRollen für EMRFS Anfragen an Amazon S3 konfigurieren.

Angeben von IAM Rollen für die Verwendung von EMRFS AWS CLI

Im Folgenden finden Sie einen JSON Beispielausschnitt für die Angabe benutzerdefinierter IAM Rollen EMRFS innerhalb einer Sicherheitskonfiguration. Es zeigt Rollenzuordnungen für die drei verschiedenen Identifier-Typen, gefolgt von einer Parameterreferenz.

```
{
  "AuthorizationConfiguration": {
    "EmrFsConfiguration": {
      "RoleMappings": [{
        "Role": "arn:aws:iam::123456789101:role/allow_EMRFS_access_for_user1",
        "IdentifierType": "User",
        "Identifiers": [ "user1" ]
      },{
        "Role": "arn:aws:iam::123456789101:role/allow_EMRFS_access_to_MyBuckets",
        "IdentifierType": "Prefix",
        "Identifiers": [ "s3://MyBucket/","s3://MyOtherBucket/" ]
      },{
        "Role": "arn:aws:iam::123456789101:role/allow_EMRFS_access_for_AdminGroup",
        "IdentifierType": "Group",
        "Identifiers": [ "AdminGroup" ]
      }]
    }
  }
}
```

Parameter	Beschreibung
`"AuthorizationConfiguration":`	Erforderlich
`"EmrFsConfiguration":`	Erforderlich Enthält Rollenzuordnungen.
`"RoleMappings":`	Erforderlich Enthält eine oder mehrere Rollenzuordnungsdefinitionen. Rollenzuordnungen werden in der Reihenfolge bewertet, in der sie von oben nach unten angezeigt werden. Wenn eine Rollenzuweisung für einen EMRFS Datenaufruf in Amazon S3 als wahr bewertet wird, werden keine weiteren Rollenzuordnungen ausgewertet und die

Parameter	Beschreibung
	angegebene IAM Rolle wird für die Anfrage EMRFS verwendet. Rollenzuordnungen bestehen aus den folgenden erforderlichen Parametern:
`"Role":`	Gibt den ARN Bezeichner einer IAM Rolle im Format an. `arn:aws:iam::` *`account-id`* `:role/`*`role-name`* Dies ist die IAM Rolle, die Amazon EMR übernimmt, wenn die EMRFS Anfrage an Amazon S3 mit einer der `Identifiers` angegebenen Anforderungen übereinstimmt.

Parameter	Beschreibung
`"IdentifierType":`	Kann einer der folgenden sein: • `"User"` gibt an, dass es sich bei den Kennungen um einen oder mehrere Hadoop-Benutzer handelt, bei denen es sich um Linux-Kontobenutzer oder Kerberos-Prinzipale handeln kann. Wenn die EMRFS Anfrage von dem oder den angegebenen Benutzern stammt, wird die IAM Rolle übernommen. • `"Prefix"` gibt an, dass der Identifier ein Amazon-S3-Speicherort ist. Die IAM Rolle wird für Anrufe an den Standort oder die Standorte mit den angegebenen Präfixen übernommen. Das Präfix `s3://mybucket/` entspricht beispielsweise `s3://mybucket/mydir` und `s3://mybucket/yetanotherdir` . • `"Group"` gibt an, dass es sich bei den Identifikatoren um eine oder mehrere Hadoop-Gruppen handelt. Die IAM Rolle wird übernommen, wenn die Anfrage von einem Benutzer in der oder den angegebenen Gruppen stammt.
`"Identifiers":`	Gibt einen oder mehrere Kennungen des entsprechenden Kennungstyps an. Trennen Sie mehrere Bezeichner durch Kommas ohne Leerzeichen.

Metadaten-Serviceanfragen an EC2 Amazon-Instances konfigurieren

Instance-Metadaten sind Daten über eine Instance, mit denen Sie die ausgeführte Instance konfigurieren und verwalten können. Sie können mit einer der folgenden Methoden auf Instance-Metadaten aus einer laufenden Instance zugreifen:

- Instance Metadata Service Version 1 (IMDSv1) — eine Anforderungs-/Antwortmethode
- Instanz-Metadatendienst Version 2 (IMDSv2) — eine sitzungsorientierte Methode

Während Amazon IMDSv1 sowohl als auch EC2 unterstütztIMDSv2, EMR unterstützt Amazon IMDSv2 in Amazon EMR 5.23.1, 5.27.1, 5.32 oder höher und 6.2 oder höher. In diesen Versionen werden EMR Amazon-Komponenten IMDSv2 für alle IMDS Aufrufe verwendet. Für IMDS Aufrufe in Ihrem Anwendungscode können Sie IMDSv1 sowohl als auch verwenden oder das so konfigurierenIMDSv2, IMDS dass es nur aus IMDSv2 Sicherheitsgründen verwendet wird. Wenn Sie angeben, dass dies verwendet werden IMDSv2 muss, funktioniert IMDSv1 es nicht mehr.

Weitere Informationen finden Sie unter Konfiguration des Instance-Metadaten-Service im EC2Amazon-Benutzerhandbuch.

> **Note**
>
> In früheren Amazon EMR 5.x- oder 6.x-Versionen IMDSv1 führt das Ausschalten zu einem Fehler beim Starten des Clusters, da EMR Amazon-Komponenten IMDSv1 für alle IMDS Aufrufe verwendet werden. Stellen Sie beim Ausschalten sicherIMDSv1, dass jede benutzerdefinierte Software, die verwendet wird, auf aktualisiert IMDSv1 ist. IMDSv2

Spezifizieren Sie die Konfiguration des Instance Metadata Services mit dem AWS CLI

Das Folgende ist ein JSON Beispielausschnitt für die Angabe des Amazon EC2 Instance Metadata Service (IMDS) innerhalb einer Sicherheitskonfiguration. Die Verwendung einer benutzerdefinierten Sicherheitskonfiguration ist optional.

```
{
  "InstanceMetadataServiceConfiguration" : {
      "MinimumInstanceMetadataServiceVersion": integer,
      "HttpPutResponseHopLimit": integer
   }
}
```

Parameter	Beschreibung
`"InstanceMetadataServiceConfiguration":`	Wenn Sie dies nicht IMDS innerhalb einer Sicherheitskonfiguration angeben und eine

Parameter	Beschreibung
	EMR Amazon-Version verwenden, die dies erfordertIMDSv1, verwendet Amazon EMR standardmäßig IMDSv1 als Mindestversion des Instance-Metadatendienstes. Wenn Sie Ihre eigene Konfiguration verwenden möchten, sind die beiden folgenden Parameter erforderlich.
`"MinimumInstanceMetadataServiceVersion":`	Erforderlich Geben Sie 1 oder 2 an. Der Wert 1 erlaubt IMDSv1 undIMDSv2. Der Wert 2 erlaubt nurIMDSv2.
`"HttpPutResponseHopLimit":`	Erforderlich Das gewünschte HTTP PUT Response-Hop-Limit für Instance-Metadaten anfragen. Je größer die Zahl ist, desto weiter können sich die Instance-Metadatenanfragen bewegen. Standard: 1. Einen Ganzzahlwert von 1 bis 64 angeben.

Die Konfiguration des Instance Metadata Services mit der Konsole angeben

Sie können die Verwendung von IMDS für einen Cluster konfigurieren, wenn Sie ihn von der EMR Amazon-Konsole aus starten.

So konfigurieren Sie die IMDS Verwendung der Konsole:

1. Wenn Sie auf der Seite Sicherheitskonfigurationen eine neue Sicherheitskonfiguration erstellen, wählen Sie unter der Einstellung EC2Instanz-Metadatendienst die Option EC2Instanz-Metadatendienst konfigurieren aus. Diese Konfiguration wird nur in Amazon EMR 5.23.1, 5.27.1, 5.32 oder höher und 6.2 oder höher unterstützt.
2. Für Minimum Instance Metadata Service Version wählen Sie eine der folgenden Optionen aus:
   - Schalten Sie die Option aus IMDSv1 und lassen Sie sie nur zuIMDSv2, wenn Sie nur auf diesem Cluster zulassen möchten. IMDSv2 Weitere Informationen finden Sie unter Umstellung auf die Nutzung des Instance-Metadaten-Service Version 2 im EC2Amazon-Benutzerhandbuch.

- Erlauben Sie beides IMDSv1 und IMDSv2 auf dem Cluster, wenn Sie IMDSv2 auf diesem Cluster sitzungsorientiert sein möchten. IMDSv1

3. Denn IMDSv2 Sie können auch die zulässige Anzahl von Netzwerk-Hops für das Metadaten-Token konfigurieren, indem Sie das Put-Response-Hop-Limit auf eine Ganzzahl zwischen und festlegen. HTTP 1 64

Weitere Informationen finden Sie unter Konfiguration des Instance-Metadaten-Service im EC2Amazon-Benutzerhandbuch.

Weitere Informationen finden Sie unter Instance-Details konfigurieren und Instance-Metadaten-Service konfigurieren im EC2Amazon-Benutzerhandbuch.

Angabe einer Sicherheitskonfiguration für einen Cluster

Sie können bei der Erstellung eines Clusters die Verschlüsselungseinstellungen festlegen, indem Sie eine Sicherheitskonfiguration angeben. Sie können das AWS Management Console oder das verwenden AWS CLI.

Console

Um eine Sicherheitskonfiguration mit der Konsole anzugeben

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und anschließend Create cluster aus.
3. Suchen Sie unter Sicherheitskonfiguration und Berechtigungen das Feld Sicherheitskonfiguration. Wählen Sie das Dropdownmenü oder klicken Sie auf Durchsuchen, um den Namen einer Sicherheitskonfiguration auszuwählen, die Sie zuvor erstellt haben. Wählen Sie alternativ VPC erstellen, um eine VPC zu erstellen, die Sie für Ihren Cluster verwenden können.
4. Wählen Sie alle anderen Optionen aus, die für Ihren Cluster gelten.
5. Um Ihren Cluster jetzt zu starten, wählen Sie Cluster erstellen aus.

CLI

Um eine Sicherheitskonfiguration mit dem anzugeben AWS CLI

- Verwenden Sie `aws emr create-cluster` und Sie können mithilfe von `--security-configuration` *`MySecConfig`* wahlweise eine Sicherheitskonfiguration verwenden, wobei *`MySecConfig`* der Name der Sicherheitskonfiguration ist, wie im folgenden Beispiel dargestellt. Der angegebene `--release-label` muss 4.8.0 oder höher sein und `--instance-type` kann als jeder verfügbare Typ ausgewählt werden.

```
aws emr create-cluster --instance-type m5.xlarge --release-label emr-5.0.0 --
security-configuration mySecConfig
```

Datenschutz bei Amazon EMR

Das Modell der AWS gemeinsamen Verantwortung gilt für den Datenschutz bei AmazonEMR. AWS Ist, wie in diesem Modell beschrieben, für den Schutz der globalen Infrastruktur verantwortlich, auf der die gesamte AWS Cloud läuft. Sie sind dafür verantwortlich, die Kontrolle über Ihre in dieser Infrastruktur gehosteten Inhalte zu behalten. Dieser Inhalt umfasst die Sicherheitskonfiguration und die Verwaltungsaufgaben für die AWS , die Sie verwenden. Weitere Informationen zum Datenschutz finden Sie im Abschnitt Datenschutz FAQ. Informationen zum Datenschutz in Europa finden Sie im Amazon-Modell der gemeinsamen Verantwortung und im GDPR Blogbeitrag im AWS Security Blog.

Aus Datenschutzgründen empfehlen wir Ihnen, Ihre AWS Kontoanmeldeinformationen zu schützen und individuelle Konten bei einzurichten AWS Identity and Access Management. So erhält jeder Benutzer nur die Berechtigungen, die zum Durchführen seiner Aufgaben erforderlich sind. Außerdem sollten Sie die Daten mit folgenden Methoden schützen:

- Verwenden Sie für jedes Konto eine Multi-Faktor-Authentifizierung (MFA).
- Wird verwendetTLS, um mit AWS Ressourcen zu kommunizieren. Wir benötigen TLS 1.2.
- Einrichtung API und Protokollierung von Benutzeraktivitäten mit AWS CloudTrail.
- Verwenden Sie AWS Verschlüsselungslösungen zusammen mit allen Standardsicherheitskontrollen innerhalb der AWS Dienste.
- Verwenden Sie erweiterte verwaltete Sicherheitsservices wie Amazon Macie, die dabei helfen, in Amazon S3 gespeicherte persönliche Daten zu erkennen und zu sichern.

- Wenn Sie FIPS 140-2 validierte kryptografische Module für den Zugriff AWS über eine Befehlszeilenschnittstelle oder eine benötigenAPI, verwenden Sie einen Endpunkt. FIPS Weitere Informationen zu den verfügbaren FIPS Endpunkten finden Sie unter Federal Information Processing Standard () 140-2. FIPS

Wir empfehlen dringend, in Freitextfeldern wie z. B. im Feld Name keine sensiblen, identifizierenden Informationen wie Kontonummern von Kunden einzugeben. Dies gilt auch, wenn Sie mit Amazon EMR oder anderen AWS Diensten über die Konsole, API AWS CLI, oder arbeiten AWS SDKs. Alle Daten, die Sie bei Amazon EMR oder anderen Diensten eingeben, werden möglicherweise zur Aufnahme in die Diagnoseprotokolle aufgenommen. Wenn Sie einem externen Server eine URL zur Verfügung stellen, geben Sie keine Anmeldeinformationen an, URL um Ihre Anfrage an diesen Server zu validieren.

Verschlüsseln von Daten im Ruhezustand und im Transit

Die Datenverschlüsselung verhindert, dass nicht autorisierte Benutzer Daten auf einem Cluster und in den dazugehörigen Datenspeichersystemen lesen können. Dies gilt für auf persistenten Medien gespeicherte Daten, auch als Daten im Ruhezustand bezeichnet, und für Daten, die während der Übertragung im Netzwerk möglicherweise abgefangen werden, auch als Daten während der Übertragung bezeichnet.

Ab EMR Amazon-Version 4.8.0 können Sie EMR Amazon-Sicherheitskonfigurationen verwenden, um Datenverschlüsselungseinstellungen für Cluster einfacher zu konfigurieren. Sicherheitskonfigurationen bieten Einstellungen, um die Sicherheit für Daten während der Übertragung und Speicherung von Daten auf Amazon Elastic Block Store (AmazonEBS) -Volumes und EMRFS auf Amazon S3 zu aktivieren.

Optional können Sie ab der EMR Amazon-Version 4.1.0 und höher die transparente Verschlüsselung konfigurierenHDFS, die nicht mithilfe von Sicherheitskonfigurationen konfiguriert ist. Weitere Informationen finden Sie unter Transparente Verschlüsselung HDFS bei Amazon EMR im Amazon EMR Release Guide.

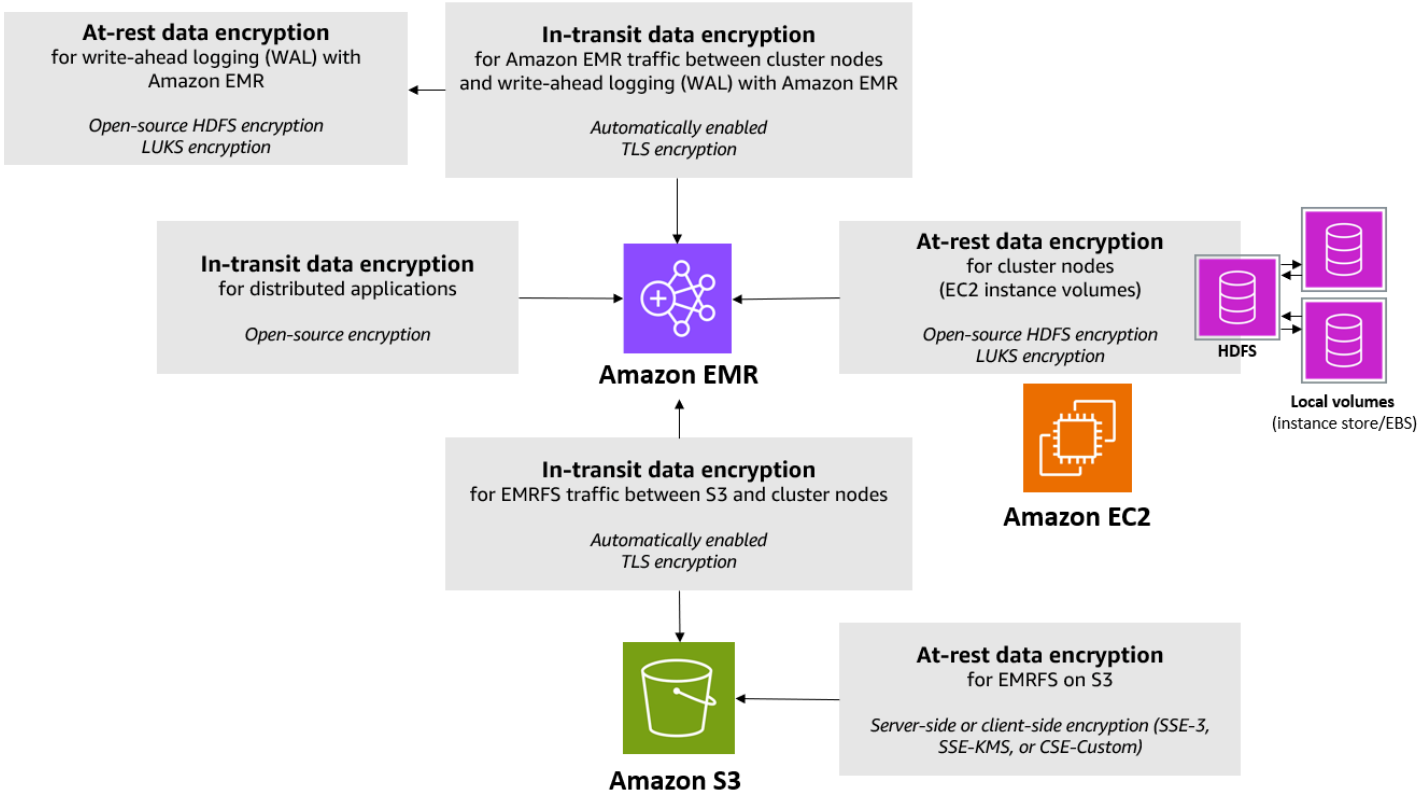
Themen

- Verschlüsselungsoptionen
- Schlüssel und Zertifikate für die Datenverschlüsselung erstellen

Verschlüsselungsoptionen

Mit EMR Amazon-Versionen 4.8.0 und höher können Sie eine Sicherheitskonfiguration verwenden, um Einstellungen für die Verschlüsselung von Daten im Ruhezustand, Daten während der Übertragung oder beidem festzulegen. Wenn Sie die Verschlüsselung von Daten im Ruhezustand aktivieren, können Sie wählen, ob Sie EMRFS Daten in Amazon S3, Daten auf lokalen Festplatten oder beides verschlüsseln möchten. Jede Sicherheitskonfiguration, die Sie erstellen, wird in Amazon und EMR nicht in der Cluster-Konfiguration gespeichert, sodass Sie eine Konfiguration problemlos wiederverwenden können, um Datenverschlüsselungseinstellungen anzugeben, wann immer Sie einen Cluster erstellen. Weitere Informationen finden Sie unter Eine Sicherheitskonfiguration erstellen.

Das folgende Diagramm zeigt die verschiedenen Datenverschlüsselungsoptionen, die für die Sicherheitskonfigurationen zur Verfügung stehen.



Die folgenden Verschlüsselungsoptionen stehen ebenfalls zur Verfügung und werden nicht mit einer Sicherheitskonfiguration konfiguriert:

- Optional können Sie bei EMR Amazon-Versionen 4.1.0 und höher wählen, ob Sie die transparente Verschlüsselung in HDFS konfigurieren möchten. Weitere Informationen finden Sie unter Transparente Verschlüsselung HDFS bei Amazon EMR im Amazon EMR Release Guide.

- Wenn Sie eine Release-Version von Amazon verwendenEMR, die keine Sicherheitskonfigurationen unterstützt, können Sie die Verschlüsselung für EMRFS Daten in Amazon S3 manuell konfigurieren. Weitere Informationen finden Sie unter Amazon S3 S3-Verschlüsselung mithilfe von EMRFS Eigenschaften angeben.
- Wenn Sie eine EMR Amazon-Version vor 5.24.0 verwenden, wird ein verschlüsseltes EBS Root-Geräte-Volume nur unterstützt, wenn Sie ein benutzerdefiniertes Volume verwenden. AMI Weitere Informationen finden Sie unter Erstellen eines benutzerdefinierten Volumes AMI mit einem verschlüsselten EBS Amazon-Root-Geräte-Volume im Amazon EMR Management Guide.

> **Note**
>
> Ab EMR Amazon-Version 5.24.0 können Sie eine Sicherheitskonfigurationsoption verwenden, um EBS Root-Geräte und Speichervolumes zu verschlüsseln, wenn Sie dies AWS KMS als Ihren Schlüsselanbieter angeben. Weitere Informationen finden Sie unter Verschlüsselung lokaler Datenträger.

Die Datenverschlüsselung erfordert Aktivierungsschlüssel und Zertifikate. Eine Sicherheitskonfiguration bietet Ihnen die Flexibilität, aus mehreren Optionen zu wählen, darunter Schlüssel AWS Key Management Service, die von Amazon S3 verwaltet werden, sowie Schlüssel und Zertifikate von benutzerdefinierten Anbietern, die Sie bereitstellen. Bei der Nutzung AWS KMS als Schlüsselanbieter fallen Gebühren für die Speicherung und Verwendung von Verschlüsselungsschlüsseln an. Weitere Informationen finden Sie unter AWS KMS Preise.

Bevor Sie die Verschlüsselungsoptionen angeben, legen Sie fest, welche Verwaltungssysteme Sie für die Schlüssel und Zertifikate verwenden möchten. Auf diese Weise können Sie zunächst die Schlüssel und Zertifikate bzw. die von Ihnen bestimmten Anbieter erstellen, die Sie als Teil der Verschlüsselungseinstellungen verwenden möchten.

Verschlüsselung im Ruhezustand für EMRFS Daten in Amazon S3

Die Amazon S3-Verschlüsselung funktioniert mit den Amazon EMR File System (EMRFS) - Objekten, die aus Amazon S3 gelesen und in Amazon S3 geschrieben wurden. Sie geben die serverseitige Amazon S3 S3-Verschlüsselung (SSE) oder die clientseitige Verschlüsselung (CSE) als Standardverschlüsselungsmodus an, wenn Sie die Verschlüsselung im Ruhezustand aktivieren. Optional können Sie verschiedene Verschlüsselungsmethoden für einzelne Buckets mithilfe von Per bucket encryption overrides (Bucket-weises Überschreiben der Verschlüsselung) angeben.

Unabhängig davon, ob die Amazon S3-Verschlüsselung aktiviert ist, verschlüsselt Transport Layer Security (TLS) die EMRFS Objekte, die zwischen EMR Clusterknoten und Amazon S3 übertragen werden. Weitere Informationen zur Amazon S3 S3-Verschlüsselung finden Sie unter Schützen von Daten durch Verschlüsselung im Amazon Simple Storage Service-Benutzerhandbuch.

> **Note**
>
> Bei der Nutzung AWS KMS fallen Gebühren für die Speicherung und Verwendung von Verschlüsselungsschlüsseln an. Weitere Informationen finden Sie unter AWS KMS - Preisgestaltung.

Serverseitige Verschlüsselung im Amazon S3

Wenn Sie die Amazon-S3-Verschlüsselung einrichten, verschlüsselt Amazon S3 die Daten auf der Objektebene, während die Daten auf den Datenträger geschrieben werden, und entschlüsselt sie, wenn auf sie zugegriffen wird. Weitere Informationen SSE dazu finden Sie unter Schützen von Daten mithilfe serverseitiger Verschlüsselung im Amazon Simple Storage Service-Benutzerhandbuch.

Bei der Angabe SSE in Amazon können Sie zwischen zwei verschiedenen Schlüsselverwaltungssystemen wählenEMR:

- SSE-S3 — Amazon S3 verwaltet Schlüssel für Sie.
- SSE- KMS — Sie verwenden eine AWS KMS key , um Richtlinien einzurichten, die für Amazon geeignet sindEMR. Weitere Informationen zu den wichtigsten Anforderungen für Amazon EMR finden Sie unter AWS KMS keys Zur Verschlüsselung verwenden.

SSEmit vom Kunden bereitgestellten Schlüsseln (SSE-C) ist nicht für die Verwendung mit Amazon verfügbar. EMR

Clientseitige Verschlüsselung für Amazon S3

Bei der clientseitigen Amazon S3 S3-Verschlüsselung erfolgt die Amazon S3 S3-Verschlüsselung und Entschlüsselung im EMRFS Client auf Ihrem Cluster. Objekte werden vor dem Hochladen nach Amazon S3 verschlüsselt und nach dem Herunterladen entschlüsselt. Der von Ihnen festgelegte Anbieter stellt den vom Client verwendeten Verschlüsselungsschlüssel bereit. Der Client kann die von AWS KMS (CSE-KMS) bereitgestellten Schlüssel oder eine benutzerdefinierte Java-Klasse verwenden, die den clientseitigen Stammschlüssel (-C) bereitstellt. CSE Die

Verschlüsselungsspezifikationen zwischen CSE - KMS und CSE -C unterscheiden sich geringfügig, abhängig vom angegebenen Anbieter und den Metadaten des Objekts, das entschlüsselt oder verschlüsselt wird. Weitere Informationen zu diesen Unterschieden finden Sie unter Schützen von Daten durch clientseitige Verschlüsselung im Entwicklerhandbuch von Amazon Simple Storage Service.

> **Note**
>
> Amazon S3 stellt CSE lediglich sicher, dass die mit Amazon S3 ausgetauschten EMRFS Daten verschlüsselt sind. Nicht alle Daten auf Cluster-Instance-Volumes sind verschlüsselt. Da Hue es nicht verwendetEMRFS, werden Objekte, die der Hue S3 File Browser in Amazon S3 schreibt, außerdem nicht verschlüsselt.

Verschlüsselung im Ruhezustand für Daten in Amazon EMR WAL

Wenn Sie serverseitige Verschlüsselung (SSE) für die Write-Ahead-Protokollierung (WAL) einrichten, EMR verschlüsselt Amazon Daten im Ruhezustand. Bei der Angabe SSE in Amazon können Sie zwischen zwei verschiedenen Schlüsselverwaltungssystemen wählenEMR:

SSE-EMR-WAL

Amazon EMR verwaltet die Schlüssel für Sie. Standardmäßig EMR verschlüsselt Amazon die Daten, die Sie bei Amazon gespeichert haben, EMR WAL mitSSE-EMR-WAL.

SSE-KMS-WAL

Sie verwenden einen AWS KMS Schlüssel, um Richtlinien einzurichten, die für Amazon gelten EMRWAL. Weitere Informationen zu den wichtigsten Anforderungen für Amazon EMR finden Sie unterAWS KMS keys Für die Verschlüsselung verwenden.

Sie können Ihren eigenen Schlüssel nicht verwendenSSE, wenn Sie WAL bei Amazon aktivierenEMR. Weitere Informationen finden Sie unter Write-Ahead-Logs (WAL) für Amazon. EMR

Verschlüsselung lokaler Datenträger

Die folgenden Mechanismen arbeiten zusammen, um lokale Festplatten zu verschlüsseln, wenn Sie die lokale Festplattenverschlüsselung mithilfe einer EMR Amazon-Sicherheitskonfiguration aktivieren.

Open-Source-Verschlüsselung HDFS

HDFStauscht während der verteilten Verarbeitung Daten zwischen Clusterinstanzen aus. Außerdem werden Daten von Instance-Speicher-Volumes und den an Instances angehängten EBS Volumes gelesen und in diese geschrieben. Wenn Sie die lokale Laufwerksverschlüsselung aktivieren, werden die folgenden Open-Source-Hadoop-Verschlüsselungsoptionen aktiviert:

- Secure Hadoop RPC ist auf eingestelltPrivacy, was Simple Authentication Security Layer (SASL) verwendet.
- Die Datenverschlüsselung bei HDFS Blockdatenübertragung ist auf AES 256-Verschlüsselung eingestellt true und für diese konfiguriert.

> **Note**
>
> Sie können zusätzlich die Apache Hadoop-Verschlüsselung verwenden, indem Sie die Verschlüsselung während der Übertragung aktivieren. Weitere Informationen finden Sie unter Verschlüsselung während der Übertragung. Diese Verschlüsselungseinstellungen aktivieren keine HDFS transparente Verschlüsselung, die Sie manuell konfigurieren können. Weitere Informationen finden Sie unter Transparente Verschlüsselung HDFS bei Amazon EMR im Amazon EMR Release Guide.

Instance-Speicher-Verschlüsselung

Bei EC2 Instance-Typen, die NVMe based SSDs als Instance-Speicher-Volume verwenden, wird die NVMe Verschlüsselung unabhängig von den EMR Amazon-Verschlüsselungseinstellungen verwendet. Weitere Informationen finden Sie in den NVMeSSDBänden im EC2Amazon-Benutzerhandbuch. Für andere Instance-Speicher-Volumes EMR verwendet Amazon, LUKS um das Instance-Speicher-Volume zu verschlüsseln, wenn die lokale Festplattenverschlüsselung aktiviert ist, unabhängig davon, ob EBS Volumes verschlüsselt EBS sind oderLUKS.

EBSVolumenverschlüsselung

Wenn Sie einen Cluster in einer Region erstellen, in der die EC2 Amazon-Verschlüsselung von EBS Volumes standardmäßig für Ihr Konto aktiviert ist, werden EBS Volumes verschlüsselt, auch wenn die lokale Festplattenverschlüsselung nicht aktiviert ist. Weitere Informationen finden Sie unter Standardverschlüsselung im EC2Amazon-Benutzerhandbuch. Wenn die lokale Festplattenverschlüsselung in einer Sicherheitskonfiguration aktiviert ist, haben die EMR Amazon-

Einstellungen Vorrang vor den EC2 encryption-by-default Amazon-Einstellungen für EC2 Cluster-Instances.

Die folgenden Optionen sind verfügbar, um EBS Volumes mithilfe einer Sicherheitskonfiguration zu verschlüsseln:

- EBSVerschlüsselung — Ab EMR Amazon-Version 5.24.0 können Sie wählen, ob Sie die Verschlüsselung aktivieren EBS möchten. Die EBS Verschlüsselungsoption verschlüsselt das EBS Root-Geräte-Volume und die angeschlossenen Speichervolumes. Die EBS Verschlüsselungsoption ist nur verfügbar, wenn Sie dies AWS Key Management Service als Ihren Schlüsselanbieter angeben. Wir empfehlen die Verwendung von EBS Verschlüsselung.
- LUKSVerschlüsselung — Wenn Sie sich für die LUKS Verschlüsselung für EBS Amazon-Volumes entscheiden, gilt die LUKS Verschlüsselung nur für angehängte Speichervolumes, nicht für das Root-Geräte-Volume. Weitere Informationen zur LUKS Verschlüsselung finden Sie in der LUKSFestplattenspezifikation.

  Für Ihren Schlüsselanbieter können Sie eine für Amazon geeignete AWS KMS key With-Richtlinie oder eine benutzerdefinierte Java-Klasse einrichtenEMR, die die Verschlüsselungsartefakte bereitstellt. Bei der Nutzung AWS KMS fallen Gebühren für die Speicherung und Verwendung von Verschlüsselungsschlüsseln an. Weitere Informationen finden Sie unter AWS KMS Preise.

> **Note**
>
> Um zu überprüfen, ob die EBS Verschlüsselung auf Ihrem Cluster aktiviert ist, wird empfohlen, `DescribeVolumes` API Call zu verwenden. Weitere Informationen finden Sie unter DescribeVolumes. Bei der Ausführung `lsblk` auf dem Cluster wird nur der LUKS Verschlüsselungsstatus und nicht der EBS Verschlüsselungsstatus überprüft.

Verschlüsselung während der Übertragung

Bei der Verschlüsselung während der Übertragung sind mehrere Verschlüsselungsmechanismen aktiviert. Dies sind Open-Source-Funktionen, die anwendungsspezifisch sind und je nach Amazon-Version variieren können. EMR Mithilfe von Sicherheitskonfigurationen können die folgenden anwendungsspezifischen Verschlüsselungsfeatures Apache aktiviert werden. Weitere Informationen finden Sie unter Konfigurieren von Anwendungen.

Hadoop

- Hadoop verwendet MapReduce verschlüsselten Shuffle. TLS
- Secure Hadoop RPC ist auf „Datenschutz“ eingestellt und verwendet SASL (in Amazon aktiviert, EMR wenn die Verschlüsselung im Ruhezustand aktiviert ist).
- Die Datenverschlüsselung bei HDFS Blockdatenübertragung verwendet AES 256 (aktiviert in Amazon, EMR wenn die Verschlüsselung im Ruhezustand in der Sicherheitskonfiguration aktiviert ist).
- Weitere Informationen finden Sie unter Hadoop in Secure Mode in der Apache-Hadoop-Dokumentation.

HBase

- Wenn Kerberos aktiviert ist, wird die `hbase.rpc.protection`-Eigenschaft für verschlüsselte Kommunikation auf `privacy` gesetzt.
- Weitere Informationen finden Sie unter Clientseitige Konfiguration für sicheren Betrieb in der Apache-Dokumentation. HBase
- Weitere Informationen zu Kerberos mit Amazon finden Sie EMR unter. Verwenden Sie Kerberos für die Authentifizierung bei Amazon EMR

Hive

- JDBC/Die ODBC Client-Kommunikation mit HiveServer 2 (HS2) wird mithilfe von SSL Konfigurationen in EMR Amazon-Versionen 6.9.0 und höher verschlüsselt.
- Weitere Informationen finden Sie im Abschnitt SSLVerschlüsselung der Apache Hive-Dokumentation.

Spark

- Die interne RPC Kommunikation zwischen Spark-Komponenten, wie dem Block Transfer Service und dem externen Shuffle Service, wird in EMR Amazon-Versionen 5.9.0 und höher mit der AES -256-Chiffre verschlüsselt. In früheren Versionen wurde die interne RPC Kommunikation SASL mit DIGEST — als Chiffre verschlüsselt. MD5

- HTTPDie Protokollkommunikation mit Benutzeroberflächen wie Spark History Server und HTTPS -fähigen Dateiservern wird mithilfe der Konfiguration von Spark verschlüsselt. SSL Weitere Informationen finden Sie in der Spark-Dokumentation unter SSLKonfiguration.
- Weitere Informationen finden Sie unter Sicherheitseinstellungen von Spark in der Apache-Spark-Dokumentation.

Tez

- Der Tez-Shuffle-Handler verwendet TLS ()`tez.runtime.ssl.enable`.

Presto

- Die interne Kommunikation zwischen Presto-Knoten verwendetSSL/TLS(nur EMR Amazon-Version 5.6.0 und höher).

Für die Verwendung der Verschlüsselungsartefakte bei der Verschlüsselung von Daten während der Übertragung stehen Ihnen zwei Optionen zur Verfügung: die Bereitstellung einer ZIP-Datei mit den Zertifikaten, die Sie auf Amazon S3 hochladen, oder der Verweis auf eine benutzerdefinierte Java-Klasse, die Verschlüsselungsartefakte bereitstellt. Weitere Informationen finden Sie unter Bereitstellung von Zertifikaten für die Verschlüsselung von Daten während der Übertragung mit EMR Amazon-Verschlüsselung.

Schlüssel und Zertifikate für die Datenverschlüsselung erstellen

Bevor Sie Verschlüsselungsoptionen unter Verwendung einer Sicherheitskonfiguration angeben, legen Sie zunächst den Anbieter der Schlüssel und Verschlüsselungsartefakte fest. Sie können beispielsweise einen benutzerdefinierten Anbieter verwenden AWS KMS , den Sie erstellen. Erstellen Sie als Nächstes die erforderlichen Schlüssel oder den Schlüsselanbieter, wie in diesem Abschnitt beschrieben.

Bereitstellung von Schlüsseln für die Verschlüsselung ruhender Daten bei Amazon EMR

Sie können AWS Key Management Service (AWS KMS) oder einen benutzerdefinierten Schlüsselanbieter für die Verschlüsselung von Daten im Ruhezustand in Amazon EMR verwenden. Bei der Nutzung AWS KMS fallen Gebühren für die Speicherung und Verwendung von Verschlüsselungsschlüsseln an. Weitere Informationen finden Sie unter AWS KMS Preise.

Dieses Thema enthält wichtige Richtliniendetails für einen KMS Schlüssel, der mit Amazon verwendet werden sollEMR, sowie Richtlinien und Codebeispiele für das Schreiben einer benutzerdefinierten Schlüsselanbieterklasse für die Amazon S3 S3-Verschlüsselung. Weitere Informationen zum Erstellen von -Schlüsseln finden Sie unter Erstellen von Schlüsseln im AWS Key Management Service -Entwicklerhandbuch.

AWS KMS keys Für die Verschlüsselung verwenden

Der AWS KMS Verschlüsselungsschlüssel muss in derselben Region wie Ihre EMR Amazon-Cluster-Instance und die verwendeten Amazon S3-Buckets erstellt werden. EMRFS Wenn sich der von Ihnen angegebene Schlüssel in einem anderen Konto befindet als dem, das Sie zur Konfiguration eines Clusters verwenden, müssen Sie den Schlüssel mit seinem ARN angeben.

Die Rolle für das EC2 Amazon-Instance-Profil muss über Berechtigungen zur Verwendung des von Ihnen angegebenen KMS Schlüssels verfügen. Die Standardrolle für das Instance-Profil in Amazon EMR ist`EMR_EC2_DefaultRole`. Wenn Sie eine andere Rolle für das Instance-Profil oder IAM Rollen für EMRFS Anfragen an Amazon S3 verwenden, stellen Sie sicher, dass jede Rolle je nach Bedarf als Schlüsselbenutzer hinzugefügt wird. Dadurch erhält die Rolle die Erlaubnis, den KMS Schlüssel zu verwenden. Weitere Informationen finden Sie unter Verwenden wichtiger Richtlinien im AWS Key Management Service Entwicklerhandbuch und Konfigurieren von IAM Rollen für EMRFS Anfragen an Amazon S3.

Sie können das verwenden AWS Management Console , um Ihr Instance-Profil oder EC2 Instance-Profil zur Liste der Schlüsselbenutzer für den angegebenen KMS Schlüssel hinzuzufügen, oder Sie können das AWS CLI oder an verwenden, AWS SDK um eine entsprechende Schlüsselrichtlinie anzuhängen.

Beachten Sie, dass Amazon nur symmetrische KMS Schlüssel EMR unterstützt. Sie können keinen asymmetrischen KMS Schlüssel verwenden, um ruhende Daten in einem EMR Amazon-Cluster zu verschlüsseln. Hilfe bei der Bestimmung, ob ein KMS Schlüssel symmetrisch oder asymmetrisch ist, finden Sie unter Identifizieren symmetrischer und asymmetrischer Schlüssel. KMS

Im Folgenden wird beschrieben, wie Sie das standardmäßige EMR Amazon-Instance-Profil `EMR_EC2_DefaultRole` als Hauptbenutzer mithilfe von hinzufügen AWS Management Console. Es wird davon ausgegangen, dass Sie bereits einen KMS Schlüssel erstellt haben. Informationen zum Erstellen eines neuen KMS Schlüssels finden Sie unter Schlüssel erstellen im AWS Key Management Service Entwicklerhandbuch.

Um das EC2 Instance-Profil für Amazon EMR zur Liste der Benutzer von Verschlüsselungsschlüsseln hinzuzufügen

1. Melden Sie sich bei der AWS Key Management Service (AWS KMS) -Konsole an AWS Management Console und öffnen Sie sie unter https://console.aws.amazon.com/kms.
2. Um das zu ändern AWS-Region, verwenden Sie die Regionsauswahl in der oberen rechten Ecke der Seite.
3. Wählen Sie den Alias des KMS Schlüssels aus, den Sie ändern möchten.
4. Wählen Sie auf der Seite mit den Schlüsseldetails unter Key Users (Schlüsselbenutzer(die Option Add (Hinzufügen) aus.
5. Wählen Sie die entsprechende Rolle im Dialogfeld Add key users (Schlüsselbenutzer hinzufügen) aus. Der Name der Standardrolle lautet `EMR_EC2_DefaultRole`.
6. Wählen Sie Hinzufügen aus.

Aktivierung der EBS Verschlüsselung durch Bereitstellung zusätzlicher Berechtigungen für KMS Schlüssel

Ab EMR Amazon-Version 5.24.0 können Sie EBS Root-Geräte und Speichervolumes mithilfe einer Sicherheitskonfigurationsoption verschlüsseln. Um diese Option zu aktivieren, müssen Sie Ihren Schlüsselanbieter angeben AWS KMS . Darüber hinaus müssen Sie der Servicerolle die von `EMR_DefaultRole` Ihnen angegebenen Berechtigungen zur Verwendung der von AWS KMS key Ihnen angegebenen Rechte erteilen.

Sie können das verwenden AWS Management Console , um die Servicerolle zur Liste der Schlüsselbenutzer für den angegebenen KMS Schlüssel hinzuzufügen, oder Sie können das AWS CLI oder ein verwenden, AWS SDK um eine entsprechende Schlüsselrichtlinie anzuhängen.

Das folgende Verfahren beschreibt, wie Sie AWS Management Console die standardmäßige EMR Amazon-Servicerolle `EMR_DefaultRole` als Schlüsselbenutzer hinzufügen können. Es wird davon ausgegangen, dass Sie bereits einen KMS Schlüssel erstellt haben. Informationen zum Erstellen eines neuen KMS Schlüssels finden Sie unter Schlüssel erstellen im AWS Key Management Service Entwicklerhandbuch.

Um die EMR Amazon-Servicerolle zur Liste der Benutzer von Verschlüsselungsschlüsseln hinzuzufügen

1. Melden Sie sich bei der AWS Key Management Service (AWS KMS) -Konsole an AWS Management Console und öffnen Sie sie unter https://console.aws.amazon.com/kms.

2. Um das zu ändern AWS-Region, verwenden Sie die Regionsauswahl in der oberen rechten Ecke der Seite.
3. Wählen Sie links Vom Kunden verwaltete Schlüssel aus.
4. Wählen Sie den Alias des KMS Schlüssels aus, den Sie ändern möchten.
5. Wählen Sie auf der Seite mit den Schlüsseldetails unter Key Users (Schlüsselbenutzer(die Option Add (Hinzufügen) aus.
6. Wählen Sie im Abschnitt Schlüsselbenutzer hinzufügen die entsprechende Rolle aus. Der Name der Standard-Servicerolle für Amazon EMR lautet`EMR_DefaultRole`.
7. Wählen Sie Hinzufügen aus.

Erstellen eines benutzerdefinierten Schlüsselanbieters

Wenn Sie eine Sicherheitskonfiguration verwenden, müssen Sie einen anderen Anbieterklassennamen für die Verschlüsselung lokaler Datenträger und die Amazon-S3-Verschlüsselung angeben. Die Anforderungen für den benutzerdefinierten Schlüsselanbieter hängen davon ab, ob Sie die lokale Festplattenverschlüsselung und die Amazon S3 S3-Verschlüsselung sowie die EMR Amazon-Release-Version verwenden.

Abhängig von der Art der Verschlüsselung, die Sie bei der Erstellung eines benutzerdefinierten Schlüsselanbieters verwenden, muss die Anwendung auch unterschiedliche EncryptionMaterialsProvider Schnittstellen implementieren. Beide Schnittstellen sind in der Version 1.11.0 AWS SDK für Java und höher verfügbar.

- Um die Amazon S3 S3-Verschlüsselung zu implementieren, verwenden Sie das Modell com.amazonaws.services.s3.model. EncryptionMaterialsProvider Schnittstelle.
- Verwenden Sie die Datei com.amazonaws.services.elasticmapreduce.spi.security, um die lokale Festplattenverschlüsselung zu implementieren. EncryptionMaterialsProvider Schnittstelle.

Sie können jede Strategie verwenden, um Verschlüsselungsmaterial für die Implementierung bereitzustellen. Sie können sich beispielsweise dafür entscheiden, statisches Verschlüsselungsmaterial bereitzustellen oder es in ein komplexeres Schlüsselverwaltungssystem zu integrieren.

Wenn Sie die Amazon S3 S3-Verschlüsselung verwenden, müssen Sie die Verschlüsselungsalgorithmen AES/GCM/NoPaddingfür benutzerdefinierte Verschlüsselungsmaterialien verwenden.

Wenn Sie die lokale Festplattenverschlüsselung verwenden, variiert der Verschlüsselungsalgorithmus, der für benutzerdefinierte Verschlüsselungsmaterialien verwendet werden soll, je nach EMR Version. Für Amazon EMR 7.0.0 und niedriger müssen Sie AES/GCM/NoPaddingverwenden. Für Amazon EMR 7.1.0 und höher müssen Sie verwenden AES.

Die EncryptionMaterialsProvider Klasse ruft Verschlüsselungsmaterial nach Verschlüsselungskontext ab. Amazon EMR füllt zur Laufzeit Informationen zum Verschlüsselungskontext aus, damit der Anrufer die richtigen Verschlüsselungsmaterialien für die Rücksendung ermitteln kann.

Example Beispiel: Verwendung eines benutzerdefinierten Schlüsselanbieters für die Amazon S3 S3-Verschlüsselung mit EMRFS

Wenn Amazon die Verschlüsselungsmaterialien von der EncryptionMaterialsProvider Klasse EMR abruft, um die Verschlüsselung durchzuführen, füllt das materialsDescription Argument EMRFS optional mit zwei Feldern auf: dem Amazon S3 URI für das Objekt und dem des Clusters, die JobFlowId von der EncryptionMaterialsProvider Klasse verwendet werden können, um Verschlüsselungsmaterialien selektiv zurückzugeben.

Beispielsweise kann der Anbieter unterschiedliche Schlüssel für verschiedene Amazon S3 URI S3-Präfixe zurückgeben. Es ist die Beschreibung der zurückgegebenen Verschlüsselungsmaterialien, die letztendlich mit dem Amazon S3 S3-Objekt gespeichert werden, und nicht der materialsDescription Wert, der vom Anbieter generiert EMRFS und an diesen weitergegeben wird. Beim Entschlüsseln eines Amazon S3 S3-Objekts wird die Beschreibung des Verschlüsselungsmaterials an die EncryptionMaterialsProvider Klasse übergeben, sodass sie wiederum selektiv den passenden Schlüssel zur Entschlüsselung des Objekts zurückgeben kann.

Eine EncryptionMaterialsProvider Referenzimplementierung finden Sie weiter unten. Ein weiterer benutzerdefinierter Anbieter EMRFSRSAEncryptionMaterialsProvider,, ist erhältlich bei GitHub.

```
import com.amazonaws.services.s3.model.EncryptionMaterials;
import com.amazonaws.services.s3.model.EncryptionMaterialsProvider;
import com.amazonaws.services.s3.model.KMSEncryptionMaterials;
import org.apache.hadoop.conf.Configurable;
import org.apache.hadoop.conf.Configuration;

import java.util.Map;

/**
 * Provides KMSEncryptionMaterials according to Configuration
 */
```

```
public class MyEncryptionMaterialsProviders implements EncryptionMaterialsProvider,
 Configurable{
  private Configuration conf;
  private String kmsKeyId;
  private EncryptionMaterials encryptionMaterials;

  private void init() {
    this.kmsKeyId = conf.get("my.kms.key.id");
    this.encryptionMaterials = new KMSEncryptionMaterials(kmsKeyId);
  }

  @Override
  public void setConf(Configuration conf) {
    this.conf = conf;
    init();
  }

  @Override
  public Configuration getConf() {
    return this.conf;
  }

  @Override
  public void refresh() {

  }

  @Override
  public EncryptionMaterials getEncryptionMaterials(Map<String, String>
 materialsDescription) {
    return this.encryptionMaterials;
  }

  @Override
  public EncryptionMaterials getEncryptionMaterials() {
    return this.encryptionMaterials;
  }
}
```

Bereitstellung von Zertifikaten für die Verschlüsselung von Daten während der Übertragung mit EMR Amazon-Verschlüsselung

Mit der EMR Amazon-Version 4.8.0 oder höher haben Sie zwei Möglichkeiten, Artefakte für die Verschlüsselung von Daten während der Übertragung mithilfe einer Sicherheitskonfiguration anzugeben:

- Sie können PEM Zertifikate manuell erstellen, sie in eine ZIP-Datei aufnehmen und dann in Amazon S3 auf die .zip-Datei verweisen.
- Sie können einen benutzerdefinierten Zertifikatanbieter als Java-Klasse implementieren. Sie geben die JAR Datei der Anwendung in Amazon S3 an und geben dann den vollständigen Klassennamen des Anbieters an, wie er in der Anwendung deklariert ist. Die Klasse muss die TLSArtifactsProviderSchnittstelle implementieren, die ab AWS SDK for Java Version 1.11.0 verfügbar ist.

Amazon lädt EMR automatisch Artefakte auf jeden Knoten im Cluster herunter und verwendet sie später, um die Open-Source-Verschlüsselungsfunktionen für die Übertragung zu implementieren. Weitere Informationen zu den verfügbaren Optionen finden Sie unter Verschlüsselung während der Übertragung.

Verwendung von Zertifikaten PEM

Wenn Sie eine .zip-Datei für die Verschlüsselung während der Übertragung angeben, erwartet die Sicherheitskonfiguration, dass die PEM Dateien in der .zip-Datei genau so benannt werden, wie sie unten angezeigt werden:

Zertifikate für Verschlüsselung von Daten während der Übertragung

Dateiname	Erforderlich/optional	Details
privateKey.pem	Erforderlich	Privater Schlüssel
certificateChain.pem	Erforderlich	Zertifikatskette
trustedCertificates.pem	Optional	Erforderlich, wenn das bereitgestellte Zertifikat nicht entweder von der standardmäßigen vertrauenswürdige

Dateiname	Erforderlich/optional	Details
		Java-Stammzertifizierungsstelle (Certification Authority, CA) oder einer CA-Zwischenzertifizierungsstelle, die eine Verbindung zur Java-Standard-Stammzertifizierungsstelle herstellen kann, signiert wurde. Das standardmäßige vertrauenswürdige Stammverzeichnis von Java CAs finden Sie in. `jre/lib/security/cacerts`

Wahrscheinlich möchten Sie die private PEM Schlüsseldatei als Platzhalterzertifikat konfigurieren, das den Zugriff auf die VPC Amazon-Domain ermöglicht, in der sich Ihre Cluster-Instances befinden. Wenn sich Ihr Cluster beispielsweise in der Region us-east-1 (N. Virginia) befindet, könnten Sie in der Zertifikatskonfiguration einen allgemeinen Namen angeben, der durch die Angabe von `CN=*.ec2.internal` in der Zertifikatsubjektdefinition Zugriff auf den Cluster gewährt. Wenn sich Ihr Cluster in der Region us-west-2 (Oregon) befindet, könnten Sie `CN=*.us-west-2.compute.internal` angeben.

Wenn die bereitgestellte PEM Datei im Verschlüsselungsartefakt kein Platzhalterzeichen in der CN für die Domain enthält, müssen Sie den Wert von to ändern. `hadoop.ssl.hostname.verifier` `ALLOW_ALL` Dies erfolgt mit der `core-site` Klassifizierung beim Senden von Konfigurationen an einen Cluster oder durch Hinzufügen dieses Werts zur Datei `core-site.xml`. Diese Änderung ist erforderlich, da die standardmäßige Hostnamen-Verifizierung keinen Hostnamen ohne Platzhalter akzeptiert, was zu einem Fehler führt. Weitere Informationen zur EMR Cluster-Konfiguration innerhalb eines Amazon VPC finden Sie unterNetzwerk konfigurieren.

Das folgende Beispiel zeigt, wie Open SSL verwendet wird, um ein selbstsigniertes X.509-Zertifikat mit einem privaten RSA 1024-Bit-Schlüssel zu generieren. Der Schlüssel ermöglicht den Zugriff auf die EMR Amazon-Cluster-Instances des Emittenten in der Region `us-west-2` (Oregon), wie durch den *`*.us-west-2.compute.internal`* Domainnamen als allgemeinen Namen angegeben.

Es können weitere optionale Subjektelemente wie Land (Country, C), Status (Status, S), Gebietsschema (Locale, L) usw. angegeben werden. Da ein selbstsigniertes Zertifikat generiert wird, kopiert der zweite Befehl im Beispiel die Datei `certificateChain.pem` zur Datei `trustedCertificates.pem`. Der dritte Befehl verwendet `zip` zum Erstellen der Datei `my-certs.zip`, die die Zertifikate enthält.

> **Important**
>
> Dieses Beispiel dient nur zur proof-of-concept Veranschaulichung. Die Verwendung von selbstsignierten Zertifikaten wird nicht empfohlen und stellt ein potenzielles Sicherheitsrisiko dar. Verwenden Sie eine vertrauenswürdige Zertifizierungsstelle (CA), um die Zertifikate für Produktionssysteme auszustellen.

```
$ openssl req -x509 -newkey rsa:1024 -keyout privateKey.pem -out certificateChain.pem
 -days 365 -nodes -subj '/C=US/ST=Washington/L=Seattle/O=MyOrg/OU=MyDept/CN=*.us-
west-2.compute.internal'
$ cp certificateChain.pem trustedCertificates.pem
$ zip -r -X my-certs.zip certificateChain.pem privateKey.pem trustedCertificates.pem
```

AWS Identity and Access Management für Amazon EMR

AWS Identity and Access Management (IAM) hilft einem Administrator AWS -Service , den Zugriff auf AWS Ressourcen sicher zu kontrollieren. IAMAdministratoren kontrollieren, wer authentifiziert (angemeldet) und autorisiert werden kann (über Berechtigungen verfügt), um EMR Amazon-Ressourcen zu nutzen. IAMist eine AWS -Service , die Sie ohne zusätzliche Kosten verwenden können.

Themen

- Zielgruppe
- Authentifizierung mit Identitäten
- Verwalten des Zugriffs mit Richtlinien
- So EMR arbeitet Amazon mit IAM
- EMRSchritte zu Runtime-Rollen für Amazon
- IAMServicerollen für EMR Amazon-Berechtigungen für AWS Dienste und Ressourcen konfigurieren

- Beispiele für EMR identitätsbasierte Richtlinien von Amazon

Zielgruppe

Wie Sie AWS Identity and Access Management (IAM) verwenden, hängt von der Arbeit ab, die Sie bei Amazon erledigenEMR.

Servicebenutzer — Wenn Sie den EMR Amazon-Service für Ihre Arbeit verwenden, stellt Ihnen Ihr Administrator die Anmeldeinformationen und Berechtigungen zur Verfügung, die Sie benötigen. Da Sie für Ihre Arbeit mehr EMR Amazon-Funktionen verwenden, benötigen Sie möglicherweise zusätzliche Berechtigungen. Wenn Sie die Fuktionsweise der Zugriffskontrolle nachvollziehen, wissen Sie bereits, welche Berechtigungen Sie von Ihrem Administrator anzufordern müssen. Wenn Sie auf eine Funktion in Amazon nicht zugreifen könnenEMR, finden Sie weitere Informationen unterFehlerbehebung Amazon EMR Amazon-Identität und -Zugriff.

Service-Administrator — Wenn Sie in Ihrem Unternehmen für die EMR Amazon-Ressourcen verantwortlich sind, haben Sie wahrscheinlich vollen Zugriff auf AmazonEMR. Es ist Ihre Aufgabe, zu bestimmen, auf welche EMR Amazon-Funktionen und -Ressourcen Ihre Servicebenutzer zugreifen sollen. Anschließend müssen Sie Anfragen an Ihren IAM Administrator richten, um die Berechtigungen Ihrer Servicebenutzer zu ändern. Lesen Sie die Informationen auf dieser Seite, um die grundlegenden Konzepte von zu verstehenIAM. Weitere Informationen darüber, wie Ihr Unternehmen Amazon nutzen IAM kannEMR, finden Sie unterSo EMR arbeitet Amazon mit IAM.

IAMAdministrator — Wenn Sie ein IAM Administrator sind, möchten Sie vielleicht mehr darüber erfahren, wie Sie Richtlinien schreiben können, um den Zugriff auf Amazon zu verwaltenEMR. Beispiele für EMR identitätsbasierte Amazon-Richtlinien, die Sie in verwenden könnenIAM, finden Sie unter. Beispiele für EMR identitätsbasierte Richtlinien von Amazon

Authentifizierung mit Identitäten

Authentifizierung ist die Art und Weise, wie Sie sich AWS mit Ihren Identitätsdaten anmelden. Sie müssen als IAM Benutzer authentifiziert (angemeldet AWS) sein oder eine IAM Rolle übernehmen. Root-Benutzer des AWS-Kontos

Sie können sich AWS als föderierte Identität anmelden, indem Sie Anmeldeinformationen verwenden, die über eine Identitätsquelle bereitgestellt wurden. AWS IAM Identity Center (IAMIdentity Center-) Nutzer, die Single-Sign-On-Authentifizierung Ihres Unternehmens und Ihre Google- oder Facebook-Anmeldeinformationen sind Beispiele für föderierte Identitäten. Wenn Sie sich als föderierte Identität

anmelden, hat Ihr Administrator zuvor einen Identitätsverbund mithilfe von Rollen eingerichtet. IAM Wenn Sie AWS mithilfe eines Verbunds darauf zugreifen, übernehmen Sie indirekt eine Rolle.

Je nachdem, welcher Benutzertyp Sie sind, können Sie sich beim AWS Management Console oder beim AWS Zugangsportal anmelden. Weitere Informationen zur Anmeldung finden Sie AWS unter So melden Sie sich bei Ihrem an AWS-Konto im AWS-Anmeldung Benutzerhandbuch.

Wenn Sie AWS programmgesteuert darauf zugreifen, AWS stellt es ein Software Development Kit (SDK) und eine Befehlszeilenschnittstelle (CLI) bereit, mit der Sie Ihre Anfragen mithilfe Ihrer Anmeldeinformationen kryptografisch signieren können. Wenn Sie keine AWS Tools verwenden, müssen Sie Anfragen selbst signieren. Weitere Informationen zur Verwendung der empfohlenen Methode, um Anfragen selbst zu signieren, finden Sie im IAMBenutzerhandbuch unter AWS API Anfragen signieren.

Unabhängig von der verwendeten Authentifizierungsmethode müssen Sie möglicherweise zusätzliche Sicherheitsinformationen angeben. AWS Empfiehlt beispielsweise, die Multi-Faktor-Authentifizierung (MFA) zu verwenden, um die Sicherheit Ihres Kontos zu erhöhen. Weitere Informationen finden Sie unter Multi-Faktor-Authentifizierung im AWS IAM Identity Center Benutzerhandbuch und Verwenden der Multi-Faktor-Authentifizierung (MFA) AWS im IAM Benutzerhandbuch.

AWS-Konto Root-Benutzer

Wenn Sie einen erstellen AWS-Konto, beginnen Sie mit einer Anmeldeidentität, die vollständigen Zugriff auf alle AWS -Services Ressourcen im Konto hat. Diese Identität wird als AWS-Konto Root-Benutzer bezeichnet. Sie können darauf zugreifen, indem Sie sich mit der E-Mail-Adresse und dem Passwort anmelden, mit denen Sie das Konto erstellt haben. Wir raten ausdrücklich davon ab, den Root-Benutzer für Alltagsaufgaben zu verwenden. Schützen Sie Ihre Root-Benutzer-Anmeldeinformationen und verwenden Sie diese, um die Aufgaben auszuführen, die nur der Root-Benutzer ausführen kann. Eine vollständige Liste der Aufgaben, für die Sie sich als Root-Benutzer anmelden müssen, finden Sie im Benutzerhandbuch unter Aufgaben, für die Root-Benutzeranmeldedaten erforderlich sind. IAM

Verbundidentität

Als bewährte Methode sollten menschliche Benutzer, einschließlich Benutzer, die Administratorzugriff benötigen, für den Zugriff AWS -Services mithilfe temporärer Anmeldeinformationen den Verbund mit einem Identitätsanbieter verwenden.

Eine föderierte Identität ist ein Benutzer aus Ihrem Unternehmensbenutzerverzeichnis, einem Web-Identitätsanbieter AWS Directory Service, dem Identity Center-Verzeichnis oder einem beliebigen Benutzer, der mithilfe AWS -Services von Anmeldeinformationen zugreift, die über eine Identitätsquelle bereitgestellt wurden. Wenn föderierte Identitäten darauf zugreifen AWS-Konten, übernehmen sie Rollen, und die Rollen stellen temporäre Anmeldeinformationen bereit.

Für die zentrale Zugriffsverwaltung empfehlen wir Ihnen, AWS IAM Identity Center zu verwenden. Sie können Benutzer und Gruppen in IAM Identity Center erstellen, oder Sie können eine Verbindung zu einer Gruppe von Benutzern und Gruppen in Ihrer eigenen Identitätsquelle herstellen und diese synchronisieren, um sie in all Ihren AWS-Konten Anwendungen zu verwenden. Informationen zu IAM Identity Center finden Sie unter Was ist IAM Identity Center? im AWS IAM Identity Center Benutzerhandbuch.

IAM-Benutzer und -Gruppen

Ein IAMBenutzer ist eine Identität innerhalb Ihres Unternehmens AWS-Konto , die über spezifische Berechtigungen für eine einzelne Person oder Anwendung verfügt. Wir empfehlen, sich nach Möglichkeit auf temporäre Anmeldeinformationen zu verlassen, anstatt IAM Benutzer mit langfristigen Anmeldeinformationen wie Passwörtern und Zugriffsschlüsseln zu erstellen. Wenn Sie jedoch spezielle Anwendungsfälle haben, für die langfristige Anmeldeinformationen von IAM Benutzern erforderlich sind, empfehlen wir, die Zugriffsschlüssel abwechselnd zu verwenden. Weitere Informationen finden Sie im Benutzerhandbuch unter Regelmäßiges Rotieren von Zugriffsschlüsseln für Anwendungsfälle, für die IAM langfristige Anmeldeinformationen erforderlich sind.

Eine IAMGruppe ist eine Identität, die eine Sammlung von IAM Benutzern spezifiziert. Sie können sich nicht als Gruppe anmelden. Mithilfe von Gruppen können Sie Berechtigungen für mehrere Benutzer gleichzeitig angeben. Gruppen vereinfachen die Verwaltung von Berechtigungen, wenn es zahlreiche Benutzer gibt. Sie könnten beispielsweise eine Gruppe benennen IAMAdminsund dieser Gruppe Berechtigungen zur Verwaltung von IAM Ressourcen erteilen.

Benutzer unterscheiden sich von Rollen. Ein Benutzer ist einer einzigen Person oder Anwendung eindeutig zugeordnet. Eine Rolle kann von allen Personen angenommen werden, die sie benötigen. Benutzer besitzen dauerhafte Anmeldeinformationen. Rollen stellen temporäre Anmeldeinformationen bereit. Weitere Informationen finden Sie unter Wann sollte ein IAM Benutzer (statt einer Rolle) erstellt werden? im IAMBenutzerhandbuch.

IAMRollen

Eine IAMRolle ist eine Identität innerhalb von Ihnen AWS-Konto , für die bestimmte Berechtigungen gelten. Sie ähnelt einem IAM Benutzer, ist jedoch keiner bestimmten Person zugeordnet. Sie können vorübergehend eine IAM Rolle in der übernehmen, AWS Management Console indem Sie die Rollen wechseln. Sie können eine Rolle übernehmen, indem Sie eine AWS CLI AWS API OR-Operation aufrufen oder eine benutzerdefinierte Operation verwendenURL. Weitere Informationen zu Methoden zur Verwendung von Rollen finden Sie unter Verwenden von IAM Rollen im IAMBenutzerhandbuch.

IAMRollen mit temporären Anmeldeinformationen sind in den folgenden Situationen nützlich:

- Verbundbenutzerzugriff – Um einer Verbundidentität Berechtigungen zuzuweisen, erstellen Sie eine Rolle und definieren Berechtigungen für die Rolle. Wird eine Verbundidentität authentifiziert, so wird die Identität der Rolle zugeordnet und erhält die von der Rolle definierten Berechtigungen. Informationen zu Rollen für den Verbund finden Sie im IAMBenutzerhandbuch unter Erstellen einer Rolle für einen externen Identitätsanbieter. Wenn Sie IAM Identity Center verwenden, konfigurieren Sie einen Berechtigungssatz. Um zu kontrollieren, worauf Ihre Identitäten nach der Authentifizierung zugreifen können, korreliert IAM Identity Center den Berechtigungssatz mit einer Rolle in. IAM Informationen zu Berechtigungssätzen finden Sie unter Berechtigungssätze im AWS IAM Identity Center -Benutzerhandbuch.
- Temporäre IAM Benutzerberechtigungen — Ein IAM Benutzer oder eine Rolle kann eine IAM Rolle übernehmen, um vorübergehend verschiedene Berechtigungen für eine bestimmte Aufgabe zu übernehmen.
- Kontoübergreifender Zugriff — Sie können eine IAM Rolle verwenden, um jemandem (einem vertrauenswürdigen Principal) in einem anderen Konto den Zugriff auf Ressourcen in Ihrem Konto zu ermöglichen. Rollen stellen die primäre Möglichkeit dar, um kontoübergreifendem Zugriff zu gewähren. Bei einigen können Sie AWS -Services jedoch eine Richtlinie direkt an eine Ressource anhängen (anstatt eine Rolle als Proxy zu verwenden). Informationen zum Unterschied zwischen Rollen und ressourcenbasierten Richtlinien für den kontenübergreifenden Zugriff finden Sie IAMim Benutzerhandbuch unter Kontoübergreifender Ressourcenzugriff. IAM
- Serviceübergreifender Zugriff — Einige AWS -Services verwenden Funktionen in anderen. AWS -Services Wenn Sie beispielsweise einen Service aufrufen, ist es üblich, dass dieser Service Anwendungen in Amazon ausführt EC2 oder Objekte in Amazon S3 speichert. Ein Dienst kann dies mit den Berechtigungen des aufrufenden Prinzipals mit einer Servicerolle oder mit einer serviceverknüpften Rolle tun.
  - Zugriffssitzungen weiterleiten (FAS) — Wenn Sie einen IAM Benutzer oder eine Rolle verwenden, um Aktionen auszuführen AWS, gelten Sie als Principal. Bei einigen Services

könnte es Aktionen geben, die dann eine andere Aktion in einem anderen Service initiieren. FASverwendet die Berechtigungen des Prinzipals, der an aufruft AWS -Service, kombiniert mit der Anforderung, Anfragen AWS -Service an nachgelagerte Dienste zu stellen. FASAnfragen werden nur gestellt, wenn ein Dienst eine Anfrage erhält, für deren Abschluss Interaktionen mit anderen AWS -Services oder Ressourcen erforderlich sind. In diesem Fall müssen Sie über Berechtigungen zum Ausführen beider Aktionen verfügen. Einzelheiten zu den Richtlinien beim Stellen von FAS Anfragen finden Sie unter Zugriffssitzungen weiterleiten.

- Servicerolle — Eine Servicerolle ist eine IAMRolle, die ein Dienst übernimmt, um Aktionen in Ihrem Namen auszuführen. Ein IAM Administrator kann eine Servicerolle von innen heraus erstellen, ändern und löschenIAM. Weitere Informationen finden Sie im IAMBenutzerhandbuch unter Erstellen einer Rolle zum Delegieren von Berechtigungen AWS -Service an eine.
- Dienstbezogene Rolle — Eine dienstverknüpfte Rolle ist eine Art von Servicerolle, die mit einer verknüpft ist. AWS -Service Der Service kann die Rolle übernehmen, um eine Aktion in Ihrem Namen auszuführen. Servicebezogene Rollen erscheinen in Ihrem Dienst AWS-Konto und gehören dem Dienst. Ein IAM Administrator kann die Berechtigungen für dienstbezogene Rollen anzeigen, aber nicht bearbeiten.

- Auf Amazon ausgeführte Anwendungen EC2 — Sie können eine IAM Rolle verwenden, um temporäre Anmeldeinformationen für Anwendungen zu verwalten, die auf einer EC2 Instance ausgeführt werden und AWS API Anfragen stellen AWS CLI . Dies ist dem Speichern von Zugriffsschlüsseln innerhalb der EC2 Instance vorzuziehen. Um einer EC2 Instanz eine AWS Rolle zuzuweisen und sie allen ihren Anwendungen zur Verfügung zu stellen, erstellen Sie ein Instanzprofil, das an die Instanz angehängt ist. Ein Instanzprofil enthält die Rolle und ermöglicht Programmen, die auf der EC2 Instanz ausgeführt werden, temporäre Anmeldeinformationen abzurufen. Weitere Informationen finden Sie im IAMBenutzerhandbuch unter Verwenden einer IAM Rolle zur Erteilung von Berechtigungen für Anwendungen, die auf EC2 Amazon-Instances ausgeführt werden.

Informationen darüber, ob Sie IAM Rollen oder IAM Benutzer verwenden sollten, finden Sie im Benutzerhandbuch unter Wann sollte eine IAM Rolle (anstelle eines IAM Benutzers) erstellt werden.

Verwalten des Zugriffs mit Richtlinien

Sie steuern den Zugriff, AWS indem Sie Richtlinien erstellen und diese an AWS Identitäten oder Ressourcen anhängen. Eine Richtlinie ist ein Objekt, AWS das, wenn es einer Identität oder Ressource zugeordnet ist, deren Berechtigungen definiert. AWS wertet diese Richtlinien aus, wenn ein Prinzipal (Benutzer, Root-Benutzer oder Rollensitzung) eine Anfrage stellt. Berechtigungen in den

Richtlinien bestimmen, ob die Anforderung zugelassen oder abgelehnt wird. Die meisten Richtlinien werden in AWS Form von JSON Dokumenten gespeichert. Weitere Informationen zur Struktur und zum Inhalt von JSON Richtliniendokumenten finden Sie im IAMBenutzerhandbuch unter Überblick über JSON Richtlinien.

Administratoren können mithilfe von AWS JSON Richtlinien festlegen, wer Zugriff auf was hat. Das bedeutet, welcher Prinzipal kann Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen.

Standardmäßig haben Benutzer, Gruppen und Rollen keine Berechtigungen. Um Benutzern die Erlaubnis zu erteilen, Aktionen mit den Ressourcen durchzuführen, die sie benötigen, kann ein IAM Administrator IAM Richtlinien erstellen. Der Administrator kann dann die IAM Richtlinien zu Rollen hinzufügen, und Benutzer können die Rollen übernehmen.

IAMRichtlinien definieren Berechtigungen für eine Aktion, unabhängig von der Methode, mit der Sie den Vorgang ausführen. Angenommen, es gibt eine Richtlinie, die Berechtigungen für die `iam:GetRole`-Aktion erteilt. Ein Benutzer mit dieser Richtlinie kann Rolleninformationen aus dem AWS Management Console AWS CLI, dem oder dem abrufen AWS API.

Identitätsbasierte Richtlinien

Identitätsbasierte Richtlinien sind Dokumente mit JSON Berechtigungsrichtlinien, die Sie an eine Identität anhängen können, z. B. an einen IAM Benutzer, eine Benutzergruppe oder eine Rolle. Diese Richtlinien steuern, welche Aktionen die Benutzer und Rollen für welche Ressourcen und unter welchen Bedingungen ausführen können. Informationen zum Erstellen einer identitätsbasierten Richtlinie finden Sie unter IAMRichtlinien erstellen im Benutzerhandbuch. IAM

Identitätsbasierte Richtlinien können weiter als Inline-Richtlinien oder verwaltete Richtlinien kategorisiert werden. Inline-Richtlinien sind direkt in einen einzelnen Benutzer, eine einzelne Gruppe oder eine einzelne Rolle eingebettet. Verwaltete Richtlinien sind eigenständige Richtlinien, die Sie mehreren Benutzern, Gruppen und Rollen in Ihrem System zuordnen können. AWS-Konto Zu den verwalteten Richtlinien gehören AWS verwaltete Richtlinien und vom Kunden verwaltete Richtlinien. Informationen dazu, wie Sie zwischen einer verwalteten Richtlinie oder einer Inline-Richtlinie wählen können, finden Sie im IAMBenutzerhandbuch unter Auswahl zwischen verwalteten Richtlinien und Inline-Richtlinien.

Ressourcenbasierte Richtlinien

Ressourcenbasierte Richtlinien sind JSON Richtliniendokumente, die Sie an eine Ressource anhängen. Beispiele für ressourcenbasierte Richtlinien sind IAM Rollenvertrauensrichtlinien und

Amazon S3 S3-Bucket-Richtlinien. In Services, die ressourcenbasierte Richtlinien unterstützen, können Service-Administratoren sie verwenden, um den Zugriff auf eine bestimmte Ressource zu steuern. Für die Ressource, an welche die Richtlinie angehängt ist, legt die Richtlinie fest, welche Aktionen ein bestimmter Prinzipal unter welchen Bedingungen für diese Ressource ausführen kann. Sie müssen in einer ressourcenbasierten Richtlinie einen Prinzipal angeben. Zu den Prinzipalen können Konten, Benutzer, Rollen, Verbundbenutzer oder gehören. AWS -Services

Ressourcenbasierte Richtlinien sind Richtlinien innerhalb dieses Diensts. Sie können AWS verwaltete Richtlinien nicht IAM in einer ressourcenbasierten Richtlinie verwenden.

Zugriffskontrolllisten () ACLs

Zugriffskontrolllisten (ACLs) steuern, welche Principals (Kontomitglieder, Benutzer oder Rollen) über Zugriffsberechtigungen für eine Ressource verfügen. ACLsähneln ressourcenbasierten Richtlinien, verwenden jedoch nicht das JSON Richtliniendokumentformat.

Amazon S3 und AWS WAF Amazon VPC sind Beispiele für Dienste, die Unterstützung bietenACLs. Weitere Informationen finden Sie unter Übersicht über ACLs die Zugriffskontrollliste (ACL) im Amazon Simple Storage Service Developer Guide.

Weitere Richtlinientypen

AWS unterstützt zusätzliche, weniger verbreitete Richtlinientypen. Diese Richtlinientypen können die maximalen Berechtigungen festlegen, die Ihnen von den häufiger verwendeten Richtlinientypen erteilt werden können.

- Berechtigungsgrenzen — Eine Berechtigungsgrenze ist eine erweiterte Funktion, mit der Sie die maximalen Berechtigungen festlegen, die eine identitätsbasierte Richtlinie einer IAM Entität (IAMBenutzer oder Rolle) gewähren kann. Sie können eine Berechtigungsgrenze für eine Entität festlegen. Die daraus resultierenden Berechtigungen sind der Schnittpunkt der identitätsbasierten Richtlinien einer Entität und ihrer Berechtigungsgrenzen. Ressourcenbasierte Richtlinien, die den Benutzer oder die Rolle im Feld `Principal` angeben, werden nicht durch Berechtigungsgrenzen eingeschränkt. Eine explizite Zugriffsverweigerung in einer dieser Richtlinien setzt eine Zugriffserlaubnis außer Kraft. Weitere Informationen zu Berechtigungsgrenzen finden Sie im IAMBenutzerhandbuch unter Berechtigungsgrenzen für IAM Entitäten.
- Dienststeuerungsrichtlinien (SCPs) — SCPs sind JSON Richtlinien, die die maximalen Berechtigungen für eine Organisation oder Organisationseinheit (OU) in festlegen AWS Organizations. AWS Organizations ist ein Dienst zur Gruppierung und zentralen Verwaltung

mehrerer AWS-Konten Unternehmenseigentümer. Wenn Sie alle Funktionen in einer Organisation aktivieren, können Sie Richtlinien zur Servicesteuerung (SCPs) auf einige oder alle Ihre Konten anwenden. Das SCP schränkt die Berechtigungen für Entitäten in Mitgliedskonten ein, einschließlich der einzelnen Root-Benutzer des AWS-Kontos. Weitere Informationen zu Organizations und SCPs finden Sie unter Richtlinien zur Servicesteuerung im AWS Organizations Benutzerhandbuch.

- Sitzungsrichtlinien – Sitzungsrichtlinien sind erweiterte Richtlinien, die Sie als Parameter übergeben, wenn Sie eine temporäre Sitzung für eine Rolle oder einen verbundenen Benutzer programmgesteuert erstellen. Die resultierenden Sitzungsberechtigungen sind eine Schnittmenge der auf der Identität des Benutzers oder der Rolle basierenden Richtlinien und der Sitzungsrichtlinien. Berechtigungen können auch aus einer ressourcenbasierten Richtlinie stammen. Eine explizite Zugriffsverweigerung in einer dieser Richtlinien setzt eine Zugriffserlaubnis außer Kraft. Weitere Informationen finden Sie im IAMBenutzerhandbuch unter Sitzungsrichtlinien.

Mehrere Richtlinientypen

Wenn mehrere auf eine Anforderung mehrere Richtlinientypen angewendet werden können, sind die entsprechenden Berechtigungen komplizierter. Informationen darüber, wie AWS bestimmt wird, ob eine Anfrage zulässig ist, wenn mehrere Richtlinientypen betroffen sind, finden Sie im IAMBenutzerhandbuch unter Bewertungslogik für Richtlinien.

So EMR arbeitet Amazon mit IAM

Informieren Sie sich vor der Nutzung IAM zur Verwaltung des Zugriffs auf Amazon darüberEMR, welche IAM Funktionen für Amazon verfügbar sindEMR.

IAMFunktionen, die Sie mit Amazon verwenden können EMR

IAMFunktion	EMRAmazon-Unterstützung
Identitätsbasierte Richtlinien	Ja
Ressourcenbasierte Richtlinien	Ja
Richtlinienaktionen	Ja
Richtlinienressourcen	Ja
Bedingungsschlüssel für die Richtlinie	Ja

IAMFunktion	EMRAmazon-Unterstützung
ACLs	Nein
ABAC(Tags in Richtlinien)	Ja
Temporäre Anmeldeinformationen	Ja
Hauptberechtigungen	Ja
Servicerollen	Nein
Serviceverknüpfte Rollen	Ja

Einen allgemeinen Überblick darüber, wie Amazon EMR und andere AWS Dienste mit den meisten IAM Funktionen funktionieren, finden Sie IAM im IAMBenutzerhandbuch unter AWS Dienste, die mit funktionieren.

Identitätsbasierte Richtlinien für Amazon EMR

Unterstützt Richtlinien auf Identitätsbasis: Ja

Identitätsbasierte Richtlinien sind Dokumente mit JSON Berechtigungsrichtlinien, die Sie an eine Identität anhängen können, z. B. an einen IAM Benutzer, eine Benutzergruppe oder eine Rolle. Diese Richtlinien steuern, welche Aktionen die Benutzer und Rollen für welche Ressourcen und unter welchen Bedingungen ausführen können. Informationen zum Erstellen einer identitätsbasierten Richtlinie finden Sie unter IAMRichtlinien erstellen im Benutzerhandbuch. IAM

Mit IAM identitätsbasierten Richtlinien können Sie zulässige oder verweigerte Aktionen und Ressourcen sowie die Bedingungen angeben, unter denen Aktionen zulässig oder verweigert werden. Sie können den Prinzipal nicht in einer identitätsbasierten Richtlinie angeben, da er für den Benutzer oder die Rolle gilt, dem er zugeordnet ist. Weitere Informationen zu allen Elementen, die Sie in einer JSON Richtlinie verwenden können, finden Sie im IAMBenutzerhandbuch unter Referenz zu IAM JSON Richtlinienelementen.

Beispiele für identitätsbasierte Richtlinien für Amazon EMR

Beispiele für EMR identitätsbasierte Richtlinien von Amazon finden Sie unter. Beispiele für EMR identitätsbasierte Richtlinien von Amazon

Ressourcenbasierte Richtlinien innerhalb von Amazon EMR

Unterstützt ressourcenbasierte Richtlinien: Ja

Ressourcenbasierte Richtlinien sind JSON Richtliniendokumente, die Sie an eine Ressource anhängen. Beispiele für ressourcenbasierte Richtlinien sind IAM Rollenvertrauensrichtlinien und Amazon S3 S3-Bucket-Richtlinien. In Services, die ressourcenbasierte Richtlinien unterstützen, können Service-Administratoren sie verwenden, um den Zugriff auf eine bestimmte Ressource zu steuern. Für die Ressource, an welche die Richtlinie angehängt ist, legt die Richtlinie fest, welche Aktionen ein bestimmter Prinzipal unter welchen Bedingungen für diese Ressource ausführen kann. Sie müssen in einer ressourcenbasierten Richtlinie einen Prinzipal angeben. Zu den Prinzipalen können Konten, Benutzer, Rollen, Verbundbenutzer oder gehören. AWS -Services

Um den kontoübergreifenden Zugriff zu ermöglichen, können Sie in einer ressourcenbasierten Richtlinie ein ganzes Konto oder IAM Entitäten in einem anderen Konto als Prinzipal angeben. Durch das Hinzufügen eines kontoübergreifenden Auftraggebers zu einer ressourcenbasierten Richtlinie ist nur die halbe Vertrauensbeziehung eingerichtet. Wenn sich der Prinzipal und die Ressource unterscheiden AWS-Konten, muss ein IAM Administrator des vertrauenswürdigen Kontos auch der Prinzipalentität (Benutzer oder Rolle) die Berechtigung zum Zugriff auf die Ressource gewähren. Sie erteilen Berechtigungen, indem Sie der juristischen Stelle eine identitätsbasierte Richtlinie anfügen. Wenn jedoch eine ressourcenbasierte Richtlinie Zugriff auf einen Prinzipal in demselben Konto gewährt, ist keine zusätzliche identitätsbasierte Richtlinie erforderlich. Weitere Informationen finden Sie IAMim IAMBenutzerhandbuch unter Kontoübergreifender Ressourcenzugriff.

Politische Maßnahmen für Amazon EMR

Unterstützt Richtlinienaktionen: Ja

Administratoren können mithilfe von AWS JSON Richtlinien festlegen, wer Zugriff auf was hat. Das bedeutet, welcher Prinzipal kann Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen.

Das `Action` Element einer JSON Richtlinie beschreibt die Aktionen, mit denen Sie den Zugriff in einer Richtlinie zulassen oder verweigern können. Richtlinienaktionen haben normalerweise denselben Namen wie der zugehörige AWS API Vorgang. Es gibt einige Ausnahmen, z. B. Aktionen, für die nur eine Genehmigung erforderlich ist und für die es keinen entsprechenden Vorgang gibt. API Es gibt auch einige Operationen, die mehrere Aktionen in einer Richtlinie erfordern. Diese zusätzlichen Aktionen werden als abhängige Aktionen bezeichnet.

Schließen Sie Aktionen in eine Richtlinie ein, um Berechtigungen zur Durchführung der zugeordneten Operation zu erteilen.

Eine Liste der EMR Amazon-Aktionen finden Sie unter Aktionen, Ressourcen und Bedingungsschlüssel für Amazon EMR in der Service Authorization Reference.

Richtlinienaktionen in Amazon EMR verwenden das folgende Präfix vor der Aktion:

```
EMR
```

Um mehrere Aktionen in einer einzigen Anweisung anzugeben, trennen Sie sie mit Kommata:

```
"Action": [
      "EMR:action1",
      "EMR:action2"
         ]
```

Beispiele für EMR identitätsbasierte Richtlinien von Amazon finden Sie unter. Beispiele für EMR identitätsbasierte Richtlinien von Amazon

Politische Ressourcen für Amazon EMR

Unterstützt Richtlinienressourcen: Ja

Administratoren können mithilfe von AWS JSON Richtlinien festlegen, wer Zugriff auf was hat. Das bedeutet, welcher Prinzipal kann Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen.

Das `Resource` JSON Richtlinienelement gibt das Objekt oder die Objekte an, für die die Aktion gilt. Anweisungen müssen entweder ein – `Resource`oder ein `NotResource`-Element enthalten. Es hat sich bewährt, eine Ressource mit ihrem Amazon-Ressourcennamen (ARN) anzugeben. Sie können dies für Aktionen tun, die einen bestimmten Ressourcentyp unterstützen, der als Berechtigungen auf Ressourcenebene bezeichnet wird.

Verwenden Sie für Aktionen, die keine Berechtigungen auf Ressourcenebene unterstützen, z. B. Auflistungsoperationen, einen Platzhalter (*), um anzugeben, dass die Anweisung für alle Ressourcen gilt.

```
"Resource": "*"
```

Eine Liste der EMR Amazon-Ressourcentypen und ihrer ARNs Eigenschaften finden Sie unter Von Amazon definierte Ressourcen EMR in der Service Authorization Reference. Informationen zu den ARN Aktionen, die Sie für jede Ressource angeben können, finden Sie unter Aktionen, Ressourcen und Zustandsschlüssel für Amazon EMR.

Beispiele für EMR identitätsbasierte Richtlinien von Amazon finden Sie unter. Beispiele für EMR identitätsbasierte Richtlinien von Amazon

Schlüssel zu den Versicherungsbedingungen für Amazon EMR

Unterstützt servicespezifische Richtlinienbedingungsschlüssel: Ja

Administratoren können mithilfe von AWS JSON Richtlinien festlegen, wer Zugriff auf was hat. Das heißt, welcher Prinzipal kann Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen.

Das Element `Condition` (oder `Condition` block) ermöglicht Ihnen die Angabe der Bedingungen, unter denen eine Anweisung wirksam ist. Das Element `Condition` ist optional. Sie können bedingte Ausdrücke erstellen, die Bedingungsoperatoren verwenden, z. B. ist gleich oder kleiner als, damit die Bedingung in der Richtlinie mit Werten in der Anforderung übereinstimmt.

Wenn Sie mehrere `Condition`-Elemente in einer Anweisung oder mehrere Schlüssel in einem einzelnen `Condition`-Element angeben, wertet AWS diese mittels einer logischen `AND`-Operation aus. Wenn Sie mehrere Werte für einen einzelnen Bedingungsschlüssel angeben, AWS wertet die Bedingung mithilfe einer logischen `OR` Operation aus. Alle Bedingungen müssen erfüllt werden, bevor die Berechtigungen der Anweisung gewährt werden.

Sie können auch Platzhaltervariablen verwenden, wenn Sie Bedingungen angeben. Sie können einem IAM Benutzer beispielsweise nur dann Zugriff auf eine Ressource gewähren, wenn sie mit seinem IAM Benutzernamen gekennzeichnet ist. Weitere Informationen finden Sie im IAMBenutzerhandbuch unter IAMRichtlinienelemente: Variablen und Tags.

AWS unterstützt globale Bedingungsschlüssel und dienstspezifische Bedingungsschlüssel. Eine Übersicht aller AWS globalen Bedingungsschlüssel finden Sie unter Kontext-Schlüssel für AWS globale Bedingungen im IAMBenutzerhandbuch.

Eine Liste der EMR Amazon-Bedingungsschlüssel und Informationen darüber, welche Aktionen und Ressourcen Sie mit einem Bedingungsschlüssel verwenden können, finden Sie unter Aktionen, Ressourcen und Bedingungsschlüssel für Amazon EMR in der Service Authorization Reference.

Beispiele für EMR identitätsbasierte Richtlinien von Amazon finden Sie unter. Beispiele für EMR identitätsbasierte Richtlinien von Amazon

Zugriffskontrolllisten (ACLs) in Amazon EMR

UnterstütztACLs: Nein

Zugriffskontrolllisten (ACLs) steuern, welche Principals (Kontomitglieder, Benutzer oder Rollen) über Zugriffsberechtigungen für eine Ressource verfügen. ACLsähneln ressourcenbasierten Richtlinien, verwenden jedoch nicht das JSON Richtliniendokumentformat.

Attributbasierte Zugriffskontrolle () ABAC mit Amazon EMR

Unterstützt ABAC (Tags in Richtlinien)	Ja

Die attributbasierte Zugriffskontrolle (ABAC) ist eine Autorisierungsstrategie, bei der Berechtigungen auf der Grundlage von Attributen definiert werden. In werden AWS diese Attribute als Tags bezeichnet. Sie können Tags an IAM Entitäten (Benutzer oder Rollen) und an viele AWS Ressourcen anhängen. Das Markieren von Entitäten und Ressourcen ist der erste Schritt vonABAC. Anschließend entwerfen Sie ABAC Richtlinien, die Operationen zulassen, wenn das Tag des Prinzipals mit dem Tag auf der Ressource übereinstimmt, auf die er zugreifen möchte.

ABACist hilfreich in Umgebungen, die schnell wachsen, und hilft in Situationen, in denen die Richtlinienverwaltung umständlich wird.

Um den Zugriff auf der Grundlage von Tags zu steuern, geben Sie im Bedingungselement einer Richtlinie Tag-Informationen an, indem Sie die Schlüssel `aws:ResourceTag/`*`key-name`*, `aws:RequestTag/`*`key-name`*, oder Bedingung `aws:TagKeys` verwenden.

Wenn ein Service alle drei Bedingungsschlüssel für jeden Ressourcentyp unterstützt, lautet der Wert für den Service Ja. Wenn ein Service alle drei Bedingungsschlüssel für nur einige Ressourcentypen unterstützt, lautet der Wert Teilweise.

Weitere Informationen zu finden Sie ABAC unter Was ist? ABAC im IAMBenutzerhandbuch. Ein Tutorial mit Schritten zur Einrichtung finden Sie im ABAC Benutzerhandbuch unter Verwenden der attributebasierten Zugriffskontrolle (ABAC). IAM

Temporäre Anmeldeinformationen mit Amazon verwenden EMR

Unterstützt temporäre Anmeldeinformationen: Ja

Einige funktionieren AWS -Services nicht, wenn Sie sich mit temporären Anmeldeinformationen anmelden. Weitere Informationen, einschließlich Informationen darüber, AWS -Services wie Sie mit temporären Anmeldeinformationen arbeiten können AWS -Services , finden Sie IAM im IAMBenutzerhandbuch unter Diese Informationen.

Sie verwenden temporäre Anmeldeinformationen, wenn Sie sich mit einer anderen AWS Management Console Methode als einem Benutzernamen und einem Kennwort anmelden. Wenn Sie beispielsweise AWS über den Single Sign-On-Link (SSO) Ihres Unternehmens darauf zugreifen, werden bei diesem Vorgang automatisch temporäre Anmeldeinformationen erstellt. Sie erstellen auch automatisch temporäre Anmeldeinformationen, wenn Sie sich als Benutzer bei der Konsole anmelden und dann die Rollen wechseln. Weitere Informationen zum Rollenwechsel finden Sie unter Wechseln zu einer Rolle (Konsole) im IAMBenutzerhandbuch.

Mit dem AWS CLI oder können Sie manuell temporäre Anmeldeinformationen erstellen AWS API. Sie können diese temporären Anmeldeinformationen dann für den Zugriff verwenden AWS. AWS empfiehlt, temporäre Anmeldeinformationen dynamisch zu generieren, anstatt langfristige Zugriffsschlüssel zu verwenden. Weitere Informationen finden Sie unter Temporäre Sicherheitsanmeldeinformationen unter IAM.

Serviceübergreifende Hauptberechtigungen für Amazon EMR

Unterstützt Forward-Access-Sitzungen (FAS): Ja

Wenn Sie einen IAM Benutzer oder eine Rolle verwenden, um Aktionen auszuführen AWS, gelten Sie als Principal. Bei einigen Services könnte es Aktionen geben, die dann eine andere Aktion in einem anderen Service initiieren. FASverwendet die Berechtigungen des Prinzipals, der einen aufruft AWS -Service, kombiniert mit der Anforderung, Anfragen AWS -Service an nachgelagerte Dienste zu stellen. FASAnfragen werden nur gestellt, wenn ein Dienst eine Anfrage erhält, für deren Abschluss Interaktionen mit anderen AWS -Services oder Ressourcen erforderlich sind. In diesem Fall müssen Sie über Berechtigungen zum Ausführen beider Aktionen verfügen. Einzelheiten zu den Richtlinien beim Stellen von FAS Anfragen finden Sie unter Zugriffssitzungen weiterleiten.

Servicerollen für Amazon EMR

Unterstützt Servicerollen	Nein

Servicebezogene Rollen für Amazon EMR

Unterstützt serviceverknüpfte Rollen	Ja

Einzelheiten zum Erstellen oder Verwalten von serviceverknüpften Rollen finden Sie unter AWS Dienste, die mit funktionieren. IAM Suchen Sie in der Tabelle nach einem Service mit einem `Yes` in der Spalte Service-linked role (Serviceverknüpfte Rolle). Wählen Sie den Link Yes (Ja) aus, um die Dokumentation für die serviceverknüpfte Rolle für diesen Service anzuzeigen.

Verwenden Sie Cluster- und Notebook-Tags mit IAM Richtlinien für die Zugriffskontrolle

Die Genehmigung von EMR Amazon-Aktionen im Zusammenhang mit EMR Notebooks und EMR Clustern kann mithilfe einer tagbasierten Zugriffskontrolle mit IAM identitätsbasierten Richtlinien verfeinert werden. Sie können Bedingungsschlüssel in einem `Condition`-Element (auch als `Condition`-Block bezeichnet) verwenden, um bestimmte Aktionen nur dann zuzulassen, wenn ein Notebook, ein Cluster oder beide bestimmte Tag-Schlüssel oder Schlüssel-Wert-Kombinationen aufweisen. Sie können auch die `CreateEditor` Aktion (die ein EMR Notizbuch erstellt) und die `RunJobFlow` Aktion (die einen Cluster erstellt) einschränken, sodass bei der Erstellung der Ressource eine Anfrage für ein Tag eingereicht werden muss.

In Amazon gelten die BedingungsschlüsselEMR, die in einem `Condition` Element verwendet werden können, nur für die EMR API Amazon-Aktionen, bei denen `ClusterID` oder ein erforderlicher Anforderungsparameter `NotebookID` ist. Beispielsweise unterstützt die ModifyInstanceGroupsAktion keine Kontextschlüssel, da es sich um einen optionalen Parameter `ClusterID` handelt.

Wenn Sie ein EMR Notizbuch erstellen, wird ein Standard-Tag mit einer Schlüsselzeichenfolge angewendet, die auf den Wert der IAM Benutzer-ID `creatorUserId` gesetzt ist, die das Notizbuch erstellt hat. Dies ist nützlich, um zulässige Aktionen für das Notebook ausschließlich auf den Ersteller zu beschränken.

Die folgenden Bedingungsschlüssel sind bei Amazon erhältlichEMR:

- Verwenden Sie den Bedingungskontextschlüssel `elasticmapreduce:ResourceTag/`*`TagKeyString`*, um Benutzeraktionen in Clustern oder Notebooks mit Tags mit dem von Ihnen festgelegten *`TagKeyString`* zuzulassen oder abzulehnen.

Wenn eine Aktion sowohl `NotebookID` als auch `ClusterID` übergibt, gilt die Bedingung sowohl für den Cluster als auch das Notebook. Das bedeutet, dass beide Ressourcen dieselbe Tag-Schlüsselzeichenfolge oder Schlüssel-Wert-Kombination aufweisen müssen. Sie können das Element `Resource` verwenden, um die Anweisung nach Bedarf nur auf Cluster oder Notebooks zu beschränken. Weitere Informationen finden Sie unter Beispiele für EMR identitätsbasierte Richtlinien von Amazon.

- Verwenden Sie den `elasticmapreduce:RequestTag/`*`TagKeyString`* Bedingungskontextschlüssel, um ein bestimmtes Tag mit API Aktionen/Aufrufen anzufordern. Verwenden Sie diesen Bedingungskontextschlüssel zusammen mit der `CreateEditor`-Aktion, um festzulegen, dass bei der Erstellung von Notebooks ein Schlüssel mit *`TagKeyString`* angewendet wird.

Beispiele

Eine Liste der EMR Amazon-Aktionen finden Sie EMR im IAMBenutzerhandbuch unter Von Amazon definierte Aktionen.

EMRSchritte zu Runtime-Rollen für Amazon

Eine Runtime-Rolle ist eine AWS Identity and Access Management (IAM) -Rolle, die Sie angeben können, wenn Sie einen Job oder eine Anfrage an einen EMR Amazon-Cluster senden. Der Job oder die Abfrage, die Sie an Ihren EMR Amazon-Cluster senden, verwendet die Runtime-Rolle, um auf AWS Ressourcen wie Objekte in Amazon S3 zuzugreifen. Sie können Runtime-Rollen bei Amazon EMR für Spark- und Hive-Jobs angeben.

Sie können auch Runtime-Rollen angeben, wenn Sie eine Verbindung zu EMR Amazon-Clustern herstellen Amazon SageMaker und wenn Sie einen Amazon EMR Studio-Workspace an einen EMR Cluster anhängen. Weitere Informationen finden Sie unter Von Studio aus eine Connect zu einem EMR Amazon-Cluster herstellen undFühren Sie einen EMR Studio-Workspace mit einer Runtime-Rolle aus.

Zuvor führten EMR Amazon-Cluster EMR Amazon-Jobs oder -Abfragen mit Berechtigungen aus, die auf der IAM Richtlinie basierten, die mit dem Instance-Profil verknüpft war, das Sie zum Starten des Clusters verwendet haben. Das bedeutete, dass die Richtlinien die Vereinigung aller Berechtigungen für alle Jobs und Abfragen enthalten mussten, die auf einem EMR Amazon-Cluster ausgeführt wurden. Mit Runtime-Rollen können Sie jetzt die Zugriffskontrolle für jeden Job oder jede Abfrage einzeln verwalten, anstatt das EMR Amazon-Instance-Profil des Clusters gemeinsam zu nutzen.

Auf EMR Amazon-Clustern mit Runtime-Rollen können Sie auch eine AWS Lake Formation basierte Zugriffskontrolle auf Spark-, Hive- und Presto-Jobs und Abfragen für Ihre Data Lakes anwenden. Weitere Informationen zur Integration mit AWS Lake Formation finden Sie unter. Integrieren Sie Amazon EMR mit AWS Lake Formation

> **Note**
>
> Wenn Sie eine Runtime-Rolle für einen EMR Amazon-Schritt angeben, können die Jobs oder Abfragen, die Sie einreichen, nur auf AWS Ressourcen zugreifen, die die mit der Runtime-Rolle verknüpften Richtlinien zulassen. Diese Jobs und Abfragen können nicht auf den Instance-Metadaten-Service auf den EC2 Instances des Clusters zugreifen oder das EC2 Instance-Profil des Clusters für den Zugriff auf AWS Ressourcen verwenden.

Voraussetzungen für den Start eines EMR Amazon-Clusters mit einer Runtime-Rolle

Themen

- Schritt 1: Sicherheitskonfigurationen in Amazon einrichten EMR
- Schritt 2: Richten Sie ein EC2 Instance-Profil für den EMR Amazon-Cluster ein
- Schritt 3: Eine Vertrauensrichtlinie einrichten

Schritt 1: Sicherheitskonfigurationen in Amazon einrichten EMR

Verwenden Sie die folgende JSON Struktur, um eine Sicherheitskonfiguration für AWS Command Line Interface (AWS CLI) zu erstellen, und stellen Sie `EnableApplicationScopedIAMRole` sie auf ein`true`. Weitere Informationen zu den Sicherheitskonfigurationen finden Sie unter Sicherheitskonfigurationen zum Einrichten der Cluster-Sicherheit verwenden.

```
{
    "AuthorizationConfiguration":{
        "IAMConfiguration":{
            "EnableApplicationScopedIAMRole":true
        }
    }
}
```

Wir empfehlen, in der Sicherheitskonfiguration immer die Verschlüsselungsoptionen bei der Übertragung zu aktivieren, sodass Daten, die über das Internet übertragen werden, verschlüsselt

und nicht im Klartext übertragen werden. Sie können diese Optionen überspringen, wenn Sie keine Verbindung zu EMR Amazon-Clustern mit Runtime-Rollen aus SageMaker Runtime Studio oder EMR Studio herstellen möchten. Informationen zur Konfiguration der Datenverschlüsselung finden Sie unter Datenverschlüsselung konfigurieren.

Alternativ können Sie mit dem eine Sicherheitskonfiguration mit benutzerdefinierten Einstellungen mit AWS Management Console erstellen.

Schritt 2: Richten Sie ein EC2 Instance-Profil für den EMR Amazon-Cluster ein

EMRAmazon-Cluster verwenden die EC2 Amazon-Instance-Profilrolle, um die Runtime-Rollen zu übernehmen. Um Runtime-Rollen mit EMR Amazon-Schritten zu verwenden, fügen Sie der IAM Rolle, die Sie als Instance-Profilrolle verwenden möchten, die folgenden Richtlinien hinzu. Informationen zum Hinzufügen von Richtlinien zu einer IAM Rolle oder zum Bearbeiten einer bestehenden Inline- oder verwalteten Richtlinie finden Sie unter Hinzufügen und Entfernen von IAM Identitätsberechtigungen.

```
{
    "Version":"2012-10-17",
    "Statement":[
        {
            "Sid":"AllowRuntimeRoleUsage",
            "Effect":"Allow",
            "Action":[
                "sts:AssumeRole",
                "sts:TagSession"
            ],
            "Resource":[
                <runtime-role-ARN>
            ]
        }
    ]
}
```

Schritt 3: Eine Vertrauensrichtlinie einrichten

Legen Sie für jede IAM Rolle, die Sie als Runtime-Rolle verwenden möchten, die folgende Vertrauensrichtlinie fest und `EMR_EC2_DefaultRole` ersetzen Sie sie durch Ihre Instanzprofilrolle. Informationen zum Ändern der Vertrauensrichtlinie einer IAM Rolle finden Sie unter Vertrauensrichtlinie für Rollen ändern.

```
{
    "Sid":"AllowAssumeRole",
    "Effect":"Allow",
    "Principal":{
        "AWS":"arn:aws:iam::<AWS_ACCOUNT_ID>:role/EMR_EC2_DefaultRole"
    },
    "Action":"sts:AssumeRole"
}
```

Starten Sie einen EMR Amazon-Cluster mit rollenbasierter Zugriffskontrolle

Nachdem Sie Ihre Konfigurationen eingerichtet haben, können Sie einen EMR Amazon-Cluster mit der Sicherheitskonfiguration von startenSchritt 1: Sicherheitskonfigurationen in Amazon einrichten EMR. Um Runtime-Rollen mit EMR Amazon-Schritten zu verwenden, verwenden Sie Release Label `emr-6.7.0` oder höher und wählen Sie Hive, Spark oder beide als Cluster-Anwendung aus. Um von SageMaker Studio aus eine Verbindung herzustellen, verwenden Sie Release `emr-6.9.0` oder höher und wählen Sie Livy, Spark, Hive oder Presto als Ihre Cluster-Anwendung aus. Anweisungen zum Start Ihres Clusters finden Sie unter Angabe einer Sicherheitskonfiguration für einen Cluster.

Spark-Jobs mithilfe der EMR Amazon-Schritte einreichen

Im Folgenden finden Sie ein Beispiel für die Ausführung des in Apache Spark enthaltenen HdfsTest Beispiels. Dieser API Aufruf ist nur erfolgreich, wenn die bereitgestellte EMR Amazon-Runtime-Rolle auf die `S3_LOCATION` zugreifen kann.

```
RUNTIME_ROLE_ARN=<runtime-role-arn>
S3_LOCATION=<s3-path>
REGION=<aws-region>
CLUSTER_ID=<cluster-id>

aws emr add-steps --cluster-id $CLUSTER_ID \
--steps '[{ "Name": "Spark Example", "ActionOnFailure": "CONTINUE","HadoopJarStep":
 { "Jar":"command-runner.jar","Args" : ["spark-example","HdfsTest",
 "$S3_LOCATION"] } }]' \
--execution-role-arn $RUNTIME_ROLE_ARN \
--region $REGION
```

> **Note**
>
> Wir empfehlen, den SSH Zugriff auf den EMR Amazon-Cluster zu deaktivieren und nur Amazon den Zugriff auf den Cluster EMR `AddJobFlowSteps` API zu gewähren.

Hive-Jobs mithilfe der EMR Amazon-Schritte einreichen

Im folgenden Beispiel wird Apache Hive mit Amazon EMR Steps verwendet, um einen Job zur Ausführung der `QUERY_FILE.hql` Datei einzureichen. Diese Abfrage ist nur erfolgreich, wenn die angegebene Laufzeit-Rolle auf den Amazon-S3-Pfad der Abfragedatei zugreifen kann.

```
RUNTIME_ROLE_ARN=<runtime-role-arn>
REGION=<aws-region>
CLUSTER_ID=<cluster-id>

aws emr add-steps --cluster-id $CLUSTER_ID \
--steps '[{ "Name": "Run hive query using command-runner.jar - simple
 select","ActionOnFailure":"CONTINUE","HadoopJarStep": { "Jar": "command-
runner.jar","Args" :["hive -
f","s3://DOC_EXAMPLE_BUCKET/QUERY_FILE.hql"] } }]' \
--execution-role-arn $RUNTIME_ROLE_ARN \
--region $REGION
```

Stellen Sie über ein SageMaker Studio-Notizbuch eine Connect zu EMR Amazon-Clustern mit Runtime-Rollen her

Sie können EMR Amazon-Runtime-Rollen auf Abfragen anwenden, die Sie in EMR Amazon-Clustern von SageMaker Studio aus ausführen. Führen Sie dazu die folgenden Schritte aus.

1. Folgen Sie den Anweisungen unter Amazon SageMaker Studio starten, um ein SageMaker Studio zu erstellen.
2. Starten Sie in der SageMaker Studio-Benutzeroberfläche ein Notizbuch mit unterstützten Kerneln. Starten Sie beispielsweise ein SparkMagic Image mit einem PySpark Kernel.
3. Wählen Sie in SageMaker Studio einen EMR Amazon-Cluster und dann Connect aus.
4. Wählen Sie eine Laufzeit-Rolle und dann Verbinden aus.

Dadurch wird eine SageMaker Notebook-Zelle mit magischen Befehlen erstellt, um eine Verbindung zu Ihrem EMR Amazon-Cluster mit der ausgewählten EMR Amazon-Runtime-Rolle herzustellen.

In der Notebook-Zelle können Sie Abfragen mit Laufzeit-Rollen- und Lake-Formation-basierter Zugriffskontrolle eingeben und ausführen. Ein detaillierteres Beispiel finden Sie unter Anwenden detaillierter Datenzugriffskontrollen mit AWS Lake Formation und Amazon EMR von Amazon SageMaker Studio aus.

Steuern Sie den Zugriff auf die EMR Amazon-Runtime-Rolle

Sie können den Zugriff auf die Laufzeit-Rolle mit dem Bedingungsschlüssel `elasticmapreduce:ExecutionRoleArn` steuern. Die folgende Richtlinie ermöglicht es einem IAM Prinzipal`Caller`, eine IAM Rolle mit dem Namen oder eine beliebige IAM Rolle, die mit der Zeichenfolge beginnt`CallerTeamRole`, als Runtime-Rolle zu verwenden.

> **Important**
>
> Sie müssen eine auf dem `elasticmapreduce:ExecutionRoleArn` Kontextschlüssel basierende Bedingung erstellen, wenn Sie einem Anrufer Zugriff auf den Befehl `AddJobFlowSteps` oder gewähren `GetClusterSessionCredentials`APIs, wie das folgende Beispiel zeigt.

```
{
    "Sid":"AddStepsWithSpecificExecRoleArn",
    "Effect":"Allow",
    "Action":[
        "elasticmapreduce:AddJobFlowSteps"
    ],
    "Resource":"*",
    "Condition":{
        "StringEquals":{
            "elasticmapreduce:ExecutionRoleArn":[
                "arn:aws:iam::<AWS_ACCOUNT_ID>:role/Caller"
            ]
        },
        "StringLike":{
            "elasticmapreduce:ExecutionRoleArn":[
                "arn:aws:iam::<AWS_ACCOUNT_ID>:role/CallerTeamRole*"
            ]
        }
    }
}
```

Vertrauen zwischen Runtime-Rollen und EMR Amazon-Clustern aufbauen

Amazon EMR generiert `ExternalId` für jede Sicherheitskonfiguration mit aktivierter Laufzeit-Rollenautorisierung eine eindeutige Kennung. Diese Autorisierung ermöglicht es jedem Benutzer, eine Reihe von Laufzeit-Rollen zu besitzen, die er auf Clustern verwenden kann, die ihm gehören. In einem Unternehmen kann beispielsweise jede Abteilung ihre externe ID verwenden, um die Vertrauensrichtlinie für ihre eigenen Laufzeit-Rollen zu aktualisieren.

Sie können die externe ID bei Amazon finden EMR `DescribeSecurityConfiguration`API, wie im folgenden Beispiel gezeigt.

```
aws emr describe-security-configuration --name 'iamconfig-with-lf'{"Name": "iamconfig-
with-lf",
    "SecurityConfiguration":
        "{\"AuthorizationConfiguration\":{\"IAMConfiguration\":
{\"EnableApplicationScopedIAMRole\
        ":true,\"ApplicationScopedIAMRoleConfiguration\":{\"PropagateSourceIdentity
\":true,\"Exter
        nalId\":\"FXH5TSACFDWUCDSR3YQE2O7ETPUSM4OBCGLYWODSCUZDNZ4Y\"}},\"Lake
        FormationConfiguration\":{\"AuthorizedSessionTagValue\":\"Amazon EMR\"}}}",
    "CreationDateTime": "2022-06-03T12:52:35.308000-07:00"
}
```

Informationen zur Verwendung einer externen ID finden Sie unter So verwenden Sie eine externe ID, wenn Sie Dritten Zugriff auf Ihre AWS Ressourcen gewähren.

Audit

Um die Aktionen zu überwachen und zu steuern, die Endbenutzer mit IAM Rollen ausführen, können Sie die Quellidentitätsfunktion aktivieren. Weitere Informationen zur Quellenidentität finden Sie unter Überwachen und Steuern von Aktionen mit übernommenen Rollen.

Um die Quellidentität nachzuverfolgen, stellen Sie `ApplicationScopedIAMRoleConfiguration/PropagateSourceIdentity` in Ihrer Sicherheitskonfiguration wie folgt auf `true` ein.

```
{
    "AuthorizationConfiguration":{
        "IAMConfiguration":{
            "EnableApplicationScopedIAMRole":true,
            "ApplicationScopedIAMRoleConfiguration":{
```

```
                "PropagateSourceIdentity":true
            }
        }
    }
}
```

Wenn Sie `PropagateSourceIdentity` diese Option festlegen`true`, EMR wendet Amazon die Quellidentität aus den Anrufanmeldedaten auf eine Job- oder Abfragesitzung an, die Sie mit der Runtime-Rolle erstellen. Wenn in den Anrufanmeldedaten keine Quellidentität enthalten ist, legt Amazon die Quellidentität EMR nicht fest.

Um diese Eigenschaft zu verwenden, geben Sie wie folgt `sts:SetSourceIdentity`-Berechtigungen für Ihr Instance-Profil ein.

```
{ // PropagateSourceIdentity statement
    "Sid":"PropagateSourceIdentity",
    "Effect":"Allow",
    "Action":"sts:SetSourceIdentity",
    "Resource":[
        <runtime-role-ARN>
    ],
    "Condition":{
        "StringEquals":{
            "sts:SourceIdentity":<source-identity>
        }
    }
}
```

Sie müssen die `AllowSetSourceIdentity`-Anweisung auch zur Vertrauensrichtlinie Ihrer Laufzeit-Rollen hinzufügen.

```
{ // AllowSetSourceIdentity statement
    "Sid":"AllowSetSourceIdentity",
    "Effect":"Allow",
    "Principal":{
        "AWS":"arn:aws:iam::<AWS_ACCOUNT_ID>:role/EMR_EC2_DefaultRole"
    },
    "Action":[
        "sts:SetSourceIdentity",
        "sts:AssumeRole"
    ],
    "Condition":{
```

```
        "StringEquals":{
            "sts:SourceIdentity":<source-identity>
        }
    }
}
```

Weitere Überlegungen

> **Note**
>
> Bei der EMR Amazon-Version kann es zu zeitweiligen Ausfällen kommen`emr-6.9.0`, wenn Sie von SageMaker Studio aus eine Verbindung zu EMR Amazon-Clustern herstellen. Um dieses Problem zu lösen, können Sie den Patch mit einer Bootstrap-Aktion installieren, wenn Sie den Cluster starten. Einzelheiten zum Patch finden Sie unter Bekannte Probleme von Amazon EMR Version 6.9.0.

Beachten Sie außerdem Folgendes, wenn Sie Runtime-Rollen für Amazon konfigurierenEMR.

- Amazon EMR unterstützt Runtime-Rollen in allen kommerziellen Anwendungen AWS-Regionen.
- Amazon EMR Steps unterstützt Apache Spark- und Apache Hive-Jobs mit Runtime-Rollen, wenn Sie Release `emr-6.7.0` oder höher verwenden.
- SageMaker Studio unterstützt Spark-, Hive- und Presto-Abfragen mit Runtime-Rollen, wenn Sie Release `emr-6.9.0` oder höher verwenden.
- Die folgenden Notebook-Kernel SageMaker unterstützen Runtime-Rollen:
  - DataScience — Python-3-Kernel
  - DataScience 2.0 — Python-3-Kernel
  - DataScience 3.0 — Python-3-Kernel
  - SparkAnalytics 1.0 — SparkMagic und PySpark Kernel
  - SparkAnalytics 2.0 — SparkMagic und Kernel PySpark
  - SparkMagic — Kernel PySpark
- Amazon EMR unterstützt Schritte, die `RunJobFlow` nur zum Zeitpunkt der Clustererstellung verwendet werden. Runtime-Rollen werden dadurch API nicht unterstützt.
- Amazon unterstützt EMR keine Runtime-Rollen auf Clustern, die Sie so konfigurieren, dass sie hochverfügbar sind.

- Sie müssen Ihre Bash-Befehlsargumente umgehen, wenn Sie Befehle mit der folgenden Datei ausführen: `command-runner.jar` JAR

```
aws emr add-steps --cluster-id <cluster-id> --steps '[{"Name":"sample-
step","ActionOnFailure":"CONTINUE","Jar":"command-runner.jar","Properties":"","Args":
["bash","-c","\"aws s3 ls\""],"Type":"CUSTOM_JAR"}]' --execution-role-
arn <IAM_ROLE_ARN>
```

- Runtime-Rollen bieten keine Unterstützung für die Steuerung des Zugriffs auf Cluster-Ressourcen wie HDFS und. HMS

IAMServicerollen für EMR Amazon-Berechtigungen für AWS Dienste und Ressourcen konfigurieren

Amazon EMR und Anwendungen wie Hadoop und Spark benötigen Berechtigungen, um auf andere AWS Ressourcen zuzugreifen und Aktionen auszuführen, wenn sie ausgeführt werden. Jeder Cluster in Amazon EMR muss eine Servicerolle und eine Rolle für das EC2 Amazon-Instance-Profil haben. Weitere Informationen finden Sie unter IAMRollen und Verwenden von Instance-Profilen im IAMBenutzerhandbuch. Die mit diesen Rollen verknüpften IAM Richtlinien gewähren dem Cluster die Erlaubnis, im Namen eines Benutzers mit anderen AWS Diensten zusammenzuarbeiten.

Eine zusätzliche Rolle, die Auto Scaling-Rolle, ist erforderlich, wenn Ihr Cluster Auto Scaling in Amazon verwendetEMR. Die AWS Servicerolle für EMR Notebooks ist erforderlich, wenn Sie EMR Notebooks verwenden.

Amazon EMR bietet Standardrollen und verwaltete Standardrichtlinien, die die Berechtigungen für jede Rolle festlegen. Verwaltete Richtlinien werden von erstellt und verwaltet AWS, sodass sie automatisch aktualisiert werden, wenn sich die Serviceanforderungen ändern. Informationen zu AWS verwalteten Richtlinien finden Sie im IAMBenutzerhandbuch.

Wenn Sie zum ersten Mal einen Cluster oder ein Notebook in einem Konto erstellen, sind Rollen für Amazon noch EMR nicht vorhanden. Nachdem Sie sie erstellt haben, können Sie die Rollen, die ihnen zugewiesenen Richtlinien und die durch die Richtlinien erlaubten oder verweigerten Berechtigungen in der IAM Konsole anzeigen (https://console.aws.amazon.com/iam/). Sie können Standardrollen angeben, die Amazon erstellen und verwenden EMR soll, Sie können Ihre eigenen Rollen erstellen und diese individuell angeben, wenn Sie einen Cluster erstellen, um Berechtigungen anzupassen, und Sie können Standardrollen angeben, die verwendet werden sollen, wenn Sie einen

Cluster mit dem erstellen AWS CLI. Weitere Informationen finden Sie unter Passen Sie IAM Rollen an.

Änderung identitätsbasierter Richtlinien für Berechtigungen zur Weitergabe von Servicerollen für Amazon EMR

Die standardmäßigen verwalteten Richtlinien von Amazon mit EMR vollen Berechtigungen beinhalten `iam:PassRole` Sicherheitskonfigurationen, darunter die folgenden:

- `iam:PassRole`Berechtigungen nur für bestimmte EMR Amazon-Standardrollen.
- `iam:PassedToService`Bedingungen, die es Ihnen ermöglichen, die Richtlinie nur mit bestimmten AWS Diensten zu verwenden, z. B. `elasticmapreduce.amazonaws.com` und`ec2.amazonaws.com`.

Sie können die JSON Version der Richtlinien A mazonEMRFull AccessPolicy _v2 und A mazonEMRService Policy_v2 in der Konsole einsehen. IAM Wir empfehlen, dass Sie neue Cluster mit den verwalteten v2-Richtlinien erstellen.

Übersicht über Servicerollen

In der folgenden Tabelle sind die mit Amazon EMR verknüpften IAM Servicerollen als Kurzreferenz aufgeführt.

Funktion	Standardrolle	Beschreibung	Verwaltete Standardrichtlinie
Servicerolle für Amazon EMR (EMRRolle)	`EMR_DefaultRole_V2`	Ermöglicht AmazonEMR, andere AWS Services in Ihrem Namen aufzurufen, wenn Ressourcen bereitgestellt und Service-Level-Aktionen ausgeführt werden. Diese Rolle ist für alle Cluster erforderlich.	`AmazonEMRServicePolicy_v2` **Important** Zum Anfordern von Spot Instances ist eine serviceverknüpfte Rolle

Funktion	Standardrolle	Beschreibung	Verwaltete Standardrichtlinie
			erforderlich. Wenn diese Rolle nicht existiert, muss die EMR Amazon-Serviceroll e über die Berechtigung verfügen, sie zu erstellen , andernfal ls tritt ein Berechtig ungsfehler auf. Wenn Sie Spot Instances anfordern möchten, müssen Sie diese Richtlini e so aktualisi eren, dass sie eine Erklärung enthält, die die Erstellun g dieser serviceve rknüpften Rolle ermöglich t. Weitere Informati

Funktion	Standardrolle	Beschreibung	Verwaltete Standardrichtlinie
			onen finden Sie unter Servicerolle für Amazon EMR (EMRRolle) und serviceverknüpfte Rolle für Spot-Instance-Anfragen im EC2Amazon-Benutzerhandbuch.

Funktion	Standardrolle	Beschreibung	Verwaltete Standardrichtlinie
Servicerolle für EC2 Cluster-Instances (EC2Instance-Profil)	`EMR_EC2_DefaultRole`	Anwendungsprozesse, die auf dem Hadoop-Ökosystem auf Cluster-Instances ausgeführt werden, verwenden diese Rolle, wenn sie andere AWS Dienste aufrufen. Für den Zugriff auf Daten in Amazon S3 können Sie je nach Speicherort der Daten in Amazon S3 verschiedene Rollen angeben, die übernommen werden sollen. EMRFS Beispielsweise können mehrere Teams auf ein einzelnes „Datenspeicherkonto“ von Amazon S3 zugreifen. Weitere Informationen finden Sie unter IAMRollen für EMRFS Anfragen an Amazon S3 konfigurieren. Diese Rolle ist für alle Cluster erforderlich.	`AmazonElasticMapReduceforEC2Role`. Weitere Informationen finden Sie unter Servicerolle für EC2 Cluster-Instances (EC2Instance-Profil).

Funktion	Standardrolle	Beschreibung	Verwaltete Standardrichtlinie
Servicerolle für Auto Scaling in Amazon EMR (Auto Scaling-Rolle)	`EMR_AutoScaling_DefaultRole`	Ermöglicht zusätzliche Aktionen für dynamisch skalierte Umgebungen. Nur für Cluster erforderlich, die automatische Skalierung in Amazon verwendenEMR. Weitere Informationen finden Sie unter Verwenden der automatischen Skalierung mit einer benutzerdefinierten Richtlinie für Instance-Gruppen.	`AmazonElasticMapReduceforAutoScalingRole` . Weitere Informationen finden Sie unter Servicerolle für Auto Scaling in Amazon EMR (Auto Scaling-Rolle).

Funktion	Standardrolle	Beschreibung	Verwaltete Standardrichtlinie
Servicerolle für EMR Notebooks	`EMR_Notebooks_DefaultRole`	Stellt Berechtigungen bereit, die ein EMR Notebook benötigt, um auf andere AWS Ressourcen zuzugreifen und Aktionen auszuführen. Nur erforderlich, wenn EMR Notebooks verwendet wird.	`AmazonElasticMapReduceEditorsRole` . Weitere Informationen finden Sie unter Servicerolle für EMR Notebooks. `S3FullAccessPolicy` wird auch standardmäßig angehängt. Im Folgenden finden Sie den Inhalt dieser Richtlinie.. `{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Action": "s3:*", "Resource": "*" } ] }`

Funktion	Standardrolle	Beschreibung	Verwaltete Standardrichtlinie
Serviceverknüpfte Rolle	`AWSServiceRoleForEMRCleanup`	Amazon erstellt EMR automatisch eine servicebezogene Rolle. Wenn der Service für Amazon nicht EMR mehr in der Lage ist, EC2 Amazon-Ressourcen zu bereinigen, EMR kann Amazon diese Rolle zum Aufräumen verwenden. Wenn ein Cluster Spot-Instances verwendet, muss die Berechtigungsrichtlinie, die der Servicerolle für Amazon EMR (EMRRolle) angefügt ist, die Erstellung einer serviceverknüpften Rolle zulassen. Weitere Informationen finden Sie unter Verwenden von serviceverknüpften Rollen für Amazon EMR.	`AmazonEMRCleanupPolicy`

Themen

- IAMVon Amazon verwendete Servicerollen EMR
- Passen Sie IAM Rollen an

- IAMRollen für EMRFS Anfragen an Amazon S3 konfigurieren
- Verwenden Sie ressourcenbasierte Richtlinien für den EMR Zugriff von Amazon auf AWS Glue Data Catalog
- Verwenden Sie IAM Rollen mit Anwendungen, die AWS Dienste direkt aufrufen
- Benutzern und Gruppen gestatten, Rollen zu erstellen und zu ändern

IAMVon Amazon verwendete Servicerollen EMR

Amazon EMR verwendet IAM Service-Rollen, um Aktionen in Ihrem Namen auszuführen, wenn es um die Bereitstellung von Cluster-Ressourcen, die Ausführung von Anwendungen, die dynamische Skalierung von Ressourcen und die Erstellung und Ausführung von EMR Notebooks geht. Amazon EMR verwendet die folgenden Rollen bei der Interaktion mit anderen AWS Diensten. Jede Rolle hat eine einzigartige Funktion innerhalb von AmazonEMR. Die Themen in diesem Abschnitt beschreiben die Rollenfunktion und stellen die Standardrollen und die Berechtigungsrichtlinie für jede Rolle bereit.

Wenn Sie in Ihrem Cluster Anwendungscode haben, der AWS Dienste direkt aufruft, müssen Sie möglicherweise den verwendenSDK, um Rollen anzugeben. Weitere Informationen finden Sie unter Verwenden Sie IAM Rollen mit Anwendungen, die AWS Dienste direkt aufrufen.

Themen

- Servicerolle für Amazon EMR (EMRRolle)
- Servicerolle für EC2 Cluster-Instances (EC2Instance-Profil)
- Servicerolle für Auto Scaling in Amazon EMR (Auto Scaling-Rolle)
- Servicerolle für EMR Notebooks
- Verwenden von serviceverknüpften Rollen für Amazon EMR

Servicerolle für Amazon EMR (EMRRolle)

Die EMR Amazon-Rolle definiert die zulässigen Aktionen für Amazon, EMR wenn es Ressourcen bereitstellt und Service-Level-Aufgaben ausführt, die nicht im Kontext einer EC2 Amazon-Instance ausgeführt werden, die innerhalb eines Clusters ausgeführt wird. Die Service-Rolle wird beispielsweise verwendet, um EC2 Instances bereitzustellen, wenn ein Cluster gestartet wird.

- Der Standardrollenname ist `EMR_DefaultRole_V2`.

- Die EMR angehängte standardmäßige verwaltete Richtlinie mit Geltungsbereich von Amazon ist. `EMR_DefaultRole_V2` `AmazonEMRServicePolicy_v2` Diese v2-Richtlinie ersetzt die veraltete verwaltete Standardrichtlinie `AmazonElasticMapReduceRole`.

`AmazonEMRServicePolicy_v2`hängt vom begrenzten Zugriff auf Ressourcen ab, die Amazon EMR bereitstellt oder nutzt. Wenn Sie diese Richtlinie verwenden, müssen Sie bei der Bereitstellung des Clusters das Benutzer-Tag `for-use-with-amazon-emr-managed-policies = true` übergeben. Amazon EMR verbreitet diese Tags automatisch. Darüber hinaus müssen Sie möglicherweise manuell ein Benutzer-Tag zu bestimmten Ressourcentypen hinzufügen, z. B. EC2 Sicherheitsgruppen, die nicht von Amazon erstellt wurdenEMR. Siehe Taggen von Ressourcen zur Verwendung verwalteter Richtlinien.

> **Important**
>
> Amazon EMR verwendet diese EMR Amazon-Servicerolle und die `AWSServiceRoleForEMRCleanup` Rolle, um Cluster-Ressourcen in Ihrem Konto zu bereinigen, die Sie nicht mehr verwenden, z. B. EC2 Amazon-Instances. Sie müssen Aktionen für die Rollenrichtlinien angeben, um die Ressourcen zu löschen oder zu beenden. Andernfalls EMR kann Amazon diese Bereinigungsaktionen nicht durchführen, und es können Kosten für ungenutzte Ressourcen anfallen, die im Cluster verbleiben.

Im Folgenden werden die Inhalte der aktuellen `AmazonEMRServicePolicy_v2`-Richtlinie angezeigt. Sie können den aktuellen Inhalt der `AmazonEMRServicePolicy_v2`verwalteten Richtlinie auch auf der IAM Konsole sehen.

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Sid": "CreateInTaggedNetwork",
   "Effect": "Allow",
   "Action": [
    "ec2:CreateNetworkInterface",
    "ec2:RunInstances",
    "ec2:CreateFleet",
    "ec2:CreateLaunchTemplate",
    "ec2:CreateLaunchTemplateVersion"
   ],
```

```
    "Resource": [
     "arn:aws:ec2:*:*:subnet/*",
     "arn:aws:ec2:*:*:security-group/*"
    ],
    "Condition": {
     "StringEquals": {
      "aws:ResourceTag/for-use-with-amazon-emr-managed-policies": "true"
     }
    }
   },
   {
    "Sid": "CreateWithEMRTaggedLaunchTemplate",
    "Effect": "Allow",
    "Action": [
     "ec2:CreateFleet",
     "ec2:RunInstances",
     "ec2:CreateLaunchTemplateVersion"
    ],
    "Resource": "arn:aws:ec2:*:*:launch-template/*",
    "Condition": {
     "StringEquals": {
      "aws:ResourceTag/for-use-with-amazon-emr-managed-policies": "true"
     }
    }
   },
   {
    "Sid": "CreateEMRTaggedLaunchTemplate",
    "Effect": "Allow",
    "Action": "ec2:CreateLaunchTemplate",
    "Resource": "arn:aws:ec2:*:*:launch-template/*",
    "Condition": {
     "StringEquals": {
      "aws:RequestTag/for-use-with-amazon-emr-managed-policies": "true"
     }
    }
   },
   {
    "Sid": "CreateEMRTaggedInstancesAndVolumes",
    "Effect": "Allow",
    "Action": [
     "ec2:RunInstances",
     "ec2:CreateFleet"
    ],
    "Resource": [
```

```
    "arn:aws:ec2:*:*:instance/*",
    "arn:aws:ec2:*:*:volume/*"
   ],
   "Condition": {
    "StringEquals": {
     "aws:RequestTag/for-use-with-amazon-emr-managed-policies": "true"
    }
   }
  },
  {
   "Sid": "ResourcesToLaunchEC2",
   "Effect": "Allow",
   "Action": [
    "ec2:RunInstances",
    "ec2:CreateFleet",
    "ec2:CreateLaunchTemplate",
    "ec2:CreateLaunchTemplateVersion"
   ],
   "Resource": [
    "arn:aws:ec2:*:*:network-interface/*",
    "arn:aws:ec2:*::image/ami-*",
    "arn:aws:ec2:*:*:key-pair/*",
    "arn:aws:ec2:*:*:capacity-reservation/*",
    "arn:aws:ec2:*:*:placement-group/pg-*",
    "arn:aws:ec2:*:*:fleet/*",
    "arn:aws:ec2:*:*:dedicated-host/*",
    "arn:aws:resource-groups:*:*:group/*"
   ]
  },
  {
   "Sid": "ManageEMRTaggedResources",
   "Effect": "Allow",
   "Action": [
    "ec2:CreateLaunchTemplateVersion",
    "ec2:DeleteLaunchTemplate",
    "ec2:DeleteNetworkInterface",
    "ec2:ModifyInstanceAttribute",
    "ec2:TerminateInstances"
   ],
   "Resource": "*",
   "Condition": {
    "StringEquals": {
     "aws:ResourceTag/for-use-with-amazon-emr-managed-policies": "true"
    }
```

```
    }
   },
   {
    "Sid": "ManageTagsOnEMRTaggedResources",
    "Effect": "Allow",
    "Action": [
     "ec2:CreateTags",
     "ec2:DeleteTags"
    ],
    "Resource": [
     "arn:aws:ec2:*:*:instance/*",
     "arn:aws:ec2:*:*:volume/*",
     "arn:aws:ec2:*:*:network-interface/*",
     "arn:aws:ec2:*:*:launch-template/*"
    ],
    "Condition": {
     "StringEquals": {
      "aws:ResourceTag/for-use-with-amazon-emr-managed-policies": "true"
     }
    }
   },
   {
    "Sid": "CreateNetworkInterfaceNeededForPrivateSubnet",
    "Effect": "Allow",
    "Action": [
     "ec2:CreateNetworkInterface"
    ],
    "Resource": [
     "arn:aws:ec2:*:*:network-interface/*"
    ],
    "Condition": {
     "StringEquals": {
      "aws:RequestTag/for-use-with-amazon-emr-managed-policies": "true"
     }
    }
   },
   {
    "Sid": "TagOnCreateTaggedEMRResources",
    "Effect": "Allow",
    "Action": [
     "ec2:CreateTags"
    ],
    "Resource": [
     "arn:aws:ec2:*:*:network-interface/*",
```

```
    "arn:aws:ec2:*:*:instance/*",
    "arn:aws:ec2:*:*:volume/*",
    "arn:aws:ec2:*:*:launch-template/*"
   ],
   "Condition": {
    "StringEquals": {
     "ec2:CreateAction": [
      "RunInstances",
      "CreateFleet",
      "CreateLaunchTemplate",
      "CreateNetworkInterface"
     ]
    }
   }
  },
  {
   "Sid": "TagPlacementGroups",
   "Effect": "Allow",
   "Action": [
    "ec2:CreateTags",
    "ec2:DeleteTags"
   ],
   "Resource": [
    "arn:aws:ec2:*:*:placement-group/pg-*"
   ]
  },
  {
   "Sid": "ListActionsForEC2Resources",
   "Effect": "Allow",
   "Action": [
    "ec2:DescribeAccountAttributes",
    "ec2:DescribeCapacityReservations",
    "ec2:DescribeDhcpOptions",
    "ec2:DescribeImages",
    "ec2:DescribeInstances",
    "ec2:DescribeInstanceTypeOfferings",
    "ec2:DescribeLaunchTemplates",
    "ec2:DescribeNetworkAcls",
    "ec2:DescribeNetworkInterfaces",
    "ec2:DescribePlacementGroups",
    "ec2:DescribeRouteTables",
    "ec2:DescribeSecurityGroups",
    "ec2:DescribeSubnets",
    "ec2:DescribeVolumes",
```

```
    "ec2:DescribeVolumeStatus",
    "ec2:DescribeVpcAttribute",
    "ec2:DescribeVpcEndpoints",
    "ec2:DescribeVpcs"
   ],
   "Resource": "*"
  },
  {
   "Sid": "CreateDefaultSecurityGroupWithEMRTags",
   "Effect": "Allow",
   "Action": [
    "ec2:CreateSecurityGroup"
   ],
   "Resource": [
    "arn:aws:ec2:*:*:security-group/*"
   ],
   "Condition": {
    "StringEquals": {
     "aws:RequestTag/for-use-with-amazon-emr-managed-policies": "true"
    }
   }
  },
  {
   "Sid": "CreateDefaultSecurityGroupInVPCWithEMRTags",
   "Effect": "Allow",
   "Action": [
    "ec2:CreateSecurityGroup"
   ],
   "Resource": [
    "arn:aws:ec2:*:*:vpc/*"
   ],
   "Condition": {
    "StringEquals": {
     "aws:ResourceTag/for-use-with-amazon-emr-managed-policies": "true"
    }
   }
  },
  {
   "Sid": "TagOnCreateDefaultSecurityGroupWithEMRTags",
   "Effect": "Allow",
   "Action": [
    "ec2:CreateTags"
   ],
   "Resource": "arn:aws:ec2:*:*:security-group/*",
```

```
    "Condition": {
     "StringEquals": {
      "aws:RequestTag/for-use-with-amazon-emr-managed-policies": "true",
      "ec2:CreateAction": "CreateSecurityGroup"
     }
    }
   },
   {
    "Sid": "ManageSecurityGroups",
    "Effect": "Allow",
    "Action": [
     "ec2:AuthorizeSecurityGroupEgress",
     "ec2:AuthorizeSecurityGroupIngress",
     "ec2:RevokeSecurityGroupEgress",
     "ec2:RevokeSecurityGroupIngress"
    ],
    "Resource": "*",
    "Condition": {
     "StringEquals": {
      "aws:ResourceTag/for-use-with-amazon-emr-managed-policies": "true"
     }
    }
   },
   {
    "Sid": "CreateEMRPlacementGroups",
    "Effect": "Allow",
    "Action": [
     "ec2:CreatePlacementGroup"
    ],
    "Resource": "arn:aws:ec2:*:*:placement-group/pg-*"
   },
   {
    "Sid": "DeletePlacementGroups",
    "Effect": "Allow",
    "Action": [
     "ec2:DeletePlacementGroup"
    ],
    "Resource": "*"
   },
   {
    "Sid": "AutoScaling",
    "Effect": "Allow",
    "Action": [
     "application-autoscaling:DeleteScalingPolicy",
```

```
    "application-autoscaling:DeregisterScalableTarget",
    "application-autoscaling:DescribeScalableTargets",
    "application-autoscaling:DescribeScalingPolicies",
    "application-autoscaling:PutScalingPolicy",
    "application-autoscaling:RegisterScalableTarget"
   ],
   "Resource": "*"
  },
  {
   "Sid": "ResourceGroupsForCapacityReservations",
   "Effect": "Allow",
   "Action": [
    "resource-groups:ListGroupResources"
   ],
   "Resource": "*"
  },
  {
   "Sid": "AutoScalingCloudWatch",
   "Effect": "Allow",
   "Action": [
    "cloudwatch:PutMetricAlarm",
    "cloudwatch:DeleteAlarms",
    "cloudwatch:DescribeAlarms"
   ],
   "Resource": "arn:aws:cloudwatch:*:*:alarm:*_EMR_Auto_Scaling"
  },
  {
   "Sid": "PassRoleForAutoScaling",
   "Effect": "Allow",
   "Action": "iam:PassRole",
   "Resource": "arn:aws:iam::*:role/EMR_AutoScaling_DefaultRole",
   "Condition": {
    "StringLike": {
     "iam:PassedToService": "application-autoscaling.amazonaws.com*"
    }
   }
  },
  {
   "Sid": "PassRoleForEC2",
   "Effect": "Allow",
   "Action": "iam:PassRole",
   "Resource": "arn:aws:iam::*:role/EMR_EC2_DefaultRole",
   "Condition": {
    "StringLike": {
```

```
      "iam:PassedToService": "ec2.amazonaws.com*"
    }
   }
  }
 ]
}
```

Ihre Servicerolle sollte die folgende Vertrauensrichtlinie verwenden.

> **Important**
>
> Die folgende Vertrauensrichtlinie umfasst die Schlüssel `aws:SourceArn`und die `aws:SourceAccount`globalen Bedingungsschlüssel, die die Berechtigungen einschränken, die Sie Amazon EMR für bestimmte Ressourcen in Ihrem Konto gewähren. Auf diese Weise können Sie sich vor dem Problem des verwirrten Stellvertreters schützen.

```
{
    "Version": "2008-10-17",
    "Statement": [
        {
            "Sid": "",
            "Effect": "Allow",
            "Principal": {
                "Service": "elasticmapreduce.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "<account-id>"
                },
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:elasticmapreduce:<region>:<account-id>:*"
                }
            }
        }
    ]
}
```

Servicerolle für EC2 Cluster-Instances (EC2Instance-Profil)

Die Servicerolle für EC2 Cluster-Instances (auch EC2 Instance-Profil für Amazon genanntEMR) ist eine spezielle Art von Servicerolle, die jeder EC2 Instance in einem EMR Amazon-Cluster zugewiesen wird, wenn die Instance gestartet wird. Anwendungsprozesse, die auf der Hadoop-Ökosystem ausgeführt werden, übernehmen diese Rolle für Berechtigungen für die Interaktion mit anderen AWS -Services.

Weitere Informationen zu Servicerollen für EC2 Instances finden Sie im IAMBenutzerhandbuch unter Verwenden einer IAM Rolle zum Erteilen von Berechtigungen für Anwendungen, die auf EC2 Amazon-Instances ausgeführt werden.

> **Important**
>
> Die Standard-Servicerolle für EC2 Cluster-Instances und die zugehörige verwaltete AWS Standardrichtlinie `AmazonElasticMapReduceforEC2Role` sind inzwischen veraltet, und es werden keine neuen AWS verwalteten Richtlinien bereitgestellt. Sie müssen ein Instance-Profil erstellen und angeben, um die veraltete Rolle und die Standardrichtlinie zu ersetzen.

Standardrolle und verwaltete Richtlinie

- Der Standardrollenname ist `EMR_EC2_DefaultRole`.
- Die `EMR_EC2_DefaultRole` standardmäßige verwaltete Richtlinie,`AmazonElasticMapReduceforEC2Role`, nähert sich dem Ende des Supports. Anstatt eine verwaltete Standardrichtlinie für das EC2 Instance-Profil zu verwenden, wenden Sie ressourcenbasierte Richtlinien auf S3-Buckets und andere Ressourcen an, die Amazon EMR benötigt, oder verwenden Sie Ihre eigene, vom Kunden verwaltete Richtlinie mit einer IAM Rolle als Instance-Profil. Weitere Informationen finden Sie unter Erstellen Sie eine Servicerolle für EC2 Cluster-Instances mit Berechtigungen mit den geringsten Rechten.

Im Folgenden werden die Inhalte von Version 3 von `AmazonElasticMapReduceforEC2Role` gezeigt.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
```

```
            "Resource": "*",
            "Action": [
                "cloudwatch:*",
                "dynamodb:*",
                "ec2:Describe*",
                "elasticmapreduce:Describe*",
                "elasticmapreduce:ListBootstrapActions",
                "elasticmapreduce:ListClusters",
                "elasticmapreduce:ListInstanceGroups",
                "elasticmapreduce:ListInstances",
                "elasticmapreduce:ListSteps",
                "kinesis:CreateStream",
                "kinesis:DeleteStream",
                "kinesis:DescribeStream",
                "kinesis:GetRecords",
                "kinesis:GetShardIterator",
                "kinesis:MergeShards",
                "kinesis:PutRecord",
                "kinesis:SplitShard",
                "rds:Describe*",
                "s3:*",
                "sdb:*",
                "sns:*",
                "sqs:*",
                "glue:CreateDatabase",
                "glue:UpdateDatabase",
                "glue:DeleteDatabase",
                "glue:GetDatabase",
                "glue:GetDatabases",
                "glue:CreateTable",
                "glue:UpdateTable",
                "glue:DeleteTable",
                "glue:GetTable",
                "glue:GetTables",
                "glue:GetTableVersions",
                "glue:CreatePartition",
                "glue:BatchCreatePartition",
                "glue:UpdatePartition",
                "glue:DeletePartition",
                "glue:BatchDeletePartition",
                "glue:GetPartition",
                "glue:GetPartitions",
                "glue:BatchGetPartition",
                "glue:CreateUserDefinedFunction",
```

```
                "glue:UpdateUserDefinedFunction",
                "glue:DeleteUserDefinedFunction",
                "glue:GetUserDefinedFunction",
                "glue:GetUserDefinedFunctions"
            ]
        }
    ]
}
```

Ihre Servicerolle sollte die folgende Vertrauensrichtlinie verwenden.

```
{
    "Version": "2008-10-17",
    "Statement": [
        {
            "Sid": "",
            "Effect": "Allow",
            "Principal": {
                "Service": "ec2.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

Erstellen Sie eine Servicerolle für EC2 Cluster-Instances mit Berechtigungen mit den geringsten Rechten

Als bewährte Methode empfehlen wir dringend, eine Servicerolle für EC2 Cluster-Instances und eine Berechtigungsrichtlinie zu erstellen, die über die Mindestberechtigungen für andere AWS Dienste verfügt, die für Ihre Anwendung erforderlich sind.

Die standardmäßige verwaltete Richtlinie, `AmazonElasticMapReduceforEC2Role`, bietet Berechtigungen, mit denen Sie problemlos einen ersten Cluster starten können. `AmazonElasticMapReduceforEC2Role`ist jedoch auf dem Weg, veraltet zu werden, und Amazon EMR wird keine Ersatzrichtlinie für die AWS verwaltete Standardrichtlinie für die veraltete Rolle bereitstellen. Um einen ersten Cluster zu starten, müssen Sie eine vom Kunden verwaltete, ressourcenbasierte oder ID-basierte Richtlinie bereitstellen.

Die folgenden Richtlinienerklärungen enthalten Beispiele für die Berechtigungen, die für verschiedene Funktionen von Amazon erforderlich sindEMR. Wir empfehlen, diese Berechtigungen zu verwenden,

um eine Berechtigungsrichtlinie zu erstellen, die den Zugriff auf nur diese Funktionen und Ressourcen beschränkt, die Ihr Cluster erfordert. Alle beispielhaften Richtlinienerklärungen verwenden *us-west-2* Region und die fiktive AWS Konto-ID *123456789012*. Ersetzen Sie diese je nach Bedarf für Ihren Cluster.

Weitere Informationen zum Erstellen und Angeben benutzerdefinierter Rollen finden Sie unter Passen Sie IAM Rollen an.

> **Note**
>
> Wenn Sie eine benutzerdefinierte EMR Rolle für erstellenEC2, folgen Sie dem grundlegenden Arbeitsablauf, der automatisch ein Instanzprofil mit demselben Namen erstellt. Amazon EC2 ermöglicht es Ihnen, Instance-Profile und Rollen mit unterschiedlichen Namen zu erstellen, Amazon unterstützt diese Konfiguration jedoch EMR nicht und führt bei der Erstellung des Clusters zu einem Fehler „Ungültiges Instanzprofil“.

Lesen und Schreiben von Daten in Amazon S3 mit EMRFS

Wenn eine Anwendung, die auf einem EMR Amazon-Cluster ausgeführt wird, Daten im `s3://mydata` Format referenziert, EMR verwendet Amazon das EC2 Instance-Profil, um die Anfrage zu stellen. Cluster lesen und schreiben in der Regel Daten auf diese Weise in Amazon S3, und Amazon EMR verwendet standardmäßig die mit der Service-Rolle verknüpften Berechtigungen für EC2 Cluster-Instances. Weitere Informationen finden Sie unter IAMRollen für EMRFS Anfragen an Amazon S3 konfigurieren.

Da IAM Rollen für EMRFS auf die Berechtigungen zurückgreifen, die mit der Service-Rolle für EC2 Cluster-Instances verknüpft sind, empfehlen wir als bewährte Methode, IAM Rollen für zu verwenden und die Amazon S3 S3-BerechtigungenEMRFS, die mit der EMRFS Servicerolle für EC2 Cluster-Instances verknüpft sind, zu beschränken.

Die folgende Beispielerklärung zeigt die Berechtigungen, die EMRFS erforderlich sind, um Anfragen an Amazon S3 zu stellen.

- *my-data-bucket-in-s3- for-emrfs-reads-and -schreibt* spezifiziert den Bucket in Amazon S3, in den der Cluster Daten liest und schreibt, sowie alle Unterordner mit */**. Fügen Sie nur die Buckets und Ordner hinzu, die Ihre Anwendung benötigt.

- Die Richtlinienerklärung, die dynamodb Aktionen zulässt, ist nur erforderlich, wenn die EMRFS konsistente Ansicht aktiviert ist. *E mrFSMetadata* gibt den Standardordner für die EMRFS konsistente Ansicht an.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:AbortMultipartUpload",
                "s3:CreateBucket",
                "s3:DeleteObject",
                "s3:GetBucketVersioning",
                "s3:GetObject",
                "s3:GetObjectTagging",
                "s3:GetObjectVersion",
                "s3:ListBucket",
                "s3:ListBucketMultipartUploads",
                "s3:ListBucketVersions",
                "s3:ListMultipartUploadParts",
                "s3:PutBucketVersioning",
                "s3:PutObject",
                "s3:PutObjectTagging"
            ],
            "Resource": [
                "arn:aws:s3:::my-data-bucket-in-s3-for-emrfs-reads-and-writes",
                "arn:aws:s3:::my-data-bucket-in-s3-for-emrfs-reads-and-writes/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "dynamodb:CreateTable",
                "dynamodb:BatchGetItem",
                "dynamodb:BatchWriteItem",
                "dynamodb:PutItem",
                "dynamodb:DescribeTable",
                "dynamodb:DeleteItem",
                "dynamodb:GetItem",
                "dynamodb:Scan",
                "dynamodb:Query",
```

```
                "dynamodb:UpdateItem",
                "dynamodb:DeleteTable",
                "dynamodb:UpdateTable"
            ],
            "Resource": "arn:aws:dynamodb:us-west-2:123456789012:table/EmrFSMetadata"
        },
        {
            "Effect": "Allow",
            "Action": [
                "cloudwatch:PutMetricData",
                "dynamodb:ListTables",
                "s3:ListBucket"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "sqs:GetQueueUrl",
                "sqs:ReceiveMessage",
                "sqs:DeleteQueue",
                "sqs:SendMessage",
                "sqs:CreateQueue"
            ],
            "Resource": "arn:aws:sqs:us-west-2:123456789012:EMRFS-Inconsistency-*"
        }
    ]
}
```

Archivieren von Protokolldateien in Amazon S3

Die folgende Richtlinienerklärung ermöglicht es dem EMR Amazon-Cluster, Protokolldateien am angegebenen Amazon S3-Speicherort zu archivieren. Im folgenden Beispiel, als der Cluster erstellt wurde, *s3://MyLoggingBucket/M-Protokolle yEMRCluster* wurde über den Speicherort des Protokollordners S3 in der Konsole, mithilfe der `--log-uri` Option von oder mithilfe des `LogUri` Parameters im `RunJobFlow` Befehl angegeben. AWS CLI Weitere Informationen finden Sie unter Archivieren von Protokolldateien in Amazon S3.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
```

```
            "Action": "s3:PutObject",
            "Resource": "arn:aws:s3:::MyLoggingBucket/MyEMRClusterLogs/*"
        }
    ]
}
```

Verwenden des AWS Glue-Datenkatalogs

Die folgende Richtlinienerklärung erlaubt Aktionen, die erforderlich sind, wenn Sie den AWS Glue-Datenkatalog als Metastore für Anwendungen verwenden. Weitere Informationen finden Sie unter Verwenden des AWS Glue-Datenkatalogs als Metastore für Spark SQL, Verwenden des AWS Glue-Datenkatalogs als Metastore für Hive und Verwenden von Presto mit dem AWS Glue-Datenkatalog im Amazon-Versionshandbuch. EMR

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "glue:CreateDatabase",
                "glue:UpdateDatabase",
                "glue:DeleteDatabase",
                "glue:GetDatabase",
                "glue:GetDatabases",
                "glue:CreateTable",
                "glue:UpdateTable",
                "glue:DeleteTable",
                "glue:GetTable",
                "glue:GetTables",
                "glue:GetTableVersions",
                "glue:CreatePartition",
                "glue:BatchCreatePartition",
                "glue:UpdatePartition",
                "glue:DeletePartition",
                "glue:BatchDeletePartition",
                "glue:GetPartition",
                "glue:GetPartitions",
                "glue:BatchGetPartition",
                "glue:CreateUserDefinedFunction",
                "glue:UpdateUserDefinedFunction",
                "glue:DeleteUserDefinedFunction",
```

```
                "glue:GetUserDefinedFunction",
                "glue:GetUserDefinedFunctions"
            ],
            "Resource": "*",
        }
    ]
}
```

Servicerolle für Auto Scaling in Amazon EMR (Auto Scaling-Rolle)

Die Auto Scaling Scaling-Rolle für Amazon EMR erfüllt eine ähnliche Funktion wie die Service-Rolle, ermöglicht jedoch zusätzliche Aktionen für dynamisch skalierende Umgebungen.

- Der Standardrollenname ist `EMR_AutoScaling_DefaultRole`.
- Die an `EMR_AutoScaling_DefaultRole` angefügte standardmäßige verwaltete Richtlinie ist `AmazonElasticMapReduceforAutoScalingRole`.

Der Inhalt von Version 1 `AmazonElasticMapReduceforAutoScalingRole` wird unten angezeigt.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "cloudwatch:DescribeAlarms",
                "elasticmapreduce:ListInstanceGroups",
                "elasticmapreduce:ModifyInstanceGroups"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

Ihre Servicerolle sollte die folgende Vertrauensrichtlinie verwenden.

> **Important**
>
> Die folgende Vertrauensrichtlinie umfasst die Schlüssel `aws:SourceArn`und die `aws:SourceAccount`globalen Bedingungsschlüssel, die die Berechtigungen einschränken,

die Sie Amazon EMR für bestimmte Ressourcen in Ihrem Konto gewähren. Auf diese Weise können Sie sich vor dem Problem des verwirrten Stellvertreters schützen.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "application-autoscaling.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "<account-id>"
                },
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:application-
autoscaling:<region>:<account-id>:scalable-target/*"
                }
            }
        }
    ]
}
```

Servicerolle für EMR Notebooks

Jedes EMR Notizbuch benötigt Berechtigungen, um auf andere AWS Ressourcen zuzugreifen und Aktionen auszuführen. Die mit dieser Servicerolle verknüpften IAM Richtlinien gewähren dem Notebook Berechtigungen für die Zusammenarbeit mit anderen AWS Diensten. Wenn Sie mit dem ein Notizbuch erstellen AWS Management Console, geben Sie eine AWS Servicerolle an. Sie können die Standardrolle, `EMR_Notebooks_DefaultRole`, verwenden oder eine Rolle angeben, die Sie erstellen. Wenn ein Notebook nicht vorher erstellt wurde, haben Sie die Möglichkeit, die Standardrolle zu erstellen.

- Der Standardrollenname ist `EMR_Notebooks_DefaultRole`.
- Die standardmäßig angehängten verwalteten Richtlinien zu `EMR_Notebooks_DefaultRole` sind `AmazonElasticMapReduceEditorsRole` und `S3FullAccessPolicy`.

Ihre Servicerolle sollte die folgende Vertrauensrichtlinie verwenden.

> **Important**
>
> Die folgende Vertrauensrichtlinie umfasst die Schlüssel `aws:SourceArn`und die `aws:SourceAccount`globalen Bedingungsschlüssel, die die Berechtigungen einschränken, die Sie Amazon EMR für bestimmte Ressourcen in Ihrem Konto gewähren. Auf diese Weise können Sie sich vor dem Problem des verwirrten Stellvertreters schützen.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "elasticmapreduce.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "<account-id>"
                },
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:elasticmapreduce:<region>:<account-id>:*"
                }
            }
        }
    ]
}
```

Der Inhalt von Version 1 von `AmazonElasticMapReduceEditorsRole` lautet wie folgt.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AuthorizeSecurityGroupEgress",
                "ec2:AuthorizeSecurityGroupIngress",
```

```
                "ec2:CreateSecurityGroup",
                "ec2:DescribeSecurityGroups",
                "ec2:RevokeSecurityGroupEgress",
                "ec2:CreateNetworkInterface",
                "ec2:CreateNetworkInterfacePermission",
                "ec2:DeleteNetworkInterface",
                "ec2:DeleteNetworkInterfacePermission",
                "ec2:DescribeNetworkInterfaces",
                "ec2:ModifyNetworkInterfaceAttribute",
                "ec2:DescribeTags",
                "ec2:DescribeInstances",
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "elasticmapreduce:ListInstances",
                "elasticmapreduce:DescribeCluster",
                "elasticmapreduce:ListSteps"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "ec2:CreateTags",
            "Resource": "arn:aws:ec2:*:*:network-interface/*",
            "Condition": {
                "ForAllValues:StringEquals": {
                    "aws:TagKeys": [
                        "aws:elasticmapreduce:editor-id",
                        "aws:elasticmapreduce:job-flow-id"
                    ]
                }
            }
        }
    ]
}
```

Im Folgenden sehen Sie den Inhalt von `S3FullAccessPolicy`. Das `S3FullAccessPolicy` ermöglicht es Ihrer Servicerolle für EMR Notebooks, alle Amazon S3 S3-Aktionen an Objekten in Ihrem auszuführen AWS-Konto. Wenn Sie eine benutzerdefinierte Servicerolle für EMR Notebooks erstellen, müssen Sie Ihrer Servicerolle Amazon S3 S3-Berechtigungen erteilen.

```
{
    "Version": "2012-10-17",
    "Statement": [
```

```
        {
            "Effect": "Allow",
            "Action": "s3:*",
            "Resource": "*"
        }
    ]
}
```

Sie können den Lese- und Schreibzugriff für Ihre Servicerolle auf den Amazon-S3-Standort beschränken, an dem Sie Ihre Notebookdateien speichern möchten. Verwenden Sie die folgenden Mindestberechtigungen an Amazon S3.

```
"s3:PutObject",
"s3:GetObject",
"s3:GetEncryptionConfiguration",
"s3:ListBucket",
"s3:DeleteObject"
```

Wenn Ihr Amazon-S3-Bucket verschlüsselt ist, müssen Sie die folgenden Berechtigungen für AWS Key Management Service angeben.

```
"kms:Decrypt",
"kms:GenerateDataKey",
"kms:ReEncryptFrom",
"kms:ReEncryptTo",
"kms:DescribeKey"
```

Wenn Sie Git-Repositorys mit Ihrem Notebook verknüpfen und ein Geheimnis für das Repository erstellen müssen, müssen Sie die `secretsmanager:GetSecretValue` Berechtigung in der IAM Richtlinie hinzufügen, die der Servicerolle für EMR Amazon-Notebooks beigefügt ist. Eine Beispielrichtlinie wird nachfolgend gezeigt:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": "secretsmanager:GetSecretValue",
            "Resource": "*"
```

```
        }
    ]
}
```

EMRBerechtigungen für die Servicerolle Notebooks

In dieser Tabelle sind die Aktionen aufgeführt, die EMR Notebooks mithilfe der Servicerolle durchführt, sowie die Berechtigungen, die für jede Aktion erforderlich sind.

Aktion	Berechtigungen
Richten Sie einen sicheren Netzwerkkanal zwischen einem Notebook und einem EMR Amazon-Cluster ein und führen Sie die erforderlichen Bereinigungsaktionen durch.	`"ec2:CreateNetworkInterface", "ec2:CreateNetworkInterfacePermission", "ec2:DeleteNetworkInterface", "ec2:DeleteNetworkInterfacePermission", "ec2:DescribeNetworkInterfaces", "ec2:ModifyNetworkInterfaceAttribute", "ec2:AuthorizeSecurityGroupEgress", "ec2:AuthorizeSecurityGroupIngress", "ec2:CreateSecurityGroup", "ec2:DescribeSecurityGroups", "ec2:RevokeSecurityGroupEgress", "ec2:DescribeTags", "ec2:DescribeInstances", "ec2:DescribeSubnets", "ec2:DescribeVpcs", "elasticmapreduce:ListInstances", "elasticmapreduce:DescribeCluster", "elasticmapreduce:ListSteps"`
Verwenden Sie die in gespeicherten Git-Anmeldeinformationen AWS Secrets Manager , um Git-Repositorys mit einem Notizbuch zu verknüpfen.	`"secretsmanager:GetSecretValue"`
Wenden Sie AWS Tags auf die Netzwerkschnittstelle und die Standardsicherheitsgruppen an, die EMR Notebooks bei der Einrichtu	`"ec2:CreateTags"`

Aktion	Berechtigungen
ng des sicheren Netzwerkkanals erstellt. Weitere Informationen finden Sie unter Markieren von AWS -Ressourcen.	

Greifen Sie auf Notebook-Dateien und Metadaten zu oder laden Sie sie in Amazon S3 hoch.	`"s3:PutObject",`
	`"s3:GetObject",`
	`"s3:GetEncryptionConfiguration",`
	`"s3:ListBucket",`
	`"s3:DeleteObject"`
	
	Wenn Sie einen verschlüsselten Amazon-S3-Bucket verwenden, sind die folgenden Berechtigungen erforderlich.
	
	`"kms:Decrypt",`
	`"kms:GenerateDataKey",`
	`"kms:ReEncryptFrom",`
	`"kms:ReEncryptTo",`
	`"kms:DescribeKey"`

EMRNotebooks aktualisiert AWS verwaltete Richtlinien

Hier finden Sie Informationen zu Aktualisierungen der AWS verwalteten Richtlinien für EMR Notebooks seit dem 1. März 2021.

Änderung	Beschreibung	Datum
`AmazonElasticMapReduceEditorsRole - Added permissions`	EMRNotizbücher hinzugefügt `ec2:describeVPCs` und `elastmicmapreduce:ListSteps` Berechtigungen für`AmazonElasticMapReduceEditorsRole` .	8. Februar 2023

Änderung	Beschreibung	Datum
EMRNotizbücher haben begonnen, Änderungen nachzuverfolgen	EMRNotebooks begann, Änderungen für die von AWS ihm verwalteten Richtlinien nachzuverfolgen.	8. Februar 2023

Verwenden von serviceverknüpften Rollen für Amazon EMR

Amazon EMR verwendet serviceverknüpfte Rollen AWS Identity and Access Management (IAM). Eine servicebezogene Rolle ist ein einzigartiger IAM Rollentyp, der direkt mit Amazon EMR verknüpft ist. Servicebezogene Rollen sind von Amazon vordefiniert EMR und beinhalten alle Berechtigungen, die der Service benötigt, um andere AWS Dienste in Ihrem Namen aufzurufen.

Themen

- Verwenden von dienstbezogenen Rollen für die Bereinigung
- Verwenden von serviceverknüpften Rollen für die Write-Ahead-Protokollierung

Informationen zu anderen Diensten, die dienstbezogene Rollen unterstützen, finden Sie unter AWS Dienste, die mit Services arbeiten, IAM und suchen Sie in der Spalte Servicebezogene Rollen nach den Diensten, für die Ja steht. Wählen Sie über einen Link Ja aus, um die Dokumentation zu einer serviceverknüpften Rolle für diesen Service anzuzeigen.

Verwenden von dienstbezogenen Rollen für die Bereinigung

Amazon EMR verwendet serviceverknüpfte Rollen AWS Identity and Access Management (IAM). Eine servicebezogene Rolle ist ein einzigartiger IAM Rollentyp, der direkt mit Amazon EMR verknüpft ist. Servicebezogene Rollen sind von Amazon vordefiniert EMR und beinhalten alle Berechtigungen, die der Service benötigt, um andere AWS Dienste in Ihrem Namen aufzurufen.

Serviceverknüpfte Rollen arbeiten mit der EMR Amazon-Servicerolle und dem EC2 Amazon-Instanzprofil für Amazon EMR zusammen. Weitere Informationen über die Service-Rolle und das Instance-Profil finden Sie unter IAMServicerollen für EMR Amazon-Berechtigungen für AWS Dienste und Ressourcen konfigurieren.

Eine serviceverknüpfte Rolle EMR erleichtert die Einrichtung von Amazon, da Sie die erforderlichen Berechtigungen nicht manuell hinzufügen müssen. Amazon EMR definiert die Berechtigungen seiner serviceverknüpften Rollen, und sofern nicht anders definiert, EMR kann nur Amazon seine

Rollen übernehmen. Zu den definierten Berechtigungen gehören die Vertrauensrichtlinie und die Berechtigungsrichtlinie, und diese Berechtigungsrichtlinie kann keiner anderen IAM Entität zugeordnet werden.

Sie können diese serviceverknüpfte Rolle für Amazon EMR erst löschen, nachdem Sie alle zugehörigen Ressourcen gelöscht und alle EMR Cluster im Konto beendet haben. Dadurch werden Ihre EMR Amazon-Ressourcen geschützt, sodass Sie nicht versehentlich die Zugriffsberechtigung für die Ressourcen entziehen können.

Verwenden von Rollen, die mit Services verknüpft sind, für die Bereinigung

Amazon EMR verwendet die servicebasierte AWSServiceRoleForEMRCleanupRolle, um Amazon die EMR Erlaubnis zu erteilen, EC2 Amazon-Ressourcen in Ihrem Namen zu beenden und zu löschen, falls die mit dem EMR Amazon-Dienst verknüpfte Rolle diese Funktion verliert. Amazon EMR erstellt die serviceverknüpfte Rolle automatisch während der Clustererstellung, sofern sie noch nicht vorhanden ist.

Die AWSServiceRoleForEMRCleanup serviceverknüpfte Rolle vertraut darauf, dass die folgenden Dienste die Rolle übernehmen:

- `elasticmapreduce.amazonaws.com`

Die Richtlinie für AWSServiceRoleForEMRCleanup servicebezogene Rollenberechtigungen ermöglicht es AmazonEMR, die folgenden Aktionen für die angegebenen Ressourcen durchzuführen:

- Aktion: `DescribeInstances` für `ec2`
- Aktion: `DescribeSpotInstanceRequests` für `ec2`
- Aktion: `ModifyInstanceAttribute` für `ec2`
- Aktion: `TerminateInstances` für `ec2`
- Aktion: `CancelSpotInstanceRequests` für `ec2`
- Aktion: `DeleteNetworkInterface` für `ec2`
- Aktion: `DescribeInstanceAttribute` für `ec2`
- Aktion: `DescribeVolumeStatus` für `ec2`
- Aktion: `DescribeVolumes` für `ec2`
- Aktion: `DetachVolume` für `ec2`
- Aktion: `DeleteVolume` für `ec2`

Sie müssen Berechtigungen konfigurieren, damit eine IAM Entität (z. B. ein Benutzer, eine Gruppe oder eine Rolle) eine dienstbezogene Rolle erstellen, bearbeiten oder löschen kann.

Eine serviceverknüpfte Rolle für Amazon erstellen EMR

Sie müssen die AWSServiceRoleForEMRCleanup Rolle nicht manuell erstellen.
Wenn Sie einen Cluster starten, entweder zum ersten Mal oder wenn die AWSServiceRoleForEMRCleanup serviceverknüpfte Rolle nicht vorhanden ist, EMR erstellt Amazon die AWSServiceRoleForEMRCleanup serviceverknüpfte Rolle für Sie. Sie müssen über die erforderlichen Berechtigungen verfügen, um eine serviceverknüpfte Rolle zu erstellen. Eine Beispielanweisung, mit der diese Funktion zur Berechtigungsrichtlinie einer IAM Entität (z. B. eines Benutzers, einer Gruppe oder Rolle) hinzugefügt wird, finden Sie unterVerwenden von dienstbezogenen Rollen für die Bereinigung.

> **Important**
>
> Wenn Sie Amazon EMR vor dem 24. Oktober 2017 verwendet haben, als serviceverknüpfte Rollen nicht unterstützt wurden, EMR hat Amazon die AWSServiceRoleForEMRCleanup serviceverknüpfte Rolle in Ihrem Konto erstellt. Weitere Informationen finden Sie unter Eine neue Rolle wurde in meinem IAM Konto angezeigt.

Bearbeitung einer serviceverknüpften Rolle für Amazon EMR

Amazon erlaubt Ihnen EMR nicht, die AWSServiceRoleForEMRCleanup serviceverknüpfte Rolle zu bearbeiten. Nachdem Sie eine serviceverknüpfte Rolle erstellt haben, können Sie den Namen der serviceverknüpften Rolle nicht mehr ändern, da verschiedene Entitäten möglicherweise auf die serviceverknüpfte Rolle verweisen. Sie können die Beschreibung der dienstbezogenen Rolle jedoch mithilfe von bearbeiten. IAM

Beschreibung einer dienstbezogenen Rolle bearbeiten (Konsole) IAM

Sie können die IAM Konsole verwenden, um die Beschreibung einer dienstbezogenen Rolle zu bearbeiten.

So bearbeiten Sie die Beschreibung einer serviceverknüpften Rolle (Konsole)

1. Wählen Sie im Navigationsbereich der IAM Konsole die Option Rollen aus.
2. Wählen Sie den Namen der zu ändernden Rolle.

3. Wählen Sie neben Rollenbeschreibung rechts Bearbeiten aus.
4. Geben Sie eine neue Beschreibung im Dialogfeld ein und wählen Sie Save changes (Änderungen speichern).

Bearbeiten einer mit einem Dienst verknüpften Rollenbeschreibung () IAM CLI

Sie können IAM Befehle von verwenden AWS Command Line Interface , um die Beschreibung einer dienstbezogenen Rolle zu bearbeiten.

Um die Beschreibung einer dienstbezogenen Rolle zu ändern () CLI

1. (Optional) Um die aktuelle Beschreibung einer Rolle anzuzeigen, verwenden Sie die folgenden Befehle:

   ```
   $ aws iam get-role --role-name role-name
   ```

   Verwenden Sie den Rollennamen, nicht denARN, um mit den CLI Befehlen auf Rollen zu verweisen. Wenn eine Rolle beispielsweise Folgendes hatARN:`arn:aws:iam::123456789012:role/myrole`, bezeichnen Sie die Rolle als**`myrole`**.
2. Um die Beschreibung einer serviceverknüpften Rolle zu aktualisieren, verwenden Sie einen der folgenden Befehle:

   ```
   $ aws iam update-role-description --role-name role-name --description description
   ```

Bearbeiten einer mit einem Dienst verknüpften Rollenbeschreibung () IAM API

Sie können den verwenden IAMAPI, um die Beschreibung einer dienstbezogenen Rolle zu bearbeiten.

Um die Beschreibung einer dienstbezogenen Rolle zu ändern () API

1. (Optional) Um die aktuelle Beschreibung einer Rolle anzuzeigen, verwenden Sie den folgenden Befehl:

   IAM API: GetRole
2. Um die Beschreibung einer Rolle zu aktualisieren, verwenden Sie den folgenden Befehl:

   IAM API: UpdateRoleDescription

Löschen einer serviceverknüpften Rolle für Amazon EMR

Wenn Sie eine Funktion oder einen Dienst nicht mehr verwenden müssen, für den eine dienstbezogene Rolle erforderlich ist, empfehlen wir Ihnen, diese dienstbezogene Rolle zu löschen. Auf diese Weise haben Sie keine ungenutzte Entität, die nicht aktiv überwacht oder verwaltet wird. Sie müssen jedoch Ihre serviceverknüpfte Rolle zunächst bereinigen, bevor Sie sie löschen können.

Bereinigen einer serviceverknüpften Rolle

Bevor Sie eine dienstverknüpfte Rolle löschen könnenIAM, müssen Sie zunächst sicherstellen, dass die dienstverknüpfte Rolle keine aktiven Sitzungen hat, und alle Ressourcen entfernen, die von der dienstbezogenen Rolle verwendet werden.

Um zu überprüfen, ob die dienstverknüpfte Rolle über eine aktive Sitzung in der Konsole verfügt IAM

1. Öffnen Sie die IAM Konsole unter. https://console.aws.amazon.com/iam/
2. Wählen Sie im Navigationsbereich Rollen aus. Wählen Sie den Namen (nicht das Kontrollkästchen) der AWSServiceRoleForEMRCleanup serviceverknüpften Rolle aus.
3. Wählen Sie auf der Übersichtsseite für die ausgewählte serviceverknüpfte Rolle die Option Access Advisor aus.
4. Überprüfen Sie auf der Registerkarte Access Advisor (Advisor aufrufen) die jüngsten Aktivitäten für die serviceverknüpfte Rolle.

> **Note**
>
> Wenn Sie sich nicht sicher sind, ob Amazon EMR die AWSServiceRoleForEMRCleanup serviceverknüpfte Rolle verwendet, können Sie versuchen, die serviceverknüpfte Rolle zu löschen. Wenn der Service die serviceverknüpfte Rolle verwendet, schlägt das Löschen fehl und Sie können die Regionen anzeigen, in denen die serviceverknüpfte Rolle verwendet wird. Wenn die dienstverknüpfte Rolle verwendet wird, müssen Sie warten, bis die Sitzung beendet ist, bevor Sie die dienstverknüpfte Rolle löschen können. Die Sitzung für eine serviceverknüpfte Rolle können Sie nicht widerrufen.

Um EMR Amazon-Ressourcen zu entfernen, die verwendet werden von AWSServiceRoleForEMRCleanup

- Beenden Sie alle Cluster in Ihrem Konto. Weitere Informationen finden Sie unter Einen Cluster beenden.

Löschen einer serviceverknüpften Rolle (IAMKonsole)

Sie können die IAM Konsole verwenden, um eine dienstverknüpfte Rolle zu löschen.

So löschen Sie eine serviceverknüpfte Rolle (Konsole)

1. Öffnen Sie die IAM Konsole unter. https://console.aws.amazon.com/iam/
2. Wählen Sie im Navigationsbereich Rollen aus. Aktivieren Sie das Kontrollkästchen neben AWSServiceRoleForEMRCleanup, nicht den Namen oder die Zeile selbst.
3. Wählen Sie für Role actions oben auf der Seite Delete role aus.
4. Überprüfen Sie im Bestätigungsdialogfeld die Daten, auf die der Dienst zuletzt zugegriffen hat. Aus diesen Daten geht hervor, wann jede der ausgewählten Rollen zuletzt auf einen AWS Dienst zugegriffen hat. Auf diese Weise können Sie leichter bestätigen, ob die Rolle derzeit aktiv ist. Wählen Sie Yes, Delete, um fortzufahren.
5. Sehen Sie sich die IAM Konsolenbenachrichtigungen an, um den Fortschritt beim Löschen der dienstbezogenen Rolle zu verfolgen. Da das Löschen der IAM dienstbezogenen Rolle asynchron erfolgt, kann die Löschaufgabe erfolgreich sein oder fehlschlagen, nachdem Sie die dienstverknüpfte Rolle zum Löschen eingereicht haben. Wenn der Vorgang fehlschlägt, können Sie in den Benachrichtigungen View details oder View Resources auswählen, um zu erfahren, warum die Löschung fehlgeschlagen ist. Wenn das Löschen fehlschlägt, weil der Service Ressourcen enthält, die von der Rolle verwendet werden, enthält die Angabe des Fehlergrundes eine Liste der Ressourcen.

Löschen einer serviceverknüpften Rolle () IAM CLI

Sie können IAM Befehle von verwenden, AWS Command Line Interface um eine dienstverknüpfte Rolle zu löschen. Da eine serviceverknüpfte Rolle nicht gelöscht werden kann, wenn sie verwendet wird oder ihr Ressourcen zugeordnet sind, müssen Sie eine Löschungsanforderung übermitteln. Wenn diese Bedingungen nicht erfüllt sind, kann diese Anforderung verweigert werden.

Um eine dienstverknüpfte Rolle zu löschen () CLI

1. Sie benötigen die `deletion-task-id` aus der Antwort, um den Status der Löschaufgabe zu überprüfen. Geben Sie den folgenden Befehl ein, um eine Löschanforderung für eine serviceverknüpfte Rolle zu übermitteln:

```
$ aws iam delete-service-linked-role --role-name AWSServiceRoleForEMRCleanup
```

2. Geben Sie den folgenden Befehl ein, um den Status der Löschaufgabe zu überprüfen:

```
$ aws iam get-service-linked-role-deletion-status --deletion-task-id deletion-task-id
```

   Der Status der Löschaufgabe kann `NOT_STARTED`, `IN_PROGRESS`, `SUCCEEDED` oder `FAILED` lauten. Wenn die Löschung fehlschlägt, gibt der Aufruf den Grund zurück, sodass Sie das Problem beheben können.

Löschen einer dienstverknüpften Rolle () IAM API

Sie können die verwenden IAMAPI, um eine dienstverknüpfte Rolle zu löschen. Da eine serviceverknüpfte Rolle nicht gelöscht werden kann, wenn sie verwendet wird oder ihr Ressourcen zugeordnet sind, müssen Sie eine Löschungsanforderung übermitteln. Wenn diese Bedingungen nicht erfüllt sind, kann diese Anforderung verweigert werden.

Um eine dienstverknüpfte Rolle zu löschen () API

1. Rufen Sie an, um eine Löschanfrage für eine dienstverknüpfte Rolle einzureichen. DeleteServiceLinkedRole Geben Sie in der Anfrage den AWSServiceRoleForEMRCleanup Rollennamen an.

   Sie benötigen die `DeletionTaskId` aus der Antwort, um den Status der Löschaufgabe zu überprüfen.

2. Rufen Sie an, um den Status des Löschvorgangs zu überprüfen GetServiceLinkedRoleDeletionStatus. Geben Sie in der Anforderung die `DeletionTaskId` an.

   Der Status der Löschaufgabe kann `NOT_STARTED`, `IN_PROGRESS`, `SUCCEEDED` oder `FAILED` lauten. Wenn die Löschung fehlschlägt, gibt der Aufruf den Grund zurück, sodass Sie das Problem beheben können.

Unterstützte Regionen für AWSServiceRoleForEMRCleanup

Amazon EMR unterstützt die Nutzung der AWSServiceRoleForEMRCleanup serviceverknüpften Rolle in den folgenden Regionen.

Name der Region	Regions-ID	Support bei Amazon EMR
USA Ost (Nord-Virginia)	us-east-1	Ja
USA Ost (Ohio)	us-east-2	Ja
USA West (Nordkalifornien)	us-west-1	Ja
USA West (Oregon)	us-west-2	Ja
Asien-Pazifik (Mumbai)	ap-south-1	Ja
Asien-Pazifik (Osaka)	ap-northeast-3	Ja
Asien-Pazifik (Seoul)	ap-northeast-2	Ja
Asien-Pazifik (Singapore)	ap-southeast-1	Ja
Asien-Pazifik (Sydney)	ap-southeast-2	Ja
Asien-Pazifik (Tokyo)	ap-northeast-1	Ja
Kanada (Zentral)	ca-central-1	Ja
Europa (Frankfurt)	eu-central-1	Ja
Europa (Ireland)	eu-west-1	Ja
Europa (London)	eu-west-2	Ja
Europa (Paris)	eu-west-3	Ja
Südamerika (São Paulo)	sa-east-1	Ja

Verwenden von serviceverknüpften Rollen für die Write-Ahead-Protokollierung

Amazon EMR verwendet serviceverknüpfte Rollen AWS Identity and Access Management (IAM). Eine servicebezogene Rolle ist ein einzigartiger IAM Rollentyp, der direkt mit Amazon EMR verknüpft ist. Servicebezogene Rollen sind von Amazon vordefiniert EMR und beinhalten alle Berechtigungen, die der Service benötigt, um andere AWS Dienste in Ihrem Namen aufzurufen.

Serviceverknüpfte Rollen arbeiten mit der EMR Amazon-Servicerolle und dem EC2 Amazon-Instanzprofil für Amazon EMR zusammen. Weitere Informationen über die Service-Rolle und das Instance-Profil finden Sie unter IAMServicerollen für EMR Amazon-Berechtigungen für AWS Dienste und Ressourcen konfigurieren.

Eine serviceverknüpfte Rolle EMR erleichtert die Einrichtung von Amazon, da Sie die erforderlichen Berechtigungen nicht manuell hinzufügen müssen. Amazon EMR definiert die Berechtigungen seiner serviceverknüpften Rollen, und sofern nicht anders definiert, EMR kann nur Amazon seine Rollen übernehmen. Zu den definierten Berechtigungen gehören die Vertrauensrichtlinie und die Berechtigungsrichtlinie, und diese Berechtigungsrichtlinie kann keiner anderen IAM Entität zugeordnet werden.

Sie können diese serviceverknüpfte Rolle für Amazon EMR erst löschen, nachdem Sie die zugehörigen Ressourcen gelöscht und alle EMR Cluster im Konto beendet haben. Dadurch werden Ihre EMR Amazon-Ressourcen geschützt, sodass Sie nicht versehentlich die Zugriffsberechtigung für die Ressourcen entziehen können.

Mit dem Dienst verknüpfte Rollenberechtigungen für Write-Ahead-Logging () WAL

Amazon EMR verwendet die serviceverknüpfte Rolle AWSServiceRoleForEMRWAL, um einen Cluster-Status abzurufen.

Die AWSServiceRoleForEMRWAL dienstbezogene Rolle vertraut darauf, dass die folgenden Dienste die Rolle übernehmen:

- `emrwal.amazonaws.com`

Die `EMRDescribeClusterPolicyForEMRWAL`Berechtigungsrichtlinie für die serviceverknüpfte Rolle ermöglicht es AmazonEMR, die folgenden Aktionen für die angegebenen Ressourcen durchzuführen:

- Aktion: `DescribeCluster` für `*`

Sie müssen Berechtigungen konfigurieren, damit eine IAM Entität (in diesem Fall Amazon EMRWAL) eine serviceverknüpfte Rolle erstellen, bearbeiten oder löschen kann. Fügen Sie der Berechtigungsrichtlinie für Ihr Instance-Profil nach Bedarf die folgenden Anweisungen hinzu:

CreateServiceLinkedRole

Um es einer IAM Entität zu ermöglichen, die AWSServiceRoleForEMRWAL serviceverknüpfte Rolle zu erstellen

Fügen Sie der Berechtigungsrichtlinie für die IAM Entität, die die dienstverknüpfte Rolle erstellen muss, die folgende Anweisung hinzu:

```
{
    "Effect": "Allow",
    "Action": [
        "iam:CreateServiceLinkedRole",
        "iam:PutRolePolicy"
    ],
    "Resource": "arn:aws:iam::*:role/aws-service-role/emrwal.amazonaws.com*/
AWSServiceRoleForEMRWAL*",
    "Condition": {
        "StringLike": {
            "iam:AWSServiceName": [
                "emrwal.amazonaws.com",
                "elasticmapreduce.amazonaws.com.cn"
            ]
        }
    }
}
```

UpdateRoleDescription

Um es einer IAM Entität zu ermöglichen, die Beschreibung der AWSServiceRoleForEMRWAL dienstbezogenen Rolle zu bearbeiten

Fügen Sie der Berechtigungsrichtlinie für die IAM Entität, die die Beschreibung einer dienstbezogenen Rolle bearbeiten muss, die folgende Anweisung hinzu:

```
{
    "Effect": "Allow",
    "Action": [
        "iam:UpdateRoleDescription"
    ],
    "Resource": "arn:aws:iam::*:role/aws-service-role/emrwal.amazonaws.com*/
AWSServiceRoleForEMRWAL*",
    "Condition": {
```

```
        "StringLike": {
            "iam:AWSServiceName": [
                "emrwal.amazonaws.com",
                "elasticmapreduce.amazonaws.com.cn"
            ]
        }
    }
}
```

DeleteServiceLinkedRole

Um einer IAM Entität das Löschen der AWSServiceRoleForEMRWAL dienstbezogenen Rolle zu ermöglichen

Fügen Sie der Berechtigungsrichtlinie für die IAM Entität, die eine dienstverknüpfte Rolle löschen muss, die folgende Anweisung hinzu:

```
{
    "Effect": "Allow",
    "Action": [
        "iam:DeleteServiceLinkedRole",
        "iam:GetServiceLinkedRoleDeletionStatus"
    ],
    "Resource": "arn:aws:iam::*:role/aws-service-role/elasticmapreduce.amazonaws.com*/
AWSServiceRoleForEMRCleanup*",
    "Condition": {
        "StringLike": {
            "iam:AWSServiceName": [
                "emrwal.amazonaws.com",
                "elasticmapreduce.amazonaws.com.cn"
            ]
        }
    }
}
```

Eine serviceverknüpfte Rolle für Amazon erstellen EMR

Sie müssen die AWSServiceRoleForEMRWAL Rolle nicht manuell erstellen. Amazon EMR erstellt diese serviceverknüpfte Rolle automatisch, wenn Sie einen WAL Workspace mit EMRWAL CLI oder von erstellen AWS CloudFormation, oder erstellt HBase die serviceverknüpfte Rolle, wenn Sie einen Workspace für Amazon konfigurieren EMR WAL und die serviceverknüpfte Rolle noch nicht existiert. Sie müssen über die erforderlichen Berechtigungen verfügen, um eine servicebezogene Rolle zu

erstellen. Anweisungen, mit denen diese Funktion zur Berechtigungsrichtlinie einer IAM Entität (z. B. eines Benutzers, einer Gruppe oder einer Rolle) hinzugefügt wird, finden Sie beispielsweise im vorherigen Abschnitt. Mit dem Dienst verknüpfte Rollenberechtigungen für Write-Ahead-Logging () WAL

Bearbeitung einer serviceverknüpften Rolle für Amazon EMR

Amazon erlaubt Ihnen EMR nicht, die AWSServiceRoleForEMRWAL serviceverknüpfte Rolle zu bearbeiten. Nachdem Sie eine serviceverknüpfte Rolle erstellt haben, können Sie den Namen der serviceverknüpften Rolle nicht mehr ändern, da verschiedene Entitäten möglicherweise auf die serviceverknüpfte Rolle verweisen. Sie können die Beschreibung der dienstbezogenen Rolle jedoch mithilfe von bearbeiten. IAM

Beschreibung einer dienstbezogenen Rolle bearbeiten (Konsole) IAM

Sie können die IAM Konsole verwenden, um die Beschreibung einer dienstbezogenen Rolle zu bearbeiten.

So bearbeiten Sie die Beschreibung einer serviceverknüpften Rolle (Konsole)

1. Wählen Sie im Navigationsbereich der IAM Konsole die Option Rollen aus.
2. Wählen Sie den Namen der zu ändernden Rolle.
3. Wählen Sie neben Rollenbeschreibung rechts Bearbeiten aus.
4. Geben Sie eine neue Beschreibung im Dialogfeld ein und wählen Sie Save changes (Änderungen speichern).

Bearbeiten einer mit einem Dienst verknüpften Rollenbeschreibung () IAM CLI

Sie können IAM Befehle von verwenden AWS Command Line Interface , um die Beschreibung einer dienstbezogenen Rolle zu bearbeiten.

Um die Beschreibung einer dienstbezogenen Rolle zu ändern () CLI

1. (Optional) Um die aktuelle Beschreibung einer Rolle anzuzeigen, verwenden Sie die folgenden Befehle:

```
$ aws iam get-role --role-name role-name
```

Verwenden Sie den Rollennamen, nicht denARN, um mit den CLI Befehlen auf Rollen zu verweisen. Wenn eine Rolle beispielsweise Folgendes hatARN:`arn:aws:iam::123456789012:role/myrole`, bezeichnen Sie die Rolle als**`myrole`**.

2. Um die Beschreibung einer serviceverknüpften Rolle zu aktualisieren, verwenden Sie einen der folgenden Befehle:

```
$ aws iam update-role-description --role-name role-name --description description
```

Bearbeiten einer mit einem Dienst verknüpften Rollenbeschreibung () IAM API

Sie können den verwenden IAMAPI, um die Beschreibung einer dienstbezogenen Rolle zu bearbeiten.

Um die Beschreibung einer dienstbezogenen Rolle zu ändern () API

1. (Optional) Um die aktuelle Beschreibung einer Rolle anzuzeigen, verwenden Sie den folgenden Befehl:

   IAM API: GetRole
2. Um die Beschreibung einer Rolle zu aktualisieren, verwenden Sie den folgenden Befehl:

   IAM API: UpdateRoleDescription

Löschen einer serviceverknüpften Rolle für Amazon EMR

Wenn Sie eine Funktion oder einen Dienst nicht mehr verwenden müssen, für den eine dienstbezogene Rolle erforderlich ist, empfehlen wir Ihnen, diese dienstbezogene Rolle zu löschen. Auf diese Weise haben Sie keine ungenutzte Entität, die nicht aktiv überwacht oder verwaltet wird. Sie müssen jedoch Ihre serviceverknüpfte Rolle zunächst bereinigen, bevor Sie sie löschen können.

> **Note**
>
> Der Write-Ahead-Protokollierungsvorgang ist nicht betroffen, wenn Sie die AWSServiceRoleForEMRWAL Rolle löschen, aber Amazon löscht die erstellten Protokolle EMR nicht automatisch, sobald Ihr EMR Cluster beendet wird. Daher müssen Sie die EMR WAL Amazon-Protokolle manuell löschen, wenn Sie die serviceverknüpfte Rolle löschen.

Bereinigen einer serviceverknüpften Rolle

Bevor Sie eine serviceverknüpfte Rolle löschen können, müssen Sie zunächst bestätigen, dass die Rolle keine aktiven Sitzungen hat, und alle von der Rolle verwendeten Ressourcen entfernen. IAM

Um zu überprüfen, ob die mit dem Dienst verknüpfte Rolle über eine aktive Sitzung in der Konsole verfügt IAM

1. Öffnen Sie die IAM Konsole unter. https://console.aws.amazon.com/iam/
2. Wählen Sie im Navigationsbereich Rollen aus. Wählen Sie den Namen (nicht das Kontrollkästchen) der AWSServiceRoleForEMRWAL Rolle aus.
3. Wählen Sie auf der Seite Summary (Übersicht) für die ausgewählte Rolle die Option Access Advisor (Advisor aufrufen) aus.
4. Überprüfen Sie auf der Registerkarte Access Advisor (Advisor aufrufen) die jüngsten Aktivitäten für die serviceverknüpfte Rolle.

> **Note**
>
> Wenn Sie sich nicht sicher sind, ob Amazon EMR die AWSServiceRoleForEMRWAL Rolle verwendet, können Sie versuchen, die serviceverknüpfte Rolle zu löschen. Wenn der Service die Rolle verwendet, schlägt das Löschen fehl und Sie können die Regionen anzeigen, in denen die mit dem Service verknüpfte Rolle verwendet wird. Wenn die dienstverknüpfte Rolle verwendet wird, müssen Sie warten, bis die Sitzung beendet ist, bevor Sie die dienstverknüpfte Rolle löschen können. Die Sitzung für eine serviceverknüpfte Rolle können Sie nicht widerrufen.

Um EMR Amazon-Ressourcen zu entfernen, die verwendet werden von AWSServiceRoleForEMRWAL

- Beenden Sie alle Cluster in Ihrem Konto. Weitere Informationen finden Sie unter Einen Cluster beenden.

Löschen einer serviceverknüpften Rolle (IAMKonsole)

Sie können die IAM Konsole verwenden, um eine dienstverknüpfte Rolle zu löschen.

So löschen Sie eine serviceverknüpfte Rolle (Konsole)

1. Öffnen Sie die IAM Konsole unter. https://console.aws.amazon.com/iam/
2. Wählen Sie im Navigationsbereich Rollen aus. Aktivieren Sie das Kontrollkästchen neben AWSServiceRoleForEMRWAL, nicht den Namen oder die Zeile selbst.
3. Wählen Sie für Role actions oben auf der Seite Delete role aus.
4. Überprüfen Sie im Bestätigungsdialogfeld die Daten, auf die der Dienst zuletzt zugegriffen hat. Aus diesen Daten geht hervor, wann jede der ausgewählten Rollen zuletzt auf einen AWS Dienst zugegriffen hat. Auf diese Weise können Sie leichter bestätigen, ob die Rolle derzeit aktiv ist. Wählen Sie Yes, Delete, um fortzufahren.
5. Sehen Sie sich die IAM Konsolenbenachrichtigungen an, um den Fortschritt beim Löschen der dienstbezogenen Rolle zu verfolgen. Da das Löschen der IAM dienstbezogenen Rolle asynchron erfolgt, kann die Löschaufgabe erfolgreich sein oder fehlschlagen, nachdem Sie die Rolle zum Löschen eingereicht haben. Wenn der Vorgang fehlschlägt, können Sie in den Benachrichtigungen View details oder View Resources auswählen, um zu erfahren, warum die Löschung fehlgeschlagen ist. Wenn das Löschen fehlschlägt, weil der Service Ressourcen enthält, die von der Rolle verwendet werden, enthält die Angabe des Fehlergrundes eine Liste der Ressourcen.

Löschen einer dienstverknüpften Rolle () IAM CLI

Sie können IAM Befehle von verwenden, AWS Command Line Interface um eine dienstverknüpfte Rolle zu löschen. Da eine serviceverknüpfte Rolle nicht gelöscht werden kann, wenn sie verwendet wird oder ihr Ressourcen zugeordnet sind, müssen Sie eine Löschungsanforderung übermitteln. Wenn diese Bedingungen nicht erfüllt sind, kann diese Anforderung verweigert werden.

Um eine dienstverknüpfte Rolle zu löschen () CLI

1. Sie benötigen die `deletion-task-id` aus der Antwort, um den Status der Löschaufgabe zu überprüfen. Geben Sie den folgenden Befehl ein, um eine Löschanforderung für eine serviceverknüpfte Rolle zu übermitteln:

   ```
   $ aws iam delete-service-linked-role --role-name AWSServiceRoleForEMRWAL
   ```

2. Geben Sie den folgenden Befehl ein, um den Status der Löschaufgabe zu überprüfen:

```
$ aws iam get-service-linked-role-deletion-status --deletion-task-id deletion-task-id
```

Der Status der Löschaufgabe kann `NOT_STARTED`, `IN_PROGRESS`, `SUCCEEDED` oder `FAILED` lauten. Wenn die Löschung fehlschlägt, gibt der Aufruf den Grund zurück, sodass Sie das Problem beheben können.

Löschen einer dienstverknüpften Rolle () IAM API

Sie können die verwenden IAMAPI, um eine dienstverknüpfte Rolle zu löschen. Da eine serviceverknüpfte Rolle nicht gelöscht werden kann, wenn sie verwendet wird oder ihr Ressourcen zugeordnet sind, müssen Sie eine Löschungsanforderung übermitteln. Wenn diese Bedingungen nicht erfüllt sind, kann diese Anforderung verweigert werden.

Um eine dienstverknüpfte Rolle zu löschen () API

1. Rufen Sie an, um eine Löschanfrage für eine dienstverknüpfte Rolle einzureichen. DeleteServiceLinkedRole Geben Sie in der Anfrage den AWSServiceRoleForEMRWAL Rollennamen an.

   Sie benötigen die `DeletionTaskId` aus der Antwort, um den Status der Löschaufgabe zu überprüfen.
2. Rufen Sie an, um den Status des Löschvorgangs zu überprüfen GetServiceLinkedRoleDeletionStatus. Geben Sie in der Anforderung die `DeletionTaskId` an.

   Der Status der Löschaufgabe kann `NOT_STARTED`, `IN_PROGRESS`, `SUCCEEDED` oder `FAILED` lauten. Wenn die Löschung fehlschlägt, gibt der Aufruf den Grund zurück, sodass Sie das Problem beheben können.

Unterstützte Regionen für AWSServiceRoleForEMRWAL

Amazon EMR unterstützt die Nutzung der AWSServiceRoleForEMRWAL serviceverknüpften Rolle in den folgenden Regionen.

Name der Region	Regions-ID	Support bei Amazon EMR
USA Ost (Nord-Virginia)	us-east-1	Ja

Name der Region	Regions-ID	Support bei Amazon EMR
USA Ost (Ohio)	us-east-2	Ja
USA West (Nordkalifornien)	us-west-1	Ja
USA West (Oregon)	us-west-2	Ja
Asien-Pazifik (Mumbai)	ap-south-1	Ja
Asien-Pazifik (Singapore)	ap-southeast-1	Ja
Asien-Pazifik (Sydney)	ap-southeast-2	Ja
Asien-Pazifik (Tokyo)	ap-northeast-1	Ja
Europa (Frankfurt)	eu-central-1	Ja
Europa (Ireland)	eu-west-1	Ja

Passen Sie IAM Rollen an

Möglicherweise möchten Sie die IAM Servicerollen und -berechtigungen anpassen, um die Rechte Ihren Sicherheitsanforderungen entsprechend einzuschränken. Zum Anpassen von Berechtigungen empfehlen wir, dass Sie neue Rollen und Richtlinien erstellen. Beginnen Sie mit den Berechtigungen in den verwalteten Richtlinien für die Standardrollen (beispielsweise `AmazonElasticMapReduceforEC2Role` und `AmazonElasticMapReduceRole`). Kopieren Sie anschließend die Inhalte in die neuen Richtlinienanweisungen, modifizieren Sie die Berechtigungen entsprechend und fügen Sie die geänderten Richtlinien zu den von Ihnen erstellten Rollen hinzu. Sie müssen über die entsprechenden IAM Berechtigungen verfügen, um mit Rollen und Richtlinien arbeiten zu können. Weitere Informationen finden Sie unter Benutzern und Gruppen gestatten, Rollen zu erstellen und zu ändern.

Wenn Sie eine benutzerdefinierte EMR Rolle für erstellenEC2, folgen Sie dem grundlegenden Arbeitsablauf, der automatisch ein Instanzprofil mit demselben Namen erstellt. Amazon EC2 ermöglicht es Ihnen, Instance-Profile und Rollen mit unterschiedlichen Namen zu erstellen, Amazon unterstützt diese Konfiguration jedoch EMR nicht und führt bei der Erstellung des Clusters zu einem Fehler „Ungültiges Instanzprofil“.

> **Important**
>
> Eingebundene Richtlinien werden nicht automatisch aktualisiert, wenn sich Serviceanforderungen ändern. Beachten Sie beim Erstellen und Anhängen von Inline-Richtlinien, dass es zu Serviceaktualisierungen kommen kann, die plötzlich zu Berechtigungsfehlern führen. Weitere Informationen finden Sie unter Verwaltete Richtlinien und Inline-Richtlinien im IAMBenutzerhandbuch undGeben Sie beim Erstellen eines Clusters benutzerdefinierte IAM Rollen an.

Weitere Informationen zum Arbeiten mit IAM Rollen finden Sie in den folgenden Themen im IAMBenutzerhandbuch:

- Eine Rolle erstellen, um Berechtigungen an einen AWS Dienst zu delegieren
- Ändern einer Rolle
- Löschen einer Rolle

Geben Sie beim Erstellen eines Clusters benutzerdefinierte IAM Rollen an

Sie geben die Servicerolle für Amazon EMR und die Rolle für das EC2 Amazon-Instance-Profil an, wenn Sie einen Cluster erstellen. Der Benutzer, der Cluster erstellt, benötigt Berechtigungen zum Abrufen und Zuweisen von Rollen zu Amazon EMR und EC2 Instances. Andernfalls tritt der EC2 Fehler „Ein Konto ist nicht zum Anrufen autorisiert“ auf. Weitere Informationen finden Sie unter Benutzern und Gruppen gestatten, Rollen zu erstellen und zu ändern.

Mit der Konsole benutzerdefinierte Rollen angeben

Wenn Sie einen Cluster erstellen, können Sie mithilfe der erweiterten Optionen eine benutzerdefinierte Servicerolle für AmazonEMR, eine benutzerdefinierte Rolle für das EC2 Instance-Profil und eine benutzerdefinierte Auto Scaling Scaling-Rolle angeben. Wenn Sie Quick Options verwenden, werden die Standard-Servicerolle und die Standardrolle für das EC2 Instance-Profil angegeben. Weitere Informationen finden Sie unter IAMVon Amazon verwendete Servicerollen EMR.

Console

Um benutzerdefinierte IAM Rollen mit der Konsole anzugeben

Wenn Sie mit der Konsole einen Cluster erstellen, müssen Sie eine benutzerdefinierte Servicerolle für Amazon EMR und eine benutzerdefinierte Rolle für das EC2 Instance-Profil angeben. Weitere Informationen finden Sie unter IAMVon Amazon verwendete Servicerollen EMR.

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und anschließend Create cluster aus.
3. Suchen Sie unter Sicherheitskonfiguration und Berechtigungen die Felder IAMRolle für Instance-Profil und Servicerolle für EMR Amazon-Felder. Wählen Sie für jeden Rollentyp eine Rolle aus der Liste aus. Nur Rollen innerhalb Ihres Kontos, die die entsprechende Vertrauensstellungen für diesen Rollentyp besitzen, sind aufgeführt.
4. Wählen Sie alle anderen Optionen aus, die für Ihren Cluster gelten.
5. Um Ihren Cluster jetzt zu starten, wählen Sie Cluster erstellen aus.

Verwenden Sie die AWS CLI , um benutzerdefinierte Rollen anzugeben

Sie können eine Servicerolle für Amazon EMR und eine Servicerolle für EC2 Cluster-Instances explizit mithilfe von Optionen mit dem `create-cluster` Befehl von angeben AWS CLI. Verwenden Sie die Option `--service-role`, um die Servicerolle anzugeben. Verwenden Sie das `InstanceProfile` Argument der `--ec2-attributes` Option, um die Rolle für das EC2 Instance-Profil anzugeben.

Die Auto Scaling-Rolle wird einer separaten Option angegeben, `--auto-scaling-role`. Weitere Informationen finden Sie unter Verwenden der automatischen Skalierung mit einer benutzerdefinierten Richtlinie für Instance-Gruppen.

Um benutzerdefinierte IAM Rollen mit dem zu spezifizieren AWS CLI

- Der folgende Befehl spezifiziert die benutzerdefinierte Servicerolle: *`MyCustomServiceRoleForEMR`*, und eine benutzerdefinierte Rolle für das EC2 Instanzprofil, *`MyCustomServiceRoleForClusterEC2Instances`*, beim Starten eines Clusters. In diesem Beispiel wird die EMR Standardrolle von Amazon verwendet.

> **Note**
>
> Linux-Zeilenfortsetzungszeichen (\) sind aus Gründen der Lesbarkeit enthalten. Sie können entfernt oder in Linux-Befehlen verwendet werden. Entfernen Sie sie unter Windows oder ersetzen Sie sie durch ein Caret-Zeichen (^).

```
aws emr create-cluster --name "Test cluster" --release-label emr-7.2.0 \
--applications Name=Hive Name=Pig --service-role MyCustomServiceRoleForEMR \
--ec2-attributes InstanceProfile=MyCustomServiceRoleForClusterEC2Instances,\
KeyName=myKey --instance-type m5.xlarge --instance-count 3
```

Sie können diese Optionen verwenden, um Standardrollen explizit anzugeben, statt die Option `--use-default-roles` zu verwenden. Die `--use-default-roles` Option gibt die Servicerolle und die Rolle für das EC2 Instance-Profil an, das in der `config` Datei für definiert ist AWS CLI.

Das folgende Beispiel zeigt den Inhalt einer `config` Datei für AWS CLI die angegebenen benutzerdefinierten Rollen für AmazonEMR. Mit dieser Konfigurationsdatei wird, wenn die `--use-default-roles` Option angegeben ist, der Cluster mit dem erstellt *MyCustomServiceRoleForEMR* and *MyCustomServiceRoleForClusterEC2Instances*. Standardmäßig gibt die `config` Datei den Standard `service_role` als `AmazonElasticMapReduceRole` und den Standard `instance_profile` als anEMR_EC2_DefaultRole.

```
[default]
output = json
region = us-west-1
aws_access_key_id = myAccessKeyID
aws_secret_access_key = mySecretAccessKey
emr =
     service_role = MyCustomServiceRoleForEMR
     instance_profile = MyCustomServiceRoleForClusterEC2Instances
```

IAMRollen für EMRFS Anfragen an Amazon S3 konfigurieren

> **Note**
>
> Die auf dieser Seite beschriebene Funktion zur EMRFS Rollenzuweisung wurde mit der Einführung von Amazon S3 Access Grants in Amazon EMR 6.15.0 verbessert. Für eine skalierbare Zugriffskontrolllösung für Ihre Daten in Amazon S3 empfehlen wir, S3 Access Grants mit Amazon zu verwendenEMR.

Wenn eine Anwendung, die auf einem Cluster ausgeführt wird, Daten im `s3://mydata` Format referenziert, EMR verwendet Amazon, EMRFS um die Anfrage zu stellen. Für die Interaktion mit Amazon S3 werden die Berechtigungsrichtlinien EMRFS vorausgesetzt, die mit Ihrem EC2Amazon-Instance-Profil verknüpft sind. Es wird dasselbe EC2 Amazon-Instance-Profil verwendet, unabhängig davon, welcher Benutzer oder welche Gruppe die Anwendung ausführt, oder vom Speicherort der Daten in Amazon S3.

Wenn Sie einen Cluster mit mehreren Benutzern haben, die unterschiedliche Zugriffsebenen auf Daten in Amazon S3 benötigenEMRFS, können Sie eine Sicherheitskonfiguration mit IAM Rollen für einrichtenEMRFS. EMRFSkann je nach Benutzer oder Gruppe, die die Anfrage stellt, oder basierend auf dem Speicherort der Daten in Amazon S3 eine andere Servicerolle für EC2 Cluster-Instances annehmen. Jede IAM Rolle für EMRFS kann unterschiedliche Berechtigungen für den Datenzugriff in Amazon S3 haben. Weitere Informationen zur Servicerolle für EC2 Cluster-Instances finden Sie unterServicerolle für EC2 Cluster-Instances (EC2Instance-Profil).

Die Verwendung von benutzerdefinierten IAM Rollen für EMRFS wird in EMR Amazon-Versionen 5.10.0 und höher unterstützt. Wenn Sie eine frühere Version verwenden oder Anforderungen haben, die über die EMRFS Bereitstellung von IAM Rollen hinausgehen, können Sie stattdessen einen Anbieter für benutzerdefinierte Anmeldeinformationen erstellen. Weitere Informationen finden Sie unter Autorisieren des Zugriffs auf EMRFS Daten in Amazon S3.

Wenn Sie eine Sicherheitskonfiguration verwenden, um IAM Rollen für anzugebenEMRFS, richten Sie Rollenzuordnungen ein. Jede Rollenzuordnung spezifiziert eine IAM Rolle, die Identifikatoren entspricht. Diese Kennungen bestimmen die Grundlage für den Zugriff auf Amazon S3 überEMRFS. Die Kennungen können Benutzer, Gruppen oder Amazon-S3-Präfixe sein, die einen Datenspeicherort angeben. Wenn eine Anfrage an Amazon S3 gestellt wird und die Anfrage mit der Grundlage für den Zugriff übereinstimmt, EMRFS EMRFS lässt EC2 Cluster-Instances die entsprechende IAM Rolle für

die Anfrage übernehmen. Die mit dieser Rolle verknüpften IAM Berechtigungen gelten anstelle der IAM Berechtigungen, die der Servicerolle für EC2 Cluster-Instances zugewiesen sind.

Die Benutzer und Gruppen in einer Rollenzuordnung sind Hadoop-Benutzer und -gruppen, die auf dem Cluster definiert sind. Benutzer und Gruppen werden EMRFS im Kontext der Anwendung, die sie verwendet, weitergegeben (z. B. YARN durch Identitätswechsel). Das Amazon-S3-Präfix kann ein Bucket-Spezifizierer beliebiger Tiefe sein (z. B. `s3://mybucket` oder `s3://mybucket/myproject/mydata`). Sie können mehrere Kennungen in einer einzigen Rollenzuordnung angeben, die jedoch alle vom selben Typ sein müssen.

> **Important**
>
> IAMRollen EMRFS sorgen für die Isolierung der Benutzer der Anwendung auf Anwendungsebene. Sie bieten keine Isolierung auf Host-Ebene zwischen Benutzern auf dem Host. Jeder Benutzer mit Zugriff auf das Cluster kann die Isolation umgehen, um eine Rolle zu übernehmen.

Wenn eine Cluster-Anwendung eine Anfrage an Amazon S3 stelltEMRFS, EMRFS bewertet sie die Rollenzuordnungen in der Reihenfolge von oben nach unten, in der sie in der Sicherheitskonfiguration erscheinen. Wenn eine Anfrage, die über EMRFS gestellt wurde, keiner Kennung entspricht, EMRFS wird auf die Verwendung der Servicerolle für Cluster-Instances zurückgegriffen. EC2 Aus diesem Grund empfehlen wir, dass Sie die Richtlinien, die dieser Rolle zugeordnet werden, auf Berechtigungen in Amazon S3 begrenzen. Weitere Informationen finden Sie unter Servicerolle für EC2 Cluster-Instances (EC2Instance-Profil).

Konfigurieren von -Rollen

Bevor Sie eine Sicherheitskonfiguration mit IAM Rollen für einrichtenEMRFS, planen und erstellen Sie die Rollen und Berechtigungsrichtlinien, die den Rollen zugewiesen werden sollen. Weitere Informationen finden Sie unter Wie funktionieren Rollen für EC2 Instanzen? im IAMBenutzerhandbuch. Wir empfehlen Ihnen, bei der Erstellung von Berechtigungsrichtlinien mit der verwalteten Richtlinie zu beginnen, die der EMR Amazon-Standardrolle für zugeordnet istEC2, und diese Richtlinie dann Ihren Anforderungen entsprechend zu bearbeiten. Der standardmäßige Rollename ist `EMR_EC2_DefaultRole`, und die zu bearbeitende verwaltete Standardrichtlinie ist `AmazonElasticMapReduceforEC2Role`. Weitere Informationen finden Sie unter Servicerolle für EC2 Cluster-Instances (EC2Instance-Profil).

Aktualisieren von Vertrauensrichtlinien, um Rollenberechtigungen zu übernehmen

Jede Rolle, die EMRFS verwendet wird, muss über eine Vertrauensrichtlinie verfügen, die es der EMR Amazon-Rolle des Clusters ermöglicht, diese EC2 zu übernehmen. Ebenso EC2 muss die EMR Amazon-Rolle für des Clusters über eine Vertrauensrichtlinie verfügen, die es EMRFS Rollen ermöglicht, sie zu übernehmen.

Das folgende Beispiel für eine Vertrauensrichtlinie ist Rollen für zugeordnetEMRFS. Die Anweisung ermöglicht es der EMR Amazon-Standardrolle fürEC2, die Rolle zu übernehmen. Wenn Sie beispielsweise zwei fiktive EMRFS Rollen haben `EMRFSRole_First` und `EMRFSRole_Second` diese Richtlinienerklärung zu den Vertrauensrichtlinien für jede dieser Rollen hinzugefügt wird.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Principal":{
            "AWS":"arn:aws:iam::AWSAcctID:role/EMR_EC2_DefaultRole"
         },
         "Action":"sts:AssumeRole"
      }
   ]
}
```

Darüber hinaus wird das folgende Beispiel für eine Vertrauensrichtlinien-Erklärung `EMR_EC2_DefaultRole` zu der hinzugefügt, damit die beiden fiktiven EMRFS Rollen sie übernehmen können.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Principal":{
            "AWS": ["arn:aws:iam::AWSAcctID:role/EMRFSRole_First",
 "arn:aws:iam::AWSAcctID:role/EMRFSRole_Second"]
         },
         "Action":"sts:AssumeRole"
      }
   ]
```

```
}
```

Um die Vertrauensrichtlinie einer Rolle zu aktualisieren IAM

Öffnen Sie die IAM Konsole unter https://console.aws.amazon.com/iam/.

1. Wählen Sie Roles (Rollen), geben Sie den Namen der Rolle unter Search (Suche) ein und wählen Sie dann Role name (Rollenname) aus.
2. Wählen Sie auf der Registerkarte Trust Relationships (Vertrauensbeziehungen) Edit Trust Relationship (Vertrauensbeziehung bearbeiten) aus.
3. Fügen Sie eine Vertrauensanweisung gemäß dem Richtliniendokument und den Richtlinien oben hinzu und wählen Sie dann Vertrauensrichtlinie updaten.

Angeben einer Rolle als Schlüsselbenutzer

Wenn eine Rolle den Zugriff auf einen Speicherort in Amazon S3 zulässt, der mit einem AWS KMS key verschlüsselt ist, muss die Rolle als Schlüsselbenutzer angegeben werden. Dadurch erhält die Rolle die Erlaubnis, den KMS Schlüssel zu verwenden. Weitere Informationen finden Sie unter Schlüsselrichtlinien in AWS KMS im Entwicklerhandbuch für AWS Key Management Service .

Richten Sie eine Sicherheitskonfiguration mit IAM Rollen für ein EMRFS

> **Important**
>
> Wenn keine der von Ihnen angegebenen IAM Rollen zutrifft, EMRFS greifen Sie auf die EMR Amazon-Rolle für zurückEC2. EMRFS Sie sollten die Berechtigungen dieser Rolle auf Amazon S3 einschränken wie für Ihre Anwendung erforderlich, und dann diese benutzerdefinierte Rolle anstelle von `EMR_EC2_DefaultRole` angeben, wenn Sie einen Cluster erstellen. Weitere Informationen erhalten Sie unter Passen Sie IAM Rollen an und Geben Sie beim Erstellen eines Clusters benutzerdefinierte IAM Rollen an.

So geben Sie IAM Rollen für EMRFS Anfragen an Amazon S3 mithilfe der Konsole an

1. Erstellen Sie eine Sicherheitskonfiguration, die Rollenzuordnungen spezifiziert:
   a. Wählen Sie in der EMR Amazon-Konsole Sicherheitskonfigurationen, Erstellen aus.

b. Geben Sie in Name (Name) einen Namen für die Sicherheitskonfiguration ein. Verwenden Sie diesen Namen zum Angeben der Sicherheitskonfiguration, wenn Sie einen Cluster erstellen.

c. Wählen Sie IAM Rollen für EMRFS Anfragen an Amazon S3 verwenden.

d. Wählen Sie eine IAMRolle aus, die Sie beantragen möchten, und wählen Sie unter Basis für den Zugriff einen Identifikationstyp (Benutzer, Gruppen oder S3-Präfixe) aus der Liste aus und geben Sie die entsprechenden Kennungen ein. Wenn Sie mehrere Kennungen verwenden, trennen Sie diese durch Komma (ohne Leerzeichen dazwischen) voneinander ab. Weitere Informationen zu den einzelnen ID-Typen finden Sie unten in der JSON configuration reference.

e. Wählen Sie Add role (Rolle hinzufügen) aus, um zusätzliche Rollenzuordnungen einzurichten wie im vorherigen Schritt beschrieben.

f. Richten Sie weitere Sicherheitskonfigurationsoptionen ein wie erforderlich. Wählen Sie Create (Erstellen) aus. Weitere Informationen finden Sie unter Eine Sicherheitskonfiguration erstellen.

2. Geben Sie die Sicherheitskonfiguration an, die Sie oben erstellt haben, wenn Sie einen Cluster erstellen. Weitere Informationen finden Sie unter Angabe einer Sicherheitskonfiguration für einen Cluster.

Um IAM Rollen für EMRFS Anfragen an Amazon S3 anzugeben, verwenden Sie den AWS CLI

1. Verwenden Sie den `aws emr create-security-configuration` Befehl und geben Sie einen Namen für die Sicherheitskonfiguration und die Sicherheitskonfigurationsdetails im JSON Format an.

   Der unten gezeigte Beispielbefehl erstellt eine Sicherheitskonfiguration namens `EMRFS_Roles_Security_Configuration`. Er basiert auf einer JSON Struktur in der Datei`MyEmrfsSecConfig.json`, die in demselben Verzeichnis gespeichert ist, in dem der Befehl ausgeführt wird.

   ```
   aws emr create-security-configuration --name EMRFS_Roles_Security_Configuration --
   security-configuration file://MyEmrFsSecConfig.json.
   ```

   Verwenden Sie die folgenden Richtlinien für die Struktur der Datei `MyEmrFsSecConfig.json`. Sie können diese Struktur zusammen mit Strukturen für andere

Sicherheitskonfigurationsoptionen angeben. Weitere Informationen finden Sie unter Eine Sicherheitskonfiguration erstellen.

Im Folgenden finden Sie einen JSON Beispielausschnitt für die Angabe benutzerdefinierter IAM Rollen EMRFS innerhalb einer Sicherheitskonfiguration. Es zeigt Rollenzuordnungen für die drei verschiedenen Identifier-Typen, gefolgt von einer Parameterreferenz.

```
{
  "AuthorizationConfiguration": {
    "EmrFsConfiguration": {
      "RoleMappings": [{
        "Role": "arn:aws:iam::123456789101:role/allow_EMRFS_access_for_user1",
        "IdentifierType": "User",
        "Identifiers": [ "user1" ]
      },{
        "Role": "arn:aws:iam::123456789101:role/allow_EMRFS_access_to_MyBuckets",
        "IdentifierType": "Prefix",
        "Identifiers": [ "s3://MyBucket/","s3://MyOtherBucket/" ]
      },{
        "Role": "arn:aws:iam::123456789101:role/allow_EMRFS_access_for_AdminGroup",
        "IdentifierType": "Group",
        "Identifiers": [ "AdminGroup" ]
      }]
    }
  }
}
```

Parameter	Beschreibung
`"AuthorizationConfiguration":`	Erforderlich
`"EmrFsConfiguration":`	Erforderlich Enthält Rollenzuordnungen.

Parameter	Beschreibung
`"RoleMappings":`	Erforderlich Enthält eine oder mehrere Rollenzuordnungsdefinitionen. Rollenzuordnungen werden in der Reihenfolge bewertet, in der sie von oben nach unten angezeigt werden. Wenn eine Rollenzuweisung für einen EMRFS Datenaufruf in Amazon S3 als wahr bewertet wird, werden keine weiteren Rollenzuordnungen ausgewertet und die angegebene IAM Rolle wird für die Anfrage EMRFS verwendet . Rollenzuordnungen bestehen aus den folgenden erforderlichen Parametern:
`"Role":`	Gibt den ARN Bezeichner einer IAM Rolle im Format an. `arn:aws:iam::` *`account-id`* `:role/`*`role-name`* Dies ist die IAM Rolle, die Amazon EMR übernimmt, wenn die EMRFS Anfrage an Amazon S3 mit einer der `Identifiers` angegebenen Anforderungen übereinstimmt.

Parameter	Beschreibung
`"IdentifierType":`	Kann einer der folgenden sein: • `"User"` gibt an, dass es sich bei den Kennungen um einen oder mehrere Hadoop-Benutzer handelt, bei denen es sich um Linux-Kontobenutzer oder Kerberos-Prinzipale handeln kann. Wenn die EMRFS Anfrage von dem oder den angegebenen Benutzern stammt, wird die IAM Rolle übernommen. • `"Prefix"` gibt an, dass der Identifier ein Amazon-S3-Speicherort ist. Die IAM Rolle wird für Anrufe an den Standort oder die Standorte mit den angegebenen Präfixen übernommen. Das Präfix `s3://mybucket/` entspricht beispielsweise `s3://mybucket/mydir` und `s3://mybucket/yetanotherdir` . • `"Group"` gibt an, dass es sich bei den Identifikatoren um eine oder mehrere Hadoop-Gruppen handelt. Die IAM Rolle wird übernommen, wenn die Anfrage von einem Benutzer in der oder den angegebenen Gruppen stammt.
`"Identifiers":`	Gibt einen oder mehrere Kennungen des entsprechenden Kennungstyps an. Trennen Sie mehrere Bezeichner durch Kommas ohne Leerzeichen.

2. Verwenden Sie den Befehl `aws emr create-cluster`, um einen Cluster einzurichten, und geben Sie die Sicherheitskonfiguration an, die Sie im vorherigen Schritt erstellt haben.

   Im folgenden Beispiel wird ein Cluster erstellt, bei dem Standard-Core-Hadoop-Anwendungen installiert sind. Der Cluster verwendet die oben erstellte Sicherheitskonfiguration als

`EMRFS_Roles_Security_Configuration` und verwendet auch eine benutzerdefinierte EMR Amazon-Rolle für EC2`EC2_Role_EMR_Restrict_S3`,, die mit dem `InstanceProfile` Argument des `--ec2-attributes` Parameters angegeben wird.

> **Note**
>
> Linux-Zeilenfortsetzungszeichen (\) sind aus Gründen der Lesbarkeit enthalten. Sie können entfernt oder in Linux-Befehlen verwendet werden. Entfernen Sie sie unter Windows oder ersetzen Sie sie durch ein Caret-Zeichen (^).

```
aws emr create-cluster --name MyEmrFsS3RolesCluster \
--release-label emr-7.2.0 --ec2-attributes
 InstanceProfile=EC2_Role_EMR_Restrict_S3,KeyName=MyKey \
--instance-type m5.xlarge --instance-count 3 \
--security-configuration EMRFS_Roles_Security_Configuration
```

Verwenden Sie ressourcenbasierte Richtlinien für den EMR Zugriff von Amazon auf AWS Glue Data Catalog

Wenn Sie AWS Glue in Verbindung mit Hive, Spark oder Presto in Amazon EMR verwenden, unterstützt AWS Glue ressourcenbasierte Richtlinien zur Steuerung des Zugriffs auf Datenkatalogressourcen. Zu diesen Ressourcen gehören Datenbanken, Tabellen, Verbindungen und benutzerdefinierte Funktionen. Weitere Informationen finden Sie unter Verwenden von ressourcenbasierten Richtlinien für AWS Glue im AWS -Glue-Entwicklerhandbuch.

Wenn Sie ressourcenbasierte Richtlinien verwenden, um den Zugriff auf AWS Glue von Amazon aus zu beschränkenEMR, muss der Principal, den Sie in der Berechtigungsrichtlinie angeben, die Rolle sein, die dem EC2 Instance-Profil ARN zugeordnet ist, das bei der Erstellung eines Clusters angegeben wird. Beispielsweise können Sie für eine ressourcenbasierte Richtlinie, die an einen Katalog angehängt ist, die Rolle ARN für die Standard-Servicerolle für Cluster-Instances angeben. EC2 *`EMR_EC2_DefaultRole`* als der`Principal`, wobei das im folgenden Beispiel gezeigte Format verwendet wird:

```
arn:aws:iam::acct-id:role/EMR_EC2_DefaultRole
```

Das Tool *acct-id* kann sich von der AWS Glue-Konto-ID unterscheiden. Dies ermöglicht den Zugriff von EMR Clustern in verschiedenen Konten aus. Sie können mehrere Principals angeben, von denen jeder aus einem anderen Konto stammt.

Verwenden Sie IAM Rollen mit Anwendungen, die AWS Dienste direkt aufrufen

Anwendungen, die auf den EC2 Instanzen eines Clusters ausgeführt werden, können das EC2 Instanzprofil verwenden, um beim Aufrufen von AWS Diensten temporäre Sicherheitsanmeldeinformationen abzurufen.

Die Versionen von Hadoop, die mit EMR Amazon-Version 2.3.0 und höher verfügbar sind, wurden bereits aktualisiert, um Rollen zu verwenden. IAM Wenn Ihre Anwendung ausschließlich auf der Hadoop-Architektur läuft und keinen Service direkt aufruft AWS, sollte sie mit IAM Rollen ohne Änderung funktionieren.

Wenn Ihre Anwendung Dienste AWS direkt aufruft, müssen Sie sie aktualisieren, um IAM Rollen nutzen zu können. Das bedeutet, dass Ihre Anwendung, anstatt die Kontoanmeldeinformationen von `/etc/hadoop/conf/core-site.xml` den EC2 Instances im Cluster abzurufen, eine verwendet, SDK um mithilfe von IAM Rollen auf die Ressourcen zuzugreifen, oder die EC2 Instanz-Metadaten aufruft, um die temporären Anmeldeinformationen abzurufen.

Für den Zugriff auf AWS Ressourcen mit IAM Rollen verwenden Sie ein SDK

- In den folgenden Themen wird gezeigt, wie Sie mehrere von verwenden AWS SDKs, um mithilfe von IAM Rollen auf temporäre Anmeldeinformationen zuzugreifen. Jedes Thema beginnt mit einer Version einer Anwendung, die keine IAM Rollen verwendet, und führt Sie anschließend durch den Prozess der Konvertierung dieser Anwendung zur Verwendung von IAM Rollen.
  - Verwenden von IAM Rollen für EC2 Amazon-Instances mit dem SDK für Java im AWS SDK for Java Developer Guide
  - Verwenden von IAM Rollen für EC2 Amazon-Instances mit dem SDK for. NETim AWS SDK for .NET Entwicklerhandbuch
  - Verwenden von IAM Rollen für EC2 Amazon-Instances mit dem SDK for PHP im AWS SDK for PHP Developer Guide
  - Verwenden von IAM Rollen für EC2 Amazon-Instances mit dem SDK for Ruby im AWS SDK for Ruby Developer Guide

Um temporäre Anmeldeinformationen aus den EC2 Instance-Metadaten abzurufen

- Rufen Sie den folgenden URL Befehl von einer EC2 Instance aus auf, die mit der angegebenen IAM Rolle ausgeführt wird. Dabei werden die zugehörigen temporären Sicherheitsanmeldedaten (AccessKeyId, SecretAccessKey SessionToken, und Expiration) zurückgegeben. Das folgende Beispiel verwendet das Standard-Instance-Profil für AmazonEMR,`EMR_EC2_DefaultRole`.

```
GET http://169.254.169.254/latest/meta-data/iam/security-
credentials/EMR_EC2_DefaultRole
```

Weitere Informationen zum Schreiben von Anwendungen, die IAM Rollen verwenden, finden Sie unter Anwendungen, die auf EC2 Amazon-Instances ausgeführt werden, Zugriff auf AWS Ressourcen gewähren.

Weitere Informationen zu temporären Sicherheitsanmeldeinformationen finden Sie unter Verwenden temporärer Sicherheitsanmeldeinformationen im Handbuch Verwenden temporärer Sicherheitsanmeldeinformationen.

Benutzern und Gruppen gestatten, Rollen zu erstellen und zu ändern

IAMPrinzipale (Benutzer und Gruppen), die Rollen für einen Cluster erstellen, ändern und spezifizieren, einschließlich Standardrollen, müssen berechtigt sein, die folgenden Aktionen auszuführen. Einzelheiten zu den einzelnen Aktionen finden Sie unter Aktionen in der IAMAPIReferenz.

- `iam:CreateRole`
- `iam:PutRolePolicy`
- `iam:CreateInstanceProfile`
- `iam:AddRoleToInstanceProfile`
- `iam:ListRoles`
- `iam:GetPolicy`
- `iam:GetInstanceProfile`
- `iam:GetPolicyVersion`
- `iam:AttachRolePolicy`

- `iam:PassRole`

Die Berechtigung `iam:PassRole` gewährt die Erstellung von Clustern. Die restlichen Berechtigungen gewähren die Erstellung von Standardrollen.

Informationen zum Zuweisen von Berechtigungen zu einem Benutzer finden Sie im Benutzerhandbuch unter Ändern von Berechtigungen für einen IAM Benutzer.

Beispiele für EMR identitätsbasierte Richtlinien von Amazon

Standardmäßig sind Benutzer und Rollen nicht berechtigt, EMR Amazon-Ressourcen zu erstellen oder zu ändern. Sie können auch keine Aufgaben mit dem AWS Management Console AWS CLI, oder ausführen AWS API. Ein IAM Administrator muss IAM Richtlinien erstellen, die Benutzern und Rollen die Berechtigung gewähren, bestimmte API Operationen mit den angegebenen Ressourcen auszuführen, die sie benötigen. Der Administrator muss diese Richtlinien anschließend den Benutzern oder -Gruppen anfügen, die diese Berechtigungen benötigen.

Informationen zum Erstellen einer IAM identitätsbasierten Richtlinie anhand dieser JSON Beispieldokumente finden Sie unter Richtlinien erstellen auf der JSON Registerkarte im IAMBenutzerhandbuch.

Themen

- Bewährte Richtlinien für Amazon EMR
- Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer
- Von Amazon EMR verwaltete Richtlinien
- IAMRichtlinien für den tagbasierten Zugriff auf Cluster und Notebooks EMR
- Die Aktion wird ModifyInstanceGroup verweigert
- Fehlerbehebung Amazon EMR Amazon-Identität und -Zugriff

Bewährte Richtlinien für Amazon EMR

Identitätsbasierte Richtlinien sind sehr leistungsfähig. Sie bestimmen, ob jemand EMR Amazon-Ressourcen in Ihrem Konto erstellen, darauf zugreifen oder diese löschen kann. Für diese Aktionen können Kosten für Ihr AWS Konto anfallen. Befolgen Sie beim Erstellen oder Bearbeiten identitätsbasierter Richtlinien die folgenden Anleitungen und Empfehlungen:

- Erste Schritte mit AWS verwalteten Richtlinien — Um EMR schnell mit der Nutzung von Amazon zu beginnen, geben Sie Ihren Mitarbeitern mithilfe AWS verwalteter Richtlinien die erforderlichen Berechtigungen. Diese Richtlinien sind bereits in Ihrem Konto verfügbar und werden von AWS. Weitere Informationen finden Sie unter Erste Schritte zur Nutzung von Berechtigungen mit AWS verwalteten Richtlinien im IAMBenutzerhandbuch undVon Amazon EMR verwaltete Richtlinien.
- Gewähren Sie die geringstmöglichen Berechtigungen – Gewähren Sie beim Erstellen benutzerdefinierter Richtlinien nur die Berechtigungen, die zum Ausführen einer Aufgabe erforderlich sind. Beginnen Sie mit einem Mindestsatz von Berechtigungen und gewähren Sie zusätzliche Berechtigungen wie erforderlich. Dies ist sicherer, als mit Berechtigungen zu beginnen, die zu weit gefasst sind, und dann später zu versuchen, sie zu begrenzen. Weitere Informationen finden Sie im IAMBenutzerhandbuch unter Gewährung der geringsten Zugriffsrechte.
- MFAFür vertrauliche Operationen aktivieren — Für zusätzliche Sicherheit müssen Benutzer für den Zugriff auf vertrauliche Ressourcen oder API Vorgänge die Multi-Faktor-Authentifizierung (MFA) verwenden. Weitere Informationen finden Sie im AWSIAMBenutzerhandbuch unter Verwenden der Multi-Faktor-Authentifizierung (MFA).
- Verwenden Sie Richtlinienbedingungen, um zusätzliche Sicherheit zu bieten – Definieren Sie die Bedingungen, unter denen Ihre identitätsbasierten Richtlinien den Zugriff auf eine Ressource zulassen, soweit praktikabel. Beispielsweise können Sie Bedingungen schreiben, die eine Reihe von zulässigen IP-Adressen festlegen, von denen eine Anforderung stammen muss. Sie können auch Bedingungen schreiben, um Anfragen nur innerhalb eines bestimmten Datums- oder Zeitbereichs zuzulassen oder die Verwendung von SSL oder MFA vorzuschreiben. Weitere Informationen finden Sie im IAMBenutzerhandbuch unter IAMJSONRichtlinienelemente: Bedingung.

Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer

In diesem Beispiel wird gezeigt, wie Sie eine Richtlinie erstellen, die Benutzern die Berechtigung zum Anzeigen der eingebundenen Richtlinien und verwalteten Richtlinien gewährt, die ihrer Benutzeridentität angefügt sind. Diese Richtlinie umfasst Berechtigungen zum Ausführen dieser Aktion auf der Konsole oder programmgesteuert mithilfe von oder. AWS CLI AWS API

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"ViewOwnUserInfo",
         "Effect":"Allow",
```

```
         "Action":[
            "iam:GetUser",
            "iam:GetUserPolicy",
            "iam:ListAttachedUserPolicies",
            "iam:ListGroupsForUser",
            "iam:ListUserPolicies"
         ],
         "Resource":[
            "arn:aws:iam::*:user/${aws:username}"
         ]
      },
      {
         "Sid":"NavigateInConsole",
         "Effect":"Allow",
         "Action":[
            "iam:GetGroupPolicy",
            "iam:GetPolicy",
            "iam:GetPolicyVersion",
            "iam:ListAttachedGroupPolicies",
            "iam:ListGroupPolicies",
            "iam:ListGroups",
            "iam:ListPolicies",
            "iam:ListPolicyVersions",
            "iam:ListUsers"
         ],
         "Resource":"*"
      }
   ]
}
```

Von Amazon EMR verwaltete Richtlinien

Der einfachste Weg, vollen Zugriff oder nur Lesezugriff auf erforderliche EMR Amazon-Aktionen zu gewähren, besteht darin, die IAM verwalteten Richtlinien für Amazon zu verwenden. EMR Verwaltete Richtlinien haben den Vorteil, automatisch aktualisiert zu werden, wenn sich die Berechtigungsanforderungen ändern. Wenn Sie eingebundene Richtlinien verwenden, können Service-Veränderungen auftreten, die zu Berechtigungsfehlern führen.

Amazon EMR wird bestehende verwaltete Richtlinien (v1-Richtlinien) zugunsten neuer verwalteter Richtlinien (v2-Richtlinien) ablehnen. Die neuen verwalteten Richtlinien wurden im Hinblick auf bewährte Verfahren heruntergestuft. AWS Sobald die bestehenden verwalteten Richtlinien der Version 1 veraltet sind, können Sie diese Richtlinien keinen neuen Rollen oder Benutzern mehr

zuordnen. IAM Bestehende Rollen und Benutzer, die veraltete Richtlinien verwenden, können diese weiterhin verwenden. Die verwalteten v2-Richtlinien schränken den Zugriff mithilfe von Tags ein. Sie lassen nur bestimmte EMR Amazon-Aktionen zu und benötigen Cluster-Ressourcen, die mit einem EMR -spezifischen Schlüssel gekennzeichnet sind. Wir empfehlen Ihnen, die Dokumentation sorgfältig zu lesen, bevor Sie die neuen v2-Richtlinien verwenden.

Die v1-Richtlinien werden in der Richtlinienliste in der Konsole mit einem Warnsymbol neben ihnen als veraltet markiert. IAM Die veralteten Richtlinien werden die folgenden Merkmale aufweisen:

- Sie werden unverändert für alle gegenwärtig angefügten Benutzer, Gruppen und Rollen funktionsfähig. Alles funktioniert normal.
- Sie können nicht neuen Benutzern, Gruppen oder Rollen angefügt werden. Wenn Sie eine der Richtlinien von einer gegenwärtigen Entität trennen, können Sie sie nicht wieder anfügen.
- Nachdem Sie eine v1-Richtlinie von allen aktuellen Entitäten getrennt haben, ist die Richtlinie nicht mehr sichtbar und kann nicht mehr verwendet werden.

In der folgenden Tabelle werden die Änderungen zwischen den aktuellen Richtlinien (v1) und v2-Richtlinien zusammengefasst.

Von Amazon EMR verwaltete Richtlinienänderungen

Richtlinientyp	Richtliniennamen	Zweck der Richtlinie	Änderungen der v2-Richtlinie
EMRStandard-Servicerolle und angehängte verwaltete Richtlinie	Rollenname: EMR_DefaultRole V1-Richtlinie (wird nicht mehr unterstützt): AmazonElasticMapReduceRole(EMRServicerolle) V2-Richtlinienname (mit eingeschränktem Geltungsbereich): **AmazonEMR**	Ermöglicht AmazonEMR, andere AWS Services in Ihrem Namen aufzurufen, wenn Ressourcen bereitgestellt und Service-Level-Aktionen ausgeführt werden. Diese Rolle ist für alle Cluster erforderlich.	Die Richtlinie fügt die neue Berechtigung hinzu. `"ec2:DescribeInstanceTypeOfferings"` Dieser API Vorgang gibt eine Liste von Instance-Typen zurück, die von einer Liste der angegebenen Availability Zones unterstützt werden.

Richtlinientyp	Richtliniennamen	Zweck der Richtlinie	Änderungen der v2-Richtlinie
	ServicePolicy_v2		
IAMverwaltete Richtlinie für vollen EMR Amazon-Zugriff nach angehängtem Benutzer, Rolle oder Gruppe	V2-Richtlinienname (mit Geltungsbereich): **AmazonEMRServicePolicy_v2**	Erlaubt Benutzern volle Berechtigungen für EMR Aktionen. Beinhaltet iam: PassRole -Berechtigungen für Ressourcen.	Die Richtlinie setzt voraus, dass Benutzer Benutzer-Tags zu Ressourcen hinzufügen müssen, bevor sie diese Richtlinie verwenden können. Siehe Taggen von Ressourcen zur Verwendung verwalteter Richtlinien. Für die PassRole Aktion iam: muss die PassedToService Bedingung iam: auf den angegebenen Dienst gesetzt sein. Der Zugriff auf AmazonEC2, Amazon S3 und andere Dienste ist standardmäßig nicht erlaubt. Siehe IAMVerwaltete Richtlinie für vollen Zugriff (verwaltete Standardrichtlinie v2).

Richtlinientyp	Richtliniennamen	Zweck der Richtlinie	Änderungen der v2-Richtlinie
IAMverwaltete Richtlinie für schreibgeschützten Zugriff durch zugeordnete Benutzer, Rollen oder Gruppen	V1-Richtlinie (wird veraltet): **AmazonElasticMapReduceReadOnlyAccess** V2-Richtlinienname (mit Geltungsbereich): **AmazonEMRReadOnlyAccessPolicy_v2**	Erlaubt Benutzern nur Leseberechtigungen für EMR Amazon-Aktionen.	Mit Berechtigungen können nur bestimmte schreibgeschützte ElasticMapReduce-Aktionen ausgeführt werden. Der Zugriff auf Amazon S3 ist standardmäßig nicht zulässig. Siehe IAMVerwaltete Richtlinie für schreibgeschützten Zugriff (v2 Verwaltete Standardrichtlinie).

Richtlinientyp	Richtliniennamen	Zweck der Richtlinie	Änderungen der v2-Richtlinie
EMRStandard-Servicerolle und angehängte verwaltete Richtlinie	Rollenname: EMR_DefaultRole V1-Richtlinie (wird nicht mehr unterstützt): AmazonElasticMapReduceRole(EMRServicerolle) V2-Richtlinienname (mit eingeschränktem Geltungsbereich): **AmazonEMRServicePolicy_v2**	Ermöglicht AmazonEMR, andere AWS Services in Ihrem Namen aufzurufen, wenn Ressourcen bereitgestellt und Service-Level-Aktionen ausgeführt werden. Diese Rolle ist für alle Cluster erforderlich.	Die v2-Servicerolle und die v2-Standardrichtlinie ersetzen die veraltete Rolle und Richtlinie. Die Richtlinie setzt voraus, dass Benutzer Benutzer-Tags zu Ressourcen hinzufügen müssen, bevor sie diese Richtlinie verwenden können. Siehe Taggen von Ressourcen zur Verwendung verwalteter Richtlinien. Siehe Servicerolle für Amazon EMR (EMRRolle).

Richtlinientyp	Richtliniennamen	Zweck der Richtlinie	Änderungen der v2-Richtlinie
Servicerolle für EC2 Cluster-Instances (EC2Instance-Profil)	Rollenname: EMR_EC2 _ DefaultRole Veralteter Richtlinienname: AmazonElasticMapReduceforEC2Role	Ermöglicht Anwendungen, die auf einem EMR Cluster ausgeführt werden, auf andere AWS Ressourcen wie Amazon S3 zuzugreifen. Wenn Sie beispielsweise Apache-Spark-Aufträge ausführen, die Daten von Amazon S3 verarbeiten, muss die Richtlinie den Zugriff auf solche Ressourcen zulassen.	Sowohl die Standardrolle als auch die Standardrichtlinie werden demnächst veraltet sein. Es gibt keinen Ersatz für eine verwaltete AWS Standardrolle oder -richtlinie. Sie müssen eine ressourcen- oder identitätsbasierte Richtlinie bereitstellen. Das bedeutet, dass Anwendungen, die auf einem EMR Cluster ausgeführt werden, standardmäßig keinen Zugriff auf Amazon S3 oder andere Ressourcen haben, es sei denn, Sie fügen diese manuell zur Richtlinie hinzu. Siehe Standardrolle und verwaltete Richtlinie.

Richtlinientyp	Richtliniennamen	Zweck der Richtlinie	Änderungen der v2-Richtlinie
Andere Richtlinien EC2 für Servicerollen	Aktuelle Richtlinienamen: AmazonElasticMapReduceforAutoScalingRole AmazonElasticMapReduceEditorsRole,, mazonEMRCleanup A-Richtlinie	Stellt Berechtigungen bereit, die Amazon EMR benötigt, um auf andere AWS Ressourcen zuzugreifen und Aktionen auszuführen, wenn Auto Scaling, Notebooks oder EC2 Ressourcen bereinigt werden.	Keine Änderungen für Version 2.

Sicherung von iam: PassRole

Die standardmäßigen verwalteten Richtlinien von Amazon mit EMR vollen Berechtigungen beinhalten `iam:PassRole` Sicherheitskonfigurationen, darunter die folgenden:

- `iam:PassRole`Berechtigungen nur für bestimmte EMR Amazon-Standardrollen.
- `iam:PassedToService`Bedingungen, die es Ihnen ermöglichen, die Richtlinie nur mit bestimmten AWS Diensten zu verwenden, z. B. `elasticmapreduce.amazonaws.com` und`ec2.amazonaws.com`.

Sie können die JSON Version der Richtlinien A mazonEMRFull AccessPolicy _v2 und A mazonEMRService Policy_v2 in der Konsole einsehen. IAM Wir empfehlen, dass Sie neue Cluster mit den verwalteten v2-Richtlinien erstellen.

Wenn Sie benutzerdefinierte Richtlinien erstellen müssen, empfehlen wir ihnen, mit verwalteten Richtlinien zu beginnen und diese entsprechend Ihren Anforderungen zu bearbeiten.

Informationen zum Anhängen von Richtlinien an Benutzer (Prinzipale) finden Sie unter Arbeiten mit verwalteten Richtlinien mithilfe von AWS Management Console im Benutzerhandbuch. IAM

Taggen von Ressourcen zur Verwendung verwalteter Richtlinien

A mazonEMRService Policy_v2 und A mazonEMRFull AccessPolicy _v2 hängen vom begrenzten Zugriff auf Ressourcen ab, die Amazon bereitstellt oder verwendet. EMR Der eingeschränkte

Umfang wird dadurch erreicht, dass der Zugriff nur auf die Ressourcen beschränkt wird, denen ein vordefiniertes Benutzer-Tag zugeordnet ist. Wenn Sie eine dieser beiden Richtlinien verwenden, müssen Sie bei der Bereitstellung des Clusters das vordefinierte Benutzer-Tag `for-use-with-amazon-emr-managed-policies = true` übergeben. Amazon EMR verbreitet dieses Tag dann automatisch. Darüber hinaus müssen Sie den im folgenden Abschnitt aufgelisteten Ressourcen ein Benutzer-Tag hinzufügen. Wenn Sie die EMR Amazon-Konsole verwenden, um Ihren Cluster zu starten, finden Sie weitere Informationen unterÜberlegungen zur Verwendung der EMR Amazon-Konsole zum Starten von Clustern mit verwalteten v2-Richtlinien.

Um verwaltete Richtlinien zu verwenden, übergeben Sie das Benutzer-Tag, `for-use-with-amazon-emr-managed-policies = true` wenn Sie einen Cluster mit der CLISDK, oder einer anderen Methode bereitstellen.

Wenn Sie das Tag übergeben, gibt Amazon das EMR Tag an das private SubnetzENI, die EC2 Instance und die von Amazon erstellten EBS Volumes weiter. Amazon EMR markiert auch automatisch Sicherheitsgruppen, die es erstellt. Wenn Sie jedoch möchten, dass Amazon EMR mit einer bestimmten Sicherheitsgruppe startet, müssen Sie sie taggen. Für Ressourcen, die nicht von Amazon erstellt wurdenEMR, müssen Sie diesen Ressourcen Tags hinzufügen. Sie müssen beispielsweise EC2 Amazon-Subnetze, EC2 Sicherheitsgruppen (falls nicht von Amazon erstelltEMR) und VPCs (wenn Amazon Sicherheitsgruppen erstellen EMR soll) taggen. Um Cluster mit verwalteten v2-Richtlinien zu startenVPCs, müssen Sie diese VPCs mit dem vordefinierten Benutzertag kennzeichnen. Siehe Überlegungen zur Verwendung der EMR Amazon-Konsole zum Starten von Clustern mit verwalteten v2-Richtlinien.

Weiterverbreitetes benutzerdefiniertes Tagging

Amazon EMR kennzeichnet Ressourcen, die es mit den EMR Amazon-Tags erstellt, die Sie bei der Erstellung eines Clusters angeben. Amazon EMR wendet Tags auf die Ressourcen an, die es während der Lebensdauer des Clusters erstellt.

Amazon EMR verbreitet Benutzer-Tags für die folgenden Ressourcen:

- Privates Subnetz ENI (elastische Netzwerkschnittstellen für Servicezugriff)
- EC2Instanzen
- EBSVolumen
- EC2Vorlage starten

Automatisch getaggte Sicherheitsgruppen

Amazon EMR EC2 kennzeichnet Sicherheitsgruppen, die es erstellt, mit dem Tag, das für verwaltete v2-Richtlinien für Amazon erforderlich ist EMR`for-use-with-amazon-emr-managed-policies`, unabhängig davon, welche Tags Sie im Befehl create cluster angeben. Für eine Sicherheitsgruppe, die vor der Einführung der verwalteten v2-Richtlinien erstellt wurde, kennzeichnet Amazon die Sicherheitsgruppe EMR nicht automatisch. Wenn Sie verwaltete v2-Richtlinien mit den Standardsicherheitsgruppen verwenden möchten, die bereits im Konto vorhanden sind, müssen Sie die Sicherheitsgruppen manuell mit `for-use-with-amazon-emr-managed-policies = true` taggen.

Manuell getaggte Clusterressourcen

Sie müssen einige Cluster-Ressourcen manuell taggen, damit auf sie mit EMR Amazon-Standardrollen zugegriffen werden kann.

- Sie müssen EC2 Sicherheitsgruppen und EC2 Subnetze manuell mit dem von Amazon EMR verwalteten Richtlinien-Tag kennzeichnen`for-use-with-amazon-emr-managed-policies`.
- Sie müssen a manuell kennzeichnen, VPC wenn Amazon Standardsicherheitsgruppen erstellen EMR soll. EMRwird versuchen, eine Sicherheitsgruppe mit dem spezifischen Tag zu erstellen, falls die Standardsicherheitsgruppe noch nicht existiert.

Amazon EMR kennzeichnet automatisch die folgenden Ressourcen:

- EMR-erstellte EC2 Sicherheitsgruppen

Sie müssen die folgenden Ressourcen manuell taggen:

- EC2Subnetz
- EC2Sicherheitsgruppen

Optional können Sie die folgenden Ressourcen manuell taggen:

- VPC- nur wenn Sie möchtenEMR, dass Amazon Sicherheitsgruppen erstellt

Überlegungen zur Verwendung der EMR Amazon-Konsole zum Starten von Clustern mit verwalteten v2-Richtlinien

Sie können Cluster mit verwalteten v2-Richtlinien mithilfe der EMR Amazon-Konsole bereitstellen. Im Folgenden finden Sie einige Überlegungen, wenn Sie die Konsole zum Starten von EMR Amazon-Clustern verwenden.

- Sie müssen das vordefinierte Tag nicht übergeben. Amazon fügt das Tag EMR automatisch hinzu und leitet es an die entsprechenden Komponenten weiter.
- Bei Komponenten, die manuell markiert werden müssen, versucht die alte EMR Amazon-Konsole, sie automatisch zu kennzeichnen, sofern Sie über die erforderlichen Berechtigungen zum Taggen von Ressourcen verfügen. Wenn Sie nicht über die Berechtigungen zum Taggen von Ressourcen verfügen oder die Konsole verwenden möchten, bitten Sie Ihren Administrator, diese Ressourcen zu taggen.
- Sie können Cluster mit verwalteten v2-Richtlinien nur starten, wenn alle Voraussetzungen erfüllt sind.
- Die alte EMR Amazon-Konsole zeigt Ihnen, welche Ressourcen (VPC/Subnetze) markiert werden müssen.

IAMverwaltete Richtlinie für vollen Zugriff (verwaltete Standardrichtlinie v2)

Die verwalteten EMR Standardrichtlinien mit Geltungsbereich v2 gewähren Benutzern bestimmte Zugriffsrechte. Sie benötigen ein vordefiniertes EMR Amazon-Ressourcen-Tag und `iam:PassRole` Bedingungsschlüssel für Ressourcen, die von Amazon verwendet werdenEMR, wie z. B. die `Subnet` und, die `SecurityGroup` Sie zum Starten Ihres Clusters verwenden.

Um die erforderlichen Aktionen für Amazon zu gewährenEMR, fügen Sie die `AmazonEMRFullAccessPolicy_v2` verwaltete Richtlinie bei. Diese aktualisierte verwaltete Standardrichtlinie ersetzt die `AmazonElasticMapReduceFullAccess` verwaltete Richtlinie.

`AmazonEMRFullAccessPolicy_v2`hängt vom begrenzten Zugriff auf Ressourcen ab, die Amazon EMR bereitstellt oder nutzt. Wenn Sie diese Richtlinie verwenden, müssen Sie bei der Bereitstellung des Clusters das Benutzer-Tag `for-use-with-amazon-emr-managed-policies = true` übergeben. Amazon EMR verbreitet das Tag automatisch. Darüber hinaus müssen Sie möglicherweise manuell ein Benutzer-Tag zu bestimmten Ressourcentypen hinzufügen, z. B. EC2 Sicherheitsgruppen, die nicht von Amazon erstellt wurdenEMR. Weitere Informationen finden Sie unter Taggen von Ressourcen zur Verwendung verwalteter Richtlinien.

Die AmazonEMRFullAccessPolicy_v2-Richtlinie schützt Ressourcen, indem sie wie folgt vorgeht:

- Erfordert, dass Ressourcen mit dem vordefinierten Tag `for-use-with-amazon-emr-managed-policies` für von Amazon EMR verwaltete Richtlinien für die Clustererstellung und den EMR Amazon-Zugriff gekennzeichnet werden.
- Beschränkt die `iam:PassRole`-Aktion auf bestimmte Standardrollen und den `iam:PassedToService`-Zugriff auf bestimmte Services.
- Bietet standardmäßig keinen Zugriff mehr auf AmazonEC2, Amazon S3 und andere Dienste.

Im Folgenden finden Sie den Inhalt dieser Richtlinie.

> **Note**
>
> Sie können die Richtlinie auch über den Konsolenlink AmazonEMRFullAccessPolicy_v2 anzeigen.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "RunJobFlowExplicitlyWithEMRManagedTag",
            "Effect": "Allow",
            "Action": [
                "elasticmapreduce:RunJobFlow"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/for-use-with-amazon-emr-managed-policies": "true"
                }
            }
        },
        {
            "Sid": "ElasticMapReduceActions",
            "Effect": "Allow",
            "Action": [
                "elasticmapreduce:AddInstanceFleet",
                "elasticmapreduce:AddInstanceGroups",
                "elasticmapreduce:AddJobFlowSteps",
```

```
                "elasticmapreduce:AddTags",
                "elasticmapreduce:CancelSteps",
                "elasticmapreduce:CreateEditor",
                "elasticmapreduce:CreateSecurityConfiguration",
                "elasticmapreduce:DeleteEditor",
                "elasticmapreduce:DeleteSecurityConfiguration",
                "elasticmapreduce:DescribeCluster",
                "elasticmapreduce:DescribeEditor",
                "elasticmapreduce:DescribeJobFlows",
                "elasticmapreduce:DescribeSecurityConfiguration",
                "elasticmapreduce:DescribeStep",
                "elasticmapreduce:DescribeReleaseLabel",
                "elasticmapreduce:GetBlockPublicAccessConfiguration",
                "elasticmapreduce:GetManagedScalingPolicy",
                "elasticmapreduce:GetAutoTerminationPolicy",
                "elasticmapreduce:ListBootstrapActions",
                "elasticmapreduce:ListClusters",
                "elasticmapreduce:ListEditors",
                "elasticmapreduce:ListInstanceFleets",
                "elasticmapreduce:ListInstanceGroups",
                "elasticmapreduce:ListInstances",
                "elasticmapreduce:ListSecurityConfigurations",
                "elasticmapreduce:ListSteps",
                "elasticmapreduce:ListSupportedInstanceTypes",
                "elasticmapreduce:ModifyCluster",
                "elasticmapreduce:ModifyInstanceFleet",
                "elasticmapreduce:ModifyInstanceGroups",
                "elasticmapreduce:OpenEditorInConsole",
                "elasticmapreduce:PutAutoScalingPolicy",
                "elasticmapreduce:PutBlockPublicAccessConfiguration",
                "elasticmapreduce:PutManagedScalingPolicy",
                "elasticmapreduce:RemoveAutoScalingPolicy",
                "elasticmapreduce:RemoveManagedScalingPolicy",
                "elasticmapreduce:RemoveTags",
                "elasticmapreduce:SetTerminationProtection",
                "elasticmapreduce:StartEditor",
                "elasticmapreduce:StopEditor",
                "elasticmapreduce:TerminateJobFlows",
                "elasticmapreduce:ViewEventsFromAllClustersInConsole"
            ],
            "Resource": "*"
        },
        {
            "Sid": "ViewMetricsInEMRConsole",
```

```
            "Effect": "Allow",
            "Action": [
                "cloudwatch:GetMetricStatistics"
            ],
            "Resource": "*"
        },
        {
            "Sid": "PassRoleForElasticMapReduce",
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": [
                "arn:aws:iam::*:role/EMR_DefaultRole",
                "arn:aws:iam::*:role/EMR_DefaultRole_V2"
            ],
            "Condition": {
                "StringLike": {
                    "iam:PassedToService": "elasticmapreduce.amazonaws.com*"
                }
            }
        },
        {
            "Sid": "PassRoleForEC2",
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::*:role/EMR_EC2_DefaultRole",
            "Condition": {
                "StringLike": {
                    "iam:PassedToService": "ec2.amazonaws.com*"
                }
            }
        },
        {
            "Sid": "PassRoleForAutoScaling",
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::*:role/EMR_AutoScaling_DefaultRole",
            "Condition": {
                "StringLike": {
                    "iam:PassedToService": "application-autoscaling.amazonaws.com*"
                }
            }
        },
        {
            "Sid": "ElasticMapReduceServiceLinkedRole",
```

```
            "Effect": "Allow",
            "Action": "iam:CreateServiceLinkedRole",
            "Resource": "arn:aws:iam::*:role/aws-service-role/
elasticmapreduce.amazonaws.com*/AWSServiceRoleForEMRCleanup*",
            "Condition": {
                "StringEquals": {
                    "iam:AWSServiceName": [
                        "elasticmapreduce.amazonaws.com",
                        "elasticmapreduce.amazonaws.com.cn"
                    ]
                }
            }
        },
        {
            "Sid": "ConsoleUIActions",
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeAccountAttributes",
                "ec2:DescribeAvailabilityZones",
                "ec2:DescribeImages",
                "ec2:DescribeKeyPairs",
                "ec2:DescribeNatGateways",
                "ec2:DescribeRouteTables",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "ec2:DescribeVpcEndpoints",
                "s3:ListAllMyBuckets",
                "iam:ListRoles"
            ],
            "Resource": "*"
        }
    ]
}
```

IAMverwaltete Richtlinie für vollen Zugriff (derzeit nicht mehr unterstützt)

Die verwalteten Richtlinien `AmazonElasticMapReduceFullAccess` und `AmazonEMRFullAccessPolicy_v2` AWS Identity and Access Management (IAM) gewähren alle erforderlichen Aktionen für Amazon EMR und andere Dienste.

> **Important**
>
> Die `AmazonElasticMapReduceFullAccess` verwaltete Richtlinie ist veraltet und wird nicht mehr für die Verwendung mit Amazon empfohlen. EMR Nutzen Sie stattdessen `AmazonEMRFullAccessPolicy_v2`. Wenn der IAM Service irgendwann die v1-Richtlinie als veraltet markiert, können Sie ihn keiner Rolle zuordnen. Sie können einem Cluster jedoch eine bestehende Rolle zuordnen, auch wenn diese Rolle die veraltete Richtlinie verwendet.

Die standardmäßigen verwalteten Richtlinien von Amazon mit EMR vollen Berechtigungen beinhalten `iam:PassRole` Sicherheitskonfigurationen, darunter die folgenden:

- `iam:PassRole`Berechtigungen nur für bestimmte EMR Amazon-Standardrollen.
- `iam:PassedToService`Bedingungen, die es Ihnen ermöglichen, die Richtlinie nur mit bestimmten AWS Diensten zu verwenden, z. B. `elasticmapreduce.amazonaws.com` und`ec2.amazonaws.com`.

Sie können die JSON Version der Richtlinien A mazonEMRFull AccessPolicy _v2 und A mazonEMRService Policy_v2 in der Konsole einsehen. IAM Wir empfehlen, dass Sie neue Cluster mit den verwalteten v2-Richtlinien erstellen.

Sie können den Inhalt der veralteten v1-Richtlinie unter einsehen. AWS Management Console `AmazonElasticMapReduceFullAccess` Die `ec2:TerminateInstances` Aktion in der Richtlinie gewährt einem Benutzer oder einer Rolle die Erlaubnis, alle mit dem IAM Konto verknüpften EC2 Amazon-Instances zu kündigen. Dies schließt Instances ein, die nicht Teil eines EMR Clusters sind.

IAMverwaltete Richtlinie für schreibgeschützten Zugriff (verwaltete Standardrichtlinie v2)

Um Amazon nur Leserechte zu gewährenEMR, fügen Sie die verwaltete Richtlinie A mazonEMRRead OnlyAccessPolicy _v2 bei. Diese verwaltete Standardrichtlinie ersetzt die `AmazonElasticMapReduceReadOnlyAccess` verwaltete Richtlinie. Der Inhalt dieser Richtlinienerklärung wird im folgenden Ausschnitt dargestellt. Im Vergleich zur `AmazonElasticMapReduceReadOnlyAccess`-Richtlinie verwendet die `AmazonEMRReadOnlyAccessPolicy_v2`-Richtlinie keine Platzhalterzeichen für das `elasticmapreduce`-Element. Stattdessen beschränkt sich die standardmäßige v2-Richtlinie auf die zulässigen `elasticmapreduce`-Aktionen.

> **Note**
>
> Sie können den AWS Management Console Link auch verwenden, um die Richtlinie AmazonEMRReadOnlyAccessPolicy_v2einzusehen.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ElasticMapReduceActions",
            "Effect": "Allow",
            "Action": [
                "elasticmapreduce:DescribeCluster",
                "elasticmapreduce:DescribeEditor",
                "elasticmapreduce:DescribeJobFlows",
                "elasticmapreduce:DescribeSecurityConfiguration",
                "elasticmapreduce:DescribeStep",
                "elasticmapreduce:DescribeReleaseLabel",
                "elasticmapreduce:GetBlockPublicAccessConfiguration",
                "elasticmapreduce:GetManagedScalingPolicy",
                "elasticmapreduce:GetAutoTerminationPolicy",
                "elasticmapreduce:ListBootstrapActions",
                "elasticmapreduce:ListClusters",
                "elasticmapreduce:ListEditors",
                "elasticmapreduce:ListInstanceFleets",
                "elasticmapreduce:ListInstanceGroups",
                "elasticmapreduce:ListInstances",
                "elasticmapreduce:ListSecurityConfigurations",
                "elasticmapreduce:ListSteps",
                "elasticmapreduce:ListSupportedInstanceTypes",
                "elasticmapreduce:ViewEventsFromAllClustersInConsole"
            ],
            "Resource": "*"
        },
        {
            "Sid": "ViewMetricsInEMRConsole",
            "Effect": "Allow",
            "Action": [
                "cloudwatch:GetMetricStatistics"
            ],
            "Resource": "*"
```

```
        }
    ]
}
```

IAMverwaltete Richtlinie für schreibgeschützten Zugriff (kein veralteter Status)

Die `AmazonElasticMapReduceReadOnlyAccess`-verwaltete Richtlinie ist demnächst veraltet. Sie können diese Richtlinie nicht anhängen, wenn Sie neue Cluster starten. `AmazonElasticMapReduceReadOnlyAccess`wurde durch `AmazonEMRReadOnlyAccessPolicy_v2`die von Amazon EMR standardmäßig verwaltete Richtlinie ersetzt. Der Inhalt dieser Richtlinienerklärung wird im folgenden Ausschnitt dargestellt. Platzhalterzeichen für das `elasticmapreduce`-Element geben an, dass nur Aktionen, die mit der angegebenen Zeichenfolgen beginnen, zulässig sind. Hinweis: Da diese Richtlinie Aktionen nicht ausdrücklich verweigert, kann dennoch eine andere Richtlinienanweisung verwendet werden, um den Zugriff auf bestimmte Aktionen zu gewähren.

> **Note**
>
> Sie können die Richtlinie auch verwenden AWS Management Console , um die Richtlinie einzusehen.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticmapreduce:Describe*",
        "elasticmapreduce:List*",
        "elasticmapreduce:ViewEventsFromAllClustersInConsole",
        "s3:GetObject",
        "s3:ListAllMyBuckets",
        "s3:ListBucket",
        "sdb:Select",
        "cloudwatch:GetMetricStatistics"
      ],
      "Resource": "*"
    }
  ]
```

```
}
```

AWS verwaltete Richtlinie: EMRDescribeClusterPolicyForEMRWAL

Sie können keine Verbindungen `EMRDescribeClusterPolicyForEMRWAL` zu Ihren IAM Entitäten herstellen. Diese Richtlinie ist mit einer dienstbezogenen Rolle verknüpft, die es Amazon EMR ermöglicht, Aktionen in Ihrem Namen durchzuführen. Weitere Informationen zu dieser dienstleistungsbezogenen Rolle finden Sie unter. Verwenden von serviceverknüpften Rollen für die Write-Ahead-Protokollierung

Diese Richtlinie gewährt Amazon nur Leseberechtigungen, die es dem WAL Service ermöglichen, den Status eines Clusters EMR zu finden und zurückzugeben. Weitere Informationen zu Amazon EMR WAL finden Sie unter Write-ahead logs (WAL) für Amazon. EMR

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen:

- `emr`— Ermöglicht Principals, den Cluster-Status von Amazon EMR zu beschreiben. Dies ist erforderlich, damit Amazon bestätigen EMR kann, wann ein Cluster beendet wurde, und nach dreißig Tagen alle vom Cluster WAL hinterlassenen Protokolle bereinigen kann.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticmapreduce:DescribeCluster"
      ],
      "Resource": "*"
    }
  ]
}
```

AWS verwaltete Richtlinien für Amazon EMR

Eine AWS verwaltete Richtlinie ist eine eigenständige Richtlinie, die von erstellt und verwaltet wird AWS. AWS Verwaltete Richtlinien dienen dazu, Berechtigungen für viele gängige Anwendungsfälle bereitzustellen, sodass Sie damit beginnen können, Benutzern, Gruppen und Rollen Berechtigungen zuzuweisen.

Beachten Sie, dass AWS verwaltete Richtlinien für Ihre speziellen Anwendungsfälle möglicherweise keine Berechtigungen mit den geringsten Rechten gewähren, da sie allen AWS Kunden zur Verfügung stehen. Wir empfehlen Ihnen, die Berechtigungen weiter zu reduzieren, indem Sie kundenverwaltete Richtlinien definieren, die speziell auf Ihre Anwendungsfälle zugeschnitten sind.

Sie können die in AWS verwalteten Richtlinien definierten Berechtigungen nicht ändern. Wenn die in einer AWS verwalteten Richtlinie definierten Berechtigungen AWS aktualisiert werden, wirkt sich das Update auf alle Prinzidentitäten (Benutzer, Gruppen und Rollen) aus, denen die Richtlinie zugeordnet ist. AWS aktualisiert eine AWS verwaltete Richtlinie höchstwahrscheinlich, wenn eine neue Richtlinie eingeführt AWS -Service wird oder neue API Operationen für bestehende Dienste verfügbar werden.

Weitere Informationen finden Sie im IAMBenutzerhandbuch unter AWS Verwaltete Richtlinien.

EMRAktualisierungen der AWS verwalteten Richtlinien durch Amazon

Sehen Sie sich Details zu Aktualisierungen der AWS verwalteten Richtlinien für Amazon an, EMR seit dieser Service begonnen hat, diese Änderungen zu verfolgen.

Änderung	Beschreibung	Datum
EMRDescribeClusterPolicyForEMRWAL – Neue Richtlinie.	Es wurde eine neue Richtlinie hinzugefügt, EMR sodass Amazon den Clusterstatus für die WAL Bereinigung dreißig Tage nach Beendigung des Clusters bestimmen kann.	10. August 2023
AmazonEMRFullAccessPolicy_v2 und **AmazonEMRReadOnlyAccessPolicy_v2** – Zur Aktualisierung einer bestehenden Richtlinie	`elasticmapreduce:DescribeReleaseLabel` und `elasticmapreduce:GetAutoTerminationPolicy` hinzugefügt.	21. April 2022
AmazonEMRFullAccessPolicy_v2 – Aktualisierung auf eine bestehende Richtlinie	`ec2:DescribeImages` hinzugefügt für Verwenden Sie ein benutzerdefiniertes AMI.	15. Februar 2022

Änderung	Beschreibung	Datum
Von Amazon EMR verwaltete Richtlinien	Aktualisiert, um die Verwendung vordefinierter Benutzer-Tags zu verdeutlichen. Es wurde ein Abschnitt zur Verwendung der AWS Konsole zum Starten von Clustern mit verwalteten v2-Richtlinien hinzugefügt.	29. September 2021
AmazonEMRFullAccessPolicy_v2 – Aktualisierung auf eine bestehende Richtlinie	Die Aktionen `PassRoleForAutoScaling` und `PassRoleForEC2` wurden dahingehend geändert, dass der `StringLike` -Bedingungsoperator entsprechend `"iam:PassedToService":"application-autoscaling.amazonaws.com*"` und `"iam:PassedToService":"ec2.amazonaws.com*"` verwendet wird.	20. Mai 2021

Änderung	Beschreibung	Datum
AmazonEMRFullAccessPolicy_v2 – Aktualisierung auf eine bestehende Richtlinie	Ungültige Aktion `s3:ListBuckets` wurde entfernt und durch `s3:ListAllMyBuckets` -Aktion ersetzt. Die Erstellung von serviceverknüpften Rollen (SLR) wurde aktualisiert und ist nun explizit auf die einzige Rolle beschränkt, SLR die Amazon mit expliziten EMR Serviceprinzipien anbietet. Die ElementeSLRs, die erstellt werden können, sind genau dieselben wie vor dieser Änderung.	23. März 2021

Änderung	Beschreibung	Datum
AmazonEMRFullAccessPolicy_v2 – Neue Richtlinie.	Amazon EMR hat neue Berechtigungen für den Geltungsbereich des Zugriffs auf Ressourcen hinzugefügt und eine Voraussetzung hinzugefügt, dass Benutzer Ressourcen mit einem vordefinierten Benutzertag versehen müssen, bevor sie von Amazon EMR verwaltete Richtlinien verwenden können. `iam:PassRole` - Aktion erfordert, dass die `iam:PassedToService` - Bedingung auf den angegebenen Service gesetzt ist. Der Zugriff auf AmazonEC2, Amazon S3 und andere Dienste ist standardmäßig nicht erlaubt.	11. März 2021
AmazonEMRServicePolicy_v2 – Neue Richtlinie.	Fügt die Voraussetzung hinzu, dass Benutzer Benutzer-Tags zu Ressourcen hinzufügen müssen, bevor sie diese Richtlinie verwenden können.	11. März 2021
AmazonEMRReadOnlyAccessPolicy_v2 – Neue Richtlinie.	Mit Berechtigungen können nur bestimmte schreibgeschützte ElasticMapReduce-Aktionen ausgeführt werden. Der Zugriff auf Amazon S3 ist standardmäßig nicht zulässig.	11. März 2021

Änderung	Beschreibung	Datum
Amazon EMR hat begonnen, Änderungen zu verfolgen	Amazon EMR hat damit begonnen, Änderungen an seinen AWS verwalteten Richtlinien nachzuverfolgen.	11. März 2021

IAMRichtlinien für den tagbasierten Zugriff auf Cluster und Notebooks EMR

Sie können Bedingungen in Ihrer identitätsbasierten Richtlinie verwenden, um den Zugriff auf Cluster und EMR Notebooks anhand von Tags zu steuern.

Weitere Informationen zum Hinzufügen von Tags zu Clustern finden Sie unter Clustern EMRtaggen.

Die folgenden Beispiele zeigen verschiedene Szenarien und Möglichkeiten, Bedingungsoperatoren mit EMR Amazon-Bedingungsschlüsseln zu verwenden. Diese IAM Grundsatzerklärungen dienen nur zu Demonstrationszwecken und sollten nicht in Produktionsumgebungen verwendet werden. Es gibt mehrere Möglichkeiten für die Kombination von Richtlinienanweisungen zum Gewähren oder Verweigern von Berechtigungen entsprechend Ihren Anforderungen. Weitere Informationen zur Planung und zum Testen von IAM Richtlinien finden Sie im IAMBenutzerhandbuch.

> **Important**
>
> Das explizite Ablehnen von Berechtigungen für Markierungsaktionen stellt eine wichtige Überlegung dar. Dadurch wird verhindert, dass Benutzer eine Ressource markieren und sich dadurch selbst Berechtigungen erteilen, die Sie nicht gewähren wollten. Wenn Sie Tagging-Aktionen für eine Ressource nicht verweigern, kann ein Benutzer Tags ändern und die Absicht der tagbasierten Richtlinien umgehen.

Beispiel identitätsbasierter Richtlinienanweisungen für Cluster

Die folgenden Beispiele zeigen identitätsbasierte Berechtigungsrichtlinien, die zur Steuerung der Aktionen verwendet werden, die mit EMR Clustern zulässig sind.

> **Important**
>
> Für die `ModifyInstanceGroup` Aktion in Amazon müssen Sie EMR keine Cluster-ID angeben. Aus diesem Grund sind zusätzliche Überlegungen erforderlich, um diese Aktion

auf der Grundlage von Cluster-Tags abzulehnen. Weitere Informationen finden Sie unter Die Aktion wird ModifyInstanceGroup verweigert.

Themen

- Zulassen von Aktionen nur für Cluster mit bestimmten Tag-Werten
- Erfordert Cluster-Tagging, wenn ein Cluster erstellt wird
- Aktionen für Cluster mit einem bestimmten Tag zulassen, unabhängig vom Tag-Wert

Zulassen von Aktionen nur für Cluster mit bestimmten Tag-Werten

Die folgenden Beispiele veranschaulichen eine Richtlinie, mit der ein Benutzer Aktionen auf der Grundlage des Cluster-Tags *department* mit dem Wert *dev* durchführen, sowie Cluster mit demselben Tag markieren kann. Das letzte Richtlinienbeispiel zeigt, wie Rechte verweigert werden können, um EMR Cluster mit etwas anderem als demselben Tag zu kennzeichnen.

Im folgenden Richtlinienbeispiel versucht der `StringEquals`-Bedingungsoperator, *dev* und den Wert für das Tag *department* abzugleichen. Wenn das Tag *department* dem Cluster nicht hinzugefügt wurde oder den Wert *dev* nicht enthält, ist die Richtlinie nicht anzuwenden und die Aktionen werden von dieser Richtlinie nicht zugelassen. Wenn keine anderen Richtlinienanweisungen die Aktionen zulassen, kann der Benutzer nur mit Clustern arbeiten, die dieses Tag mit diesem Wert enthalten.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt12345678901234",
      "Effect": "Allow",
      "Action": [
        "elasticmapreduce:DescribeCluster",
        "elasticmapreduce:ListSteps",
        "elasticmapreduce:TerminateJobFlows",
        "elasticmapreduce:SetTerminationProtection",
        "elasticmapreduce:ListInstances",
        "elasticmapreduce:ListInstanceGroups",
        "elasticmapreduce:ListBootstrapActions",
        "elasticmapreduce:DescribeStep"
      ],
```

```
      "Resource": [
        "*"
      ],
      "Condition": {
        "StringEquals": {
          "elasticmapreduce:ResourceTag/department": "dev"
        }
      }
    }
  ]
}
```

Sie können auch mehrere Tag-Werte mithilfe eines Bedingungsoperators angeben. Um beispielsweise alle Aktionen in Clustern zuzulassen, in denen das Tag *department* den Wert *dev* oder *test* enthält können Sie den Bedingungsblock im vorherigen Beispiel durch Folgendes ersetzen.

```
            "Condition": {
              "StringEquals": {
                "elasticmapreduce:ResourceTag/department":["dev", "test"]
              }
            }
```

Erfordert Cluster-Tagging, wenn ein Cluster erstellt wird

Wie im oben stehenden Beispiel sucht die folgenden Beispielrichtlinie dasselbe übereinstimmende Tag: den Wert *dev* für das Tag *department*. In diesem Beispiel gibt der `RequestTag`-Bedingungsschlüssel jedoch an, dass die Richtlinie während der Tag-Erstellung gilt. Sie müssen also einen Cluster mit einem Tag erstellen, der dem angegebenen Wert entspricht.

Um einen Cluster mit einem Tag zu erstellen, benötigen Sie auch die Erlaubnis für die `elasticmapredue:AddTags`-Aktion. Bei dieser Anweisung stellt der `elasticmapreduce:ResourceTag` Bedingungsschlüssel sicher, dass IAM nur Zugriff auf Tag-Ressourcen gewährt wird, deren Wert *dev* auf dem *department* Tag steht. Das `Resource`-Element wird verwendet, um diese Berechtigung auf Clusterressourcen zu beschränken.

Für die `PassRole` Ressourcen müssen Sie die AWS Konto-ID oder den Alias, den Namen der Servicerolle in der `PassRoleForEMR` Anweisung und den Namen des Instanzprofils in der

PassRoleForEC2 Anweisung angeben. Weitere Informationen zum IAM ARN Format finden Sie IAMARNsim IAMBenutzerhandbuch.

Weitere Informationen zum Abgleichen von Tag-Schlüsselwerten finden Sie aws:RequestTag/tag-keyim IAMBenutzerhandbuch.

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Sid": "RunJobFlowExplicitlyWithTag",
        "Effect": "Allow",
        "Action": [
          "elasticmapreduce:RunJobFlow"
        ],
        "Resource": "*",
        "Condition": {
          "StringEquals": {
            "aws:RequestTag/department": "dev"
          }
        }
      },
      {
        "Sid": "AddTagsForDevClusters",
        "Effect": "Allow",
        "Action": "elasticmapreduce:AddTags",
        "Resource": "arn:aws:elasticmapreduce:*:*:cluster/*",
        "Condition": {
          "StringEquals": {
            "elasticmapreduce:ResourceTag/department": "dev"
          }
        }
      },
      {
        "Sid": "PassRoleForEMR",
        "Effect": "Allow",
        "Action": "iam:PassRole",
        "Resource": "arn:aws:iam::AccountId:role/Role-Name-With-Path",
        "Condition": {
          "StringLike": {
            "iam:PassedToService": "elasticmapreduce.amazonaws.com*"
          }
        }
```

```
      },
      {
        "Sid": "PassRoleForEC2",
        "Effect": "Allow",
        "Action": "iam:PassRole",
        "Resource": "arn:aws:iam::AccountId:role/Role-Name-With-Path",
        "Condition": {
          "StringLike": {
            "iam:PassedToService": "ec2.amazonaws.com*"
          }
        }
      }
    ]
}
```

Aktionen für Cluster mit einem bestimmten Tag zulassen, unabhängig vom Tag-Wert

Sie können auch Aktionen nur für Cluster mit einem bestimmten Tag, unabhängig vom Tag-Wert, zulassen. Dazu können Sie den `Null`-Operator verwenden. Weitere Informationen finden Sie im IAMBenutzerhandbuch unter Bedingungsoperator zur Überprüfung der Existenz von Bedingungsschlüsseln. Um beispielsweise Aktionen nur für EMR Cluster zuzulassen, die über das *`department`* Tag verfügen, unabhängig vom darin enthaltenen Wert, könnten Sie die Bedingungsblöcke im vorherigen Beispiel durch den folgenden ersetzen. Der `Null` Operator sucht nach dem Vorhandensein des Tags *`department`* in einem EMR Cluster. Wenn das Tag vorhanden ist, wird die Anweisung `Null` entsprechend der in dieser Richtlinienanweisung angegebenen Bedingung mit "false" ausgewertet und die jeweiligen Aktionen werden zugelassen.

```
"Condition": {
  "Null": {
    "elasticmapreduce:ResourceTag/department":"false"
  }
}
```

Mit der folgenden Richtlinienanweisung kann ein Benutzer nur dann einen EMR Cluster erstellen, wenn der Cluster über ein *`department`* Tag verfügt, das einen beliebigen Wert enthalten kann. Für die `PassRole` Ressource müssen Sie die AWS Konto-ID oder den Alias und den Namen der Dienstrolle angeben. Weitere Informationen zum IAM ARN Format finden Sie IAMARNsim IAMBenutzerhandbuch.

Weitere Informationen zur Angabe des Bedingungsoperators Null („falsch“) finden Sie im IAMBenutzerhandbuch unter Bedingungsoperator zur Überprüfung der Existenz von Bedingungsschlüsseln.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "CreateClusterTagNullCondition",
            "Effect": "Allow",
            "Action": [
                "elasticmapreduce:RunJobFlow"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "Null": {
                    "aws:RequestTag/department": "false"
                }
            }
        },
        {
            "Sid": "AddTagsNullCondition",
            "Effect": "Allow",
            "Action": "elasticmapreduce:AddTags",
            "Resource": "arn:aws:elasticmapreduce:*:*:cluster/*",
            "Condition": {
                "Null": {
                    "elasticmapreduce:ResourceTag/department": "false"
                }
            }
        },
        {
            "Sid": "PassRoleForElasticMapReduce",
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::AccountId:role/Role-Name-With-Path",
            "Condition": {
                "StringLike": {
                    "iam:PassedToService": "elasticmapreduce.amazonaws.com*"
                }
            }
```

```
        },
        {
            "Sid": "PassRoleForEC2",
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::AccountId:role/Role-Name-With-Path",
            "Condition": {
                "StringLike": {
                    "iam:PassedToService": "ec2.amazonaws.com*"
                }
            }
        }
    ]
}
```

Beispiel für identitätsbasierte Richtlinienerklärungen für Notebooks EMR

Die IAM Beispielrichtlinien in diesem Abschnitt veranschaulichen gängige Szenarien für die Verwendung von Schlüsseln, um zulässige Aktionen mit Notebooks einzuschränken. EMR Solange keine andere mit dem Prinzipal (Benutzer) verknüpfte Richtlinie die Aktionen zulässt, schränken die Bedingungskontextschlüssel die zulässigen Aktionen wie angegeben ein.

Example — Erlaubt nur den Zugriff auf EMR Notizbücher, die ein Benutzer auf der Grundlage von Tagging erstellt

Wenn die folgende Beispiel-Richtlinienanweisung an eine Rolle oder einen Benutzer angefügt wird, können Benutzer nur mit Notebooks arbeiten, die sie selbst erstellt haben. Diese Richtlinienanweisung verwendet das bei der Erstellung eines Notebooks angewendete Standard-Tag.

In diesem Beispiel versucht der Bedingungsoperator `StringEquals`, eine Variable, die die Benutzer-ID (`{aws:userId}`) des aktuellen Benutzers darstellt, dem Wert des Tags `creatorUserID` zuzuordnen. Wenn das Tag `creatorUserID` nicht zum Notebook hinzugefügt wurde oder den Wert der ID des aktuellen Benutzers nicht enthält, ist die Richtlinie nicht anzuwenden und die Aktionen werden von dieser Richtlinie nicht zugelassen. Wenn keine anderen Richtlinienanweisungen die Aktionen zulassen, kann der Benutzer nur mit Notebooks arbeiten, die dieses Tag mit diesem Wert enthalten.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
```

```
                "elasticmapreduce:DescribeEditor",
                "elasticmapreduce:StartEditor",
                "elasticmapreduce:StopEditor",
                "elasticmapreduce:DeleteEditor",
                "elasticmapreduce:OpenEditorInConsole"
            ],
            "Effect": "Allow",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "elasticmapreduce:ResourceTag/creatorUserId": "${aws:userId}"
                }
            }
        }
    ]
}
```

Example – Notebook-Tagging anfordern, wenn ein Notebook erstellt wird

In diesem Beispiel wird der Kontextschlüssel `RequestTag` verwendet. Die Aktion `CreateEditor` ist nur dann zulässig, wenn der Benutzer das `creatorUserID` Tag, das standardmäßig hinzugefügt wird, nicht ändert oder löscht. Die Variable $ {aws:userId} gibt die Benutzer-ID des aktuell aktiven Benutzers an. Dies ist der Standardwert des Tags.

Die Richtlinienanweisung kann verwendet werden, um sicherzustellen, dass Benutzer das Tag `createUserId` nicht entfernen und dessen Wert nicht ändern.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "elasticmapreduce:CreateEditor"
            ],
            "Effect": "Allow",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "elasticmapreduce:RequestTag/creatorUserId": "${aws:userid}"
                }
            }
        }
    ]
```

```
}
```

Dieses Beispiel erfordert, dass der Benutzer den Cluster mit einem Tag mit der Schlüsselzeichenfolge `dept` und einem der folgenden Werte erstellt: `datascience`, `analytics`, `operations`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "elasticmapreduce:CreateEditor"
            ],
            "Effect": "Allow",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "elasticmapreduce:RequestTag/dept": [
                        "datascience",
                        "analytics",
                        "operations"
                    ]
                }
            }
        }
    ]
}
```

Example – Die Notebook-Erstellung auf getaggte Cluster beschränken und Notebook-Tags anfordern

Dieses Beispiel erlaubt die Notebook-Erstellung nur, wenn das Notebook mit einem Tag erstellt wird, bei dem die Schlüsselzeichenfolge `owner` auf einen der angegebenen Werte festgelegt ist. Darüber hinaus kann das Notebook nur erstellt werden, wenn der Cluster ein Tag enthält, bei dem die Schlüsselzeichenfolge `department` auf einen der angegebenen Werte festgelegt ist.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "elasticmapreduce:CreateEditor"
            ],
```

```
            "Effect": "Allow",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "elasticmapreduce:RequestTag/owner": [
                        "owner1",
                        "owner2",
                        "owner3"
                    ],
                    "elasticmapreduce:ResourceTag/department": [
                        "dep1",
                        "dep3"
                    ]
                }
            }
        }
    ]
}
```

Example – Basierend auf Tags die Möglichkeit einschränken, ein Notebook zu starten

Dieses Beispiel schränkt die Möglichkeit, ein Notebook zu starten, auf Notebooks ein, die ein Tag enthalten, bei dem die Schlüsselzeichenfolge `owner` auf einen der angegebenen Werte festgelegt ist. Da das Element `Resource` verwendet wird, um nur den `editor` anzugeben, gilt die Bedingung nicht für den Cluster und ein Tagging ist nicht erforderlich.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "elasticmapreduce:StartEditor"
            ],
            "Effect": "Allow",
            "Resource": "arn:aws:elasticmapreduce:*:123456789012:editor/*",
            "Condition": {
                "StringEquals": {
                    "elasticmapreduce:ResourceTag/owner": [
                        "owner1",
                        "owner2"
                    ]
                }
            }
```

```
        }
    ]
}
```

Dieses Beispiel ähnelt dem obigen. Die Einschränkung gilt hier jedoch nur für getaggte Cluster, nicht für Notebooks.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "elasticmapreduce:StartEditor"
            ],
            "Effect": "Allow",
            "Resource": "arn:aws:elasticmapreduce:*:123456789012:cluster/*",
            "Condition": {
                "StringEquals": {
                    "elasticmapreduce:ResourceTag/department": [
                        "dep1",
                        "dep3"
                    ]
                }
            }
        }
    ]
}
```

Dieses Beispiel verwendet andere Notebook- und Cluster-Tags. Es ermöglicht das Starten eines Notebooks nur, wenn Folgendes zutrifft:

- Das Notebook enthält ein Tag, bei dem die Schlüsselzeichenfolge `owner` auf einen der angegebenen Wert festgelegt ist.

  – und –

- Der Cluster enthält ein Tag, bei dem die Schlüsselzeichenfolge `department` auf einen der angegebenen Wert festgelegt ist.

```
{
    "Version": "2012-10-17",
    "Statement": [
```

```
        {
            "Action": [
                "elasticmapreduce:StartEditor"
            ],
            "Effect": "Allow",
            "Resource": "arn:aws:elasticmapreduce:*:123456789012:editor/*",
            "Condition": {
                "StringEquals": {
                    "elasticmapreduce:ResourceTag/owner": [
                        "user1",
                        "user2"
                    ]
                }
            }
        },
        {
            "Action": [
                "elasticmapreduce:StartEditor"
            ],
            "Effect": "Allow",
            "Resource": "arn:aws:elasticmapreduce:*:123456789012:cluster/*",
            "Condition": {
                "StringEquals": {
                    "elasticmapreduce:ResourceTag/department": [
                        "datascience",
                        "analytics"
                    ]
                }
            }
        }
    ]
}
```

Example – Basierend auf Tags die Möglichkeit einschränken, den Notebook-Editor zu öffnen

In diesem Beispiel kann der Notebook-Editor nur geöffnet werden, wenn Folgendes zutrifft:

- Das Notebook enthält ein Tag, bei dem die Schlüsselzeichenfolge `owner` auf einen der angegebenen Wert festgelegt ist.

  – und –

- Der Cluster enthält ein Tag, bei dem die Schlüsselzeichenfolge `department` auf einen der angegebenen Wert festgelegt ist.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "elasticmapreduce:OpenEditorInConsole"
            ],
            "Effect": "Allow",
            "Resource": "arn:aws:elasticmapreduce:*:123456789012:editor/*",
            "Condition": {
                "StringEquals": {
                    "elasticmapreduce:ResourceTag/owner": [
                        "user1",
                        "user2"
                    ]
                }
            }
        },
        {
            "Action": [
                "elasticmapreduce:OpenEditorInConsole"
            ],
            "Effect": "Allow",
            "Resource": "arn:aws:elasticmapreduce:*:123456789012:cluster/*",
            "Condition": {
                "StringEquals": {
                    "elasticmapreduce:ResourceTag/department": [
                        "datascience",
                        "analytics"
                    ]
                }
            }
        }
    ]
}
```

Die Aktion wird ModifyInstanceGroup verweigert

Die ModifyInstanceGroupsAktion in Amazon EMR erfordert nicht, dass Sie bei der Aktion eine Cluster-ID angeben. Stattdessen können Sie nur eine Instance-Gruppen-ID angeben. Aus diesem Grund hat eine scheinbar einfache Ablehnungsrichtlinie für diese Aktion, die auf der Cluster-ID oder

einem Cluster-Tag basiert, möglicherweise nicht die beabsichtigte Wirkung. Betrachten Sie die folgende Beispielrichtlinie.

```
{
 "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "elasticmapreduce:ModifyInstanceGroups"
            ],
            "Effect": "Allow",
            "Resource": "*"
        },
        {
            "Action": [
                "elasticmapreduce:ModifyInstanceGroups"
            ],
            "Effect": "Deny",
            "Resource": "arn:aws:elasticmapreduce:us-east-1:123456789012:cluster/
j-12345ABCDEFG67"
        }
    ]

}
```

Wenn ein Benutzer, dem diese Richtlinie zugewiesen ist, eine `ModifyInstanceGroup`-Aktion ausführt und nur die Instance-Gruppen-ID angibt, gilt die Richtlinie nicht. Da die Aktion für alle anderen Ressourcen zulässig ist, ist die Aktion erfolgreich.

Eine Lösung für dieses Problem besteht darin, der Identität eine Richtlinienerklärung beizufügen, die ein NotResourceElement verwendet, um jede `ModifyInstanceGroup` Aktion abzulehnen, die ohne Cluster-ID ausgeführt wurde. Die folgende Beispielrichtlinie fügt eine solche Deny-Anweisung hinzu, sodass jede `ModifyInstanceGroups`-Anfrage fehlschlägt, sofern keine Cluster-ID angegeben ist. Da eine Identität bei der Aktion eine Cluster-ID angeben muss, sind Ablehnungsbefehle, die auf der Cluster-ID basieren, daher wirksam.

```
{
 "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
```

```
                "elasticmapreduce:ModifyInstanceGroups"
            ],
            "Effect": "Allow",
            "Resource": "*"
        },
        {
            "Action": [
                "elasticmapreduce:ModifyInstanceGroups"
            ],
            "Effect": "Deny",
            "Resource": "arn:aws:elasticmapreduce:us-east-1:123456789012:cluster/
j-12345ABCDEFG67"
        },
        {
            "Action": [
                "elasticmapreduce:ModifyInstanceGroups"
            ],
            "Effect": "Deny",
            "NotResource": "arn:*:elasticmapreduce:*:*:cluster/*"
        }
    ]

}
```

Ein ähnliches Problem tritt auf, wenn Sie die `ModifyInstanceGroups`-Aktion auf der Grundlage des mit einem Cluster-Tag verknüpften Werts ablehnen möchten. Die Lösung ist ähnlich. Zusätzlich zu einer Ablehnungs-Anweisung, die den Tag-Wert angibt, können Sie eine Richtlinienanweisung hinzufügen, die die `ModifyInstanceGroup`-Aktion ablehnt, wenn das von Ihnen angegebene Tag nicht vorhanden ist, unabhängig vom Wert.

Das folgende Beispiel zeigt eine Richtlinie, die, wenn sie an eine Identität angehängt ist, der Identität die `ModifyInstanceGroups`-Aktion aller Cluster verweigert, bei denen das Tag `department` auf `dev` gesetzt ist. Diese Anweisung ist nur aufgrund der Ablehnungs-Anweisung wirksam, die die `StringNotLike`-Bedingung verwendet, um die Aktion zu verweigern, sofern das `department`-Tag nicht vorhanden ist.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "elasticmapreduce:ModifyInstanceGroups"
```

```
            ],
            "Effect": "Allow",
            "Resource": "*"
        },
        {
            "Action": [
                "elasticmapreduce:ModifyInstanceGroups"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/department": "dev"
                }
            },
            "Effect": "Deny",
            "Resource": "*"
        },
        {
            "Action": [
                "elasticmapreduce:ModifyInstanceGroups"
            ],
            "Condition": {
                "StringNotLike": {
                    "aws:ResourceTag/department": "?*"
                }
            },
            "Effect": "Deny",
            "Resource": "*"
        }
    ],
}
```

Fehlerbehebung Amazon EMR Amazon-Identität und -Zugriff

Verwenden Sie die folgenden Informationen, um häufig auftretende Probleme zu diagnostizieren und zu beheben, die bei der Arbeit mit Amazon EMR und auftreten könnenIAM.

Themen

- Ich bin nicht berechtigt, eine Aktion in Amazon durchzuführen EMR
- Ich bin nicht berechtigt, iam auszuführen: PassRole
- Ich möchte Personen außerhalb meines AWS Kontos den Zugriff auf meine EMR Amazon-Ressourcen ermöglichen

Ich bin nicht berechtigt, eine Aktion in Amazon durchzuführen EMR

Wenn Ihnen AWS Management Console mitgeteilt wird, dass Sie nicht berechtigt sind, eine Aktion durchzuführen, müssen Sie sich an Ihren Administrator wenden, um Unterstützung zu erhalten. Ihr Administrator ist die Person, die Ihnen Ihren Benutzernamen und Ihr Passwort bereitgestellt hat.

Der folgende Beispielfehler tritt auf, wenn der `mateojackson`-Benutzer versucht, die Konsole zum Anzeigen von Details zu einer fiktiven *`my-example-widget`*-Ressource zu verwenden, jedoch nicht über `EMR:`*`GetWidget`*-Berechtigungen verfügt.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 EMR:GetWidget on resource: my-example-widget
```

In diesem Fall bittet Mateo seinen Administrator um die Aktualisierung seiner Richtlinien, um unter Verwendung der Aktion *`my-example-widget`* auf die Ressource `EMR:`*`GetWidget`* zugreifen zu können.

Ich bin nicht berechtigt, iam auszuführen: PassRole

Wenn Sie eine Fehlermeldung erhalten, dass Sie nicht berechtigt sind, die `iam:PassRole` Aktion durchzuführen, müssen Ihre Richtlinien aktualisiert werden, damit Sie eine Rolle an Amazon weitergeben könnenEMR.

Einige AWS -Services ermöglichen es Ihnen, eine bestehende Rolle an diesen Service zu übergeben, anstatt eine neue Servicerolle oder eine dienstbezogene Rolle zu erstellen. Hierzu benötigen Sie Berechtigungen für die Übergabe der Rolle an den Dienst.

Der folgende Beispielfehler tritt auf, wenn ein IAM Benutzer mit dem Namen `marymajor` versucht, die Konsole zu verwenden, um eine Aktion in Amazon auszuführenEMR. Die Aktion erfordert jedoch, dass der Service über Berechtigungen verfügt, die durch eine Servicerolle gewährt werden. Mary besitzt keine Berechtigungen für die Übergabe der Rolle an den Dienst.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

In diesem Fall müssen die Richtlinien von Mary aktualisiert werden, um die Aktion `iam:PassRole` ausführen zu können.

Wenn Sie Hilfe benötigen, wenden Sie sich an Ihren AWS Administrator. Ihr Administrator hat Ihnen Ihre Anmeldeinformationen zur Verfügung gestellt.

Ich möchte Personen außerhalb meines AWS Kontos den Zugriff auf meine EMR Amazon-Ressourcen ermöglichen

Sie können eine Rolle erstellen, die Benutzer in anderen Konten oder Personen außerhalb Ihrer Organisation für den Zugriff auf Ihre Ressourcen verwenden können. Sie können festlegen, wem die Übernahme der Rolle anvertraut wird. Für Dienste, die ressourcenbasierte Richtlinien oder Zugriffskontrolllisten (ACLs) unterstützen, können Sie diese Richtlinien verwenden, um Personen Zugriff auf Ihre Ressourcen zu gewähren.

Weitere Informationen dazu finden Sie hier:

- Informationen darüber, ob Amazon diese Funktionen EMR unterstützt, finden Sie unterSo EMR arbeitet Amazon mit IAM.
- Informationen dazu, wie Sie Zugriff auf Ihre Ressourcen gewähren können, AWS-Konten die Ihnen gehören, finden Sie im IAM Benutzerhandbuch unter Gewähren des Zugriffs auf einen anderen IAMBenutzer AWS-Konto , den Sie besitzen.
- Informationen dazu, wie Sie Dritten Zugriff auf Ihre Ressourcen gewähren können AWS-Konten, finden Sie AWS-Konten im IAMBenutzerhandbuch unter Gewähren des Zugriffs für Dritte.
- Informationen dazu, wie Sie Zugriff über einen Identitätsverbund gewähren, finden Sie im Benutzerhandbuch unter Zugriff für extern authentifizierte Benutzer (Identitätsverbund). IAM
- Informationen zum Unterschied zwischen der Verwendung von Rollen und ressourcenbasierten Richtlinien für den kontenübergreifenden Zugriff finden Sie IAMim Benutzerhandbuch unter Kontoübergreifender Ressourcenzugriff. IAM

Amazon S3 Access Grants mit Amazon verwenden EMR

Übersicht über S3 Access Grants für Amazon EMR

Mit den EMR Amazon-Versionen 6.15.0 und höher bieten Amazon S3 Access Grants eine skalierbare Zugriffskontrolllösung, mit der Sie den Zugriff auf Ihre Amazon S3 S3-Daten von Amazon erweitern können. EMR Wenn Sie für Ihre S3-Daten eine komplexe oder umfangreiche Berechtigungskonfiguration haben, können Sie mit S3 Access Grants die S3-Datenberechtigungen für Benutzer, Gruppen, Rollen und Anwendungen in Ihrem Cluster skalieren.

Verwenden Sie S3 Access Grants, um den Zugriff auf Amazon S3 S3-Daten über die Berechtigungen hinaus zu erweitern, die von der Runtime-Rolle oder den IAM Rollen, die den Identitäten mit Zugriff

auf Ihren EMR Cluster zugewiesen sind, gewährt werden. Weitere Informationen finden Sie unter Verwalten des Zugriffs mit S3 Access Grants im Benutzerhandbuch zu Amazon S3.

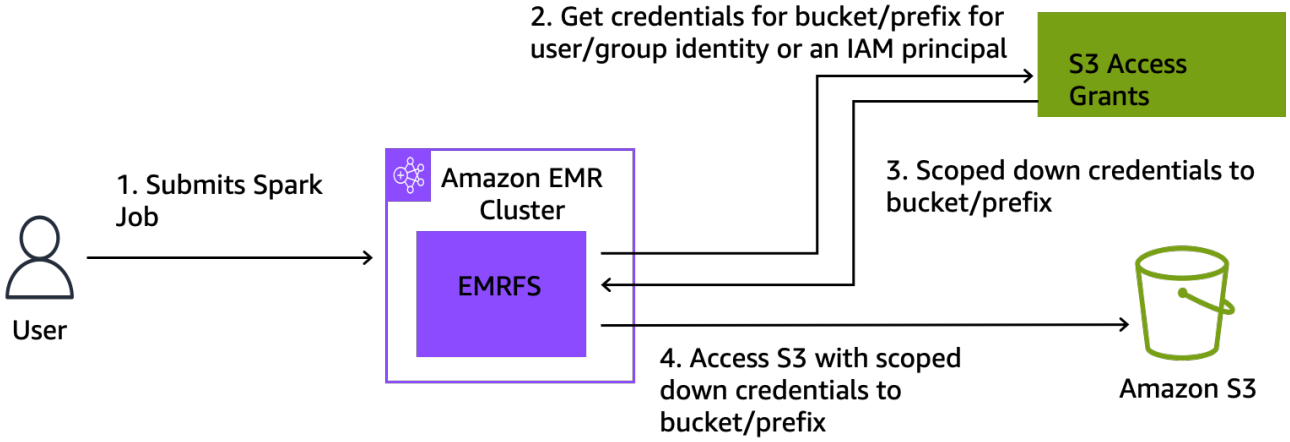
Schritte zur Verwendung von S3 Access Grants mit anderen EMR Amazon-Bereitstellungen finden Sie in der folgenden Dokumentation:

- Verwenden von S3 Access Grants mit Amazon EMR auf EKS
- Verwenden von S3 Access Grants mit Amazon EMR Serverless

So EMR arbeitet Amazon mit S3 Access Grants

EMRAmazon-Versionen 6.15.0 und höher bieten eine native Integration mit S3 Access Grants. Sie können S3 Access Grants bei Amazon aktivieren EMR und Spark-Jobs ausführen. Wenn ein Spark-Auftrag eine Anfrage für S3-Daten stellt, stellt Amazon S3 temporäre Anmeldeinformationen bereit, die auf den jeweiligen Bucket, das Präfix oder das Objekt beschränkt sind.

Im Folgenden finden Sie einen allgemeinen Überblick darüber, wie Amazon Zugriff auf Daten EMR erhält, die durch S3 Access Grants geschützt sind.



1. Ein Benutzer reicht einen Amazon EMR Spark-Job ein, der in Amazon S3 gespeicherte Daten verwendet.
2. Amazon EMR stellt eine Anfrage für S3 Access Grants, um im Namen dieses Benutzers Zugriff auf den Bucket, das Präfix oder das Objekt zu gewähren.
3. Amazon S3 gibt temporäre Anmeldeinformationen in Form eines AWS Security Token Service (STS) -Tokens für den Benutzer zurück. Das Token ist für den Zugriff auf den S3-Bucket, das S3-Präfix oder das S3-Objekt vorgesehen.
4. Amazon EMR verwendet das STS Token, um Daten von S3 abzurufen.

5. Amazon EMR empfängt die Daten von S3 und gibt die Ergebnisse an den Benutzer zurück.

Überlegungen zu S3 Access Grants mit Amazon EMR

Beachten Sie die folgenden Verhaltensweisen und Einschränkungen, wenn Sie S3 Access Grants mit Amazon verwendenEMR.

Feature-Unterstützung

- S3 Access Grants wird mit EMR Amazon-Versionen 6.15.0 und höher unterstützt.
- Spark ist die einzige unterstützte Abfrage-Engine, wenn Sie S3 Access Grants mit Amazon verwendenEMR.
- Delta Lake und Hudi sind die einzigen unterstützten Open-Table-Formate, wenn Sie S3 Access Grants mit Amazon verwenden. EMR
- Die folgenden EMR Amazon-Funktionen werden für die Verwendung mit S3 Access Grants nicht unterstützt:
  - Apache-Iceberg-Tabellen
  - LDAPnative Authentifizierung
  - Native Apache-Ranger-Authentifizierung
  - AWS CLI Anfragen an Amazon S3, die IAM Rollen verwenden
  - S3-Zugriff über das Open-Source-Protokoll S3A
- Die `fallbackToIAM` Option wird nicht für EMR Cluster unterstützt, die vertrauenswürdige Identitätsverbreitung mit IAM Identity Center verwenden.
- S3 Access Grants with AWS Lake Formation wird nur mit EMR Amazon-Clustern unterstützt, die auf Amazon ausgeführt EC2 werden.

Überlegungen in Bezug auf das Verhalten

- Die native Apache Ranger-Integration mit Amazon EMR bietet Funktionen, die mit S3 Access Grants als Teil des S3 Apache Ranger-Plug-ins EMRFS übereinstimmen. Wenn Sie Apache Ranger für eine differenzierte Zugriffskontrolle (FGAC) verwenden, empfehlen wir, dieses Plugin anstelle von S3 Access Grants zu verwenden.
- Amazon EMR stellt einen Anmeldedaten-Cache bereit, EMRFS um sicherzustellen, dass ein Benutzer innerhalb eines Spark-Jobs nicht wiederholt dieselben Anmeldeinformationen

anfordern muss. Daher fordert Amazon EMR immer das Standardrecht an, wenn es Anmeldeinformationen anfordert. Weitere Informationen finden Sie unter Zugriff auf S3-Daten anfordern im Benutzerhandbuch zu Amazon S3.

- Für den Fall, dass ein Benutzer eine Aktion ausführt, die S3 Access Grants nicht unterstützt, EMR ist Amazon so eingestellt, dass es die IAM Rolle verwendet, die für die Auftragsausführung angegeben wurde. Weitere Informationen finden Sie unter Greifen Sie auf Rollen IAM zurück.

Starten Sie einen EMR Amazon-Cluster mit S3 Access Grants

In diesem Abschnitt wird beschrieben, wie Sie einen EMR Cluster startenEC2, der auf Amazon läuft und S3 Access Grants verwendet, um den Zugriff auf Daten in Amazon S3 zu verwalten. Schritte zur Verwendung von S3 Access Grants mit anderen EMR Amazon-Bereitstellungen finden Sie in der folgenden Dokumentation:

- Verwenden von S3 Access Grants mit Amazon EMR auf EKS
- Verwenden von S3 Access Grants mit EMR Serverless

Gehen Sie wie folgt vor, um einen EMR Cluster zu startenEC2, der auf Amazon läuft und S3 Access Grants verwendet, um den Zugriff auf Daten in Amazon S3 zu verwalten.

1. Richten Sie eine Rolle zur Auftragsausführung für Ihren EMR Cluster ein. Geben Sie die erforderlichen IAM Berechtigungen an, die Sie für die Ausführung von Spark-Jobs benötigen, `s3:GetDataAccess` und`s3:GetAccessGrantsInstanceForPrefix`:

```
{
    "Effect": "Allow",
    "Action": [
    "s3:GetDataAccess",
    "s3:GetAccessGrantsInstanceForPrefix"
    ],
    "Resource": [     //LIST ALL INSTANCE ARNS THAT THE ROLE IS ALLOWED TO QUERY
         "arn:aws_partition:s3:Region:account-id1:access-grants/default",
         "arn:aws_partition:s3:Region:account-id2:access-grants/default"
    ]
}
```

> **Note**
>
> Bei Amazon EMR erweitern S3 Access Grants die Berechtigungen, die in IAM Rollen festgelegt sind. Wenn die IAM Rollen, die Sie für die Auftragsausführung angeben, Berechtigungen für den direkten Zugriff auf S3 enthalten, können Benutzer möglicherweise auf mehr Daten zugreifen als nur auf die Daten, die Sie in S3 Access Grants definieren.

2. Verwenden Sie als Nächstes die, AWS CLI um einen Cluster mit Amazon EMR 6.15 oder höher zu erstellen und die `emrfs-site` Klassifizierung, um S3 Access Grants zu aktivieren, ähnlich dem folgenden Beispiel:

```
aws emr create-cluster
  --release-label emr-6.15.0 \
  --instance-count 3 \
  --instance-type m5.xlarge \
  --configurations '[{"Classification":"emrfs-site",
 "Properties":{"fs.s3.s3AccessGrants.enabled":"true",
 "fs.s3.s3AccessGrants.fallbackToIAM":"false"}}]'
```

S3 Access Grants mit AWS Lake Formation

Wenn Sie Amazon EMR mit der AWS Lake Formation Integration verwenden, können Sie Amazon S3 Access Grants für direkten oder tabellarischen Zugriff auf Daten in Amazon S3 verwenden.

> **Note**
>
> S3 Access Grants with AWS Lake Formation wird nur mit EMR Amazon-Clustern unterstützt, die auf Amazon ausgeführt EC2 werden.

Direkter Zugriff

Der direkte Zugriff umfasst alle Aufrufe zum Zugriff auf S3-Daten, die nicht den Service API for the AWS Glue aufrufen, den Lake Formation als Metastore bei Amazon verwendetEMR, zum Beispiel, um Folgendes aufzurufen: `spark.read`

```
spark.read.csv("s3://...")
```

Wenn Sie S3 Access Grants bei AWS Lake Formation Amazon verwendenEMR, werden alle Direktzugriffsmuster über S3 Access Grants temporäre S3-Anmeldeinformationen abgerufen.

Tabellarischer Zugriff

Ein tabellarischer Zugriff erfolgt, wenn Lake Formation den Metastore aufruft, API um auf Ihren S3-Standort zuzugreifen, z. B. um Tabellendaten abzufragen:

```
spark.sql("select * from test_tbl")
```

Wenn Sie S3 Access Grants mit AWS Lake Formation auf Amazon verwendenEMR, werden alle tabellarischen Zugriffsmuster über Lake Formation abgewickelt.

Greifen Sie auf Rollen IAM zurück

Wenn ein Benutzer versucht, eine Aktion auszuführen, die S3 Access Grants nicht unterstützt, verwendet Amazon EMR standardmäßig die IAM Rolle, die bei der `fallbackToIAM` Konfiguration für die Auftragsausführung angegeben wurde. `true` Auf diese Weise können Benutzer in Szenarien, die S3 Access Grants nicht abdeckt, auf ihre Auftragsausführungsrolle zurückgreifen, um Anmeldeinformationen für den S3-Zugriff einzugeben.

Wenn `fallbackToIAM` aktiviert ist, können Benutzer auf die Daten zugreifen, die Access Grant zulässt. Wenn es kein S3 Access Grants-Token für die Zieldaten gibt, EMR prüft Amazon, ob die entsprechende Berechtigung für die Auftragsausführungsrolle vorliegt.

> **Note**
>
> Wir empfehlen Ihnen, Ihre Zugriffsberechtigungen bei aktivierter `fallbackToIAM`-Konfiguration zu testen, auch wenn Sie planen, die Option für Produktionsworkloads zu deaktivieren. Bei Spark-Jobs gibt es andere Möglichkeiten, wie Benutzer mit ihren IAM Anmeldeinformationen auf alle Berechtigungssätze zugreifen können. Wenn sie auf EMR Clustern aktiviert sind, gewähren S3-Zuschüsse Spark-Jobs Zugriff auf S3-Standorte. Sie sollten sicherstellen, dass Sie diese S3-Standorte vor Zugriffen von außen schützenEMRFS. Sie sollten die S3-Standorte beispielsweise vor dem Zugriff durch S3-Clients schützen, die in Notebooks verwendet werden, oder durch Anwendungen, die nicht von S3 Access Grants unterstützt werden, wie Hive oder Presto.

Authentifizieren Sie sich Amazon EMR Amazon-Cluster-Knoten

SSHKunden können ein EC2 Amazon-Schlüsselpaar verwenden, um sich bei Cluster-Instances zu authentifizieren. Alternativ können Sie mit EMR Amazon-Versionen 5.10.0 und höher Kerberos so konfigurieren, dass Benutzer und SSH Verbindungen zum primären Knoten authentifiziert werden. Und mit EMR Amazon-Versionen 5.12.0 und höher können Sie sich mit authentifizieren. LDAP

Themen

Verwenden Sie ein EC2 key pair für SSH Anmeldeinformationen

EMRAmazon-Clusterknoten werden auf EC2 Amazon-Instances ausgeführt. Sie können eine Verbindung zu Clusterknoten genauso herstellen wie zu EC2 Amazon-Instances. Sie können Amazon verwendenEC2, um ein key pair zu erstellen, oder Sie können ein key pair importieren. Wenn Sie einen Cluster erstellen, können Sie das EC2 Amazon-Schlüsselpaar angeben, das für SSH Verbindungen zu allen Cluster-Instances verwendet wird. Außerdem können Sie auch einen Cluster ohne ein Schlüsselpaar erstellen. Dies geschieht normalerweise mit vorübergehenden Clusters, die starten, gewisse Schritte ausführen und dann automatisch beendet werden.

Der SSH Client, den Sie für die Verbindung mit dem Cluster verwenden, muss die private Schlüsseldatei verwenden, die diesem key pair zugeordnet ist. Dies ist eine PEM-Datei für SSH Clients, die Linux, Unix und macOS verwenden. Sie müssen die Berechtigungen so festlegen, dass nur der Schlüsselbesitzer berechtigt ist, auf die Datei zuzugreifen. Dies ist eine PPK-Datei für SSH Clients, die Windows verwenden, und die PPK-Datei wird normalerweise aus der PEM-Datei erstellt.

- Weitere Informationen zum Erstellen eines EC2 Amazon-Schlüsselpaars finden Sie unter EC2Amazon-Schlüsselpaare im EC2Amazon-Benutzerhandbuch.
- Anweisungen zur Verwendung von P uTTYgen zum Erstellen einer .ppk-Datei aus einer .pem-Datei finden Sie unter Konvertieren Ihres privaten Schlüssels mit P uTTYgen im Amazon-Benutzerhandbuch. EC2
- Weitere Informationen zum Festlegen von Zugriffsberechtigungen für PEM-Dateien und zum Herstellen einer Verbindung mit dem Primärknoten eines EMR Clusters mithilfe verschiedener Methoden — einschließlich `ssh` Linux oder macOS, Pu unter Windows oder TTY von einem

beliebigen unterstützten Betriebssystem AWS CLI aus — finden Sie unter. Connect zum Primärknoten her mit SSH

Verwenden Sie Kerberos für die Authentifizierung bei Amazon EMR

EMRAmazon-Versionen 5.10.0 und höher unterstützen Kerberos. Kerberos ist ein Netzwerkauthentifizierungsprotokoll, das eine Verschlüsselung mit geheimen Schlüsseln verwendet, um eine starke Authentifizierung bereitzustellen, sodass Passwörter oder andere Anmeldeinformationen nicht in einem unverschlüsselten Format über das Netzwerk gesendet werden.

In Kerberos werden Services und Benutzer, die sich authentifizieren müssen, als Prinzipale bezeichnet. Prinzipale befinden sich in einem Kerberos-Bereich. Innerhalb des Bereichs bietet ein Kerberos-Server, der als Key Distribution Center (KDC) bezeichnet wird, den Prinzipalen die Möglichkeit, sich zu authentifizieren. Das KDC tut dies, indem Tickets für die Authentifizierung ausgestellt werden. Der KDC verwaltet eine Datenbank mit den Prinzipalen in seinem Bereich, ihren Passwörtern und anderen administrativen Informationen zu jedem Prinzipal. A KDC kann auch Authentifizierungsdaten von Prinzipalen in anderen Bereichen akzeptieren, was als realmübergreifende Vertrauensstellung bezeichnet wird. Darüber hinaus kann ein EMR Cluster ein KDC externes System zur Authentifizierung von Prinzipalen verwenden.

Ein gängiges Szenario für die Einrichtung einer realmübergreifenden Vertrauensstellung oder die Verwendung einer externen Vertrauensstellung KDC ist die Authentifizierung von Benutzern aus einer Active Directory-Domäne. Auf diese Weise können Benutzer mit ihrem Domänenkonto auf einen EMR Cluster zugreifen, wenn sie eine Verbindung SSH zu einem Cluster herstellen oder mit Big-Data-Anwendungen arbeiten.

Wenn Sie die Kerberos-Authentifizierung verwenden, EMR konfiguriert Amazon Kerberos für die Anwendungen, Komponenten und Subsysteme, die es auf dem Cluster installiert, sodass sie sich gegenseitig authentifizieren.

> **Important**
>
> Amazon unterstützt EMR nicht AWS Directory Service for Microsoft Active Directory in einem realmübergreifenden Trust oder als externes KDC Unternehmen.

Bevor Sie Kerberos mit Amazon konfigurieren, empfehlen wir IhnenEMR, sich mit den Kerberos-Konzepten, den Diensten, die auf einem ausgeführt werdenKDC, und den Tools zur Verwaltung

von Kerberos-Diensten vertraut zu machen. Weitere Informationen finden Sie in der MITKerberos-Dokumentation, die vom Kerberos-Konsortium veröffentlicht wurde.

Themen

Unterstützte Anwendungen

Innerhalb eines EMR Clusters sind Kerberos-Prinzipale die Big-Data-Anwendungsdienste und -Subsysteme, die auf allen Clusterknoten ausgeführt werden. Amazon EMR kann die unten aufgeführten Anwendungen und Komponenten für die Verwendung von Kerberos konfigurieren. Jeder Anwendung ist ein Kerberos-Benutzer-Prinzipal zugeordnet.

Amazon unterstützt EMR keine realmübergreifenden Vertrauensstellungen mit. AWS Directory Service for Microsoft Active Directory

Amazon konfiguriert EMR nur die Open-Source-Kerberos-Authentifizierungsfunktionen für die unten aufgeführten Anwendungen und Komponenten. Alle anderen installierten Anwendungen sind nicht durch Kerberos geschützt. Dies kann zu einer Unfähigkeit der Kommunikation mit durch Kerberos geschützten Komponenten führen und Anwendungsfehler verursachen. Für Anwendungen und Komponenten, die nicht durch Kerberos geschützt sind, ist keine Authentifizierung aktiviert. Die unterstützten Anwendungen und Komponenten können je nach Amazon EMR variieren.

Die Livy-Benutzeroberfläche ist die einzige Weboberfläche, die auf dem Kerberized Cluster gehostet wird.

- Hadoop MapReduce
- Hbase
- HCatalog
- HDFS

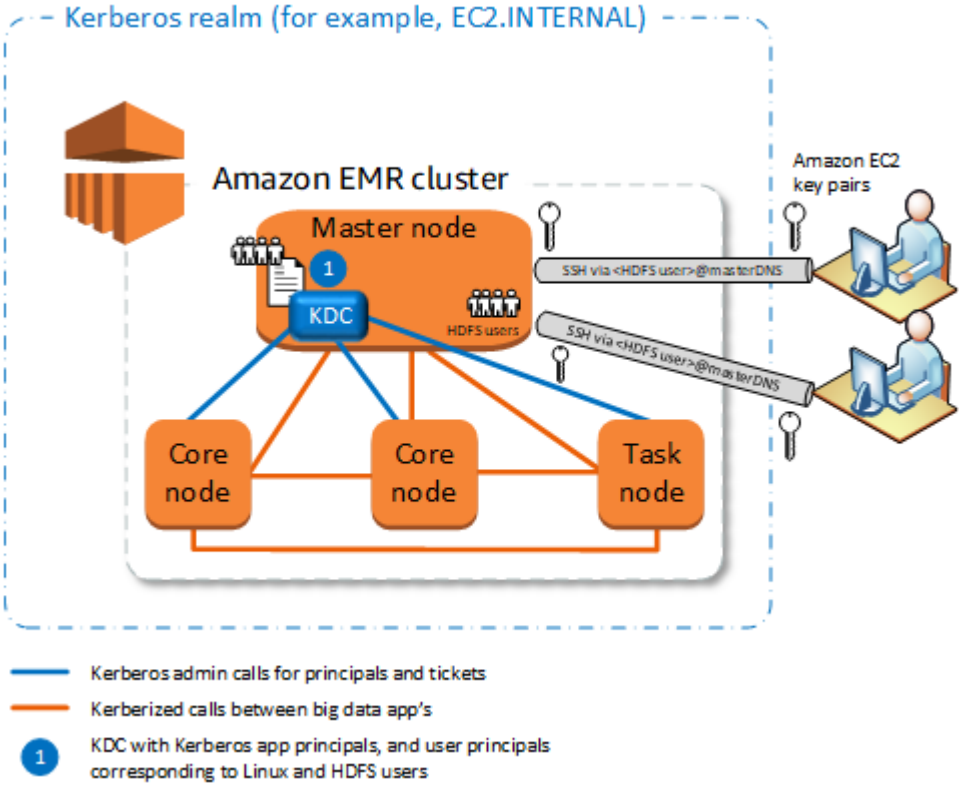
- Hive
  - Aktivieren Sie Hive nicht mit LDAP Authentifizierung. Dies kann zu Problemen bei der Kommunikation mit YARN Kerberized führen.
- Hue
  - Die Hue-Benutzerauthentifizierung wird nicht automatisch festgelegt und kann mithilfe der Konfiguration konfiguriert werden. API
  - Der Hue-Server ist durch Kerberos geschützt. Das Hue Front-End (UI) ist nicht für die Authentifizierung konfiguriert. LDAPDie Authentifizierung kann für die Hue-Benutzeroberfläche konfiguriert werden.
- Livy
  - Livy-Identitätswechsel mit kerberisierten Clustern wird in EMR Amazon-Versionen 5.22.0 und höher unterstützt.
- Oozie
- Phoenix
- Presto
  - Presto unterstützt die Kerberos-Authentifizierung in EMR Amazon-Versionen 6.9.0 und höher.
  - Um die Kerberos-Authentifizierung für Presto zu verwenden, müssen Sie die Verschlüsselung bei der Übertragung aktivieren.
- Spark
- Tez
- Trino
  - Trino unterstützt die Kerberos-Authentifizierung in EMR Amazon-Versionen 6.11.0 und höher.
  - Um die Kerberos-Authentifizierung für Trino zu verwenden, müssen Sie die Verschlüsselung bei der Übertragung aktivieren.
- YARN
- Zeppelin
  - Zeppelin ist nur mit dem Spark-Interpreter für die Verwendung von Kerberos konfiguriert. Es ist nicht für andere Interpreter konfiguriert.
  - Der Identitätswechsel von Benutzern wird für Kerberized-Zeppelin-Interpreter außer Spark nicht unterstützt.
- Zookeeper
  - Der Zookeeper-Client wird nicht unterstützt.

Kerberos-Architektur-Optionen

Wenn Sie Kerberos mit Amazon verwendenEMR, können Sie aus den in diesem Abschnitt aufgeführten Architekturen wählen. Unabhängig von der gewählten Architektur konfigurieren Sie Kerberos anhand der gleichen Schritte. Sie erstellen eine Sicherheitskonfiguration, geben die Sicherheitskonfiguration und die kompatiblen clusterspezifischen Kerberos-Optionen an, wenn Sie den Cluster erstellen, und Sie erstellen HDFS Verzeichnisse für Linux-Benutzer auf dem Cluster, die den Benutzerprinzipalen im entsprechen. KDC Weitere Informationen zu Konfigurationsoptionen und Beispielkonfigurationen für jede Architektur finden Sie unter Konfiguration von Kerberos auf Amazon EMR.

Clusterspezifisch KDC (auf dem primären Knoten) KDC

Diese Konfiguration ist mit EMR Amazon-Versionen 5.10.0 und höher verfügbar.



Vorteile

- Amazon EMR hat das volle Eigentum an derKDC.
- Das KDC auf dem EMR Cluster ist unabhängig von zentralisierten KDC Implementierungen wie Microsoft Active Directory oder AWS Managed Microsoft AD.

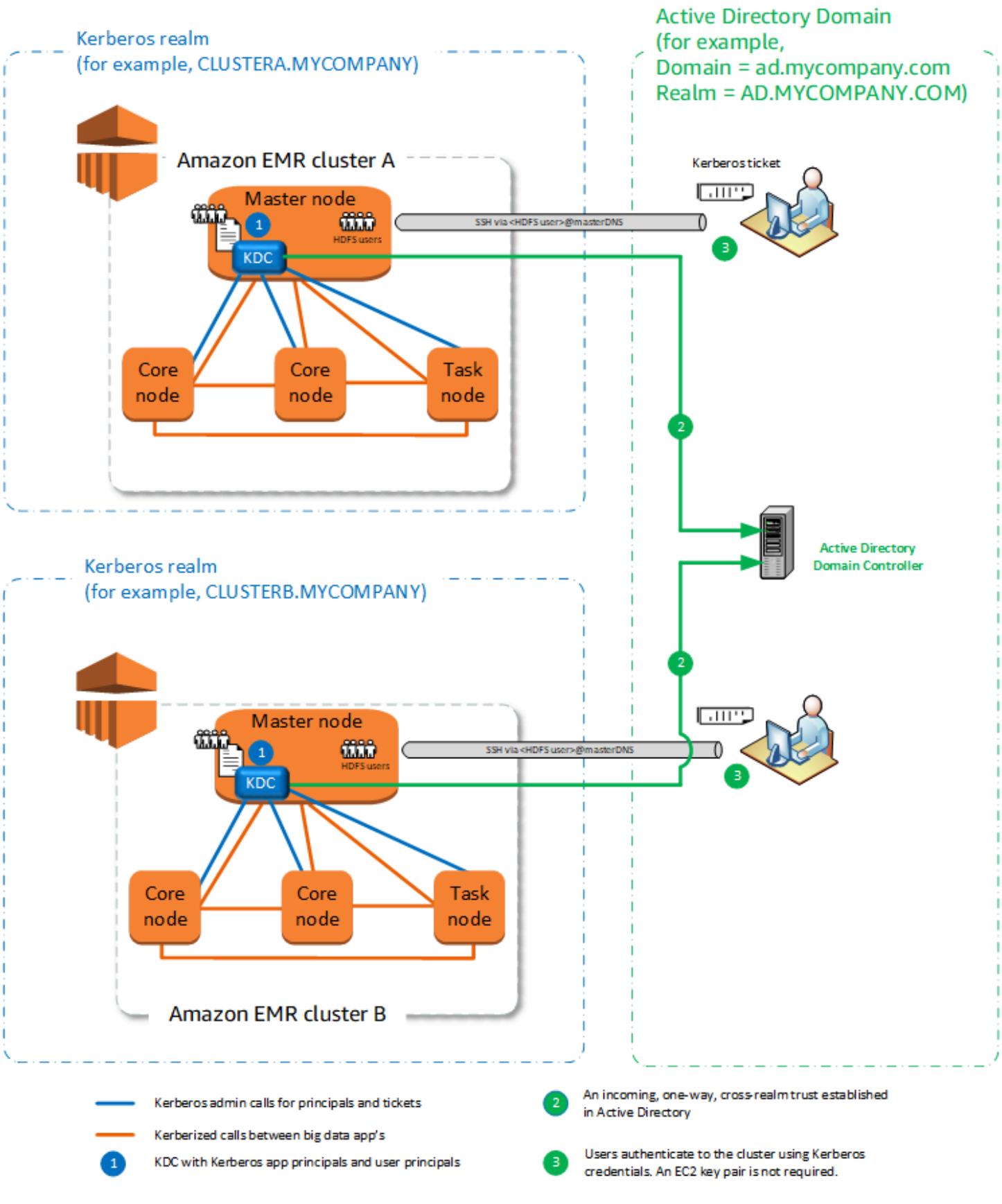
- Die Auswirkungen auf die Leistung sind minimal, da die Authentifizierung nur für lokale Knoten innerhalb des Clusters KDC verwaltet wird.
- Optional können andere Kerber-Cluster auf den KDC als externe Cluster verweisen. KDC Weitere Informationen finden Sie unter Extern KDC — primärer Knoten auf einem anderen Cluster.

Überlegungen und Einschränkungen

- Kerberos-Cluster können einander nicht authentifizieren, sodass für die Anwendungen keine Interoperabilität besteht. Wenn Clusteranwendungen zusammenarbeiten müssen, müssen Sie eine realmübergreifende Vertrauensstellung zwischen Clustern einrichten oder einen Cluster als externen KDC Cluster für andere Cluster einrichten. Wenn eine realmübergreifende Vertrauensstellung eingerichtet ist, KDCs müssen sie über unterschiedliche Kerberos-Bereiche verfügen.
- Sie müssen Linux-Benutzer auf der EC2 Instanz des primären Knotens erstellen, die den KDC Benutzerprinzipalen entsprechen, zusammen mit den HDFS Verzeichnissen für jeden Benutzer.
- Benutzerprinzipale müssen eine EC2 private Schlüsseldatei und `kinit` Anmeldeinformationen verwenden, um eine Verbindung zum Cluster herzustellen. SSH

Bereichsübergreifende Vertrauensstellung

In dieser Konfiguration authentifizieren sich Prinzipale (normalerweise Benutzer) aus einem anderen Kerberos-Bereich bei Anwendungskomponenten auf einem Kerberisierten Cluster, der über einen eigenen verfügtEMR. KDC Der Knoten KDC auf dem Primärknoten stellt KDC mithilfe eines realmübergreifenden Prinzipals, der in beiden vorhanden ist, eine Vertrauensbeziehung zu einem anderen Knoten her. KDCs Der Prinzipalname und das Passwort stimmen jeweils KDC exakt überein. Bereichsübergreifende Vertrauensstellungen kommen am häufigsten in Active Directory-Implementierungen vor, wie in der folgenden Abbildung dargestellt. Realmübergreifende Vertrauensstellungen mit einem externen MIT KDC oder einem KDC anderen EMR Amazon-Cluster werden ebenfalls unterstützt.


Active Directory Domain
(for example,
Domain = ad.mycompany.com
Realm = AD.MYCOMPANY.COM)
Kerberos realm
(for example, CLUSTERA.MYCOMPANY)
Amazon EMR cluster A
Master node
1
KDC
HDFS users
SSH via <HDFS user>@masterDNS
Kerberos ticket
3
Core node
Core node
Task node
2
Active Directory Domain Controller
Kerberos realm
(for example, CLUSTERB.MYCOMPANY)
Master node
1
KDC
HDFS users
SSH via <HDFS user>@masterDNS
2
3
Core node
Core node
Task node
Amazon EMR cluster B
Kerberos admin calls for principals and tickets
Kerberized calls between big data app's
1
KDC with Kerberos app principals and user principals
2
An incoming, one-way, cross-realm trust established in Active Directory
3
Users authenticate to the cluster using Kerberos credentials. An EC2 key pair is not required.

Vorteile

- Der EMR Cluster, auf dem der installiert KDC ist, behält den vollen Besitz von. KDC
- Mit Active Directory erstellt Amazon EMR automatisch Linux-Benutzer, die Benutzerprinzipalen aus dem KDC entsprechen. Sie müssen weiterhin HDFS Verzeichnisse für jeden Benutzer erstellen. Darüber hinaus können Benutzerprinzipale in der Active Directory-Domäne mithilfe von `kinit` Anmeldeinformationen ohne die EC2 private Schlüsseldatei auf kerberisierte Cluster zugreifen. Dies beseitigt die Notwendigkeit, die Datei mit dem privaten Schlüssel für die Cluster-Benutzer freizugeben.
- Da jeder Cluster die Authentifizierung für die Knoten im Cluster KDC verwaltet, werden die Auswirkungen der Netzwerklatenz und des Verarbeitungsaufwands für eine große Anzahl von Knoten in Clustern minimiert.

Überlegungen und Einschränkungen

- Wenn Sie eine Vertrauensstellung mit einem Active-Directory-Bereich einrichten, müssen Sie beim Erstellen des Clusters Active Directory-Benutzername und -Passwort mit Berechtigungen zum Hinzufügen von Prinzipalen zur Domain angeben.
- Bereichsübergreifende Vertrauensstellungen können nicht zwischen Kerberos-Bereichen mit demselben Namen eingerichtet werden.
- Bereichsübergreifende Vertrauensstellungen müssen explizit eingerichtet werden. Wenn Cluster A und Cluster B beispielsweise beide eine realmübergreifende Vertrauensstellung mit a einrichtenKDC, vertrauen sie einander nicht grundsätzlich, und ihre Anwendungen können sich nicht gegenseitig authentifizieren, um zusammenzuarbeiten.
- KDCsmüssen unabhängig verwaltet und koordiniert werden, sodass die Anmeldeinformationen der Benutzerprinzipale exakt übereinstimmen.

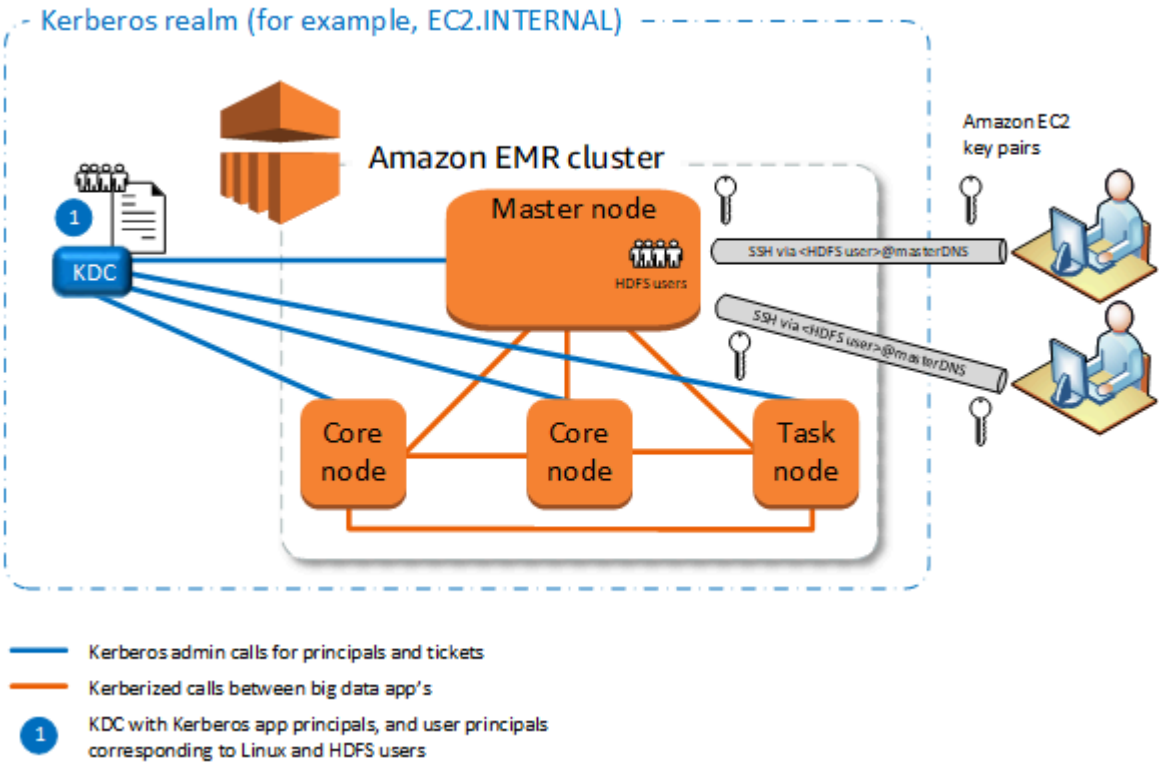
Extern KDC

Konfigurationen mit einem externen System KDC werden mit Amazon EMR 5.20.0 und höher unterstützt.

- Extern — KDC MIT KDC
- Extern KDC — primärer Knoten auf einem anderen Cluster
- Extern KDC — Cluster KDC auf einem anderen Cluster mit realmübergreifender Active Directory-Vertrauensstellung

Extern — KDC MIT KDC

Diese Konfiguration ermöglicht es einem oder mehreren EMR Clustern, Prinzipale zu verwenden, die in einem MIT KDC Server definiert und verwaltet werden.



Vorteile

- Die Verwaltung von Prinzipalen ist in einem einzigen System zusammengefasst. KDC
- Mehrere Cluster können dasselbe KDC in demselben Kerberos-Bereich verwenden. Weitere Informationen finden Sie unter Anforderungen für die Verwendung mehrerer Cluster mit demselben KDC.
- Der primäre Knoten in einem kerberisierten Cluster hat nicht die Leistungsbelastung, die mit der Wartung von verbunden ist. KDC

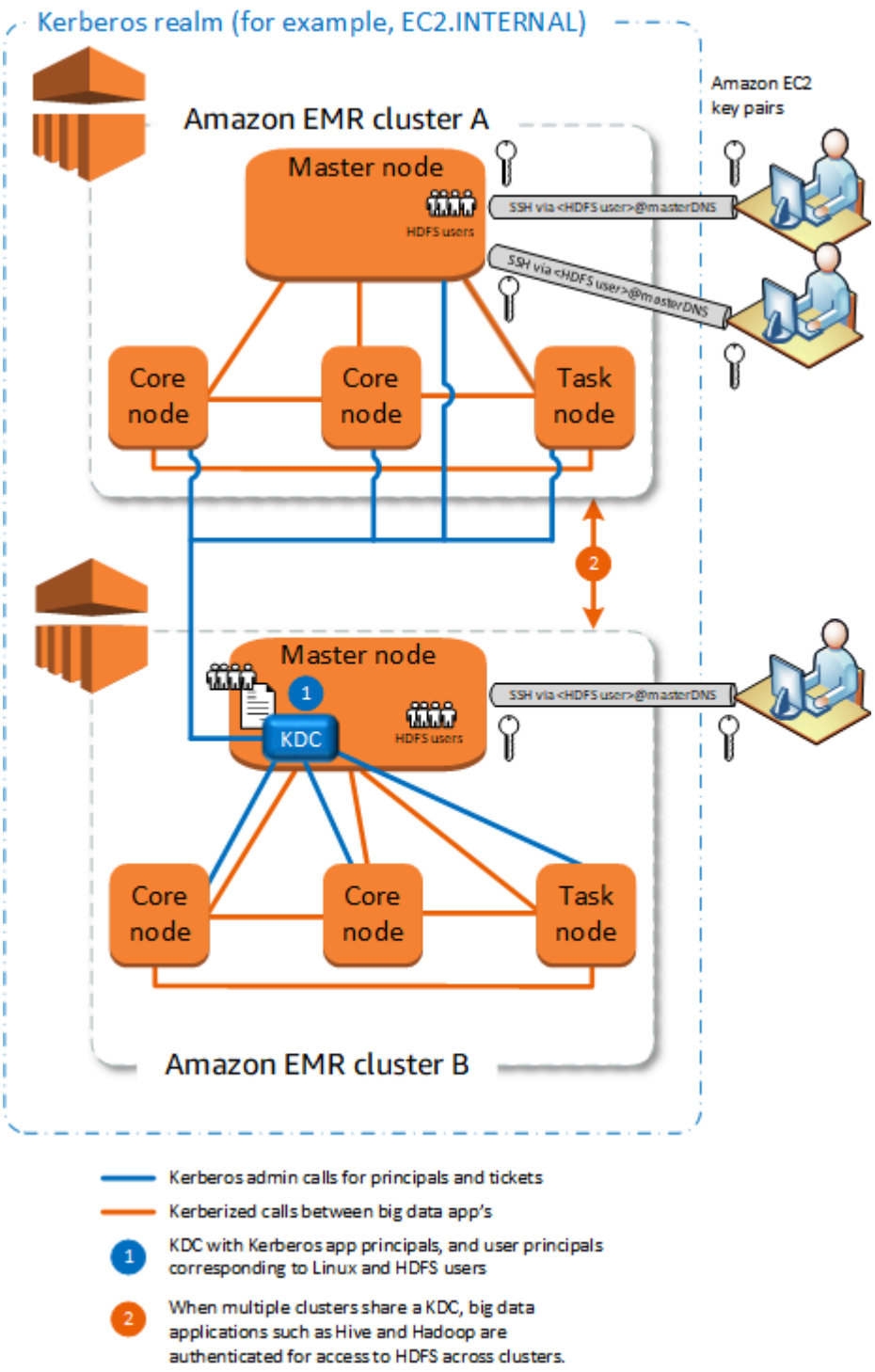
Überlegungen und Einschränkungen

- Sie müssen auf der EC2 Instanz des primären Knotens jedes Kerberized-Clusters Linux-Benutzer erstellen, die den KDC Benutzerprinzipalen entsprechen, zusammen mit den HDFS Verzeichnissen für jeden Benutzer.

- Benutzerprinzipale müssen eine EC2 private Schlüsseldatei und `kinit` Anmeldeinformationen verwenden, um eine Verbindung zu Kerberized-Clustern herzustellen. SSH
- Jeder Knoten in kerberisierten EMR Clustern muss über eine Netzwerkroute zum verfügen. KDC
- Jeder Knoten in kerberisierten Clustern belastet den externen Knoten durch die AuthentifizierungKDC, sodass sich die Konfiguration des Clusters auf die Clusterleistung auswirkt. KDC Wenn Sie die Hardware des KDC Servers konfigurieren, sollten Sie die maximale Anzahl von EMR Amazon-Knoten berücksichtigen, die gleichzeitig unterstützt werden sollen.
- Die Clusterleistung hängt von der Netzwerklatenz zwischen Knoten in kerberisierten Clustern und dem ab. KDC
- Die Fehlerbehebung kann sich aufgrund von Abhängigkeiten untereinander schwieriger gestalten.

Extern KDC — primärer Knoten auf einem anderen Cluster

Diese Konfiguration ist fast identisch mit der obigen externen MIT KDC Implementierung, mit der Ausnahme, dass sie KDC sich auf dem primären Knoten eines EMR Clusters befindet. Weitere Informationen erhalten Sie unter Clusterspezifisch KDC (auf dem primären Knoten) KDC und Tutorial: Konfigurieren einer bereichsübergreifenden Vertrauensstellung mit einer Active-Directory-Domain.



Vorteile

- Die Verwaltung von Principals ist in einem einzigen KDC System zusammengefasst.

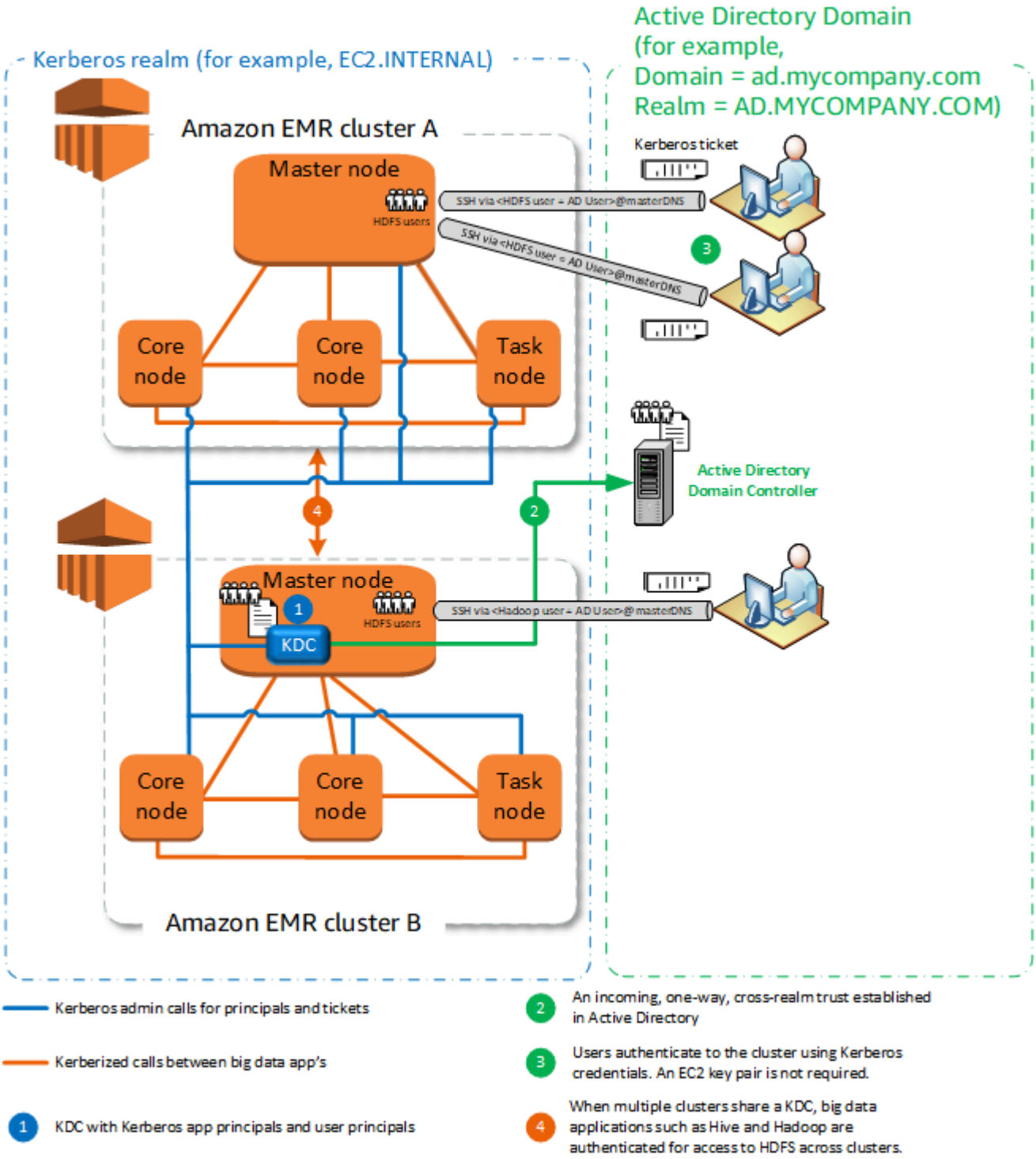
- Mehrere Cluster können dasselbe KDC in demselben Kerberos-Bereich verwenden. Weitere Informationen finden Sie unter Anforderungen für die Verwendung mehrerer Cluster mit demselben KDC.

Überlegungen und Einschränkungen

- Sie müssen Linux-Benutzer auf der EC2 Instanz des primären Knotens jedes Kerberized-Clusters erstellen, die den KDC Benutzerprinzipalen entsprechen, zusammen mit den HDFS Verzeichnissen für jeden Benutzer.
- Benutzerprinzipale müssen eine EC2 private Schlüsseldatei und `kinit` Anmeldeinformationen verwenden, um eine Verbindung zu Kerberized-Clustern herzustellen. SSH
- Jeder Knoten in jedem EMR Cluster muss über eine Netzwerkroute zum verfügen. KDC
- Jeder EMR Amazon-Knoten in kerberisierten Clustern belastet den externen Knoten mit einer AuthentifizierungslastKDC, sodass sich die Konfiguration des Clusters auf die KDC Cluster-Leistung auswirkt. Wenn Sie die Hardware des KDC Servers konfigurieren, sollten Sie die maximale Anzahl von EMR Amazon-Knoten berücksichtigen, die gleichzeitig unterstützt werden sollen.
- Die Cluster-Leistung hängt von der Netzwerklatenz zwischen den Knoten in den Clustern und dem abKDC.
- Die Fehlerbehebung kann sich aufgrund von Abhängigkeiten untereinander schwieriger gestalten.

Extern KDC — Cluster KDC auf einem anderen Cluster mit realmübergreifender Active Directory-Vertrauensstellung

In dieser Konfiguration erstellen Sie zunächst einen Cluster mit einem dedizierten Cluster, der über eine unidirektionale KDC realmübergreifende Vertrauensstellung mit Active Directory verfügt. Ein detailliertes Tutorial finden Sie unter Tutorial: Konfigurieren einer bereichsübergreifenden Vertrauensstellung mit einer Active-Directory-Domain. Anschließend starten Sie weitere Cluster und verweisen dabei auf den Cluster, dem die Vertrauensstellung zugewiesen wurdeKDC, als externen Cluster. KDC Ein Beispiel finden Sie unter Externer Cluster KDC mit realmübergreifender Active Directory-Vertrauensstellung. Auf diese Weise kann jeder EMR Amazon-Cluster, der das Externe verwendet, KDC um die in einer Microsoft Active Directory-Domäne definierten und verwalteten Prinzipale zu authentifizieren.



Vorteile

- Die Verwaltung von Prinzipalen ist in der Active-Directory-Domain zusammengefasst.

- Amazon EMR tritt dem Active Directory-Bereich bei, sodass keine Linux-Benutzer erstellt werden müssen, die Active Directory-Benutzern entsprechen. Sie müssen weiterhin HDFS Verzeichnisse für jeden Benutzer erstellen.
- Mehrere Cluster können dasselbe KDC in demselben Kerberos-Bereich verwenden. Weitere Informationen finden Sie unter Anforderungen für die Verwendung mehrerer Cluster mit demselben KDC.
- Benutzerprinzipale in der Active Directory-Domäne können mithilfe von `kinit` Anmeldeinformationen ohne die private Schlüsseldatei auf kerberisierte Cluster zugreifen. EC2 Dies beseitigt die Notwendigkeit, die Datei mit dem privaten Schlüssel für die Cluster-Benutzer freizugeben.
- Nur ein EMR Amazon-Primärknoten ist für die Wartung des verantwortlichKDC, und nur dieser Cluster muss mit Active Directory-Anmeldeinformationen für die realmübergreifende Vertrauensstellung zwischen dem KDC und Active Directory erstellt werden.

Überlegungen und Einschränkungen

- Jeder Knoten in jedem EMR Cluster muss über eine Netzwerkroute zum KDC und zum Active Directory-Domänencontroller verfügen.
- Jeder EMR Amazon-Knoten belastet den externen Knoten mit einer AuthentifizierungslastKDC, sodass sich die Konfiguration des Clusters KDC auf die Cluster-Leistung auswirkt. Wenn Sie die Hardware des KDC Servers konfigurieren, sollten Sie die maximale Anzahl von EMR Amazon-Knoten berücksichtigen, die gleichzeitig unterstützt werden sollen.
- Die Cluster-Leistung hängt von der Netzwerklatenz zwischen den Knoten in den Clustern und dem KDC Server ab.
- Die Fehlerbehebung kann sich aufgrund von Abhängigkeiten untereinander schwieriger gestalten.

Anforderungen für die Verwendung mehrerer Cluster mit demselben KDC

Mehrere Cluster können dasselbe KDC in demselben Kerberos-Bereich verwenden. Wenn die Cluster jedoch gleichzeitig ausgeführt werden, schlagen die Cluster möglicherweise fehl, wenn sie ServicePrincipal Kerberos-Namen verwenden, die zu Konflikten führen.

Wenn Sie mehrere gleichzeitige Cluster mit demselben externen Cluster haben, stellen Sie sicherKDC, dass die Cluster unterschiedliche Kerberos-Bereiche verwenden. Wenn die Cluster denselben Kerberos-Bereich verwenden müssen, stellen Sie sicher, dass sich die Cluster in unterschiedlichen Subnetzen befinden und dass sich ihre Bereiche nicht überschneiden. CIDR

Konfiguration von Kerberos auf Amazon EMR

Dieser Abschnitt enthält Konfigurationsdetails und Beispiele für das Einrichten von Kerberos mit gängigen Architekturen. Unabhängig von der gewählten Architektur sind die Konfigurationsgrundlagen identisch und in drei Schritte unterteilt. Wenn Sie eine externe Verbindung verwenden KDC oder eine realmübergreifende Vertrauensstellung einrichten, müssen Sie sicherstellen, dass jeder Knoten in einem Cluster über eine Netzwerkroute nach außen verfügt. Dazu gehört auch die Konfiguration der entsprechenden SicherheitsgruppenKDC, um eingehenden und ausgehenden Kerberos-Verkehr zuzulassen.

Schritt 1: Eine Sicherheitskonfiguration mit Kerberos-Eigenschaften erstellen

Die Sicherheitskonfiguration spezifiziert Details zu Kerberos und ermöglicht die KDC Wiederverwendung der Kerberos-Konfiguration bei jeder Clustererstellung. Sie können eine Sicherheitskonfiguration mit der EMR Amazon-Konsole AWS CLI, dem oder dem erstellen EMRAPI. Die Sicherheitskonfiguration kann auch andere Sicherheitsoptionen enthalten, wie beispielsweise die Verschlüsselung. Weitere Informationen zum Erstellen von Sicherheitskonfigurationen und Festlegen einer Sicherheitskonfiguration beim Erstellen eines Clusters finden Sie unter Sicherheitskonfigurationen zum Einrichten der Cluster-Sicherheit verwenden. Informationen zu Kerberos-Eigenschaften in einer Sicherheitskonfiguration finden Sie unter Kerberos-Einstellungen für Sicherheitskonfigurationen.

Schritt 2: Einen Cluster erstellen und Cluster-spezifische Kerberos-Attribute festlegen

Beim Erstellen eines Clusters legen Sie eine Kerberos-Sicherheitskonfiguration sowie Cluster-spezifische Kerberos-Optionen fest. Wenn Sie die EMR Amazon-Konsole verwenden, sind nur die Kerberos-Optionen verfügbar, die mit der angegebenen Sicherheitskonfiguration kompatibel sind. Wenn Sie Amazon AWS CLI oder Amazon verwenden, stellen Sie sicher EMRAPI, dass Sie Kerberos-Optionen angeben, die mit der angegebenen Sicherheitskonfiguration kompatibel sind. Wenn Sie beispielsweise bei der Erstellung eines Clusters mithilfe von ein Prinzipalkennwort für eine realmübergreifende Vertrauensstellung angeben und die CLI angegebene Sicherheitskonfiguration nicht mit realmübergreifenden Vertrauensparametern konfiguriert ist, tritt ein Fehler auf. Weitere Informationen finden Sie unter Kerberos-Einstellungen für Cluster.

Schritt 3: Den Cluster-Primärknoten konfigurieren

Abhängig von den Anforderungen an Ihre Architektur und Implementierung ist möglicherweise eine zusätzliche Einrichtung auf dem Cluster erforderlich. Sie können dies nach dem Erstellen oder anhand der Schritte oder Bootstrap-Aktionen während des Erstellungsvorgangs erledigen.

Für jeden Kerberos-authentifizierten Benutzer, der über Kerberos eine Verbindung zum Cluster herstellt, müssen Sie sicherstellenSSH, dass Linux-Konten erstellt werden, die dem Kerberos-Benutzer entsprechen. Wenn Benutzerprinzipale von einem Active Directory-Domänencontroller entweder als externer KDC oder über eine realmübergreifende Vertrauensstellung bereitgestellt werden, EMR erstellt Amazon automatisch Linux-Konten. Wenn Active Directory nicht verwendet wird, müssen Sie Prinzipale für jeden Benutzer erstellen, der ihrem Linux-Benutzer entspricht. Weitere Informationen finden Sie unter Konfiguration eines Clusters für Kerberos-authentifizierte Benutzer HDFS und Verbindungen SSH.

Jeder Benutzer muss außerdem über ein HDFS Benutzerverzeichnis verfügen, das er besitzt und das Sie erstellen müssen. Außerdem SSH muss es so konfiguriert sein, dass es GSSAPI aktiviert ist, um Verbindungen von Kerberos-authentifizierten Benutzern zuzulassen. GSSAPImuss auf dem Primärknoten aktiviert sein, und die SSH Client-Anwendung muss für die Verwendung konfiguriert sein. GSSAPI Weitere Informationen finden Sie unter Konfiguration eines Clusters für Kerberos-authentifizierte Benutzer HDFS und Verbindungen SSH.

Sicherheitskonfiguration und Cluster-Einstellungen für Kerberos auf Amazon EMR

Wenn Sie einen durch Kerberos geschützten Cluster erstellen, geben Sie die Sicherheitskonfiguration zusammen mit den Kerberos-Attributen an, die spezifisch für den Cluster sind. Sie können eine Gruppe nicht ohne die andere angeben, sonst tritt ein Fehler auf.

Dieses Thema bietet eine Übersicht über die für Kerberos verfügbaren Konfigurationsparameter, wenn Sie eine Sicherheitskonfiguration und einen Cluster erstellen. Darüber hinaus werden CLI Beispiele für die Erstellung kompatibler Sicherheitskonfigurationen und Cluster für gängige Architekturen bereitgestellt.

Kerberos-Einstellungen für Sicherheitskonfigurationen

Sie können mit der EMR Amazon-Konsole, dem oder dem eine Sicherheitskonfiguration erstellen AWS CLI, die Kerberos-Attribute spezifiziert. EMR API Die Sicherheitskonfiguration kann auch andere Sicherheitsoptionen enthalten, wie beispielsweise die Verschlüsselung. Weitere Informationen finden Sie unter Eine Sicherheitskonfiguration erstellen.

Verwenden Sie die folgenden Referenzen, um die verfügbaren Sicherheitskonfigurationseinstellungen für die Kerberos-Architektur zu verstehen, die Sie auswählen. Die EMR Amazon-Konsoleneinstellungen werden angezeigt. Die entsprechenden CLI Optionen finden Sie unter Angeben von Kerberos-Einstellungen mithilfe von AWS CLI oderBeispiele für Konfigurationen.

Parameter		Beschreibung
Kerberos		Gibt an, dass Kerberos für Cluster aktiviert ist, die diese Sicherheitskonfiguration verwenden. Wenn ein Cluster diese Sicherheitskonfiguration verwendet, müssen für den Cluster auch Kerberos-Einstellungen angegeben sein, andernfalls tritt ein Fehler auf.
Anbieter	Cluster-spezifisch KDC	Gibt an, dass Amazon KDC auf dem primären Knoten eines Clusters, der diese Sicherheitskonfiguration verwendet, eine EMR erstellt. Sie geben den Realm-Namen und das KDC Admin-Passwort an, wenn Sie den Cluster erstellen. Sie können bei Bedarf KDC von anderen Clustern aus darauf verweisen. Erstellen Sie diese Cluster mit einer anderen Sicherheitskonfiguration, geben Sie eine externe KDC Konfiguration an und verwenden Sie den Bereichsnamen und das KDC Administratorkennwort, die Sie für den dedizierten Cluster angeben. KDC
	Extern KDC	Nur mit Amazon EMR 5.20.0 und höher verfügbar. Gibt an, dass Cluster, die diese Sicherheitskonfigu ration verwenden, Kerberos-Prinzipale mithilfe eines Servers außerhalb des Clusters authentifizieren. KDC A KDC wird auf dem Cluster nicht erstellt. Wenn Sie den Cluster erstellen, geben Sie den Bereichsnamen und das KDC Administratorkennwort für den externen Cluster anKDC.
Gültigkeitsdauer des Tickets		Optional. Gibt den Zeitraum an, für den ein von der ausgestelltes Kerberos-Ticket auf Clustern gültig KDC ist, die diese Sicherheitskonfiguration verwenden. Ticket-Gültigkeitsdauern werden aus Sicherhei tsgründen beschränkt. Cluster-Anwendungen und Services verlängern Tickets automatisch, wenn sie ablaufen. Benutzer, die SSH mithilfe von Kerberos-

Parameter		Beschreibung
		Anmeldeinformationen eine Verbindung zum Cluster herstellen, müssen von der Befehlszeile des primären Knotens `kinit` aus starten, um das Ticket zu verlängern, nachdem ein Ticket abgelaufen ist.
Bereichsübergreifende Vertrauensstellung		Gibt eine bereichsübergreifende Vertrauensstellung zwischen einem Cluster, der ausschließlich Clustern zugeordnet KDC ist, die diese Sicherheitskonfiguration verwenden, und einem Cluster KDC in einem anderen Kerberos-Bereich an. Prinzipale (in der Regel Benutzer) aus einem anderen Bereich werden gegenüber Clustern authentifiziert, die diese Konfiguration verwenden. Eine zusätzliche Konfiguration im anderen Kerberos-Bereich ist erforderlich. Weitere Informationen finden Sie unter Tutorial: Konfigurieren einer bereichsübergreifenden Vertrauensstellung mit einer Active-Directory-Domain.
Realitätsübergreifende Vertrauenseigenschaften	Bereich	Gibt den Kerberos-Bereichsnamen des anderen Bereichs in der Vertrauensstellung an. Gemäß der Konvention sind Kerberos-Bereichsnamen mit dem Domainnamen identisch, jedoch ausschließlich in Großbuchstaben.
	Domain	Gibt den Domain-Namen des anderen Bereichs in der Vertrauensstellung an.

Parameter		Beschreibung
	Admin-Server	Gibt den vollqualifizierten Domänennamen (FQDN) oder die IP-Adresse des Admin-Servers im anderen Bereich der Vertrauensstellung an. Der Admin-Server und der KDC Server laufen in der Regel auf demselben Computer mit demselbenFQDN, kommunizieren aber über unterschiedliche Ports. Falls kein Port angegeben ist, wird Port 749 verwendet , da es dabei um den Kerberos-Standard handelt. Sie können optional einen Port angeben (beispielsweise `domain.example.com :749`).
	KDCServer	Gibt den vollqualifizierten Domänennamen (FQDN) oder die IP-Adresse des KDC Servers im anderen Bereich der Vertrauensstellung an. Der KDC Server und der Admin-Server laufen normalerweise auf demselben Computer mit denselbenFQDN, verwenden jedoch unterschiedliche Ports. Falls kein Port angegeben ist, wird Port 88 verwendet , da es dabei um den Kerberos-Standard handelt. Sie können optional einen Port angeben (beispielsweise `domain.example.com :88`).
Extern KDC		Gibt an, dass externe KDC Cluster vom Cluster verwendet werden.

Parameter			Beschreibung
Externe KDC Eigenschaften	Admin-Server		Gibt den vollqualifizierten Domänennamen (FQDN) oder die IP-Adresse des externen Admin-Servers an. Der Admin-Server und der KDC Server laufen in der Regel auf demselben Computer mit denselben AnschlüssenFQDN, kommunizieren aber über unterschiedliche Ports. Falls kein Port angegeben ist, wird Port 749 verwendet, da es dabei um den Kerberos-Standard handelt. Sie können optional einen Port angeben (beispielsweise `domain.example.com :749`).
	KDCServer		Gibt den vollqualifizierten Domänennamen (FQDN) des externen KDC Servers an. Der KDC Server und der Admin-Server laufen normalerweise auf demselben Computer mit denselbenFQDN, verwenden jedoch unterschiedliche Ports. Falls kein Port angegeben ist, wird Port 88 verwendet, da es dabei um den Kerberos-Standard handelt. Sie können optional einen Port angeben (beispielsweise `domain.example.com :88`).
	Active-Directory-Integration		Gibt an, dass die Kerberos-Prinzipalauthentifizierung in eine Microsoft-Active-Directory-Domain integriert ist.
	Active-Directory-Integrationseigenschaften	Active-Directory-Bereich	Gibt den Kerberos-Bereichsnamen der Active-Directory-Domain an. Gemäß der Konvention sind Kerberos-Bereichsnamen in der Regel identisch mit dem Domainnamen, jedoch ausschließlich in Großbuchstaben.
		Active-Directory-Domain	Gibt den Active-Directory-Domainnamen an.

Parameter			Beschreibung
		Active-Directory-Server	Gibt den vollqualifizierten Domänennamen (FQDN) des Microsoft Active Directory-Domänencontrollers an.

Kerberos-Einstellungen für Cluster

Sie können Kerberos-Einstellungen angeben, wenn Sie einen Cluster mit der EMR Amazon-Konsole AWS CLI, dem oder dem erstellen. EMR API

Verwenden Sie die folgenden Referenzen, um die verfügbaren Clusterkonfigurationseinstellungen für die Kerberos-Architektur zu verstehen, die Sie auswählen. Die EMR Amazon-Konsoleneinstellungen werden angezeigt. Die entsprechenden CLI Optionen finden Sie unterBeispiele für Konfigurationen.

Parameter	Beschreibung
Bereich	Der Kerberos-Bereichsname für den Cluster. Die Kerberos-Konvention ist, denselben Namen wie den Domain-Namen zu verwenden, aber in Großbuchstaben. Beispielsweise für die Domain `ec2.internal` mit `EC2.INTERNAL` als Bereichsnamen.
KDCAdmin-Passwort	Das im Cluster verwendete Passwort für `kadmin` oder `kadmin.local` . Dabei handelt es sich um Befehlszeilen-Schnittstellen zum Kerberos V5-Verwaltungssystem, das Kerberos-Prinzipale, Passwortrichtlinien und Keytabs für den Cluster verwaltet.
Prinzipal-Passwort für bereichsübergreifende Vertrauensstellungen (optional)	Erforderlich, wenn eine bereichsübergreifende Vertrauensstellung eingerichtet wird. Das Passwort für die bereichsübergreifende Vertrauensstellung, die über alle Bereiche

Parameter	Beschreibung
	hinweg identisch sein muss. Verwenden Sie ein sicheres Passwort.
Benutzer für die Verbindung mit der Active-Directory-Domain (optional)	Erforderlich bei Verwendung von Active Directory in einer bereichsübergreifenden Vertrauensstellung. Dies ist der Benutzeranmeldename eines Active-Directory-Kontos mit der Berechtigung, der Domain Computer hinzuzufügen. Amazon EMR verwendet diese Identität, um den Cluster mit der Domain zu verbinden. Weitere Informationen finden Sie unter the section called “Schritt 3: Fügen Sie der Domäne für den EMR Cluster Konten hinzu”.
Passwort für die Verbindung mit der Active-Directory-Domain (optional)	Das Passwort für den Benutzer für die Verbindung mit der Active-Directory-Domain. Weitere Informationen finden Sie unter the section called “Schritt 3: Fügen Sie der Domäne für den EMR Cluster Konten hinzu”.

Beispiele für Konfigurationen

Die folgenden Beispiele zeigen Sicherheitskonfigurationen und Clusterkonfigurationen für gängige Szenarien. AWS CLI Befehle werden der Kürze halber dargestellt.

Lokal KDC

Die folgenden Befehle erstellen einen Cluster mit einem dedizierten Cluster, der auf dem primären Knoten KDC ausgeführt wird. Eine zusätzliche Konfiguration auf dem Cluster ist erforderlich. Weitere Informationen finden Sie unter Konfiguration eines Clusters für Kerberos-authentifizierte Benutzer HDFS und Verbindungen SSH.

Sicherheitskonfiguration erstellen

```
aws emr create-security-configuration --name LocalKDCSecurityConfig \
--security-configuration '{"AuthenticationConfiguration": \
{"KerberosConfiguration": {"Provider": "ClusterDedicatedKdc",\
"ClusterDedicatedKdcConfiguration": {"TicketLifetimeInHours": 24 }}}}'
```

Erstellen eines Clusters

```
aws emr create-cluster --release-label emr-7.2.0 \
--instance-count 3 --instance-type m5.xlarge \
--applications Name=Hadoop Name=Hive --ec2-attributes
 InstanceProfile=EMR_EC2_DefaultRole,KeyName=MyEC2Key \
--service-role EMR_DefaultRole \
--security-configuration LocalKDCSecurityConfig \
--kerberos-attributes Realm=EC2.INTERNAL,KdcAdminPassword=MyPassword
```

Clusterspezifisch mit realmübergreifendem Active KDC Directory-Vertrauen

Mit den folgenden Befehlen wird ein Cluster mit einem dedizierten Cluster erstellt, der auf dem primären Knoten KDC ausgeführt wird und über eine bereichsübergreifende Vertrauensstellung zu einer Active Directory-Domäne verfügt. Zusätzliche Konfiguration auf dem Cluster und in Active Directory ist erforderlich. Weitere Informationen finden Sie unter Tutorial: Konfigurieren einer bereichsübergreifenden Vertrauensstellung mit einer Active-Directory-Domain.

Sicherheitskonfiguration erstellen

```
aws emr create-security-configuration --name LocalKDCWithADTrustSecurityConfig \
--security-configuration '{"AuthenticationConfiguration": \
{"KerberosConfiguration": {"Provider": "ClusterDedicatedKdc", \
"ClusterDedicatedKdcConfiguration": {"TicketLifetimeInHours": 24, \
"CrossRealmTrustConfiguration": {"Realm":"AD.DOMAIN.COM", \
"Domain":"ad.domain.com", "AdminServer":"ad.domain.com", \
"KdcServer":"ad.domain.com"}}}}}'
```

Erstellen eines Clusters

```
aws emr create-cluster --release-label emr-7.2.0 \
--instance-count 3 --instance-type m5.xlarge --applications Name=Hadoop Name=Hive \
--ec2-attributes InstanceProfile=EMR_EC2_DefaultRole,KeyName=MyEC2Key \
--service-role EMR_DefaultRole --security-configuration KDCWithADTrustSecurityConfig \
--kerberos-attributes Realm=EC2.INTERNAL,KdcAdminPassword=MyClusterKDCAdminPassword,\
ADDomainJoinUser=ADUserLogonName,ADDomainJoinPassword=ADUserPassword,\
```

```
CrossRealmTrustPrincipalPassword=MatchADTrustPassword
```

Extern KDC auf einem anderen Cluster

Mit den folgenden Befehlen wird ein Cluster erstellt, der auf einen speziellen Cluster KDC auf dem Primärknoten eines anderen Clusters verweist, um die Prinzipale zu authentifizieren. Eine zusätzliche Konfiguration auf dem Cluster ist erforderlich. Weitere Informationen finden Sie unter Konfiguration eines Clusters für Kerberos-authentifizierte Benutzer HDFS und Verbindungen SSH.

Sicherheitskonfiguration erstellen

```
aws emr create-security-configuration --name ExtKDCOnDifferentCluster \
--security-configuration '{"AuthenticationConfiguration": \
{"KerberosConfiguration": {"Provider": "ExternalKdc", \
"ExternalKdcConfiguration": {"KdcServerType": "Single", \
"AdminServer": "MasterDNSOfKDCMaster:749", \
"KdcServer": "MasterDNSOfKDCMaster:88"}}}}'
```

Erstellen eines Clusters

```
aws emr create-cluster --release-label emr-7.2.0 \
--instance-count 3 --instance-type m5.xlarge \
--applications Name=Hadoop Name=Hive \
--ec2-attributes InstanceProfile=EMR_EC2_DefaultRole,KeyName=MyEC2Key \
--service-role EMR_DefaultRole --security-configuration ExtKDCOnDifferentCluster \
--kerberos-attributes Realm=EC2.INTERNAL,KdcAdminPassword=KDCOnMasterPassword
```

Externer Cluster KDC mit realmübergreifender Active Directory-Vertrauensstellung

Die folgenden Befehle erstellen einen Cluster ohne. KDC Der Cluster verweist auf einen dedizierten Cluster, der auf dem Primärknoten eines anderen Clusters KDC ausgeführt wird, um die Prinzipale zu authentifizieren. Dieser KDC verfügt über eine bereichsübergreifende Vertrauensstellung mit einem Active Directory-Domänencontroller. Eine zusätzliche Konfiguration auf dem Primärknoten mit dem KDC ist erforderlich. Weitere Informationen finden Sie unter Tutorial: Konfigurieren einer bereichsübergreifenden Vertrauensstellung mit einer Active-Directory-Domain.

Sicherheitskonfiguration erstellen

```
aws emr create-security-configuration --name ExtKDCWithADIntegration \
--security-configuration '{"AuthenticationConfiguration": \
{"KerberosConfiguration": {"Provider": "ExternalKdc", \
```

```
"ExternalKdcConfiguration": {"KdcServerType": "Single", \
"AdminServer": "MasterDNSofClusterKDC:749", \
"KdcServer": "MasterDNSofClusterKDC.com:88", \
"AdIntegrationConfiguration": {"AdRealm":"AD.DOMAIN.COM", \
"AdDomain":"ad.domain.com", \
"AdServer":"ad.domain.com"}}}}}'
```

Erstellen eines Clusters

```
aws emr create-cluster --release-label emr-7.2.0 \
--instance-count 3 --instance-type m5.xlarge --applications Name=Hadoop Name=Hive \
--ec2-attributes InstanceProfile=EMR_EC2_DefaultRole,KeyName=MyEC2Key \
--service-role EMR_DefaultRole --security-configuration ExtKDCWithADIntegration \
--kerberos-attributes Realm=EC2.INTERNAL,KdcAdminPassword=KDCOnMasterPassword,\
ADDomainJoinUser=MyPrivilegedADUserName,ADDomainJoinPassword=PasswordForADDomainJoinUser
```

Konfiguration eines Clusters für Kerberos-authentifizierte Benutzer HDFS und Verbindungen SSH

Amazon EMR erstellt Kerberos-authentifizierte Benutzerclients für die Anwendungen, die auf dem Cluster ausgeführt werden, z. B. für den Benutzer, den `hadoop` Benutzer und andere. `spark` Sie können auch Benutzer hinzufügen, die mit Kerberos für Cluster-Prozesse authentifiziert werden. Authentifizierte Benutzer können dann eine Verbindung mit dem Cluster mit ihren Kerberos-Anmeldeinformationen einrichten und mit den Anwendungen arbeiten. Damit sich ein Benutzer am Cluster authentifizieren kann, sind die folgenden Konfigurationen erforderlich:

- Ein Linux-Konto, das dem Kerberos-Prinzipal in entspricht, muss auf dem Cluster vorhanden sein. KDC Amazon EMR tut dies automatisch in Architekturen, die in Active Directory integriert sind.
- Sie müssen für jeden HDFS Benutzer ein Benutzerverzeichnis auf dem primären Knoten erstellen und dem Benutzer Berechtigungen für das Verzeichnis erteilen.
- Sie müssen den SSH Dienst so konfigurieren, dass er auf dem primären Knoten aktiviert GSSAPI ist. Darüber hinaus müssen Benutzer über einen SSH Client mit GSSAPI aktivierter Option verfügen.

Hinzufügen von Linux-Benutzern und Kerberos-Prinzipalen zum Primärknoten

Wenn Sie Active Directory nicht verwenden, müssen Sie Linux-Konten auf dem primären Clusterknoten erstellen und dem die Prinzipale für diese Linux-Benutzer hinzufügen. KDC Dazu gehört ein Prinzipal im KDC für den Primärknoten. Zusätzlich zu den Benutzerprinzipalen benötigt das auf dem Primärknoten KDC laufende System einen Principal für den lokalen Host.

Wenn Ihre Architektur die Active Directory-Integration beinhaltet, werden Linux-Benutzer und -Prinzipale auf dem lokalen KDC System, sofern zutreffend, automatisch erstellt. Sie können diesen Schritt überspringen. Weitere Informationen erhalten Sie unter Bereichsübergreifende Vertrauensstellung und Extern KDC — Cluster KDC auf einem anderen Cluster mit realmübergreifender Active Directory-Vertrauensstellung.

> **Important**
>
> Die KDC geht zusammen mit der Datenbank der Prinzipale verloren, wenn der Primärknoten beendet wird, weil der Primärknoten kurzlebigen Speicher verwendet. Wenn Sie Benutzer für SSH Verbindungen erstellen, empfehlen wir Ihnen, eine realmübergreifende Vertrauensstellung mit einem externen System einzurichten, das für hohe Verfügbarkeit konfiguriert ist. KDC Wenn Sie Benutzer für SSH Verbindungen mithilfe von Linux-Konten erstellen, automatisieren Sie alternativ den Kontoerstellungsprozess mithilfe von Bootstrap-Aktionen und -Skripts, sodass er wiederholt werden kann, wenn Sie einen neuen Cluster erstellen.

Am einfachsten lassen sich Benutzer und KDC Principals hinzufügen, wenn Sie einen Schritt an den Cluster senden, nachdem Sie ihn oder wenn Sie den Cluster erstellt haben. Alternativ können Sie eine Verbindung zum Primärknoten herstellen, indem Sie ein EC2 key pair als `hadoop` Standardbenutzer verwenden, um die Befehle auszuführen. Weitere Informationen finden Sie unter Connect zum Primärknoten her mit SSH.

Im folgenden Beispiel wird einem Cluster ein bereits vorhandenes Bash-Skript `configureCluster.sh` übergeben, das auf seine Cluster-ID verweist. Das Skript wird in Amazon S3 gespeichert.

```
aws emr add-steps --cluster-id <j-2AL4XXXXXX5T9> \
--steps Type=CUSTOM_JAR,Name=CustomJAR,ActionOnFailure=CONTINUE,\
Jar=s3://region.elasticmapreduce/libs/script-runner/script-runner.jar,\
Args=["s3://DOC-EXAMPLE-BUCKET/configureCluster.sh"]
```

Das folgende Beispiel veranschaulicht den Inhalt des `configureCluster.sh`-Skripts. Das Skript kümmert sich auch um das Erstellen von HDFS Benutzerverzeichnissen und das Aktivieren GSSAPI von BenutzerverzeichnissenSSH, die in den folgenden Abschnitten behandelt werden.

```
#!/bin/bash
```

```
#Add a principal to the KDC for the primary node, using the primary node's returned
 host name
sudo kadmin.local -q "ktadd -k /etc/krb5.keytab host/`hostname -f`"
#Declare an associative array of user names and passwords to add
declare -A arr
arr=([lijuan]=pwd1 [marymajor]=pwd2 [richardroe]=pwd3)
for i in ${!arr[@]}; do
    #Assign plain language variables for clarity
     name=${i}
     password=${arr[${i}]}

     # Create a principal for each user in the primary node and require a new password
 on first logon
     sudo kadmin.local -q "addprinc -pw $password +needchange $name"

     #Add hdfs directory for each user
     hdfs dfs -mkdir /user/$name

     #Change owner of each user's hdfs directory to that user
     hdfs dfs -chown $name:$name /user/$name
done

# Enable GSSAPI authentication for SSH and restart SSH service
sudo sed -i 's/^.*GSSAPIAuthentication.*$/GSSAPIAuthentication yes/' /etc/ssh/
sshd_config
sudo sed -i 's/^.*GSSAPICleanupCredentials.*$/GSSAPICleanupCredentials yes/' /etc/ssh/
sshd_config
sudo systemctl restart sshd
```

HDFSBenutzerverzeichnisse hinzufügen

Damit sich Ihre Benutzer beim Cluster anmelden können, um Hadoop-Jobs auszuführen, müssen Sie HDFS Benutzerverzeichnisse für ihre Linux-Konten hinzufügen und jedem Benutzer das Eigentum an seinem Verzeichnis zuweisen.

Das Senden eines Schritts an den Cluster, nachdem Sie ihn oder wenn Sie den Cluster erstellt haben, ist die einfachste Methode, HDFS Verzeichnisse zu erstellen. Alternativ können Sie eine Verbindung zum Primärknoten herstellen, indem Sie ein EC2 key pair als `hadoop` Standardbenutzer verwenden, um die Befehle auszuführen. Weitere Informationen finden Sie unter Connect zum Primärknoten her mit SSH.

Im folgenden Beispiel wird einem Cluster ein bereits vorhandenes Bash-Skript `AddHDFSUsers.sh` übergeben, das auf seine Cluster-ID verweist. Das Skript wird in Amazon S3 gespeichert.

```
aws emr add-steps --cluster-id <j-2AL4XXXXXX5T9> \
--steps Type=CUSTOM_JAR,Name=CustomJAR,ActionOnFailure=CONTINUE,\
Jar=s3://region.elasticmapreduce/libs/script-runner/script-runner.jar,Args=["s3://DOC-EXAMPLE-BUCKET/AddHDFSUsers.sh"]
```

Das folgende Beispiel veranschaulicht den Inhalt des `AddHDFSUsers.sh`-Skripts.

```
#!/bin/bash
# AddHDFSUsers.sh script

# Initialize an array of user names from AD, or Linux users created manually on the
 cluster
ADUSERS=("lijuan" "marymajor" "richardroe" "myusername")

# For each user listed, create an HDFS user directory
# and change ownership to the user

for username in ${ADUSERS[@]}; do
     hdfs dfs -mkdir /user/$username
     hdfs dfs -chown $username:$username /user/$username
done
```

Aktiviert GSSAPI für SSH

Damit Kerberos-authentifizierte Benutzer eine Verbindung zum Primärknoten herstellen könnenSSH, muss für den SSH Dienst die Authentifizierung aktiviert sein. GSSAPI Führen Sie zur Aktivierung GSSAPI die folgenden Befehle von der Befehlszeile des primären Knotens aus oder verwenden Sie einen Schritt, um ihn als Skript auszuführen. Nach der Neukonfiguration SSH müssen Sie den Dienst neu starten.

```
sudo sed -i 's/^.*GSSAPIAuthentication.*$/GSSAPIAuthentication yes/' /etc/ssh/
sshd_config
sudo sed -i 's/^.*GSSAPICleanupCredentials.*$/GSSAPICleanupCredentials yes/' /etc/ssh/
sshd_config
sudo systemctl restart sshd
```

Wird verwendetSSH, um eine Verbindung zu kerberisierten Clustern herzustellen

In diesem Abschnitt werden die Schritte für einen Kerberos-authentifizierten Benutzer veranschaulicht, um eine Verbindung zum Primärknoten eines Clusters herzustellen. EMR

Auf jedem Computer, der für eine SSH Verbindung verwendet wird, müssen SSH Client- und Kerberos-Clientanwendungen installiert sein. Linux-Computer enthalten diese höchstwahrscheinlich standardmäßig. Open SSH ist beispielsweise auf den meisten Linux-, Unix- und MacOS-Betriebssystemen installiert. Sie können nach einem SSH Client suchen, indem Sie ihn ssh in der Befehlszeile eingeben. Wenn Ihr Computer den Befehl nicht erkennt, installieren Sie einen SSH Client, um eine Verbindung zum Primärknoten herzustellen. Das SSH Open-Projekt bietet eine kostenlose Implementierung der gesamten SSH Toolsuite. Weitere Informationen finden Sie auf der SSHOpen-Website. Windows-Benutzer können Anwendungen wie Pu TTY als SSH Client verwenden.

Weitere Hinweise zu SSH Verbindungen finden Sie unterVerbinden mit einem Cluster.

SSHverwendet GSSAPI für die Authentifizierung von Kerberos-Clients, und Sie müssen die GSSAPI Authentifizierung für den SSH Dienst auf dem primären Clusterknoten aktivieren. Weitere Informationen finden Sie unter Aktiviert GSSAPI für SSH. SSHClients müssen ebenfalls verwenden. GSSAPI

In den folgenden Beispielen für *MasterPublicDNS* verwenden Sie den Wert, der für Master public DNS auf der Registerkarte Zusammenfassung im Bereich Cluster-Details angezeigt wird — zum Beispiel *ec2-11-222-33-44.compute-1.amazonaws.com*.

Voraussetzung für krb5.conf (nicht Active Directory)

Wenn Sie eine Konfiguration ohne Active Directory-Integration verwenden, muss jeder Client-Computer zusätzlich zu den SSH Client- und Kerberos-Clientanwendungen über eine Kopie der Datei verfügen, die mit der `/etc/krb5.conf` `/etc/krb5.conf` Datei auf dem primären Clusterknoten übereinstimmt.

So kopieren Sie die krb5.conf-Datei

1. Wird verwendetSSH, um mithilfe eines EC2 key pair und des `hadoop` Standardbenutzers eine Verbindung zum Primärknoten herzustellen, z. B. `hadoop@`*MasterPublicDNS* Detaillierte Anweisungen finden Sie unter Verbinden mit einem Cluster.
2. Kopieren Sie vom Primärknoten die Inhalte der `/etc/krb5.conf`-Datei. Weitere Informationen finden Sie unter Verbinden mit einem Cluster.
3. Erstellen Sie auf jedem Client-Computer, der eine Verbindung zum Cluster herstellt, eine identische `/etc/krb5.conf`-Datei basierend auf der Kopie, die Sie im vorigen Schritt erstellt haben.

Mit Kinit und SSH

Immer wenn ein Benutzer eine Verbindung von einem Client-Computer aus mithilfe von Kerberos-Anmeldeinformationen herstellt, muss der Benutzer zuerst Kerberos-Tickets für seinen Benutzer auf dem Client-Computer verlängern. Darüber hinaus muss der SSH Client für die Verwendung der GSSAPI Authentifizierung konfiguriert sein.

Wird verwendet, SSH um eine Verbindung zu einem EMR Kerber-Cluster herzustellen

1. Verwenden Sie `kinit` zum Verlängern Ihres Kerberos-Tickets, wie im folgenden Beispiel gezeigt

   ```
   kinit user1
   ```

2. Verwenden Sie einen `ssh` Client zusammen mit dem Prinzipal, den Sie im Clusternamen KDC oder im Active Directory-Benutzernamen erstellt haben. Stellen Sie sicher, dass die GSSAPI Authentifizierung aktiviert ist, wie in den folgenden Beispielen gezeigt.

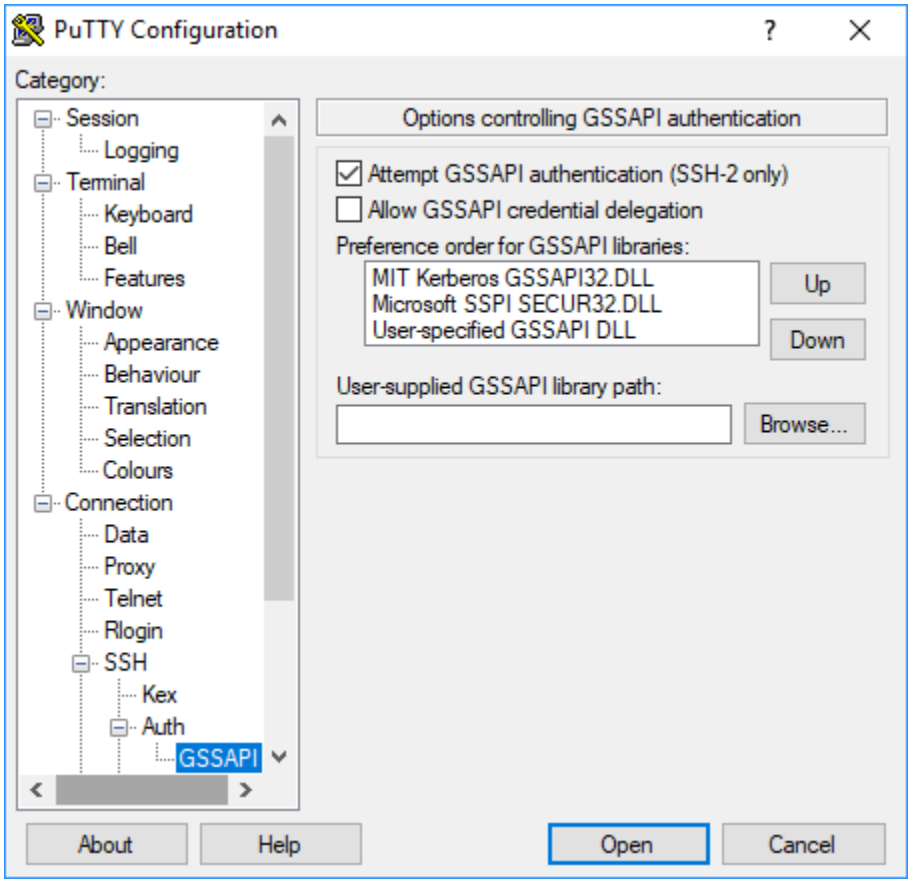
   Beispiel: Linux-Benutzer

   Die `-K` Option spezifiziert die GSSAPI Authentifizierung.

   ```
   ssh -K user1@MasterPublicDNS
   ```

   Beispiel: Windows-Benutzer (PuTTY)

   Stellen Sie sicher, dass die GSSAPI Authentifizierungsoption für die Sitzung wie folgt aktiviert ist:



Tutorial: Konfigurieren Sie einen dedizierten Cluster KDC

Dieses Thema führt Sie durch die Erstellung eines Clusters mit einem für den Cluster bestimmten Schlüsselverteilungszentrum (KDC), das manuelle Hinzufügen von Linux-Konten zu allen Clusterknoten, das Hinzufügen von Kerberos-Prinzipalen zum auf dem primären Knoten und das Sicherstellen, dass KDC auf den Client-Computern ein Kerberos-Client installiert ist.

Weitere Informationen zur EMR Unterstützung von Kerberos durch Amazon sowie Links zur MIT Kerberos-Dokumentation finden Sie unter. KDC Verwenden Sie Kerberos für die Authentifizierung bei Amazon EMR

Schritt 1: Den durch Kerberos geschützten Cluster erstellen

1. Erstellen Sie eine Sicherheitskonfiguration, die Kerberos aktiviert. Das folgende Beispiel zeigt einen `create-security-configuration` Befehl mit dem AWS CLI , der die Sicherheitskonfiguration als Inline-Struktur spezifiziert. JSON Sie können auch auf eine lokal gespeicherte Datei verweisen.

```
aws emr create-security-configuration --name MyKerberosConfig \
--security-configuration '{"AuthenticationConfiguration": {"KerberosConfiguration":
{"Provider": "ClusterDedicatedKdc", "ClusterDedicatedKdcConfiguration":
 {"TicketLifetimeInHours": 24}}}}'
```

2. Erstellen Sie einen Cluster, der auf die Sicherheitskonfiguration verweist, Kerberos-Attribute für die Cluster einrichtet, und Linux-Konten unter Verwendung einer Bootstrap-Aktion hinzufügt. Das folgende Beispiel zeigt den Befehl `create-cluster` unter Verwendung der AWS CLI. Der Befehl bezieht sich auf die Sicherheitskonfiguration, die Sie oben erstellt haben, `MyKerberosConfig`. Er referenziert auch ein einfaches Skript `createlinuxusers.sh`, als Bootstrap-Aktion, das Sie erstellen und zu Amazon S3 hochladen, bevor Sie den Cluster erstellen.

   ```
   aws emr create-cluster --name "MyKerberosCluster" \
   --release-label emr-7.2.0 \
   --instance-type m5.xlarge \
   --instance-count 3 \
   --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole,KeyName=MyEC2KeyPair \
   --service-role EMR_DefaultRole \
   --security-configuration MyKerberosConfig \
   --applications Name=Hadoop Name=Hive Name=Oozie Name=Hue Name=HCatalog Name=Spark \
   --kerberos-attributes Realm=EC2.INTERNAL,\
   KdcAdminPassword=MyClusterKDCAdminPwd \
   --bootstrap-actions Path=s3://DOC-EXAMPLE-BUCKET/createlinuxusers.sh
   ```

   Dar folgende Code zeigt den Inhalt des `createlinuxusers.sh`-Skripts, das jedem Knoten im Cluster user1, user2 und user3 hinzufügt. Im nächsten Schritt fügen Sie diese Benutzer als KDC Principals hinzu.

   ```
   #!/bin/bash
   sudo adduser user1
   sudo adduser user2
   sudo adduser user3
   ```

Schritt 2: Fügen Sie Principals zu dem hinzuKDC, erstellen Sie HDFS Benutzerverzeichnisse und konfigurieren Sie SSH

Für die KDC Ausführung auf dem primären Knoten muss ein Principal für den lokalen Host und für jeden Benutzer, den Sie auf dem Cluster erstellen, hinzugefügt werden. Sie können auch HDFS

Verzeichnisse für jeden Benutzer erstellen, wenn dieser eine Verbindung zum Cluster herstellen und Hadoop-Jobs ausführen muss. Konfigurieren Sie den SSH Dienst auf ähnliche Weise so, dass die GSSAPI Authentifizierung aktiviert wird, die für Kerberos erforderlich ist. Starten Sie den Dienst nach GSSAPI der SSH Aktivierung neu.

Die einfachste Möglichkeit dafür ist, dem Cluster ein Skript zu übergeben. Das folgende Beispiel übergibt dem Cluster ein bash-Skript `configurekdc.sh`, das Sie im vorherigen Schritt erstellt haben, wobei Sie die Cluster-ID angeben. Das Skript wird in Amazon S3 gespeichert. Alternativ können Sie mithilfe eines EC2 key pair eine Verbindung zum Primärknoten herstellen, um die Befehle auszuführen oder den Schritt während der Clustererstellung abzusenden.

```
aws emr add-steps --cluster-id <j-2AL4XXXXXX5T9> --steps
 Type=CUSTOM_JAR,Name=CustomJAR,ActionOnFailure=CONTINUE,Jar=s3://
myregion.elasticmapreduce/libs/script-runner/script-runner.jar,Args=["s3://DOC-EXAMPLE-
BUCKET/configurekdc.sh"]
```

Das folgende Beispiel veranschaulicht den Inhalt des `configurekdc.sh`-Skripts.

```
#!/bin/bash
#Add a principal to the KDC for the primary node, using the primary node's returned
 host name
sudo kadmin.local -q "ktadd -k /etc/krb5.keytab host/`hostname -f`"
#Declare an associative array of user names and passwords to add
declare -A arr
arr=([user1]=pwd1 [user2]=pwd2 [user3]=pwd3)
for i in ${!arr[@]}; do
    #Assign plain language variables for clarity
     name=${i}
     password=${arr[${i}]}

     # Create principal for sshuser in the primary node and require a new password on
 first logon
     sudo kadmin.local -q "addprinc -pw $password +needchange $name"

     #Add user hdfs directory
     hdfs dfs -mkdir /user/$name

     #Change owner of user's hdfs directory to user
     hdfs dfs -chown $name:$name /user/$name
done

# Enable GSSAPI authentication for SSH and restart SSH service
```

```
sudo sed -i 's/^.*GSSAPIAuthentication.*$/GSSAPIAuthentication yes/' /etc/ssh/
sshd_config
sudo sed -i 's/^.*GSSAPICleanupCredentials.*$/GSSAPICleanupCredentials yes/' /etc/ssh/
sshd_config
sudo systemctl restart sshd
```

Die Benutzer, die Sie hinzugefügt haben, sollten jetzt in der Lage sein, mithilfe von eine Verbindung zum Cluster herzustellenSSH. Weitere Informationen finden Sie unter Wird verwendetSSH, um eine Verbindung zu kerberisierten Clustern herzustellen.

Tutorial: Konfigurieren einer bereichsübergreifenden Vertrauensstellung mit einer Active-Directory-Domain

Wenn Sie eine realmübergreifende Vertrauensstellung einrichten, ermöglichen Sie Prinzipalen (normalerweise Benutzern) aus einem anderen Kerberos-Bereich, sich bei Anwendungskomponenten auf dem Cluster zu authentifizieren. EMR Das dem Cluster zugeordnete Schlüsselverteilungszentrum (KDC) baut KDC mithilfe eines realmübergreifenden Prinzipals, der in beiden vorhanden ist, eine Vertrauensbeziehung zu einem anderen auf. KDCs Der Name des Prinzipals und das Passwort stimmen genau überein.

Eine realmübergreifende Vertrauensstellung setzt voraus, dass sie einander über das Netzwerk KDCs erreichen und die Domainnamen der jeweils anderen Seite auflösen können. Im Folgenden finden Sie Schritte zum Einrichten einer realmübergreifenden Vertrauensstellung mit einem Microsoft AD-Domänencontroller, der als EC2 Instanz ausgeführt wird, sowie ein Beispiel für eine Netzwerkkonfiguration, die die erforderliche Konnektivität und Domänennamenauflösung bereitstellt. Jede Netzwerkkonfiguration, die den erforderlichen Netzwerkverkehr zwischen KDCs den Geräten zulässt, ist zulässig.

Optional können Sie, nachdem Sie eine realmübergreifende Vertrauensstellung mit Active Directory unter Verwendung eines KDC Clusters eingerichtet haben, einen weiteren Cluster mit einer anderen Sicherheitskonfiguration erstellen, um den KDC auf dem ersten Cluster als externen KDC Cluster zu referenzieren. Ein Beispiel für die Einrichtung einer Sicherheitskonfiguration und eines Clusters finden Sie unter Externer Cluster KDC mit realmübergreifender Active Directory-Vertrauensstellung.

Weitere Informationen zur EMR Unterstützung von Kerberos durch Amazon sowie Links zur MIT Kerberos-Dokumentation finden Sie unter. KDC Verwenden Sie Kerberos für die Authentifizierung bei Amazon EMR

> **Important**
>
> Amazon unterstützt EMR keine realmübergreifenden Vertrauensstellungen mit. AWS Directory Service for Microsoft Active Directory

Schritt 1: Richten Sie das VPC UND-Subnetz ein

In den folgenden Schritten wird das Erstellen eines VPC UND-Subnetzes veranschaulicht, sodass der dedizierte Cluster den Active Directory-Domänencontroller erreichen und dessen Domänennamen auflösen KDC kann. In diesen Schritten erfolgt die Domänennamenauflösung, indem der Active Directory-Domänencontroller als Domänennamenserver im Optionssatz referenziert wird. DHCP Weitere Informationen finden Sie unter Schritt 5: Verwenden Sie einen DHCP Optionssatz, um den Active Directory-Domänencontroller als VPC DNS Server anzugeben.

Der KDC und der Active Directory-Domänencontroller müssen in der Lage sein, die Domänennamen der jeweils anderen Person aufzulösen. Dadurch kann Amazon EMR Computer mit der Domain verbinden und die entsprechenden Linux-Konten und SSH -Parameter auf Cluster-Instances automatisch konfigurieren.

Wenn Amazon den Domainnamen nicht auflösen EMR kann, können Sie die Vertrauensstellung anhand der IP-Adresse des Active Directory-Domänencontrollers referenzieren. Sie müssen jedoch manuell Linux-Konten hinzufügen, entsprechende Principals zum Cluster-Dedicated KDC hinzufügen und die Konfiguration vornehmen. SSH

So richten Sie das UND-Subnetz ein VPC

1. Erstellen Sie ein Amazon VPC mit einem einzigen öffentlichen Subnetz. Weitere Informationen finden Sie unter Schritt 1: Erstellen des VPC im Amazon-Handbuch „VPCErste Schritte“.

   > **Important**
   >
   > Wenn Sie einen Microsoft Active Directory-Domänencontroller verwenden, wählen Sie einen CIDR Block für den EMR Cluster aus, sodass alle IPv4 Adressen weniger als neun Zeichen lang sind (z. B. 10.0.0.0/16). Das liegt daran, dass die DNS Namen der Clustercomputer verwendet werden, wenn die Computer dem Active Directory-Verzeichnis beitreten. AWS weist DNSHostnamen auf der Grundlage der IPv4 Adresse zu, sodass längere IP-Adressen zu DNS Namen mit mehr als 15 Zeichen führen können. Für Active Directory gilt ein Limit von 15 Zeichen für die Registrierung der Namen der hinzugefügten Computer, und es kürzt längere Namen, was zu unvorhersehbaren Fehlern führen kann.

2. Entfernt den DHCP Standardoptionssatz, der dem zugewiesen ist. VPC Weitere Informationen finden Sie unter Ändern von a VPC zu „Keine DHCP Optionen“. Später fügen Sie einen neuen hinzu, der den Active Directory-Domänencontroller als DNS Server angibt.
3. Vergewissern Sie sich, dass die DNS Unterstützung für aktiviert istVPC, d. h., dass sowohl DNS Hostnamen als auch DNS Auflösung aktiviert sind. Standardmäßig sind sie aktiviert. Weitere Informationen finden Sie unter Aktualisierung der DNS Unterstützung für Ihren VPC.
4. Vergewissern Sie VPC sich, dass Ihr Internet-Gateway angeschlossen ist. Dies ist die Standardeinstellung. Weitere Informationen finden Sie unter Erstellen und Anfügen eines Internet-Gateways.

   > **Note**
   >
   > In diesem Beispiel wird ein Internet-Gateway verwendet, weil Sie einen neuen Domänencontroller für einrichtenVPC. Für Ihre Anwendung ist möglicherweise kein Internet-Gateway erforderlich. Die einzige Voraussetzung ist, dass der dedizierte Cluster auf den Active Directory-Domänencontroller zugreifen KDC kann.

5. Erstellen Sie eine benutzerdefinierte Routing-Tabelle, fügen Sie eine Route zum Internet-Gateway hinzu, und ordnen Sie ihn dann Ihrem Subnetz zu. Weitere Informationen finden Sie unter Erstellen einer benutzerdefinierten Routing-Tabelle.

6. Wenn Sie die EC2 Instanz für den Domänencontroller starten, muss sie über eine statische öffentliche IPv4 Adresse verfügen, über die Sie eine Verbindung herstellen können. RDP Der einfachste Weg, dies zu tun, besteht darin, Ihr Subnetz so zu konfigurieren, dass öffentliche IPv4 Adressen automatisch zugewiesen werden. Dies ist nicht die Standardeinstellung, wenn ein Subnetz erstellt wird. Weitere Informationen finden Sie unter Ändern des Attributs für die öffentliche IPv4 Adressierung Ihres Subnetzes. Optional können Sie die Adresse zuweisen, wenn Sie die Instance starten. Weitere Informationen finden Sie unter Zuweisen einer öffentlichen IPv4 Adresse beim Instance-Start.
7. Wenn Sie fertig sind, notieren Sie sich Ihr Subnetz VPC und Ihr IDs Subnetz. Sie benötigen sie später, wenn Sie den Active-Directory-Domain-Controller und den Cluster starten.

Schritt 2: Den Active-Directory-Domain-Controller starten und installieren

1. Starten Sie eine EC2 Instanz, die auf Microsoft Windows Server 2016 Base basiertAMI. Wir empfehlen einen m4.xlarge oder einen besseren Instance-Typ. Weitere Informationen finden Sie unter Starten einer AWS Marketplace Instance im EC2Amazon-Benutzerhandbuch.
2. Notieren Sie sich die Gruppen-ID der Sicherheitsgruppe, die der EC2 Instance zugeordnet ist. Sie benötigen sie für Schritt 6: Starten Sie einen kerberisierten Cluster EMR. Wir verwenden *sg-012xrlmdomain345*. Alternativ können Sie verschiedene Sicherheitsgruppen für den EMR Cluster und diese Instanz angeben, die den Datenverkehr zwischen ihnen ermöglicht. Weitere Informationen finden Sie unter EC2Amazon-Sicherheitsgruppen für Linux-Instances im EC2Amazon-Benutzerhandbuch.
3. Stellen Sie mithilfe von Connect zur EC2 Instanz herRDP. Weitere Informationen finden Sie unter Herstellen einer Verbindung zu Ihrer Windows-Instance im EC2Amazon-Benutzerhandbuch.
4. Starten Sie Server Manager, um die Domain-Services-Rolle von Active Directory auf dem Server zu installieren und zu konfigurieren. Machen Sie den Server zu einem Domain-Controller und weisen Sie einen Domain-Namen zu (wir verwenden in diesem Beispiel hier *ad.domain.com*). Notieren Sie sich den Domainnamen, da Sie ihn später benötigen, wenn Sie die EMR Sicherheitskonfiguration und den Cluster erstellen. Wenn Sie noch keine Erfahrung mit der Einrichtung von Active Directory haben, können Sie den Anweisungen in So richten Sie Active Directory (AD) in Windows Server 2016 ein folgen.

   Die Instance startet neu, wenn Sie fertig sind.

Schritt 3: Fügen Sie der Domäne für den EMR Cluster Konten hinzu

RDPzum Active Directory-Domänencontroller, um Konten unter Active Directory-Benutzer und -Computer für jeden Clusterbenutzer zu erstellen. Weitere Informationen finden Sie unter Erstellen eines Benutzerkontos in Active-Directory-Benutzern und -Computern auf der Website Microsoft Learn. Notieren Sie den Wert für User logon name (Benutzeranmeldename) jedes Benutzers. Sie benötigen diese später, wenn Sie den Cluster konfigurieren.

Darüber hinaus erstellen Sie ein Konto mit ausreichenden Berechtigungen, um der Domain Computer hinzuzufügen. Sie geben dieses Konto an, wenn Sie einen Cluster erstellen. Amazon EMR verwendet es, um Cluster-Instances mit der Domain zu verbinden. Sie geben dieses Konto und sein Passwort in Schritt 6: Starten Sie einen kerberisierten Cluster EMR an. Für die Delegation von Join-Berechtigungen des Computers an das Konto empfehlen wir das Erstellen einer Gruppe mit Join-Berechtigungen und die anschließende Zuweisung des Benutzers zu der Gruppe. Weitere Informationen finden Sie unter Delegieren von Berechtigungen für den Verzeichniszugang im AWS Directory Service -Administratorhandbuch.

Schritt 4: Eine eingehende Vertrauensstellung auf dem Active-Directory-Domain-Controller konfigurieren

Mit den folgenden Beispielbefehlen wird eine Vertrauensstellung in Active Directory hergestellt. Dabei handelt es sich um eine unidirektionale, eingehende, nicht transitive Realm-Vertrauensstellung mit dem zugewiesenen Cluster. KDC Das Beispiel, das wir für den Bereich des Clusters verwenden, ist *EC2.INTERNAL*. Ersetzen Sie den *KDC-FQDN* wobei der öffentliche DNS Name für den EMR Amazon-Primärknoten aufgeführt ist, auf dem sich der befindetKDC. Der Parameter `passwordt` gibt das cross-realm principal password (Passwort des bereichsübergreifenden Prinzipals) an, das Sie beim Erstellen eines Clusters zusammen mit dem realm (Bereich) des Clusters angeben. Der Bereichsname wird von dem Standard-Domain-Namen `us-east-1` für den Cluster abgeleitet. Die `Domain` ist die Active-Directory-Domain, in der Sie die Vertrauensstellung erstellen. Sie wird gemäß Konvention in Kleinbuchstaben angegeben. Im Beispiel wird *ad.domain.com* verwendet.

Öffnen Sie die Windows-Eingabeaufforderung mit Administrator-Berechtigungen und geben Sie die folgenden Befehle zum Erstellen der Vertrauensstellung auf dem Active-Directory-Domain-Controller ein:

```
C:\Users\Administrator> ksetup /addkdc EC2.INTERNAL KDC-FQDN
C:\Users\Administrator> netdom trust EC2.INTERNAL /Domain:ad.domain.com /add /realm /
passwordt:MyVeryStrongPassword
C:\Users\Administrator> ksetup /SetEncTypeAttr EC2.INTERNAL AES256-CTS-HMAC-SHA1-96
```

Schritt 5: Verwenden Sie einen DHCP Optionssatz, um den Active Directory-Domänencontroller als VPC DNS Server anzugeben

Nachdem der Active Directory-Domänencontroller konfiguriert ist, müssen Sie den VPC so konfigurieren, dass er als Domänennamenserver für die Namensauflösung in Ihrem verwendet wirdVPC. Fügen Sie dazu einen DHCP Optionssatz hinzu. Geben Sie einen Wert in Domainname als Domainnamen für Ihren Cluster ein, z. B. `ec2.internal`, wenn sich Ihr Cluster in us-east-1 befindet, oder *`region`*`.compute.internal` für andere Regionen. Für Domänennamenserver müssen Sie die IP-Adresse des Active Directory-Domänencontrollers (der vom Cluster aus erreichbar sein muss) als ersten Eintrag angeben, gefolgt von AmazonProvidedDNS(z. B. ***`xx.xx.xx.xx`***, AmazonProvided DNS). Weitere Informationen finden Sie unter DHCPOptionssätze ändern.

Schritt 6: Starten Sie einen kerberisierten Cluster EMR

1. Erstellen Sie in Amazon eine SicherheitskonfigurationEMR, die den Active Directory-Domänencontroller angibt, den Sie in den vorherigen Schritten erstellt haben. Ein Beispielbefehl ist nachfolgend gezeigt. Ersetzen Sie die Domain *`ad.domain.com`* durch den Namen der Domain, die Sie in Schritt 2: Den Active-Directory-Domain-Controller starten und installieren angegeben haben.

```
aws emr create-security-configuration --name MyKerberosConfig \
--security-configuration '{
  "AuthenticationConfiguration": {
    "KerberosConfiguration": {
      "Provider": "ClusterDedicatedKdc",
      "ClusterDedicatedKdcConfiguration": {
        "TicketLifetimeInHours": 24,
        "CrossRealmTrustConfiguration": {
          "Realm": "AD.DOMAIN.COM",
          "Domain": "ad.domain.com",
          "AdminServer": "ad.domain.com",
          "KdcServer": "ad.domain.com"
        }
      }
    }
  }
}'
```

2. Erstellen Sie den Cluster mit den folgenden Attributen:
   - Verwenden Sie die `--security-configuration`-Option, um die Sicherheitskonfiguration anzugeben, die Sie erstellt haben. Wir verwenden *`MyKerberosConfig`* im Beispiel.

- Verwenden Sie die `SubnetId`-Eigenschaft der `--ec2-attributes option`, um das Subnetz anzugeben, das Sie in Schritt 1: Richten Sie das VPC UND-Subnetz ein erstellt haben. Wir verwenden *`step1-subnet`* im Beispiel.
- Verwenden Sie die `AdditionalMasterSecurityGroups` und `AdditionalSlaveSecurityGroups` der `--ec2-attributes`-Option, um anzugeben, dass die Sicherheitsgruppe, die dem AD-Domain-Controller von Schritt 2: Den Active-Directory-Domain-Controller starten und installieren zugeordnet ist, dem Cluster-Primärknoten sowie dem Core- und dem Aufgabenknoten zugeordnet ist. Wir verwenden *`sg-012xrlmdomain345`* im Beispiel.

Verwenden Sie `--kerberos-attributes`, um die folgenden Cluster-spezifischen Kerberos-Attribute anzugeben:

- Den Bereich für den Cluster, den Sie bei der Einrichtung des Active-Directory-Domain-Controllers angegeben haben.
- Das Prinzipal-Passwort der bereichsübergreifenden Vertrauensstellung, das Sie als `passwordt` in Schritt 4: Eine eingehende Vertrauensstellung auf dem Active-Directory-Domain-Controller konfigurieren angegeben haben.
- A`KdcAdminPassword`, mit der Sie den KDC Cluster-Dedicated administrieren können.
- Der Benutzeranmeldename und das Passwort des Active Directory-Kontos mit Computer-Join-Berechtigungen, die Sie in Schritt 3: Fügen Sie der Domäne für den EMR Cluster Konten hinzu erstellt haben.

Das folgende Beispiel startet einen Cluster mit Schutz durch Kerberos.

```
aws emr create-cluster --name "MyKerberosCluster" \
--release-label emr-5.10.0 \
--instance-type m5.xlarge \
--instance-count 3 \
--ec2-attributes InstanceProfile=EMR_EC2_DefaultRole,KeyName=MyEC2KeyPair,\
SubnetId=step1-subnet, AdditionalMasterSecurityGroups=sg-012xrlmdomain345,
AdditionalSlaveSecurityGroups=sg-012xrlmdomain345\
--service-role EMR_DefaultRole \
--security-configuration MyKerberosConfig \
--applications Name=Hadoop Name=Hive Name=Oozie Name=Hue Name=HCatalog Name=Spark \
--kerberos-attributes Realm=EC2.INTERNAL,\
KdcAdminPassword=MyClusterKDCAdminPwd,\
ADDomainJoinUser=ADUserLogonName,ADDomainJoinPassword=ADUserPassword,\
CrossRealmTrustPrincipalPassword=MatchADTrustPwd
```

Schritt 7: Erstellen Sie HDFS Benutzer und legen Sie Berechtigungen für den Cluster für Active Directory-Konten fest

Beim Einrichten einer Vertrauensbeziehung mit Active Directory EMR erstellt Amazon Linux-Benutzer auf dem Cluster für jedes Active Directory-Konto. Beispielsweise hat der Benutzeranmeldename `LiJuan` in Active Directory ein Linux-Benutzerkonto von `lijuan`. Active Directory-Benutzernamen können Großbuchstaben, aber Linux ignoriert die Groß-/Kleinschreibung von Active Directory.

Damit sich Ihre Benutzer beim Cluster anmelden können, um Hadoop-Jobs auszuführen, müssen Sie HDFS Benutzerverzeichnisse für ihre Linux-Konten hinzufügen und jedem Benutzer das Eigentum an seinem Verzeichnis zuweisen. Zu diesem Zweck empfehlen wir, dass Sie ein Skript ausführen, das Sie in Amazon S3 als Cluster-Schritt gespeichert haben. Alternativ können Sie die Befehle aus dem folgenden Skript von der Befehlszeile auf dem Primärknoten aus ausführen. Verwenden Sie das EC2 key pair, das Sie bei der Erstellung des Clusters angegeben haben, um SSH als Hadoop-Benutzer eine Verbindung zum primären Knoten herzustellen. Weitere Informationen finden Sie unter Verwenden Sie ein EC2 key pair für SSH Anmeldeinformationen.

Führen Sie den folgenden Befehl aus, um dem Cluster, der ein Skript ausführt, einen Schritt hinzuzufügen: *AddHDFSUsers.sh*.

```
aws emr add-steps --cluster-id <j-2AL4XXXXXX5T9> \
--steps Type=CUSTOM_JAR,Name=CustomJAR,ActionOnFailure=CONTINUE,\
Jar=s3://region.elasticmapreduce/libs/script-runner/script-runner.jar,Args=["s3://DOC-EXAMPLE-BUCKET/AddHDFSUsers.sh"]
```

Der Inhalt der Datei *AddHDFSUsers.sh* ist wie folgt.

```
#!/bin/bash
# AddHDFSUsers.sh script

# Initialize an array of user names from AD or Linux users and KDC principals created
 manually on the cluster
ADUSERS=("lijuan" "marymajor" "richardroe" "myusername")

# For each user listed, create an HDFS user directory
# and change ownership to the user

for username in ${ADUSERS[@]}; do
     hdfs dfs -mkdir /user/$username
     hdfs dfs -chown $username:$username /user/$username
done
```

Hadoop-Gruppen zugeordnete Active-Directory-Gruppen

Amazon EMR verwendet System Security Services Daemon (SSD), um Active Directory-Gruppen Hadoop-Gruppen zuzuordnen. Um die Gruppenzuordnungen nach der Anmeldung am Primärknoten zu bestätigen, wie in Wird verwendetSSH, um eine Verbindung zu kerberisierten Clustern herzustellen beschrieben, können Sie den Befehl `hdfs groups` ausführen, um zu bestätigen, dass Active Directory-Gruppen, zu denen Ihr Active Directory-Konto gehört, Hadoop-Gruppen für die entsprechenden Hadoop-Benutzer auf dem Cluster zugeordnet wurden. Sie können auch die Gruppenzuordnungen anderer Benutzer überprüfen, indem Sie einen oder mehrere Benutzernamen im Befehl angeben, z. B. `hdfs groups lijuan`. Weitere Informationen finden Sie unter Gruppen im Apache HDFS Commands Guide.

Verwenden Sie Active Directory oder LDAP Server für die Authentifizierung bei Amazon EMR

Mit EMR Amazon-Versionen 6.12.0 und höher können Sie das LDAP over SSL (LDAPS) - Protokoll verwenden, um einen Cluster zu starten, der nativ in Ihren Corporate Identity-Server integriert ist. LDAP(Lightweight Directory Access Protocol) ist ein offenes, herstellerneutrales Anwendungsprotokoll, das auf Daten zugreift und diese verwaltet. LDAPwird häufig für die Benutzerauthentifizierung gegenüber Corporate-Identity-Servern verwendet, die auf Anwendungen wie Active Directory (AD) und Open gehostet werden. LDAP Mit dieser nativen Integration können Sie Ihren LDAP Server verwenden, um Benutzer bei Amazon EMR zu authentifizieren.

Zu den Höhepunkten der EMR LDAP Amazon-Integration gehören:

- Amazon EMR konfiguriert die unterstützten Anwendungen so, dass sie sich in Ihrem Namen mit LDAP Authentifizierung authentifizieren.
- Amazon EMR konfiguriert und verwaltet die Sicherheit für die unterstützten Anwendungen mit dem Kerberos-Protokoll. Sie müssen keine Befehle oder Skripte eingeben.
- Durch die Apache Ranger-Autorisierung erhalten Sie eine detaillierte Zugriffskontrolle (FGAC) für die Hive Metastore-Datenbank und -Tabellen. Weitere Informationen finden Sie unter Integrieren Sie Amazon EMR mit Apache Ranger.
- Wenn Sie LDAP Anmeldeinformationen für den Zugriff auf einen Cluster benötigen, erhalten Sie eine detaillierte Zugriffskontrolle (FGAC) darüber, wer auf Ihre Cluster zugreifen kann. EMR SSH

Die folgenden Seiten bieten einen konzeptionellen Überblick, die Voraussetzungen und die Schritte zum Starten eines EMR Clusters mit der EMR LDAP Amazon-Integration.

Themen

Überblick über LDAP mit Amazon EMR

Das Lightweight Directory Access Protocol (LDAP) ist ein Softwareprotokoll, mit dem Netzwerkadministratoren den Zugriff auf Daten verwalten und kontrollieren, indem sie Benutzer im Netzwerk eines Unternehmens authentifizieren. Das LDAP Protokoll speichert Informationen in einer hierarchischen Verzeichnisstruktur. Weitere Informationen finden Sie unter Basic LDAP Concepts auf LDAP.com.

Innerhalb eines Unternehmensnetzwerks verwenden viele Anwendungen das LDAP Protokoll möglicherweise zur Benutzerauthentifizierung. Mit der EMR LDAP Amazon-Integration können EMR Cluster nativ dasselbe LDAP Protokoll mit einer zusätzlichen Sicherheitskonfiguration verwenden.

Es gibt zwei Hauptimplementierungen des LDAP Protokolls, die Amazon EMR unterstützt: Active Directory und Open LDAP. Andere Implementierungen sind zwar möglich, aber die meisten passen zu denselben Authentifizierungsprotokollen wie Active Directory oder Open. LDAP

Active Directory (AD)

Active Directory (AD) ist ein Verzeichnisservice von Microsoft für Windows-Domainnetzwerke. AD ist in den meisten Windows Server-Betriebssystemen enthalten und kann mit Clients über die LDAPS Protokolle LDAP und kommunizieren. Zur Authentifizierung EMR versucht Amazon, eine Benutzerbindung mit Ihrer AD-Instance mit dem Benutzerprinzipalnamen (UPN) als definiertem Namen und Passwort herzustellen. Der UPN verwendet das Standardformat`username@domain_name`.

Öffnen LDAP

Open LDAP ist eine kostenlose Open-Source-Implementierung des LDAP Protokolls. Zur Authentifizierung EMR versucht Amazon, eine Benutzerbindung mit Ihrer LDAP Open-Instance mit dem vollqualifizierten Domainnamen (FQDN)

als definiertem Namen und Passwort herzustellen. Der FQDN verwendet das Standardformat`username_attribute=username,LDAP_user_search_base`. In der Regel lautet der `username_attribute` Wert `uid`, und der `LDAP_user_search_base` Wert enthält die Attribute des Baums, der zum Benutzer führt. Beispiel, `ou=People,dc=example,dc=com`.

Andere kostenlose und quelloffene Implementierungen des LDAP Protokolls folgen in der Regel einem ähnlichen FQDN Schema wie Open LDAP für die eindeutigen Namen ihrer Benutzer.

LDAPKomponenten für Amazon EMR

Sie können Ihren LDAP Server verwenden, um sich bei Amazon EMR und allen Anwendungen, die der Benutzer direkt auf dem EMR Cluster verwendet, über die folgenden Komponenten zu authentifizieren.

Secret Agent

Der Secret Agent ist ein Cluster-Prozess, der alle Benutzeranfragen authentifiziert. Der Secret Agent erstellt im Namen der unterstützten Anwendungen im Cluster die Benutzerbindung zu Ihrem LDAP Server. EMR Der Secret-Agent wird unter Benutzer `emrsecretagent` ausgeführt und schreibt Protokolle in das Verzeichnis `/emr/secretagent/log`. Diese Protokolle enthalten Details zum Status der Authentifizierungsanfrage jedes Benutzers und zu allen Fehlern, die bei der Benutzerauthentifizierung auftreten können.

Daemon für Systemsicherheitsdienste () SSSD

SSSDist ein Daemon, der auf jedem Knoten eines Clusters mit LDAP aktivierter EMR Option ausgeführt wird. SSSDerstellt und verwaltet einen UNIX Benutzer, der Ihre Remote-Unternehmensidentität mit jedem Knoten synchronisiert. YARNbasierte Anwendungen wie Hive und Spark erfordern, dass auf jedem Knoten, der eine Abfrage für einen UNIX Benutzer ausführt, ein lokaler Benutzer vorhanden ist.

Anwendungsunterstützung und Überlegungen LDAP zu Amazon EMR

Unterstützte Anwendungen mit LDAP für Amazon EMR

> **Important**
>
> Die auf dieser Seite aufgeführten Anwendungen sind die einzigen Anwendungen, die Amazon EMR unterstütztLDAP. Um die Clustersicherheit zu gewährleisten, können Sie nur LDAP -kompatible Anwendungen einbeziehen, wenn Sie einen EMR Cluster mit LDAP

aktivierter Option erstellen. Wenn Sie versuchen, andere, nicht unterstützte Anwendungen zu installieren, lehnt Amazon EMR Ihre Anfrage für einen neuen Cluster ab.

Die EMR Amazon-Versionen 6.12 und höher unterstützen die LDAP Integration mit den folgenden Anwendungen:

- Apache Livy
- Apache Hive bis HiveServer 2 () HS2
- Trino
- Presto
- Hue

Sie können auch die folgenden Anwendungen auf einem EMR Cluster installieren und sie so konfigurieren, dass sie Ihren Sicherheitsanforderungen entsprechen:

- Apache Spark
- Apache Hadoop

Unterstützte Funktionen von LDAP für Amazon EMR

Sie können die folgenden EMR Amazon-Funktionen mit der LDAP Integration verwenden:

> **Note**
>
> Um die Sicherheit der LDAP Anmeldeinformationen zu gewährleisten, müssen Sie die Verschlüsselung während der Übertragung verwenden, um den Datenfluss innerhalb und außerhalb des Clusters zu sichern. Weitere Informationen über Verschlüsselung während der Übertragung finden Sie unter Verschlüsseln von Daten im Ruhezustand und im Transit.

- Verschlüsselung während der Übertragung (erforderlich) und im Ruhestand
- Instance-Gruppen, Instance-Flotten und Spot Instances
- Neukonfiguration von Anwendungen auf einem laufenden Cluster
- EMRFSserverseitige Verschlüsselung () SSE

Nicht unterstützte Funktionen

Beachten Sie die folgenden Einschränkungen, wenn Sie die EMR LDAP Amazon-Integration verwenden:

- Amazon EMR deaktiviert Schritte für Cluster mit LDAP aktivierter Option.
- Amazon unterstützt EMR keine Runtime-Rollen und AWS Lake Formation Integrationen für Cluster mit LDAP aktivierter Option.
- Amazon unterstützt LDAP Start EMR nichtTLS.
- Amazon unterstützt den Hochverfügbarkeitsmodus (Cluster mit mehreren Primärknoten) EMR nicht für Cluster mit LDAP aktivierter Option.
- Sie können Bindungsanmeldedaten oder Zertifikate für Cluster mit LDAP aktivierter Option nicht rotieren. Wenn eines dieser Felder rotiert wurde, empfehlen wir, einen neuen Cluster mit den aktualisierten Bindungsanmeldeinformationen oder Zertifikaten zu starten.
- Sie müssen exakte Suchbasen mit verwendenLDAP. Die LDAP Benutzer- und Gruppensuchbasis unterstützt keine LDAP Suchfilter.

Konfigurieren und starten Sie einen EMR Cluster mit LDAP

In diesem Abschnitt wird beschrieben, wie Amazon EMR für die Verwendung mit LDAP Authentifizierung konfiguriert wird.

Themen

AWS Secrets Manager Berechtigungen zur EMR Amazon-Instance-Rolle hinzufügen

Amazon EMR verwendet eine IAM Service-Rolle, um in Ihrem Namen Aktionen zur Bereitstellung und Verwaltung von Clustern durchzuführen. Die Servicerolle für EC2 Cluster-Instances, auch EC2Instance-Profil für Amazon genanntEMR, ist eine spezielle Art von Servicerolle, die Amazon jeder EC2 Instance in einem Cluster beim Start EMR zuweist.

Um Berechtigungen für einen EMR Cluster für die Interaktion mit Amazon S3 S3-Daten und anderen AWS Diensten zu definieren, definieren Sie ein benutzerdefiniertes EC2 Amazon-Instance-Profil und

nicht das, `EMR_EC2_DefaultRole` wenn Sie Ihren Cluster starten. Weitere Informationen erhalten Sie unter Servicerolle für EC2 Cluster-Instances (EC2Instance-Profil) und Passen Sie IAM Rollen an.

Fügen Sie dem EC2 Standard-Instance-Profil die folgenden Anweisungen hinzu, damit Amazon Sitzungen EMR taggen und auf die zugreifen kann AWS Secrets Manager , in denen LDAP Zertifikate gespeichert sind.

```
    {
      "Sid": "AllowAssumeOfRolesAndTagging",
      "Effect": "Allow",
      "Action": ["sts:TagSession", "sts:AssumeRole"],
      "Resource": [
        "arn:aws:iam::111122223333:role/LDAP_DATA_ACCESS_ROLE_NAME",
        "arn:aws:iam::111122223333:role/LDAP_USER_ACCESS_ROLE_NAME"
      ]
    },
    {
        "Sid": "AllowSecretsRetrieval",
        "Effect": "Allow",
        "Action": "secretsmanager:GetSecretValue",
        "Resource": [
            "arn:aws:secretsmanager:us-east-1:111122223333:secret:LDAP_SECRET_NAME*",
            "arn:aws:secretsmanager:us-
east-1:111122223333:secret:ADMIN_LDAP_SECRET_NAME*"
        ]
    }
```

> **Note**
>
> Ihre Cluster-Anfragen schlagen fehl, wenn Sie bei der Festlegung von Secrets Manager-Berechtigungen das *-Platzhalterzeichen am Ende des geheimen Namens vergessen. Der Platzhalter steht für die geheimen Versionen.
> Sie sollten den Geltungsbereich der AWS Secrets Manager Richtlinie auch auf die Zertifikate beschränken, die Ihr Cluster für die Bereitstellung von Instances benötigt.

Erstellen Sie die EMR Amazon-Sicherheitskonfiguration für die LDAP Integration

Bevor Sie einen EMR Cluster mit LDAP Integration starten können, verwenden Sie die Schritte unter, Eine Sicherheitskonfiguration erstellen um eine EMR Amazon-Sicherheitskonfiguration für den Cluster zu erstellen. Füllen Sie die folgenden Konfigurationen im `LDAPConfiguration` Block

darunter `AuthenticationConfiguration` oder in den entsprechenden Feldern im Abschnitt EMR Sicherheitskonfigurationen der Amazon-Konsole aus:

`EnableLDAPAuthentication`

Konsolenoption: Authentifizierungsprotokoll: LDAP

Um die LDAP Integration zu verwenden, setzen Sie diese Option auf `true` oder wählen Sie sie als Authentifizierungsprotokoll aus, wenn Sie einen Cluster in der Konsole erstellen. Standardmäßig `EnableLDAPAuthentication` ist dies, `true` wenn Sie eine Sicherheitskonfiguration in der EMR Amazon-Konsole erstellen.

`LDAPServerURL`

Konsolenoption: LDAPServerstandort

Der Standort des LDAP Servers einschließlich des Präfixes:`ldaps://`*`location_of_server`*.

`BindCertificateARN`

Konsolenoption: LDAPSSLZertifikat

Das AWS Secrets Manager ARN, das das Zertifikat zum Signieren des SSL Zertifikats enthält, das der LDAP Server verwendet. Wenn Ihr LDAP Server von einer öffentlichen Zertifizierungsstelle (CA) signiert ist, können Sie eine AWS Secrets Manager ARN leere Datei bereitstellen. Weitere Informationen zum Speichern Ihres Zertifikats in Secrets Manager finden Sie unter TLSSpeichern Sie Zertifikate in AWS Secrets Manager.

`BindCredentialsARN`

Konsolenoption: LDAPServer-Bind-Anmeldeinformationen

Und AWS Secrets Manager ARN das enthält die Bindungsanmeldeinformationen für den LDAP Admin-Benutzer. Die Anmeldeinformationen werden als JSON Objekt gespeichert. In diesem Geheimnis gibt es nur ein Schlüssel-Wert-Paar. Der Schlüssel in dem Paar ist der Benutzername und der Wert ist das Passwort. Beispiel, `{"uid=admin,cn=People,dc=example,dc=com": "AdminPassword1"}`. Dies ist ein optionales Feld, es sei denn, Sie aktivieren die SSH Anmeldung für Ihren EMR Cluster. In vielen Konfigurationen benötigen Active Directory-Instanzen Bind-Anmeldeinformationen, um Benutzer synchronisieren SSSD zu können.

`LDAPAccessFilter`

Konsolenoption: LDAPZugriffsfilter

Gibt die Teilmenge der Objekte auf Ihrem LDAP Server an, die sich authentifizieren können. Wenn Sie beispielsweise allen Benutzern mit der `posixAccount` Objektklasse auf Ihrem LDAP Server Zugriff gewähren möchten, definieren Sie den Zugriffsfilter als. `(objectClass=posixAccount)`

`LDAPUserSearchBase`

Konsolenoption: LDAPBenutzer-Suchbasis

Die Suchbasis, zu der Ihre Benutzer auf Ihrem LDAP Server gehören. Beispiel, `cn=People,dc=example,dc=com`.

`LDAPGroupSearchBase`

Konsolenoption: LDAPGruppensuchbasis

Die Suchbasis, zu der Ihre Gruppen auf Ihrem LDAP Server gehören. Beispiel, `cn=Groups,dc=example,dc=com`.

`EnableSSHLogin`

Konsolenoption: SSHAnmeldung

Gibt an, ob die Kennwortauthentifizierung mit LDAP Anmeldeinformationen zulässig ist oder nicht. Wir empfehlen nicht, dass Sie diese Option aktiviert lassen. Schlüsselpaare sind eine sicherere Route, um den Zugriff auf EMR Cluster zu ermöglichen. Dieses Feld ist optional und standardmäßig auf `false` gesetzt.

`LDAPServerType`

Konsolenoption: LDAPServertyp

Gibt den LDAP Servertyp an, mit dem Amazon EMR eine Verbindung herstellt. Unterstützte Optionen sind Active Directory und OpenLDAP. Andere LDAP Servertypen könnten funktionieren, aber Amazon unterstützt offiziell EMR keine anderen Servertypen. Weitere Informationen finden Sie unter LDAPKomponenten für Amazon EMR.

`ActiveDirectoryConfigurations`

Ein erforderlicher Unterblock für Sicherheitskonfigurationen, die den Active-Directory-Servertyp verwenden.

`ADDomain`

Konsolenoption: Active-Directory-Domain

Der Domänenname, der zur Erstellung des Benutzerprinzipalnamens (UPN) für die Benutzerauthentifizierung mit Sicherheitskonfigurationen verwendet wurde, die den Active Directory-Servertyp verwenden.

Überlegungen zu Sicherheitskonfigurationen mit LDAP und Amazon EMR

- Um eine Sicherheitskonfiguration mit EMR LDAP Amazon-Integration zu erstellen, müssen Sie die Verschlüsselung bei der Übertragung verwenden. Informationen zur Verschlüsselung der Daten während der Übertragung finden Sie unter Verschlüsseln von Daten im Ruhezustand und im Transit.
- Sie können die Kerberos-Konfiguration nicht in derselben Sicherheitskonfiguration definieren. Amazon EMR stellt automatisch KDC ein Thar zur Verfügung und verwaltet das Admin-Passwort dafürKDC. Benutzer können nicht auf dieses Admin-Passwort zugreifen.
- Sie können keine IAM Runtime-Rollen AWS Lake Formation in derselben Sicherheitskonfiguration definieren.
- Die `LDAPServerURL` muss das `ldaps://`-Protokoll in ihrem Wert enthalten.
- Der `LDAPAccessFilter` darf nicht leer sein.

Verwendung LDAP mit der Apache Ranger-Integration für Amazon EMR

Mit der LDAP Integration für Amazon EMR können Sie Apache Ranger weiter integrieren. Wenn Sie Ihre LDAP Benutzer in Ranger abrufen, können Sie diese Benutzer dann einem Apache Ranger-Policy-Server zuordnen, um sie in Amazon EMR und andere Anwendungen zu integrieren. Definieren Sie dazu das `RangerConfiguration` Feld innerhalb `AuthorizationConfiguration` der Sicherheitskonfiguration, das Sie mit Ihrem Cluster verwenden. LDAP Weitere Informationen zum Festlegen der Sicherheitskonfiguration finden Sie unter Erstellen Sie die EMR Sicherheitskonfiguration.

Wenn Sie es LDAP mit Amazon verwendenEMR, müssen Sie `KerberosConfiguration` bei der EMR Amazon-Integration für Apache Ranger keine angeben.

Starten Sie einen EMR Cluster, der sich authentifiziert mit LDAP

Gehen Sie wie folgt vor, um einen EMR Cluster mit LDAP oder Active Directory zu starten.

1. Einrichten Ihrer Umgebung:

- Stellen Sie sicher, dass die Knoten in Ihrem EMR Cluster mit Amazon S3 kommunizieren können und AWS Secrets Manager. Weitere Informationen darüber, wie Sie Ihre EC2 Instance-Profilrolle ändern können, um mit diesen Diensten zu kommunizieren, finden Sie unterAWS Secrets Manager Berechtigungen zur EMR Amazon-Instance-Rolle hinzufügen.
- Wenn Sie planen, Ihren EMR Cluster in einem privaten Subnetz zu betreiben, sollten Sie VPC Amazon-Endpunkte verwenden oder Network Address Transalation (NAT) verwenden, um die Kommunikation mit S3 AWS PrivateLink und VPC Secrets Manager zu konfigurieren. Weitere Informationen finden Sie unter VPCEndpoints AWS PrivateLink und NAT Instances im Amazon VPC Getting Started Guide.
- Stellen Sie sicher, dass zwischen Ihrem EMR Cluster und dem LDAP Server eine Netzwerkverbindung besteht. Ihre EMR Cluster müssen über das Netzwerk auf Ihren LDAP Server zugreifen. Die Primär-, Kern- und Taskknoten für den Cluster kommunizieren mit dem LDAP Server, um Benutzerdaten zu synchronisieren. Wenn Ihr LDAP Server auf Amazon läuftEC2, aktualisieren Sie die EC2 Sicherheitsgruppe so, dass sie Datenverkehr vom EMR Cluster akzeptiert. Weitere Informationen finden Sie unter AWS Secrets Manager Berechtigungen zur EMR Amazon-Instance-Rolle hinzufügen.

2. Erstellen Sie eine EMR Amazon-Sicherheitskonfiguration für die LDAP Integration. Weitere Informationen finden Sie unter Erstellen Sie die EMR Amazon-Sicherheitskonfiguration für die LDAP Integration.
3. Nachdem Sie nun eingerichtet sind, gehen Sie wie unter Starten Sie einen EMR Amazon-Cluster beschrieben vor, um Ihren Cluster mit den folgenden Konfigurationen zu starten:

   - Wählen Sie Amazon EMR Version 6.12 oder höher aus. Wir empfehlen Ihnen, die neueste EMR Amazon-Version zu verwenden.
   - Geben Sie nur Anwendungen für Ihren Cluster an oder wählen Sie sie aus, die dies unterstützenLDAP. Eine Liste der von Amazon LDAP unterstützten Anwendungen finden Sie EMR unterAnwendungsunterstützung und Überlegungen LDAP zu Amazon EMR.
   - Verwenden Sie die Sicherheitskonfiguration, die Sie im vorherigen Schritt erstellt haben.

Beispiele für die Verwendung LDAP mit Amazon EMR

Sobald Sie einen EMR Cluster bereitgestellt haben, der LDAP Integration verwendet, können Sie Ihre LDAP Anmeldeinformationen über den integrierten Authentifizierungsmechanismus für Benutzernamen und Passwörter für jede unterstützte Anwendung bereitstellen. Diese Seite zeigt einige Beispiele.

Verwenden Sie die LDAP Authentifizierung mit Apache Hive

Example – Apache Hive

Der folgende Beispielbefehl startet eine Apache Hive-Sitzung über HiveServer 2 und Beeline:

```
beeline -u "jdbc:hive2://$HOSTNAME:10000/default;ssl=true;sslTrustStore=
$TRUSTSTORE_PATH;trustStorePassword=$TRUSTSTORE_PASS"  -n LDAP_USERNAME -
p LDAP_PASSWORD
```

Verwendung der LDAP Authentifizierung mit Apache Livy

Example – Apache Livy

Der folgende Beispielbefehl startet eine Livy-Sitzung über c. URL Ersetzen Sie *ENCODED-KEYPAIR* durch eine Base64-kodierte Zeichenfolge für `username:password`.

```
curl -X POST --data '{"proxyUser":"LDAP_USERNAME","kind": "pyspark"}' -H "Content-Type:
 application/json" -H "Authorization: Basic ENCODED-KEYPAIR" DNS_OF_PRIMARY_NODE:8998/
sessions
```

Verwendung der LDAP Authentifizierung mit Presto

Example – Presto

Der folgende Beispielbefehl startet eine Presto-Sitzung über Presto: CLI

```
presto-cli --user "LDAP_USERNAME" --password --catalog hive
```

Nachdem Sie diesen Befehl ausgeführt haben, geben Sie das LDAP Passwort an der Eingabeaufforderung ein.

Verwenden Sie die LDAP Authentifizierung mit Trino

Example – Trino

Der folgende Beispielbefehl startet eine Trino-Sitzung über Trino: CLI

```
trino-cli --user "LDAP_USERNAME" --password --catalog hive
```

Nachdem Sie diesen Befehl ausgeführt haben, geben Sie das LDAP Passwort an der Eingabeaufforderung ein.

Verwenden Sie die LDAP Authentifizierung mit Hue

Sie können über einen SSH Tunnel, den Sie auf dem Cluster erstellen, auf die Hue-Benutzeroberfläche zugreifen, oder Sie können einen Proxyserver einrichten, der die Verbindung zu Hue öffentlich überträgt. Da Hue standardmäßig nicht im HTTPS Modus läuft, empfehlen wir die Verwendung einer zusätzlichen Verschlüsselungsebene, um sicherzustellen, dass die Kommunikation zwischen den Clients und der Hue-Benutzeroberfläche verschlüsselt istHTTPS. Dadurch wird die Wahrscheinlichkeit verringert, dass Sie versehentlich Benutzeranmeldeinformationen im Klartext preisgeben.

Um die Hue-Benutzeroberfläche zu verwenden, öffnen Sie die Hue-Benutzeroberfläche in Ihrem Browser und geben Sie Ihren LDAP Nutzernamen und Ihr Passwort ein, um sich anzumelden. Wenn die Anmeldeinformationen korrekt sind, meldet Hue Sie an und verwendet Ihre Identität, um Sie bei allen unterstützten Anwendungen zu authentifizieren.

Wird SSH für die Passwortauthentifizierung und Kerberos-Tickets für andere Anwendungen verwendet

> **Important**
>
> Wir empfehlen nicht, die Passwortauthentifizierung für den Zugang zu einem EMR Cluster SSH zu verwenden.

Sie können Ihre LDAP Anmeldeinformationen verwenden, SSH um einem EMR Cluster beizutreten. Stellen Sie dazu die `EnableSSHLogin` Konfiguration `true` in der EMR Amazon-Sicherheitskonfiguration, die Sie zum Starten des Clusters verwenden, auf ein. Verwenden Sie dann den folgenden Befehl, SSH um den Cluster nach dem Start aufzurufen:

```
ssh username@EMR_PRIMARY_DNS_NAME
```

Nachdem Sie diesen Befehl ausgeführt haben, geben Sie das LDAP Passwort an der Eingabeaufforderung ein.

Amazon EMR bietet ein Cluster-Skript, das es Benutzern ermöglicht, eine Kerberos-Keytab-Datei und ein Ticket für die Verwendung mit unterstützten Anwendungen zu generieren, die Anmeldeinformationen nicht direkt akzeptieren. LDAP Zu diesen Anwendungen gehören Spark `spark-submit` und. SQL PySpark

Führen Sie `ldap-kinit` aus und befolgen Sie die Eingabeaufforderungen. Wenn die Authentifizierung erfolgreich ist, wird die Kerberos-Keytab-Datei mit einem gültigen Kerberos-Ticket in Ihrem Home-Verzeichnis angezeigt. Verwenden Sie das Kerberos-Ticket, um Anwendungen wie in jeder Kerberized-Umgebung auszuführen.

Integrieren Sie Amazon EMR mit AWS IAM Identity Center

Mit EMR Amazon-Versionen 6.15.0 und höher können Sie Identitäten von verwenden, um sich bei einem AWS IAM Identity Center Amazon-Cluster zu authentifizieren. EMR Die folgenden Abschnitte bieten einen konzeptionellen Überblick, die Voraussetzungen und die erforderlichen Schritte, um einen EMR Cluster mit Identity Center-Integration zu starten.

Themen

Übersicht

Die vertrauenswürdige Weitergabe von Identitäten über IAM Identity Center kann Ihnen dabei helfen, die Identitäten Ihrer Mitarbeiter sicher zu erstellen oder zu verknüpfen und deren Zugriff über AWS Konten und Anwendungen hinweg zentral zu verwalten. Mit dieser Funktion kann sich ein Benutzer bei der Anwendung anmelden, die Trusted Identity Propagation verwendet, und diese Anwendung kann die Identität des Benutzers bei Anfragen weitergeben, die sie stellt, um auf Daten in AWS Diensten zuzugreifen, die ebenfalls Trusted Identity Propagation verwenden. Da der Zugriff auf der Identität eines Benutzers basiert, müssen Benutzer keine lokalen Benutzeranmeldedaten für die Datenbank verwenden oder eine IAM Rolle übernehmen, um auf Daten zuzugreifen.

Identity Center ist der empfohlene Ansatz für die Authentifizierung und Autorisierung von Mitarbeitern AWS für Unternehmen jeder Größe und Art. Mit Identity Center können Sie Benutzeridentitäten in Ihrer vorhandenen Identitätsquelle AWS, einschließlich Microsoft Active Directory, Okta, Ping Identity, Google Workspace und Microsoft Entra ID (ehemals Azure AD) JumpCloud, erstellen und verwalten oder eine Verbindung zu Ihrer vorhandenen Identitätsquelle herstellen.

Weitere Informationen finden Sie unter Was ist? AWS IAM Identity Center und Verbreitung vertrauenswürdiger Identitäten zwischen Anwendungen im AWS IAM Identity Center Benutzerhandbuch.

Features und Vorteile

Die EMR Amazon-Integration mit IAM Identity Center bietet die folgenden Vorteile:

- Amazon EMR stellt Anmeldeinformationen bereit, um Ihre Identity Center-Identität an einen EMR Cluster weiterzuleiten.
- Amazon EMR konfiguriert alle unterstützten Anwendungen so, dass sie sich mit den Cluster-Anmeldeinformationen authentifizieren.
- Amazon EMR konfiguriert und verwaltet die unterstützte Anwendungssicherheit mit dem Kerberos-Protokoll, ohne dass Sie Befehle oder Skripts benötigen.
- Die Möglichkeit, die Amazon-S3-Autorisierung auf Präfixebene mit Identity-Center-Identitäten auf von S3 Access Grants verwalteten S3-Präfixen durchzusetzen.
- Die Möglichkeit, die Autorisierung auf Tabellenebene mit Identity Center-Identitäten für AWS Lake Formation verwaltete AWS Glue-Tabellen durchzusetzen.

Erste Schritte mit der AWS IAM Identity Center Integration für Amazon EMR

Dieser Abschnitt hilft Ihnen bei der Konfiguration von Amazon EMR für die Integration mit AWS IAM Identity Center.

Themen

- Eine Identity-Center-Instance erstellen
- Erstellen Sie eine IAM Rolle für Identity Center
- Eine Identity-Center-fähige Sicherheitskonfiguration erstellen
- Einen Identity-Center-fähigen Cluster erstellen und starten
- Lake Formation für einen IAM Identity Center-fähigen EMR Cluster konfigurieren
- Arbeiten mit S3 Access Grants auf einem IAM Identity Center-fähigen EMR Cluster

Eine Identity-Center-Instance erstellen

Wenn Sie noch keine haben, erstellen Sie dort, AWS-Region wo Sie Ihren EMR Cluster starten möchten, eine Identity Center-Instance. Eine Identity-Center-Instance kann nur in einer einzigen Region für ein AWS-Konto existieren.

Verwenden Sie den folgenden AWS CLI Befehl, um eine neue Instanz mit dem Namen zu erstellen*MyInstance*:

```
aws sso-admin create-instance --name MyInstance
```

Erstellen Sie eine IAM Rolle für Identity Center

Um Amazon zu EMR integrieren AWS IAM Identity Center, erstellen Sie eine IAM Rolle, die sich über den EMR Cluster bei Identity Center authentifiziert. Unter der Haube EMR verwendet Amazon SigV4 Anmeldeinformationen, um die Identity Center-Identität an nachgelagerte Dienste weiterzuleiten, wie AWS Lake Formation z. Ihre Rolle sollte auch über die entsprechenden Berechtigungen zum Aufrufen der nachgelagerten Services verfügen.

Verwenden Sie die folgende Berechtigungsrichtlinie zum Erstellen der Rolle:

```
{
  "Statement": [
    {
      "Sid": "IdCPermissions",
      "Effect": "Allow",
      "Action": [
        "sso-oauth:*"
      ],
      "Resource": "*"
    },
    {
      "Sid": "GlueandLakePermissions",
      "Effect": "Allow",
      "Action": [
        "glue:*",
        "lakeformation:GetDataAccess"
      ],
      "Resource": "*"
    },
    {
```

```
      "Sid": "AccessGrantsPermissions",
      "Effect": "Allow",
      "Action": [
        "s3:GetDataAccess",
        "s3:GetAccessGrantsInstanceForPrefix"
      ],
      "Resource": "*"
    }
  ]
}
```

Die Vertrauensrichtlinie für diese Rolle ermöglicht es der InstanceProfile-Rolle, sie die Rolle übernehmen zu lassen.

```
{
    "Sid": "AssumeRole",
    "Effect": "Allow",
    "Principal": {
        "AWS": "arn:aws:iam::12345678912:role/EMR_EC2_DefaultRole"
    },
    "Action": [
        "sts:AssumeRole",
        "sts:SetContext"
    ]
}
```

Wenn die Rolle keine vertrauenswürdigen Anmeldeinformationen hat und auf eine durch Lake Formation geschützte Tabelle zugreift, setzt Amazon den Wert `principalId` der angenommenen Rolle EMR automatisch auf. *`userID`*`-untrusted` Im Folgenden finden Sie einen Auszug aus einem Ereignis, das den anzeigt. CloudTrail `principalId`

```
{
    "eventVersion": "1.09",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "ABCDEFGH1JKLMNO2PQR3TU:5000-untrusted",
        "arn": "arn:aws:sts::123456789012:assumed-role/EMR_TIP/5000-untrusted",
        "accountId": "123456789012",
        "accessKeyId": "ABCDEFGH1IJKLMNOPQ7R3"
        ...
```

Eine Identity-Center-fähige Sicherheitskonfiguration erstellen

Um einen EMR Cluster mit IAM Identity Center-Integration zu starten, verwenden Sie den folgenden Beispielbefehl, um eine EMR Amazon-Sicherheitskonfiguration zu erstellen, für die Identity Center aktiviert ist. Jede Konfiguration wird im Folgenden erklärt.

```
aws emr create-security-configuration --name "IdentityCenterConfiguration-with-lf-
accessgrants" --region "us-west-2" --security-configuration '{
 "AuthenticationConfiguration":{
  "IdentityCenterConfiguration":{
   "EnableIdentityCenter":true,
   "IdentityCenterApplicationAssigmentRequired":false,
   "IdentityCenterInstanceARN": "arn:aws:sso:::instance/ssoins-123xxxxxxxxx789",
   "IAMRoleForEMRIdentityCenterApplicationARN": "arn:aws:iam::123456789012:role/tip-
role"
     }
 },
 "AuthorizationConfiguration": {
  "LakeFormationConfiguration": {
   "EnableLakeFormation": true
  }
 },
 "EncryptionConfiguration": {
  "EnableInTransitEncryption": true,
  "EnableAtRestEncryption": false,
  "InTransitEncryptionConfiguration": {
   "TLSCertificateConfiguration": {
    "CertificateProviderType": "PEM",
    "S3Object": "s3://my-bucket/cert/my-certs.zip"
   }
  }
 }
}'
```

- **`EnableIdentityCenter`** – (erforderlich) Aktiviert die Identity-Center-Integration.
- **`IdentityCenterApplicationARN`**— (erforderlich) Die Identity Center-InstanzARN.
- **`IAMRoleForEMRIdentityCenterApplicationARN`**— (erforderlich) Die IAM Rolle, die Identity Center-Token aus dem Cluster beschafft.
- **`IdentityCenterApplicationAssignmentRequired`** – (boolean) Legt fest, ob für die Nutzung der Identity-Center-Anwendung eine Zuweisung erforderlich ist. Der Standardwert ist `true`.

- **`AuthorizationConfiguration`/`LakeFormationConfiguration`**— Konfigurieren Sie optional die Autorisierung:
  - **`EnableLakeFormation`** – Aktivieren Sie die Lake-Formation-Autorisierung auf dem Cluster.

Um die Identity Center-Integration mit Amazon zu aktivierenEMR, müssen Sie `EncryptionConfiguration` und angeben`IntransitEncryptionConfiguration`.

Einen Identity-Center-fähigen Cluster erstellen und starten

Nachdem Sie nun die IAM Rolle eingerichtet haben, die sich bei Identity Center authentifiziert, und eine EMR Amazon-Sicherheitskonfiguration erstellt haben, für die Identity Center aktiviert ist, können Sie Ihren identitätsbewussten Cluster erstellen und starten. Schritte zum Starten Ihres Clusters mit der erforderlichen Sicherheitskonfiguration finden Sie unter Angabe einer Sicherheitskonfiguration für einen Cluster.

Lesen Sie optional den folgenden Abschnitt, wenn Sie Ihren Identity Center-fähigen Cluster mit anderen Sicherheitsoptionen verwenden möchten, die Amazon EMR unterstützt:

- Arbeiten mit S3 Access Grants auf einem IAM Identity Center-fähigen EMR Cluster
- Lake Formation für einen IAM Identity Center-fähigen EMR Cluster konfigurieren

Lake Formation für einen IAM Identity Center-fähigen EMR Cluster konfigurieren

Sie können die AWS Lake FormationIntegration in Ihren AWS IAM Identity Center aktivierten EMR Cluster durchführen.

Stellen Sie zunächst sicher, dass Sie eine Identity-Center-Instance in derselben Region wie Ihr Cluster eingerichtet haben. Weitere Informationen finden Sie unter Eine Identity-Center-Instance erstellen. Sie finden die Instanz ARN in der IAM Identity Center-Konsole, wenn Sie sich die Instanzdetails ansehen, oder verwenden Sie den folgenden Befehl, um Details für all Ihre Instances von der aus anzuzeigenCLI:

```
aws sso-admin list-instances
```

Verwenden Sie dann die ARN und Ihre AWS Konto-ID mit dem folgenden Befehl, um Lake Formation so zu konfigurieren, dass es mit IAM Identity Center kompatibel ist:

```
aws lakeformation create-lake-formation-identity-center-configuration --cli-input-json
 file://create-lake-fromation-idc-config.json
json input:
{
    "CatalogId": "account-id/org-account-id",
    "InstanceArn": "identity-center-instance-arn"
}
```

Rufen Sie jetzt `put-data-lake-settings` auf und aktivieren Sie `AllowFullTableExternalDataAccess` mit Lake Formation:

```
aws lakeformation put-data-lake-settings --cli-input-json file://put-data-lake-
settings.json
json input:
{
    "DataLakeSettings": {
        "DataLakeAdmins": [
            {
                "DataLakePrincipalIdentifier": "admin-ARN"
            }
        ],
        "CreateDatabaseDefaultPermissions": [...],
        "CreateTableDefaultPermissions": [...],
        "AllowExternalDataFiltering": true,
        "AllowFullTableExternalDataAccess": true
    }
}
```

Erteilen Sie abschließend der Identität des Benutzers, der auf ARN den EMR Cluster zugreift, vollständige Tabellenberechtigungen. Das ARN enthält die Benutzer-ID von Identity Center. Navigieren Sie in der Konsole zu Identity Center, wählen Sie Users (Benutzer) und dann den Benutzer aus, um dessen allgemeine Informationseinstellungen anzuzeigen.

Kopieren Sie die Benutzer-ID und fügen Sie sie in das folgende Feld ARN ein*user-id*:

```
arn:aws:identitystore:::user/user-id
```

> **Note**
>
> Abfragen auf dem EMR Cluster funktionieren nur, wenn die IAM Identity Center-Identität vollen Tabellenzugriff auf die geschützte Lake Formation-Tabelle hat. Wenn die Identität keinen vollständigen Tabellenzugriff hat, schlägt die Abfrage fehl.

Verwenden Sie den folgenden Befehl, um dem Benutzer vollen Tabellenzugriff zu gewähren:

```
aws lakeformation grant-permissions --cli-input-json file://grantpermissions.json
json input:
{
    "Principal": {
        "DataLakePrincipalIdentifier": "arn:aws:identitystore:::user/user-id"
    },
    "Resource": {
        "Table": {
            "DatabaseName": "tip_db",
            "Name": "tip_table"
        }
    },
    "Permissions": [
        "ALL"
    ],
    "PermissionsWithGrantOption": [
        "ALL"
    ]
}
```

Arbeiten mit S3 Access Grants auf einem IAM Identity Center-fähigen EMR Cluster

Sie können S3 Access Grants in Ihren AWS IAM Identity Center aktivierten EMR Cluster integrieren.

Verwenden Sie S3 Access Grants, um den Zugriff auf Ihre Datensätze von Clustern aus zu autorisieren, die Identity Center verwenden. Erstellen Sie Zuschüsse, um die Berechtigungen zu erweitern, die Sie für IAM Benutzer, Gruppen, Rollen oder für ein Unternehmensverzeichnis festlegen. Weitere Informationen finden Sie unter Verwenden von S3 Access Grants mit Amazon EMR.

Themen

- S3-Access-Grants-Instance und -Standort erstellen

- Grants für Identity-Center-Identitäten erstellen

S3-Access-Grants-Instance und -Standort erstellen

Falls Sie noch keine haben, erstellen Sie dort, AWS-Region wo Sie Ihren EMR Cluster starten möchten, eine S3 Access Grants-Instance.

Verwenden Sie den folgenden AWS CLI Befehl, um eine neue Instanz mit dem Namen zu erstellen*MyInstance*:

```
aws s3control-access-grants create-access-grants-instance \
--account-id 12345678912 \
--identity-center-arn "identity-center-instance-arn" \
```

Erstellen Sie dann einen S3-Access-Grants-Standort und ersetzen Sie die roten Werte durch Ihre eigenen:

```
aws s3control-access-grants create-access-grants-location \
--account-id 12345678912 \
--location-scope s3:// \
--iam-role-arn "access-grant-role-arn" \
--region aa-example-1
```

> **Note**
>
> Definieren Sie den `iam-role-arn` Parameter als `accessGrantRole`ARN.

Grants für Identity-Center-Identitäten erstellen

Erstellen Sie abschließend die Grants für die Identitäten, die Zugriff auf Ihren Cluster haben:

```

aws s3control-access-grants create-access-grant \
--account-id 12345678912 \
--access-grants-location-id "default" \
--access-grants-location-configuration S3SubPrefix="s3-bucket-prefix"
--permission READ \
--grantee GranteeType=DIRECTORY_USER,GranteeIdentifier="your-identity-center-user-id"
```

Beispielausgabe:

```
{
"CreatedAt": "2023-09-21T23:47:24.870000+00:00",
"AccessGrantId": "1234-12345-1234-1234567",
"AccessGrantArn": "arn:aws:s3:aa-example-1-1:123456789012:access-grants/default/grant/
xxxx1234-1234-5678-1234-1234567890",
"Grantee": {
"GranteeType": "DIRECTORY_USER",
"GranteeIdentifier": "5678-56789-5678-567890"
},
"AccessGrantsLocationId": "default",
"AccessGrantsLocationConfiguration": {
"S3SubPrefix": "myprefix/*"
},
"Permission": "READ",
"GrantScope": "s3://myprefix/*"
}
```

Überlegungen und Einschränkungen für Amazon EMR bei der Identity Center-Integration

Beachten Sie die folgenden Punkte, wenn Sie IAM Identity Center mit Amazon verwendenEMR:

- Die Verbreitung vertrauenswürdiger Identitäten über Identity Center wird auf Amazon EMR 6.15.0 und höher und nur mit Apache Spark unterstützt.
- Um EMR Cluster mit vertrauenswürdiger Identitätsverbreitung zu aktivieren, müssen Sie die verwenden, AWS CLI um eine Sicherheitskonfiguration zu erstellen, für die vertrauenswürdige Identitätsverbreitung aktiviert ist, und diese Sicherheitskonfiguration verwenden, wenn Sie Ihren Cluster starten. Weitere Informationen finden Sie unter Eine Identity-Center-fähige Sicherheitskonfiguration erstellen.
- EMRCluster, die Trusted Identity Propagation verwenden, können nur Dienste aufrufen, die auch Trusted Identity Propagation verwenden.
- Für EMR Cluster, die die Weitergabe vertrauenswürdiger Identitäten verwenden, AWS Lake Formation ist nur eine Zugriffskontrolle auf Tabellenebene verfügbar, die auf basiert.
- Bei EMR Clustern, die vertrauenswürdige Identitätsverbreitung verwenden, gehören zu den Vorgängen, die die Zugriffskontrolle auf der Grundlage von Lake Formation mit Apache Spark unterstützen `SELECT`ALTER `TABLE`, und`DROP TABLE`.

- Bei EMR Clustern, die vertrauenswürdige Identitätsverbreitung verwenden, enthalten Lake Formation Formation-basierte Zugriffskontrollen, die von Apache Spark nicht unterstützt werden, `INSERT` Anweisungen.
- Die Weitergabe vertrauenswürdiger Identitäten mit Amazon EMR wird in den folgenden Bereichen unterstützt AWS-Regionen:
  - `ap-east-1` – Asien-Pazifik (Hongkong)
  - `ap-northeast-1` – Asien-Pazifik (Tokio)
  - `ap-northeast-2` – Asien-Pazifik (Seoul)
  - `ap-south-1` – Asien-Pazifik (Mumbai)
  - `ap-southeast-1` – Asien-Pazifik (Singapur)
  - `ap-southeast-2` – Asien-Pazifik (Sydney)
  - `ca-central-1` – Kanada (Zentral)
  - `eu-central-1` – Europa (Frankfurt)
  - `eu-north-1` – Europa (Stockholm)
  - `eu-west-1` – Europa (Irland)
  - `eu-west-2` – Europa (London)
  - `eu-west-3` – Europa (Paris)
  - `me-south-1` – Naher Osten (Bahrain)
  - `sa-east-1` – Südamerika (São Paulo)
  - `us-east-1` – USA Ost (Nord-Virginia)
  - `us-east-2` – USA Ost (Ohio)
  - `us-west-1` – USA West (Nordkalifornien)
  - `us-west-2` – USA West (Oregon)

Integrieren Sie Amazon EMR mit AWS Lake Formation

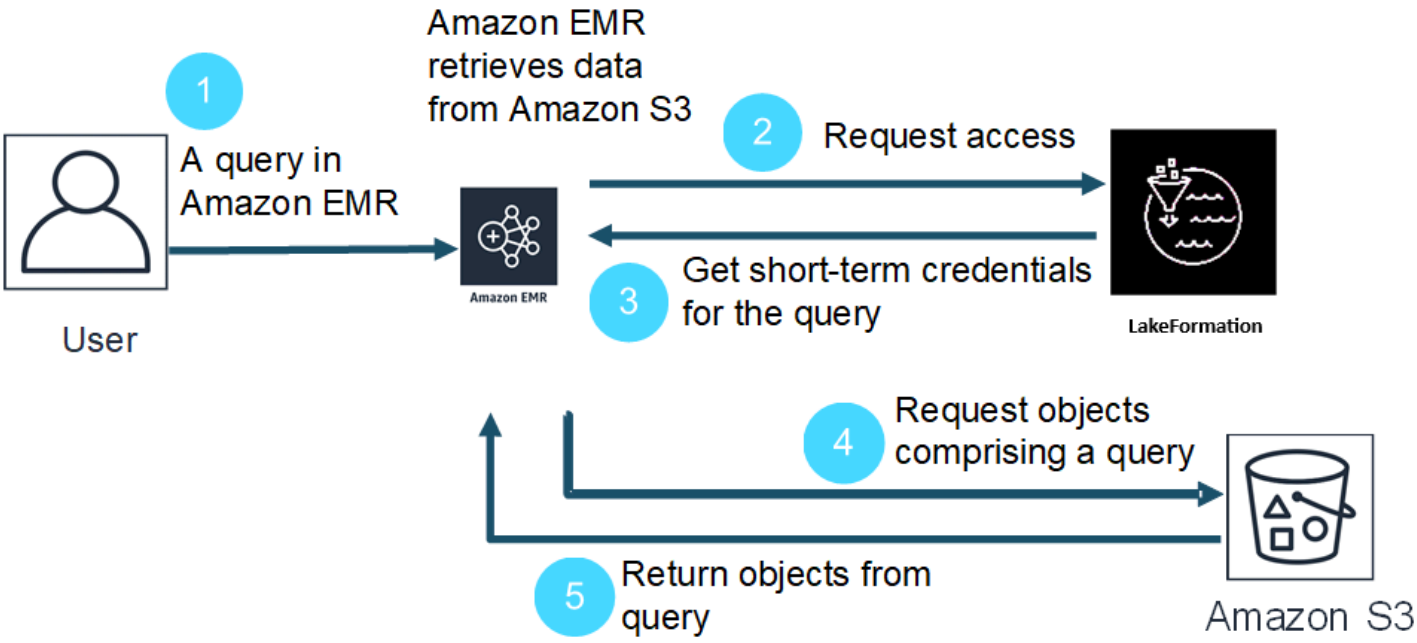
AWS Lake Formation ist ein verwalteter Service, der Sie dabei unterstützt, Daten in einem Amazon Simple Storage Service (S3) Data Lake zu entdecken, zu katalogisieren, zu bereinigen und zu sichern. Lake Formation bietet einen detaillierten Zugriff auf Spaltenebene auf Datenbanken und Tabellen im Glue-Datenkatalog. AWS Weitere Informationen finden Sie unter Was ist AWS Lake Formation?

Mit EMR Amazon-Version 6.7.0 und höher können Sie die auf Lake Formation basierende Zugriffskontrolle auf Spark-, Hive- und Presto-Jobs anwenden, die Sie an Amazon-Cluster senden. EMR Für die Integration mit Lake Formation müssen Sie einen EMR Cluster mit einer Runtime-Rolle erstellen. Eine Runtime-Rolle ist eine AWS Identity and Access Management (IAM) -Rolle, die Sie EMR Amazon-Jobs oder -Abfragen zuordnen. Amazon verwendet diese Rolle EMR dann für den Zugriff auf AWS Ressourcen. Weitere Informationen finden Sie unter EMRSchritte zu Runtime-Rollen für Amazon.

Wie Amazon mit Lake Formation EMR zusammenarbeitet

Nachdem Sie Amazon EMR in Lake Formation integriert haben, können Sie Abfragen an EMR Amazon-Cluster mit `StepAPI`oder mit SageMaker Studio ausführen. Anschließend bietet Lake Formation Zugriff auf Daten über temporäre Anmeldeinformationen für AmazonEMR. Dieser Prozess wird als Anmeldeinformationsvergabe bezeichnet. Weitere Informationen finden Sie unter Was ist AWS Lake Formation?

Im Folgenden finden Sie einen allgemeinen Überblick darüber, wie Amazon Zugriff auf Daten EMR erhält, die durch die Sicherheitsrichtlinien von Lake Formation geschützt sind.



1. Ein Benutzer sendet eine EMR Amazon-Anfrage für Daten in Lake Formation.
2. Amazon EMR fordert temporäre Anmeldeinformationen von Lake Formation an, um den Benutzerdaten Zugriff zu gewähren.
3. Lake Formation gibt temporäre Anmeldeinformationen zurück.

4. Amazon EMR sendet die Abfrageanforderung zum Abrufen von Daten aus Amazon S3.
5. Amazon EMR empfängt die Daten von Amazon S3, filtert sie und gibt Ergebnisse zurück, die auf den Benutzerberechtigungen basieren, die der Benutzer in Lake Formation definiert hat.

Weitere Informationen zum Hinzufügen von Benutzern und Gruppen zu Lake Formation-Richtlinien finden Sie unter Erteilen von Datenkatalogberechtigungen.

Voraussetzungen

Sie müssen die folgenden Anforderungen erfüllen, bevor Sie Amazon EMR und Lake Formation integrieren können:

- Aktivieren Sie die Autorisierung von Runtime-Rollen in Ihrem EMR Amazon-Cluster.
- Verwenden Sie den AWS Glue-Datenkatalog als Ihren Metadatenspeicher.
- Definieren und verwalten Sie in Lake Formation Berechtigungen für den Zugriff auf Datenbanken, Tabellen und Spalten im AWS Glue Data Catalog. Weitere Informationen finden Sie unter Was ist AWS Lake Formation?

Themen

Aktivieren Sie Lake Formation mit Amazon EMR

Wenn Sie mit Amazon EMR 6.15.0 und höher Spark-Jobs auf Amazon auf EC2 Clustern ausführen, die EMR auf Daten im AWS Glue-Datenkatalog zugreifen, können Sie AWS Lake Formation damit Berechtigungen auf Tabellen-, Zeilen-, Spalten- und Zellenebene auf Hudi-, Iceberg- oder Delta Lake-basierte Tabellen anwenden.

In diesem Abschnitt erfahren Sie, wie Sie eine Sicherheitskonfiguration erstellen und Lake Formation für die Zusammenarbeit mit Amazon einrichtenEMR. Wir gehen auch darauf ein, wie Sie einen Cluster mit der Sicherheitskonfiguration starten, die Sie für Lake Formation erstellt haben.

Schritt 1: Richten Sie eine Runtime-Rolle für Ihren EMR Cluster ein

Um eine Runtime-Rolle für Ihren EMR Cluster zu verwenden, müssen Sie eine Sicherheitskonfiguration erstellen. Mit einer Sicherheitskonfiguration können Sie konsistente Sicherheits-, Autorisierungs- und Authentifizierungsoptionen für alle Ihre Cluster anwenden.

1. Erstellen Sie eine Datei mit dem Namen `lf-runtime-roles-sec-cfg.json` und den folgenden Inhalten.

```
{
    "AuthorizationConfiguration": {
        "IAMConfiguration": {
            "EnableApplicationScopedIAMRole": true,
            "ApplicationScopedIAMRoleConfiguration": {
                "PropagateSourceIdentity": true
            }
        },
        "LakeFormationConfiguration": {
            "AuthorizedSessionTagValue": "Amazon EMR"
        }
    },
    "EncryptionConfiguration": {
            "EnableInTransitEncryption": true,
            "InTransitEncryptionConfiguration": {
            "TLSCertificateConfiguration": {<certificate-configuration>}
        }
    }
}
```

2. Um als Nächstes sicherzustellen, dass das Sitzungs-Tag Lake Formation autorisieren kann, setzen Sie die `LakeFormationConfiguration/AuthorizedSessionTagValue`-Eigenschaft auf `Amazon EMR`.
3. Verwenden Sie den folgenden Befehl, um die EMR Amazon-Sicherheitskonfiguration zu erstellen.

```
aws emr create-security-configuration \
--name 'iamconfig-with-iam-lf' \
--security-configuration file://lf-runtime-roles-sec-cfg.json
```

Alternativ können Sie die EMRAmazon-Konsole verwenden, um eine Sicherheitskonfiguration mit benutzerdefinierten Einstellungen zu erstellen.

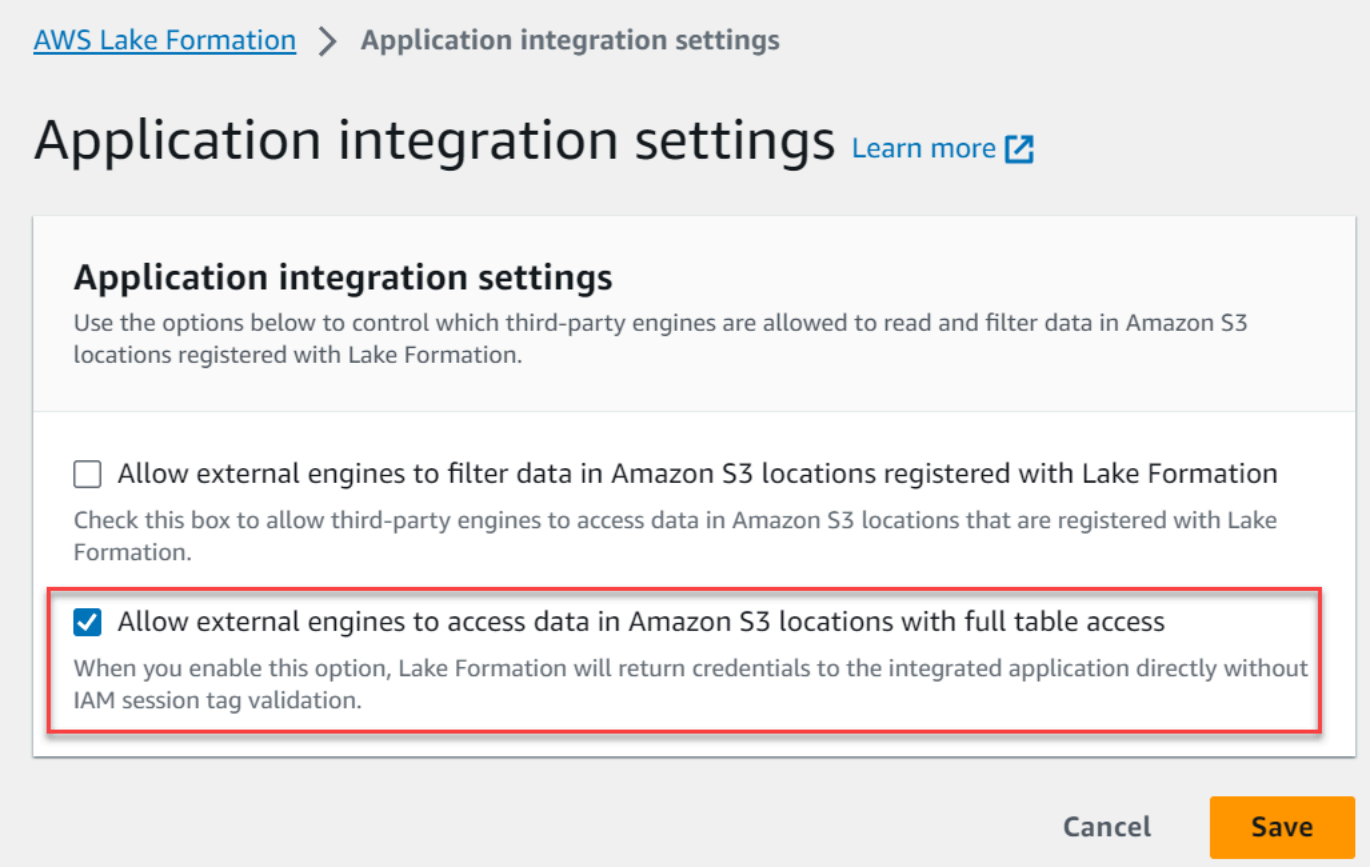
Schritt 2: Starten Sie einen EMR Amazon-Cluster

Jetzt sind Sie bereit, einen EMR Cluster mit der Sicherheitskonfiguration zu starten, die Sie im vorherigen Schritt erstellt haben. Weitere Informationen zum Erstellen einer Sicherheitskonfiguration finden Sie unter Sicherheitskonfigurationen zum Einrichten der Cluster-Sicherheit verwenden und EMRSchritte zu Runtime-Rollen für Amazon.

Schritt 3a: Lake Formation Formation-basierte Berechtigungen auf Tabellenebene mit Amazon-Runtime-Rollen einrichten EMR

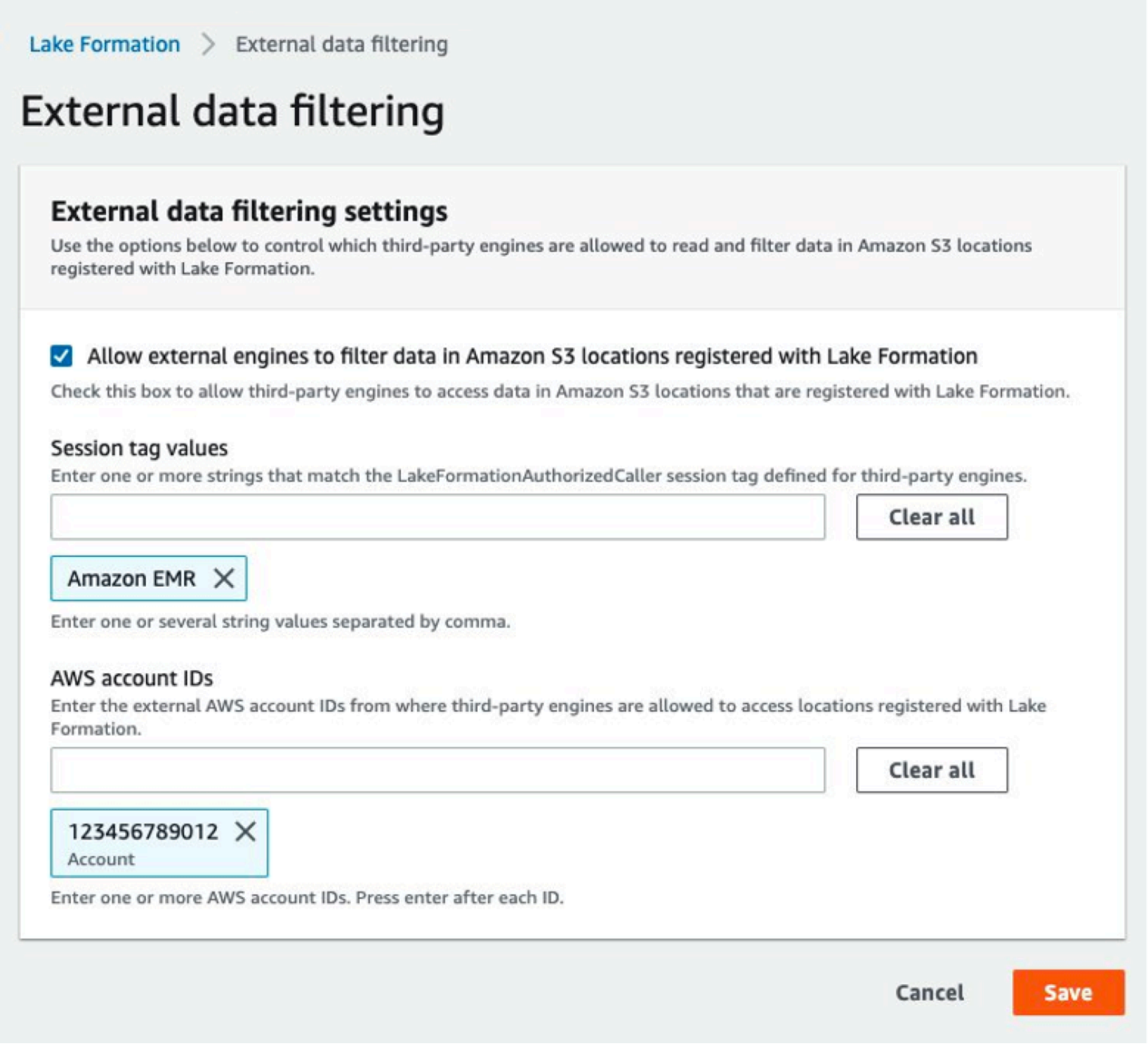
Wenn Sie keine differenzierte Zugriffskontrolle auf Spalten-, Zeilen- oder Zellenebene benötigen, können Sie mit Glue-Datenkatalog Berechtigungen auf Tabellenebene einrichten. Um den Zugriff auf Tabellenebene zu aktivieren, navigieren Sie zur AWS Lake Formation Konsole und wählen Sie im Bereich Administration in der Seitenleiste die Option Anwendungsintegrationseinstellungen aus. Aktivieren Sie dann die folgende Option und wählen Sie Save (Speichern) aus:

Allow external engines to access data in Amazon S3 locations with full table access (Externen Engines den Zugriff auf Daten an Amazon-S3-Standorten mit vollständigem Tabellenzugriff erlauben)



Schritt 3b: Auf Lake Formation basierende Berechtigungen auf Spalten-, Zeilen- oder Zellenebene mit Amazon-Runtime-Rollen einrichten EMR

Um Berechtigungen auf Tabellen- und Spaltenebene mit Lake Formation anzuwenden, muss der Data-Lake-Administrator für Lake Formation `Amazon EMR` als Wert für die Sitzungs-Tag-Konfiguration `AuthorizedSessionTagValue` festlegen. Lake Formation verwendet dieses Sitzungs-Tag, um Anrufer zu autorisieren und Zugriff auf den Data Lake zu gewähren. Sie können dieses Sitzungs-Tag im Abschnitt Externe Datenfilterung der Lake-Formation-Konsole festlegen. Ersetzen *123456789012* mit Ihrer eigenen ID. AWS-Konto



Schritt 4: AWS Glue- und Lake Formation Formation-Grants für EMR Amazon-Runtime-Rollen konfigurieren

Um mit der Einrichtung der auf Lake Formation basierenden Zugriffskontrolle mit EMR Amazon-Runtime-Rollen fortzufahren, müssen Sie AWS Glue- und Lake Formation Formation-Grants für EMR Amazon-Runtime-Rollen konfigurieren. Damit Ihre IAM Runtime-Rollen mit Lake Formation interagieren können, gewähren Sie ihnen Zugriff mit `lakeformation:GetDataAccess` und`glue:Get*`.

Lake Formation Formation-Berechtigungen kontrollieren den Zugriff auf AWS Glue Data Catalog-Ressourcen, Amazon S3 S3-Standorte und die zugrunde liegenden Daten an diesen Standorten. IAMBerechtigungen kontrollieren den Zugriff auf Lake Formation und AWS Glue APIs sowie auf Ressourcen. Obwohl Sie möglicherweise über die Lake Formation Formation-Berechtigung verfügen, auf eine Tabelle im Datenkatalog (SELECT) zuzugreifen, schlägt Ihr Vorgang fehl, wenn Sie nicht über die IAM Berechtigung für die verfügen `glue:Get*`API. Weitere Informationen zur Zugriffskontrolle für Lake Formation finden Sie unter Übersicht über die Zugriffskontrolle für Lake Formation.

1. Erstellen Sie die Datei e`mr-runtime-roles-lake-formation-policy.json` mit folgendem Inhalt.

```
{
    "Version": "2012-10-17",
    "Statement": {
        "Sid": "LakeFormationManagedAccess",
        "Effect": "Allow",
        "Action": [
            "lakeformation:GetDataAccess",
            "glue:Get*",
            "glue:Create*",
            "glue:Update*"
        ],
        "Resource": "*"
    }
}
```

2. Erstellen Sie die zugehörige IAM Richtlinie.

```
aws iam create-policy \
--policy-name emr-runtime-roles-lake-formation-policy \
```

```
--policy-document file://emr-runtime-roles-lake-formation-policy.json
```

3. Um diese Richtlinie Ihren IAM Runtime-Rollen zuzuweisen, folgen Sie den Schritten unter AWS Lake Formation Berechtigungen verwalten.

Sie können jetzt Laufzeit-Rollen und Lake Formation verwenden, um Berechtigungen auf Tabellen- und Spaltenebene anzuwenden. Sie können auch eine Quellidentität verwenden, um Aktionen zu steuern und Vorgänge zu überwachen AWS CloudTrail. Ein detailliertes end-to-end Beispiel finden Sie in den EMRSchritten Einführung von Runtime-Rollen für Amazon.

Apache Hudi und Lake Formation

Die EMR Amazon-Versionen 6.15.0 und höher bieten Unterstützung für eine differenzierte Zugriffskontrolle, die auf Apache Hudi basiert, wenn Sie Daten AWS Lake Formation mit Spark lesen und schreiben. SQL Amazon EMR unterstützt die Zugriffskontrolle auf Tabellen-, Zeilen-, Spalten- und Zellenebene mit Apache Hudi. Mit dieser Funktion können Sie Snapshot-Abfragen für copy-on-write Tabellen ausführen, um den neuesten Snapshot der Tabelle zu einem bestimmten Commit- oder Komprimierungszeitpunkt abzufragen.

Derzeit muss ein Lake Formation-fähiger EMR Amazon-Cluster die Commit-Zeitspalte von Hudi abrufen, um inkrementelle Abfragen und Zeitreiseabfragen durchzuführen. Die `timestamp as of` Syntax und die Funktion von Spark werden nicht unterstützt. `Spark.read()` Die richtige Syntax ist`select * from table where _hoodie_commit_time <= point_in_time`. Weitere Informationen finden Sie unter Point-in-Time-Time-Travel-Abfragen in der Hudi-Tabelle.

Die folgende Unterstützungsmatrix listet einige Kernfeatures von Apache Hudi mit Lake Formation auf:

	Kopieren Sie beim Schreiben	Beim Lesen zusammenführen (MoR)
Snapshot-Abfragen — Spark SQL	✓	✓
Leseoptimierte Abfragen — Spark SQL	✓	✓
Inkrementelle Abfragen	✓	✓
Zeitreiseabfragen	✓	✓

	Kopieren Sie beim Schreiben	Beim Lesen zusammenführen (MoR)
Metadaten-Tabellen	✓	✓
DML**INSERT**Befehle	✓	✓
DDLBefehle		
Spark-Datenquellenabfragen		
Spark-Datenquellenschreibvorgänge		

Abfragen von Hudi-Tabellen

In diesem Abschnitt wird gezeigt, wie Sie die oben beschriebenen unterstützten Abfragen auf einem Lake-Formation-fähigen Cluster ausführen können. Bei der Tabelle sollte es sich um eine registrierte Katalogtabelle handeln.

1. Verwenden Sie die folgenden Befehle, um die Spark-Shell zu starten.

```
spark-sql
--jars /usr/lib/hudi/hudi-spark-bundle.jar \
--conf spark.serializer=org.apache.spark.serializer.KryoSerializer \
--conf
spark.sql.catalog.spark_catalog=org.apache.spark.sql.hudi.catalog.HoodieCatalog \
--conf
spark.sql.extensions=org.apache.spark.sql.hudi.HoodieSparkSessionExtension,com.amazonaws.em
 \
--conf spark.sql.catalog.spark_catalog.lf.managed=true
```

   Wenn Sie möchten, dass Lake Formation einen Datensatzserver zur Verwaltung Ihres Spark-Katalogs verwendet, setzen Sie `spark.sql.catalog.<managed_catalog_name>.lf.managed` ihn auf true.

2. Verwenden Sie die folgenden Befehle, um den neuesten Snapshot der copy-on-write Tabellen abzufragen.

```
SELECT * FROM my_hudi_cow_table
```

```
spark.read.table("my_hudi_cow_table")
```

3. Um die neuesten komprimierten Daten von MOR-Tabellen abzufragen, können Sie die leseoptimierte Tabelle mit dem Suffix `_ro` abfragen:

```
SELECT * FROM my_hudi_mor_table_ro
```

```
spark.read.table("my_hudi_mor_table_ro")
```

> **Note**
>
> Die Leistung von Lesevorgängen auf Lake Formation-Clustern kann aufgrund von Optimierungen, die nicht unterstützt werden, langsamer sein. Zu diesen Features gehören das Auflisten von Dateien auf der Grundlage von Hudi-Metadaten und das Überspringen von Daten. Wir empfehlen Ihnen, die Leistung Ihrer Anwendung zu testen, um sicherzustellen, dass sie Ihrem SLA entspricht.

Apache Iceberg und Lake Formation

Die EMR Amazon-Versionen 6.15.0 und höher bieten Unterstützung für eine differenzierte Zugriffskontrolle, die auf Apache Iceberg basiert, wenn Sie Daten AWS Lake Formation mit Spark lesen und schreiben. SQL Amazon EMR unterstützt die Zugriffskontrolle auf Tabellen-, Zeilen-, Spalten- und Zellenebene mit Apache Iceberg. Mit dieser Funktion können Sie Snapshot-Abfragen für copy-on-write Tabellen ausführen, um den neuesten Snapshot der Tabelle zu einem bestimmten Commit- oder Komprimierungszeitpunkt abzufragen.

Wenn Sie das Iceberg-Format verwenden möchten, legen Sie die folgenden Konfigurationen fest. Ersetzen Sie *DB_LOCATION* durch den Amazon-S3-Pfad, in dem sich Ihre Iceberg-Tabellen befinden, und ersetzen Sie die Platzhalter für Region und Konto-ID durch Ihre eigenen Werte.

```
spark-sql \
--conf
 spark.sql.extensions=org.apache.iceberg.spark.extensions.IcebergSparkSessionExtensions,com.ama

--conf spark.sql.catalog.iceberg_catalog=org.apache.iceberg.spark.SparkCatalog
--conf spark.sql.catalog.iceberg_catalog.warehouse=s3://DB_LOCATION
```

```
--conf spark.sql.catalog.iceberg_catalog.catalog-
impl=org.apache.iceberg.aws.glue.GlueCatalog
--conf spark.sql.catalog.iceberg_catalog.io-impl=org.apache.iceberg.aws.s3.S3FileIO
--conf spark.sql.catalog.iceberg_catalog.glue.account-id=ACCOUNT_ID
--conf spark.sql.catalog.iceberg_catalog.glue.id=ACCOUNT_ID
--conf spark.sql.catalog.iceberg_catalog.client.assume-role.region=AWS_REGION
--conf spark.sql.secureCatalog=iceberg_catalog
```

Wenn Sie möchten, dass Lake Formation einen Datensatzserver zur Verwaltung Ihres Spark-Katalogs verwendet, setzen Sie `spark.sql.catalog.<managed_catalog_name>.lf.managed` ihn auf true.

Sie sollten auch darauf achtenNOT, die folgenden Einstellungen für die Übernahme der Rolle zu übergeben:

```
--conf spark.sql.catalog.my_catalog.client.assume-role.region
--conf spark.sql.catalog.my_catalog.client.assume-role.arn
--conf spark.sql.catalog.my_catalog.client.assume-
role.tags.LakeFormationAuthorizedCaller
```

Die folgende Unterstützungsmatrix listet einige Kernfeatures von Apache Iceberg mit Lake Formation auf:

	Kopieren Sie beim Schreiben	Beim Lesen zusammenf ühren (MoR)
Snapshot-Abfragen — Spark SQL	✓	✓
Leseoptimierte Abfragen — Spark SQL	✓	✓
Inkrementelle Abfragen	✓	✓
Zeitreiseabfragen	✓	✓
Metadaten-Tabellen	✓	✓
DML**`INSERT`**Befehle	✓	✓
DDLBefehle		

	Kopieren Sie beim Schreiben	Beim Lesen zusammenführen (MoR)
Spark-Datenquellenabfragen		
Spark-Datenquellenschreibvorgänge		

Delta Lake und Lake Formation

Die EMR Amazon-Versionen 6.15.0 und höher bieten Unterstützung für eine differenzierte Zugriffskontrolle, die auf Delta Lake basiert, wenn Sie Daten AWS Lake Formation mit Spark lesen und schreiben. SQL Amazon EMR unterstützt mit Delta Lake die Zugriffskontrolle auf Tabellen-, Zeilen-, Spalten- und Zellenebene. Mit dieser Funktion können Sie Snapshot-Abfragen für copy-on-write Tabellen ausführen, um den neuesten Snapshot der Tabelle zu einem bestimmten Commit- oder Komprimierungszeitpunkt abzufragen.

Führen Sie den folgenden Befehl aus, um Delta Lake mit Lake Formation zu verwenden.

```
spark-sql \
--conf
 spark.sql.extensions=io.delta.sql.DeltaSparkSessionExtension,com.amazonaws.emr.recordserver.co
 \
--conf spark.sql.catalog.spark_catalog=org.apache.spark.sql.delta.catalog.DeltaCatalog
 \
--conf spark.sql.catalog.spark_catalog.lf.managed=true
```

Wenn Sie möchten, dass Lake Formation einen Datensatzserver zur Verwaltung Ihres Spark-Katalogs verwendet, setzen Sie `spark.sql.catalog.<managed_catalog_name>.lf.managed` ihn auf true.

Die folgende Unterstützungsmatrix listet einige Kernfeatures von Delta Lake mit Lake Formation auf:

	Kopieren Sie beim Schreiben	Beim Lesen zusammenführen (MoR)
Snapshot-Abfragen — Spark SQL	✓	✓
Leseoptimierte Abfragen — Spark SQL	✓	✓

	Kopieren Sie beim Schreiben	Beim Lesen zusammenführen (MoR)
Inkrementelle Abfragen	Nicht unterstützt	Nicht unterstützt
Zeitreiseabfragen	Nicht unterstützt	Nicht unterstützt
Metadaten-Tabellen	✓	✓
DML**INSERT**Befehle	✓	✓
DDLBefehle		
Spark-Datenquellenabfragen		
Spark-Datenquellenschreibvorgänge		

Erstellen einer Delta Lake-Tabelle im AWS Glue Data Catalog

Amazon EMR mit Lake Formation unterstützt keine DDL Befehle und die Erstellung von Delta-Tabellen. Gehen Sie wie folgt vor, um Tabellen im AWS Glue-Datenkatalog zu erstellen.

1. Verwenden Sie das folgende Beispiel, um eine Delta-Tabelle zu erstellen. Stellen Sie sicher, dass Ihr S3-Standort existiert.

```
spark-sql \
--conf "spark.sql.extensions=io.delta.sql.DeltaSparkSessionExtension" \
--conf
 "spark.sql.catalog.spark_catalog=org.apache.spark.sql.delta.catalog.DeltaCatalog"

> CREATE DATABASE if not exists <DATABASE_NAME> LOCATION 's3://<S3_LOCATION>/
transactionaldata/native-delta/<DATABASE_NAME>/';
> CREATE TABLE <TABLE_NAME> (x INT, y STRING, z STRING) USING delta;
> INSERT INTO <TABLE_NAME> VALUES (1, 'a1', 'b1');
```

2. Um die Details Ihrer Tabelle zu sehen, gehen Sie zu https://console.aws.amazon.com/glue/.
3. Erweitern Sie in der linken Navigationsleiste den Datenkatalog, wählen Sie Tabellen und dann die Tabelle aus, die Sie erstellt haben. Unter Schema sollten Sie sehen, dass die Delta-Tabelle, die Sie mit Spark erstellt haben, alle Spalten in einem Datentyp von `array<string>` in AWS Glue speichert.

4. Um Filter auf Spalten- und Zellenebene in Lake Formation zu definieren, entfernen Sie die `col` Spalte aus Ihrem Schema und fügen Sie dann die Spalten hinzu, die sich in Ihrem Tabellenschema befinden. Fügen Sie in diesem Beispiel die Spalten `xy`, und hinzu. `z`

Überlegungen zu Amazon EMR mit Lake Formation

Beachten Sie Folgendes, wenn Sie Amazon EMR mit verwenden AWS Lake Formation.

- Zugriffskontrolle auf Tabellenebene ist auf Clustern mit EMR Amazon-Versionen 6.13 und höher verfügbar.
- Eine differenzierte Zugriffskontrolle auf Zeilen-, Spalten- und Zellenebene ist für Cluster mit EMR Amazon-Versionen 6.15 und höher verfügbar.
- Benutzer mit Zugriff auf eine Tabelle können auf alle Eigenschaften dieser Tabelle zugreifen. Wenn Sie eine auf Lake Formation basierende Zugriffskontrolle für eine Tabelle haben, überprüfen Sie die Tabelle, um sicherzustellen, dass die Eigenschaften keine vertraulichen Daten oder Informationen enthalten.
- EMRAmazon-Cluster mit Lake Formation unterstützen den Fallback von Spark auf die HDFS Erfassung von Tabellenstatistiken durch Spark nicht. Dies trägt normalerweise zur Optimierung der Abfrageleistung bei.
- Zu den Vorgängen, die Zugriffskontrollen auf der Grundlage von Lake Formation mit nicht verwalteten Apache-Spark-Tabellen unterstützen, gehören `INSERT INTO` und `INSERT OVERWRITE`.
- Zu den Vorgängen, die auf Lake Formation mit Apache Spark und Apache Hive basierende Zugriffskontrollen unterstützen `SELECT`, `DESCRIBE`, `SHOW DATABASE`, `SHOW TABLE`, `SHOW COLUMN` und `SHOW PARTITION`.
- Amazon unterstützt EMR keine Zugriffskontrolle für die folgenden auf Lake Formation basierenden Operationen:
  - Schreibt in geregelte Tabellen
  - Amazon unterstützt EMR nicht`CREATE TABLE`. Amazon EMR 6.10.0 und höher werden unterstützt`ALTER TABLE`.
  - DMLandere Anweisungen als `INSERT` Befehle.
- Es gibt Leistungsunterschiede zwischen derselben Abfrage mit und ohne Lake-Formation-basierte Zugriffskontrolle.
- Sie können Amazon nur EMR mit Lake Formation für Spark-Jobs verwenden.

Integrieren Sie Amazon EMR mit Apache Ranger

Ab Amazon EMR 5.32.0 können Sie einen Cluster starten, der nativ in Apache Ranger integriert ist. Apache Ranger ist ein Open-Source-Framework zur Aktivierung, Überwachung und Verwaltung einer umfassenden Datensicherheit auf der gesamten Hadoop-Plattform. Weitere Informationen finden Sie unter Apache Ranger. Dank der nativen Integration können Sie Ihren eigenen Apache Ranger verwenden, um eine detaillierte Datenzugriffskontrolle auf Amazon durchzusetzen. EMR

Dieser Abschnitt bietet einen konzeptionellen Überblick über die EMR Amazon-Integration mit Apache Ranger. Es enthält auch die Voraussetzungen und Schritte, die für den Start eines in Apache Ranger integrierten EMR Amazon-Clusters erforderlich sind.

Die native Integration von Amazon EMR mit Apache Ranger bietet die folgenden Hauptvorteile:

- Präzise Zugriffskontrolle für Hive Metastore-Datenbanken und -Tabellen, mit der Sie Datenfilterungsrichtlinien auf Datenbank-, Tabellen- und Spaltenebene für Apache Spark- und Apache Hive-Anwendungen definieren können. Filterung und Datenmaskierung auf Zeilenebene werden von Hive-Anwendungen unterstützt.
- Die Möglichkeit, Ihre bestehenden Hive-Richtlinien direkt mit Amazon EMR for Hive-Anwendungen zu verwenden.
- Zugriffskontrolle auf Amazon S3 S3-Daten auf Präfix- und Objektebene, sodass Sie Datenfilterrichtlinien für den Zugriff auf S3-Daten mithilfe des EMR Dateisystems definieren können.
- Die Möglichkeit, CloudWatch Logs für zentralisierte Prüfungen zu verwenden.
- Amazon EMR installiert und verwaltet die Apache Ranger-Plugins in Ihrem Namen.

Apache Ranger

Apache Ranger ist ein Framework zur Aktivierung, Überwachung und Verwaltung einer umfassenden Datensicherheit auf der gesamten Hadoop-Plattform.

Apache Ranger bietet folgende Features:

- Zentralisierte Sicherheitsadministration zur Verwaltung aller sicherheitsrelevanten Aufgaben in einer zentralen Benutzeroberfläche oder mithilfe von RESTAPIs.

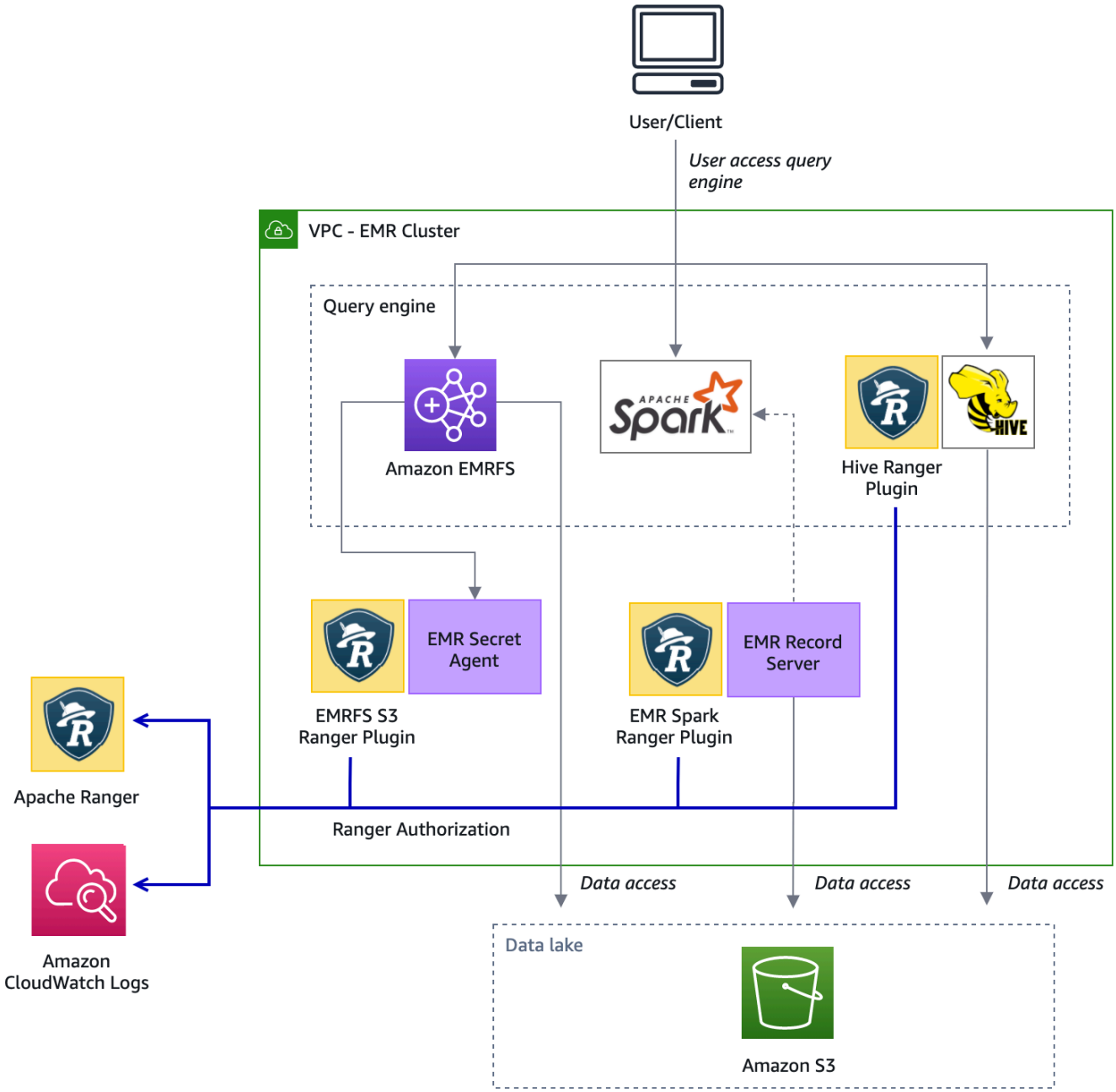
- Detaillierte Autorisierung zur Durchführung einer bestimmten Aktion oder Operation mit einer Hadoop-Komponente oder einem Hadoop-Tool, die über ein zentrales Administrationstool verwaltet wird.
- Eine standardisierte Autorisierungsmethode für alle Hadoop-Komponenten.
- Verbesserte Unterstützung für verschiedene Autorisierungsmethoden.
- Zentralisierte Prüfung des Benutzerzugriffs und der administrativen Aktionen (sicherheitsbezogen) innerhalb aller Komponenten von Hadoop.

Apache Ranger verwendet zwei Schlüsselkomponenten für die Autorisierung:

- Apache-Ranger-Richtlinien-Admin-Server – Mit diesem Server können Sie die Autorisierungsrichtlinien für Hadoop-Anwendungen definieren. Bei der Integration mit Amazon EMR können Sie Richtlinien für Apache Spark und Hive für den Zugriff auf Hive Metastore und für den Zugriff auf das Amazon EMRS3-Datendateisystem () definieren und durchsetzen. EMRFS Sie können einen neuen Apache Ranger Policy Admin-Server einrichten oder einen vorhandenen Apache Ranger Policy Admin-Server für die Integration mit Amazon EMR verwenden.
- Apache-Ranger-Plugin – Dieses Plugin validiert den Zugriff eines Benutzers anhand der Autorisierungsrichtlinien, die im Apache-Ranger-Richtlinien-Admin-Server definiert sind. Amazon EMR installiert und konfiguriert das Apache Ranger-Plugin automatisch für jede Hadoop-Anwendung, die in der Apache Ranger-Konfiguration ausgewählt wurde.

Themen

Architektur der EMR Amazon-Integration mit Apache Ranger



EMRAmazon-Komponenten

Amazon EMR ermöglicht eine differenzierte Zugriffskontrolle mit Apache Ranger über die folgenden Komponenten. Im Architekturdiagramm finden Sie eine visuelle Darstellung dieser EMR Amazon-Komponenten mit den Apache Ranger-Plugins.

Geheimagent — Der Geheimagent speichert Geheimnisse sicher und verteilt Geheimnisse an andere EMR Amazon-Komponenten oder -Anwendungen. Bei diesen geheimen Daten („Secrets“) kann es sich beispielsweise um temporäre Anmeldeinformationen von Benutzern, um Verschlüsselungsschlüssel oder um Kerberos-Tickets handeln. Der Secret Agent läuft auf jedem Knoten im Cluster und fängt Aufrufe an den Instance Metadata Service ab. Bei Anfragen an die Rollenanmeldedaten des Instance-Profils vergibt der Secret Agent je nach dem anfragenden Benutzer und den angeforderten Ressourcen Anmeldeinformationen, nachdem er die Anfrage mit dem EMRFS S3 Ranger-Plugin autorisiert hat. Der Secret-Agent wird unter Benutzer `emrsecretagent` ausgeführt und schreibt Protokolle in das Verzeichnis /emr/secretagent/log directory. Der Prozess benötigt eine bestimmte Zusammenstellung von `iptables`-Regeln, um zu funktionieren. Es ist wichtig sicherzustellen, dass `iptables` nicht deaktiviert ist. Wenn Sie die `iptables` Konfiguration anpassen, müssen die NAT Tabellenregeln beibehalten und unverändert bleiben.

EMRDatensatzserver — Der Datensatzserver empfängt Anfragen zum Zugriff auf Daten von Spark. Anschließend autorisiert es Anfragen, indem es die angeforderten Ressourcen an das Spark Ranger-Plugin für Amazon weiterleitet. EMR Der Datensatzserver liest Daten von Amazon S3 und gibt gefilterte Daten zurück, auf die der Benutzer gemäß der Ranger-Richtlinie zugreifen darf. Der Record-Server läuft auf jedem Knoten im Cluster als Benutzer emr_record_server und schreibt Protokolle in das Verzeichnis /var/log/. emr-record-server

Anwendungsunterstützung und Einschränkungen

Unterstützte Anwendungen

Die Integration zwischen Amazon EMR und Apache Ranger, bei der Ranger-Plugins EMR installiert werden, unterstützt derzeit die folgenden Anwendungen:

- Apache Spark (verfügbar mit EMR 5.32+ und 6.3+) EMR
- Apache Hive (verfügbar mit EMR 5.32+ und 6.3+) EMR
- S3-Zugriff über EMRFS (verfügbar mit EMR 5.32+ und 6.3+) EMR

Die folgenden Anwendungen können auf einem EMR Cluster installiert werden und müssen möglicherweise entsprechend Ihren Sicherheitsanforderungen konfiguriert werden:

- Apache Hadoop (verfügbar mit EMR 5.32+ und EMR 6.3+, einschließlich und) YARN HDFS
- Apache Livy (verfügbar mit 5.32+ und 6.3+) EMR EMR

- Apache Zeppelin (verfügbar mit 5.32+ und 6.3+) EMR EMR
- Apache Hue (verfügbar mit 5.32+ und 6.3+) EMR EMR
- Ganglia (verfügbar mit EMR 5.32+ und 6.3+) EMR
- HCatalog(Verfügbar mit EMR 5.32+ und 6.3+) EMR
- Mahout (Verfügbar mit EMR 5.32+ und 6.3+) EMR
- MXNet(Verfügbar mit EMR 5.32+ und 6.3+) EMR
- TensorFlow (Verfügbar mit EMR 5.32+ und 6.3+) EMR
- Tez (Verfügbar mit EMR 5.32+ und 6.3+) EMR
- Trino (Verfügbar mit 6.7+) EMR
- ZooKeeper (Verfügbar mit EMR 5.32+ und 6.3+) EMR

> **Important**
>
> Die oben aufgeführten Anwendungen sind die einzigen Anwendungen, die derzeit unterstützt werden. Um die Clustersicherheit zu gewährleisten, dürfen Sie einen EMR Cluster nur mit den Anwendungen in der obigen Liste erstellen, wenn Apache Ranger aktiviert ist.
> Andere Anwendungen werden derzeit nicht unterstützt. Um die Sicherheit Ihres Clusters zu gewährleisten, führt der Versuch, andere Anwendungen zu installieren, zur Ablehnung Ihres Clusters.

Unterstützte Funktionen

Die folgenden EMR Amazon-Funktionen können mit Amazon EMR und Apache Ranger verwendet werden:

- Verschlüsselung bei Speicherung und Übertragung
- Kerberos-Authentifizierung (erforderlich)
- Instance-Gruppen, Instance-Flotten und Spot Instances
- Neukonfiguration von Anwendungen auf einem laufenden Cluster
- EMRFSserverseitige Verschlüsselung () SSE

> **Note**
>
> Die EMR Amazon-Verschlüsselungseinstellungen geltenSSE. Weitere Informationen finden Sie unter Verschlüsselungsoptionen.

Einschränkungen der Anwendung

Bei der Integration von Amazon EMR und Apache Ranger sind mehrere Einschränkungen zu beachten:

- Sie können die Konsole derzeit nicht verwenden, um eine Sicherheitskonfiguration zu erstellen, die die AWS Ranger-Integrationsoption in der spezifiziert. AWS GovCloud (US) Region Die Sicherheitskonfiguration kann mit dem CLI vorgenommen werden.
- Kerberos muss in Ihrem Cluster installiert sein.
- Bei Anwendungen UIs (Benutzeroberflächen) wie der YARN Resource HDFS NameNode Manager-Benutzeroberfläche, der Benutzeroberfläche und der Livy-Benutzeroberfläche ist standardmäßig keine Authentifizierung aktiviert.
- Die HDFS Standardberechtigungen `umask` sind so konfiguriert, dass erstellte Objekte `world wide readable` standardmäßig auf eingestellt sind.
- Amazon unterstützt den Hochverfügbarkeitsmodus (mehrere Primärversionen) mit Apache Ranger EMR nicht.
- Weitere Einschränkungen finden Sie unter Einschränkungen für jede Anwendung.

> **Note**
>
> Die EMR Amazon-Verschlüsselungseinstellungen geltenSSE. Weitere Informationen finden Sie unter Verschlüsselungsoptionen.

Einschränkungen des Plugins

Jedes Plugin hat spezifische Einschränkungen. Die Einschränkungen des Apache-Hive-Plugins finden Sie unter Einschränkungen des Apache-Hive-Plugins. Die Einschränkungen des Apache-Spark-Plugins finden Sie unter Einschränkungen des Apache-Spark-Plugins. Die Einschränkungen des EMRFS S3-Plug-ins finden Sie unter Einschränkungen des EMRFSS3-Plug-ins.

Amazon EMR für Apache Ranger einrichten

Bevor Sie Apache Ranger installieren, überprüfen Sie die Informationen in diesem Abschnitt, um sicherzustellen, dass Amazon ordnungsgemäß konfiguriert EMR ist.

Themen

Richten Sie den Ranger-Admin-Server ein

Für die EMR Amazon-Integration müssen die Apache Ranger-Anwendungs-Plugins überTLS/SSLmit dem Admin-Server kommunizieren.

Voraussetzung: Aktivierung des Ranger Admin-Servers SSL

Apache Ranger auf Amazon EMR erfordert eine bidirektionale SSL Kommunikation zwischen Plugins und dem Ranger Admin-Server. Um sicherzustellen, dass Plugins mit dem Apache Ranger-Server kommunizierenSSL, aktivieren Sie das folgende Attribut in ranger-admin-site .xml auf dem Ranger Admin-Server.

```
<property>
    <name>ranger.service.https.attrib.ssl.enabled</name>
    <value>true</value>
</property>
```

Darüber hinaus sind die folgenden Konfigurationen erforderlich.

```
<property>
    <name>ranger.https.attrib.keystore.file</name>
    <value>_<PATH_TO_KEYSTORE>_</value>
```

```
</property>

<property>
    <name>ranger.service.https.attrib.keystore.file</name>
    <value>_<PATH_TO_KEYSTORE>_</value>
</property>

<property>
    <name>ranger.service.https.attrib.keystore.pass</name>
    <value>_<KEYSTORE_PASSWORD>_</value>
</property>

<property>
    <name>ranger.service.https.attrib.keystore.keyalias</name>
    <value><PRIVATE_CERTIFICATE_KEY_ALIAS></value>
</property>

<property>
    <name>ranger.service.https.attrib.clientAuth</name>
    <value>want</value>
</property>

<property>
    <name>ranger.service.https.port</name>
    <value>6182</value>
</property>
```

TLSZertifikate

Die Apache Ranger-Integration mit Amazon EMR erfordert, dass der Datenverkehr von EMR Amazon-Knoten zum Ranger Admin-Server verschlüsselt wird und dass Ranger-Plug-ins sich beim Apache Ranger-Server über TLS eine wechselseitige bidirektionale Authentifizierung authentifizieren. TLS Der EMR Amazon-Service benötigt das öffentliche Zertifikat Ihres Ranger Admin-Servers (im vorherigen Beispiel angegeben) und das private Zertifikat.

Zertifikate für das Apache-Ranger-Plugin

Öffentliche TLS Zertifikate des Apache Ranger-Plug-ins müssen für den Apache Ranger Admin-Server zugänglich sein, um zu überprüfen, ob die Plugins eine Verbindung herstellen. Es gibt drei verschiedene Methoden, dies zu tun.

Methode 1: Konfigurieren Sie einen Truststore auf dem Apache-Ranger-Admin-Server

Füllen Sie die folgenden Konfigurationen in ranger-admin-site .xml aus, um einen Truststore zu konfigurieren.

```
<property>
    <name>ranger.truststore.file</name>
    <value><LOCATION TO TRUSTSTORE></value>
</property>

<property>
    <name>ranger.truststore.password</name>
    <value><PASSWORD FOR TRUSTSTORE></value>
</property>
```

Methode 2: Laden Sie das Zertifikat in den Truststore von Java cacerts

Wenn Ihr Ranger Admin-Server in seinen JVM Optionen keinen Truststore angibt, können Sie die öffentlichen Zertifikate des Plug-ins im standardmäßigen Cacerts-Speicher ablegen.

Methode 3: Erstellen Sie einen Truststore und geben Sie ihn als Teil der Optionen an JVM

Ändern Sie die `{RANGER_HOME_DIRECTORY}/ews/ranger-admin-services.sh` innerhalb von `JAVA_OPTS`, dass sie "`-Djavax.net.ssl.trustStore=<TRUSTSTORE_LOCATION>`" und "`-Djavax.net.ssl.trustStorePassword=<TRUSTSTORE_PASSWORD>`" einschließt. Fügen Sie beispielsweise die folgende Zeile nach dem vorhandenen JAVA _ OPTS hinzu.

```
JAVA_OPTS=" ${JAVA_OPTS} -Djavax.net.ssl.trustStore=${RANGER_HOME}/truststore/
truststore.jck -Djavax.net.ssl.trustStorePassword=changeit"
```

> **Note**
>
> Diese Spezifikation kann das Truststore-Passwort offenlegen, wenn sich ein Benutzer beim Apache- Ranger Admin-Server anmelden und laufende Prozesse sehen kann, z. B. wenn er den `ps`-Befehl verwendet.

Verwenden selbstsignierter Zertifikate

Selbstsignierte Zertifikate werden als Zertifikate nicht empfohlen. Selbstsignierte Zertifikate können nicht gesperrt werden, und selbstsignierte Zertifikate entsprechen möglicherweise nicht den internen Sicherheitsanforderungen.

Installation der Servicedefinition

Eine Servicedefinition wird vom Ranger-Admin-Server verwendet, um die Attribute von Richtlinien für eine Anwendung zu beschreiben. Die Richtlinien werden dann in einem Richtlinien-Repository gespeichert, sodass die Clients sie herunterladen können.

Um Dienstdefinitionen konfigurieren zu können, müssen REST Aufrufe an den Ranger Admin-Server getätigt werden. Informationen zu den APIs erforderlichen Anforderungen finden Sie im folgenden Abschnitt unter Apache Ranger P ublicAPIsv 2.

Installation der Servicedefinition von Apache Spark

Informationen zur Installation der Servicedefinition von Apache Spark finden Sie unter Apache Spark Plugin.

EMRFSService Definition wird installiert

Informationen zur Installation der S3-Servicedefinition für Amazon EMR finden Sie unterEMRFSS3-Plugin.

Verwenden der Hive-Servicedefinition

Apache Hive kann die bestehende Ranger-Servicedefinition verwenden, die im Lieferumfang von Apache Ranger 2.0 und höher enthalten ist. Weitere Informationen finden Sie unter Apache- Hive-Plugin.

Regeln für den Netzwerkverkehr

Wenn Apache Ranger in Ihren EMR Cluster integriert ist, muss der Cluster mit zusätzlichen Servern kommunizieren und AWS.

Alle EMR Amazon-Knoten, einschließlich Kern- und Task-Knoten, müssen in der Lage sein, mit den Apache Ranger Admin-Servern zu kommunizieren, um Richtlinien herunterzuladen. Wenn Ihr Apache Ranger Admin auf Amazon läuftEC2, müssen Sie die Sicherheitsgruppe aktualisieren, um Datenverkehr vom EMR Cluster entgegennehmen zu können.

Zusätzlich zur Kommunikation mit dem Ranger Admin-Server müssen alle Knoten in der Lage sein, mit den folgenden AWS Diensten zu kommunizieren:

- Amazon S3
- AWS KMS (wenn Sie EMRFS SSE - KMS verwenden)
- Amazon CloudWatch

- AWS STS

Wenn Sie planen, Ihren EMR Cluster in einem privaten Subnetz zu betreiben, konfigurieren Sie die VPC so, dass Sie mit diesen Diensten entweder AWS PrivateLink über VPCEndpunkte im VPCAmazon-Benutzerhandbuch oder mithilfe der Network Address Translation (NAT) -Instance im VPCAmazon-Benutzerhandbuch kommunizieren können.

IAMRollen für die native Integration mit Apache Ranger

Die Integration zwischen Amazon EMR und Apache Ranger basiert auf drei Schlüsselrollen, die Sie erstellen sollten, bevor Sie Ihren Cluster starten:

- Ein benutzerdefiniertes EC2 Amazon-Instanzprofil für Amazon EMR
- Eine IAM Rolle für Apache Ranger Engines
- Eine IAM Rolle für andere Dienste AWS

Dieser Abschnitt gibt einen Überblick über diese Rollen und die Richtlinien, die Sie für jede IAM Rolle angeben müssen. Informationen zum Erstellen dieser Rollen finden Sie unter Richten Sie den Ranger-Admin-Server ein.

EC2-Instance-Profil

Amazon EMR verwendet eine IAM Service-Rolle, um in Ihrem Namen Aktionen zur Bereitstellung und Verwaltung von Clustern durchzuführen. Die Servicerolle für EC2 Cluster-Instances, auch EC2 Instance-Profil für Amazon genanntEMR, ist eine spezielle Art von Servicerolle, die jeder EC2 Instance in einem Cluster beim Start zugewiesen wird.

Um Berechtigungen für die EMR Cluster-Interaktion mit Amazon S3 S3-Daten und mit dem durch Apache Ranger und andere AWS Dienste geschützten Hive-Metastore zu definieren, definieren Sie ein benutzerdefiniertes EC2 Instance-Profil, das anstelle des `EMR_EC2_DefaultRole` beim Starten Ihres Clusters verwendet werden soll.

Weitere Informationen erhalten Sie unter Servicerolle für EC2 Cluster-Instances (EC2Instance-Profil) und Passen Sie IAM Rollen an.

Sie müssen die folgenden Anweisungen zum EC2 Standard-Instance-Profil hinzufügen, EMR damit Amazon Sitzungen taggen und auf die Dateien zugreifen kann, in AWS Secrets Manager denen TLS Zertifikate gespeichert sind.

```
    {
      "Sid": "AllowAssumeOfRolesAndTagging",
      "Effect": "Allow",
      "Action": ["sts:TagSession", "sts:AssumeRole"],
      "Resource": [
        "arn:aws:iam::<AWS_ACCOUNT_ID>:role/<RANGER_ENGINE-
PLUGIN_DATA_ACCESS_ROLE_NAME>",
        "arn:aws:iam::<AWS_ACCOUNT_ID>:role/<RANGER_USER_ACCESS_ROLE_NAME>"
      ]
    },
    {
        "Sid": "AllowSecretsRetrieval",
        "Effect": "Allow",
        "Action": "secretsmanager:GetSecretValue",
        "Resource": [

 "arn:aws:secretsmanager:<REGION>:<AWS_ACCOUNT_ID>:secret:<PLUGIN_TLS_SECRET_NAME>*",

 "arn:aws:secretsmanager:<REGION>:<AWS_ACCOUNT_ID>:secret:<ADMIN_RANGER_SERVER_TLS_SECRET_NAME>
        ]
    }
```

> **Note**
>
> Vergessen Sie bei den Secrets- Manager-Berechtigungen nicht den Platzhalter („*“) am Ende des geheimen Namens, da Ihre Anfragen sonst fehlschlagen. Der Platzhalter gilt für geheime Versionen.

> **Note**
>
> Beschränken Sie den Geltungsbereich der AWS Secrets Manager Richtlinie auf die Zertifikate, die für die Bereitstellung erforderlich sind.

IAMRolle für Apache Ranger

Diese Rolle stellt Anmeldeinformationen für vertrauenswürdige Ausführungs-Engines wie Apache Hive und Amazon EMR Record Server bereit, um auf Amazon S3 S3-Daten zuzugreifen. Verwenden

Sie nur diese Rolle, um auf Amazon S3 S3-Daten, einschließlich aller KMS Schlüssel, zuzugreifen, wenn Sie S3 SSE - verwendenKMS.

Diese Rolle muss mit der im folgenden Beispiel angegebenen Mindestrichtlinie erstellt werden.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "CloudwatchLogsPermissions",
      "Action": [
        "logs:CreateLogGroup",
        "logs:CreateLogStream",
        "logs:PutLogEvents"
      ],
      "Effect": "Allow",
      "Resource": [

 "arn:aws:logs:<REGION>:<AWS_ACCOUNT_ID>:<CLOUDWATCH_LOG_GROUP_NAME_IN_SECURITY_CONFIGURATION>:
      ]
    },
    {
      "Sid": "BucketPermissionsInS3Buckets",
      "Action": [
        "s3:CreateBucket",
        "s3:DeleteBucket",
        "s3:ListAllMyBuckets",
        "s3:ListBucket"
      ],
      "Effect": "Allow",
      "Resource": [
*"arn:aws:s3:::bucket1",
        "arn:aws:s3:::bucket2"*
        ]
    },
    {
      "Sid": "ObjectPermissionsInS3Objects",
      "Action": [
        "s3:GetObject",
        "s3:DeleteObject",
        "s3:PutObject"
      ],
      "Effect": "Allow",
      "Resource": [
```

```
 *"arn:aws:s3:::bucket1/*",
          "arn:aws:s3:::bucket2/*"
*        ]
    }
  ]
}
```

> **Important**
>
> Das Sternchen „*“ am Ende der CloudWatch Protokollressource muss enthalten sein, um die Erlaubnis zu erteilen, in die Protokollstreams zu schreiben.

> **Note**
>
> Wenn Sie EMRFS Consistency View oder SSE S3-Verschlüsselung verwenden, fügen Sie den DynamoDB-Tabellen und KMS -Schlüsseln Berechtigungen hinzu, damit die Ausführungsmodule mit diesen Engines interagieren können.

Die IAM Rolle für Apache Ranger wird von der EC2 Instanzprofilrolle übernommen. Verwenden Sie das folgende Beispiel, um eine Vertrauensrichtlinie zu erstellen, die es ermöglicht, dass die IAM Rolle für Apache Ranger von der EC2 Instanzprofilrolle übernommen wird.

```
   {
     "Sid": "",
     "Effect": "Allow",
     "Principal": {
       "AWS": "arn:aws:iam::<AWS_ACCOUNT_ID>:role/<EC2 INSTANCE PROFILE ROLE NAME eg.
EMR_EC2_DefaultRole>"
     },
     "Action": ["sts:AssumeRole", "sts:TagSession"]
   }
```

IAMRolle für andere Dienste AWS

Mit dieser Rolle erhalten Benutzer, denen die Ausführungsmodule nicht vertrauen, bei Bedarf Anmeldeinformationen für die Interaktion mit AWS Diensten. Verwenden Sie diese IAM Rolle nicht, um Zugriff auf Amazon S3 S3-Daten zu gewähren, es sei denn, es handelt sich um Daten, auf die alle Benutzer zugreifen können sollten.

Diese Rolle wird von der EC2 Instance-Profilrolle übernommen. Verwenden Sie das folgende Beispiel, um eine Vertrauensrichtlinie zu erstellen, die es ermöglicht, dass die IAM Rolle für Apache Ranger von der EC2 Instanzprofilrolle übernommen wird.

```
   {
     "Sid": "",
     "Effect": "Allow",
     "Principal": {
       "AWS": "arn:aws:iam::<AWS_ACCOUNT_ID>:role/<EC2 INSTANCE PROFILE ROLE NAME eg.
EMR_EC2_DefaultRole>"
     },
     "Action": ["sts:AssumeRole", "sts:TagSession"]
   }
```

Validieren Ihrer Berechtigungen

Anweisungen zum Überprüfen von Berechtigungen finden Sie unter Fehlerbehebung für Apache Ranger.

Erstellen Sie die EMR Sicherheitskonfiguration

Eine EMR Amazon-Sicherheitskonfiguration für Apache Ranger erstellen

Bevor Sie einen in Apache Ranger integrierten EMR Amazon-Cluster starten, erstellen Sie eine Sicherheitskonfiguration.

Console

So erstellen Sie eine Sicherheitskonfiguration, die die AWS -Ranger-Integrationsoption angibt

1. Wählen Sie in der EMR Amazon-Konsole Sicherheitskonfigurationen und anschließend Erstellen aus.
2. Geben Sie in Name (Name) einen Namen für die Sicherheitskonfiguration ein. Verwenden Sie diesen Namen zum Angeben der Sicherheitskonfiguration, wenn Sie einen Cluster erstellen.
3. Wählen Sie unter AWS -Ranger-Integration die Option Aktivieren einer von Apache Ranger verwalteten feinkörnigen Zugriffskontrolle.
4. Wählen Sie Ihre IAMRolle aus, auf die sich Apache Ranger bewerben soll. Weitere Informationen finden Sie unter IAMRollen für die native Integration mit Apache Ranger.
5. Wählen Sie Ihre IAMRolle aus, damit sich andere AWS Dienste bewerben können.

6. Konfigurieren Sie die Plugins so, dass sie eine Verbindung zum Ranger Admin-Server herstellen, indem Sie den Secret Manager ARN für den Admin-Server und die Adresse eingeben.
7. Wählen Sie die Anwendungen aus, um Ranger-Plugins zu konfigurieren. Füllen Sie den Secret Manager ausARN, der das private TLS Zertifikat für das Plugin enthält.

   Wenn Sie Apache Spark oder Apache Hive nicht konfigurieren und diese als Anwendung für Ihren Cluster ausgewählt wurden, schlägt die Anfrage fehl.
8. Richten Sie weitere Sicherheitskonfigurationsoptionen ein wie erforderlich. Wählen Sie Create (Erstellen) aus. Sie müssen die Kerberos-Authentifizierung mithilfe des dedizierten oder externen Clusters aktivieren. KDC

> **Note**
>
> Sie können die Konsole derzeit nicht verwenden, um eine Sicherheitskonfiguration zu erstellen, die die AWS Ranger-Integrationsoption in der spezifiziert. AWS GovCloud (US) Region Die Sicherheitskonfiguration kann mit dem CLI vorgenommen werden.

CLI

Wie Sie eine Sicherheitskonfiguration für die Apache Ranger-Integration erstellen

1. *<ACCOUNT ID>*Ersetzen Sie es durch Ihre AWS Konto-ID.
2. Ersetzen Sie *<REGION>* durch die Region, in der sich die Ressource befindet.
3. Geben Sie einen Wert für an`TicketLifetimeInHours`, um den Zeitraum zu bestimmen, für den ein von der ausgestelltes Kerberos-Ticket gültig KDC ist.
4. Geben Sie die Adresse des Ranger-Admin-Servers für `AdminServer`URL an.

```
{
    "AuthenticationConfiguration": {
        "KerberosConfiguration": {
            "Provider": "ClusterDedicatedKdc",
            "ClusterDedicatedKdcConfiguration": {
                "TicketLifetimeInHours": 24
            }
        }
```

```
   },
   "AuthorizationConfiguration":{
     "RangerConfiguration":{
        "AdminServerURL":"https://_<RANGER ADMIN SERVER IP>_:6182",
        "RoleForRangerPluginsARN":"arn:aws:iam::_<ACCOUNT ID>_:role/_<RANGER PLUGIN
DATA ACCESS ROLE NAME>_",
        "RoleForOtherAWSServicesARN":"arn:aws:iam::_<ACCOUNT ID>_:role/_<USER
ACCESS ROLE NAME>_",
        "AdminServerSecretARN":"arn:aws:secretsmanager:_<REGION>_:_<ACCOUNT
ID>_:secret:_<SECRET NAME THAT PROVIDES ADMIN SERVERS PUBLIC TLS CERTIFICATE
WITHOUT VERSION>_",
        "RangerPluginConfigurations":[
           {
              "App":"Spark",
              "ClientSecretARN":"arn:aws:secretsmanager:_<REGION>_:_<ACCOUNT
ID>_:secret:_<SECRET NAME THAT PROVIDES SPARK PLUGIN PRIVATE TLS CERTIFICATE
WITHOUT VERSION>_",
              "PolicyRepositoryName":"<SPARK SERVICE NAME eg. amazon-emr-spark>"
           },
           {
              "App":"Hive",
              "ClientSecretARN":"arn:aws:secretsmanager:_<REGION>_:_<ACCOUNT
ID>_:secret:_<SECRET NAME THAT PROVIDES Hive PLUGIN PRIVATE TLS CERTIFICATE WITHOUT
VERSION>_",
              "PolicyRepositoryName":"<HIVE SERVICE NAME eg. Hivedev>"
           },
           {
              "App":"EMRFS-S3",
              "ClientSecretARN":"arn:aws:secretsmanager:_<REGION>_:_<ACCOUNT
ID>_:secret:_<SECRET NAME THAT PROVIDES EMRFS S3 PLUGIN PRIVATE TLS CERTIFICATE
WITHOUT VERSION>_",
              "PolicyRepositoryName":"<EMRFS S3 SERVICE NAME eg amazon-emr-emrfs>"
           },
      {
              "App":"Trino",
              "ClientSecretARN":"arn:aws:secretsmanager:_<REGION>_:_<ACCOUNT
ID>_:secret:_<SECRET NAME THAT PROVIDES TRINO PLUGIN PRIVATE TLS CERTIFICATE
WITHOUT VERSION>_",
              "PolicyRepositoryName":"<TRINO SERVICE NAME eg amazon-emr-trino>"
           }
        ],
        "AuditConfiguration":{
           "Destinations":{
              "AmazonCloudWatchLogs":{
```

```
                    "CloudWatchLogGroup":"arn:aws:logs:<REGION>:_<ACCOUNT ID>_:log-
group:_<LOG GROUP NAME FOR AUDIT EVENTS>_"
                }
            }
        }
    }
}
```

Dies PolicyRespositoryNames sind die Dienstnamen, die in Ihrem Apache Ranger-Adminbereich angegeben sind.

Erstellen Sie mit dem folgenden Befehl eine EMR Amazon-Sicherheitskonfiguration. Ersetzen Sie Sicherheitskonfiguration durch einen beliebigen Namen. Wählen Sie diese Konfiguration beim Erstellen Ihres Clusters nach Namen aus.

```
aws emr create-security-configuration \
--security-configuration file://./security-configuration.json \
--name security-configuration
```

Konfigurieren von zusätzlichen Sicherheits-Features

Um Amazon sicher EMR mit Apache Ranger zu integrieren, konfigurieren Sie die folgenden EMR Sicherheitsfunktionen:

- Aktivieren Sie die Kerberos-Authentifizierung mithilfe des dedizierten oder externen Clusters. KDC Detaillierte Anweisungen finden Sie unter Verwenden Sie Kerberos für die Authentifizierung bei Amazon EMR.
- (Optional) Aktivieren Sie die Verschlüsselung von Daten während der Übertragung oder im Ruhezustand. Weitere Informationen finden Sie unter Verschlüsselungsoptionen.

Weitere Informationen finden Sie unter Sicherheit bei Amazon EMR.

TLSSpeichern Sie Zertifikate in AWS Secrets Manager

Die auf einem EMR Amazon-Cluster installierten Ranger-Plug-ins und der Ranger Admin-Server müssen miteinander kommunizieren, TLS um sicherzustellen, dass Policy-Daten und andere gesendete Informationen nicht gelesen werden können, wenn sie abgefangen werden. EMRschreibt außerdem vor, dass sich die Plugins beim Ranger Admin-Server authentifizieren, indem sie ein

eigenes Zertifikat bereitstellen und eine bidirektionale Authentifizierung durchführen. TLS TLS Für dieses Setup mussten vier Zertifikate erstellt werden: zwei Paare von privaten und öffentlichen Zertifikaten. TLS Anweisungen zur Installation des Zertifikats auf Ihrem Ranger Admin-Server finden Sie unter Richten Sie den Ranger-Admin-Server ein. Um das Setup abzuschließen, benötigen die auf dem EMR Cluster installierten Ranger-Plugins zwei Zertifikate: das öffentliche TLS Zertifikat Ihres Admin-Servers und das private Zertifikat, das das Plugin zur Authentifizierung gegenüber dem Ranger-Admin-Server verwendet. Um diese TLS Zertifikate bereitzustellen, müssen sie sich in der Sicherheitskonfiguration befinden AWS Secrets Manager und in einer EMR Sicherheitskonfiguration bereitgestellt werden.

> **Note**
>
> Es wird dringend empfohlen, aber nicht vorgeschrieben, für jede Ihrer Anwendungen ein Zertifikatspaar zu erstellen, um die Auswirkungen zu begrenzen, falls eines der Plugin-Zertifikate kompromittiert wird.

> **Note**
>
> Sie müssen Zertifikate vor ihrem Ablaufdatum nachverfolgen und rotieren.

Zertifikatformat

Das Importieren der Zertifikate in das AWS Secrets Manager ist identisch, unabhängig davon, ob es sich um das private Plugin-Zertifikat oder das öffentliche Ranger-Administratorzertifikat handelt. Vor dem Import der TLS Zertifikate müssen die Zertifikate im PEM 509x-Format vorliegen.

Ein Beispiel für ein öffentliches Zertifikat hat das folgende Format:

```
-----BEGIN CERTIFICATE-----
...Certificate Body...
-----END CERTIFICATE-----
```

Ein Beispiel für ein privates Zertifikat hat das folgende Format:

```
-----BEGIN PRIVATE KEY-----
...Private Certificate Body...
-----END PRIVATE KEY-----
```

```
-----BEGIN CERTIFICATE-----
...Trust Certificate Body...
-----END CERTIFICATE-----
```

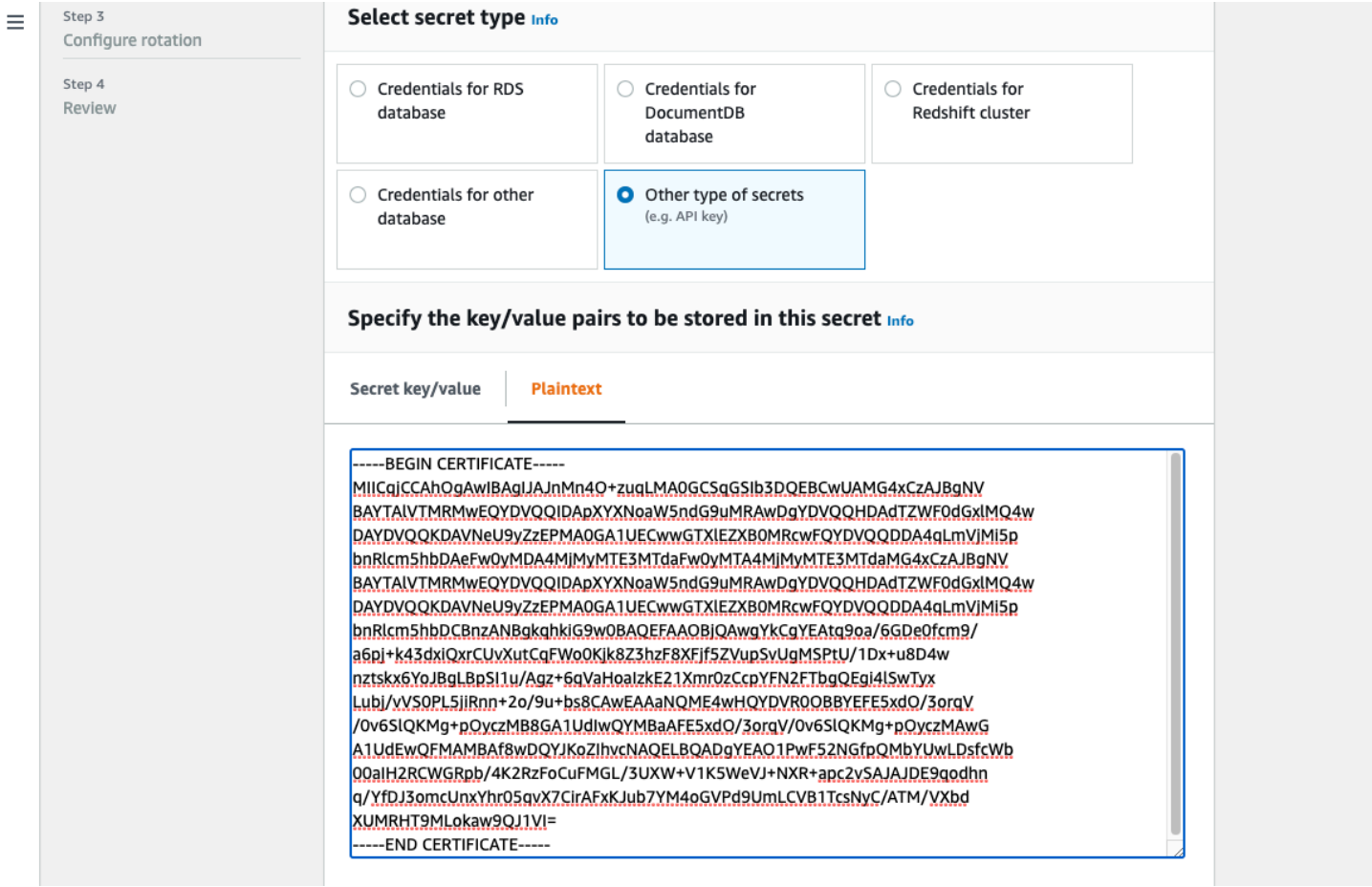
Das private Zertifikat sollte auch ein Vertrauenszertifikat enthalten.

Mit folgendem Befehl können Sie prüfen, ob die Zertifikate das richtige Format haben:

```
openssl x509 -in <PEM FILE> -text
```

Ein Zertifikat in AWS Secrets Manager importieren

Wenn Sie Ihr Geheimnis im Secrets Manager erstellen, wählen Sie unter Geheimtyp die Option Andere Art von Geheimnissen und fügen Sie Ihr PEM codiertes Zertifikat in das Klartext-Feld ein.



Starten Sie einen EMR Cluster

Bevor Sie einen EMR Amazon-Cluster mit Apache Ranger starten, stellen Sie sicher, dass jede Komponente die folgenden Mindestanforderungen an die Version erfüllt:

- Amazon EMR 5.32.0 oder höher oder 6.3.0 oder höher. Wir empfehlen Ihnen, die neueste EMR Amazon-Release-Version zu verwenden.
- Apache Ranger Admin-Server 2.x.

Führen Sie folgende Schritte aus.

- Installieren Sie Apache Ranger, wenn das noch nicht geschehen ist. Weitere Informationen finden Sie unter Installation von Apache Ranger 0.5.0.
- Stellen Sie sicher, dass zwischen Ihrem EMR Amazon-Cluster und dem Apache Ranger Admin-Server eine Netzwerkverbindung besteht. Siehe Richten Sie den Ranger-Admin-Server ein
- Erstellen Sie die erforderlichen IAM Rollen. Siehe IAMRollen für die native Integration mit Apache Ranger.
- Erstellen Sie eine EMR Sicherheitskonfiguration für die Apache Ranger-Installation. Weitere Informationen finden Sie unter Erstellen Sie die EMR Sicherheitskonfiguration.

Zeppelin für Apache Ranger-fähige Amazon-Cluster konfigurieren EMR

Das Thema behandelt die Konfiguration von Apache Zeppelin für einen Apache Ranger-fähigen EMR Amazon-Cluster, sodass Sie Zeppelin als Notizbuch für die interaktive Datenexploration verwenden können. Zeppelin ist in den EMR Amazon-Release-Versionen 5.0.0 und höher enthalten. Frühere Versionen enthalten Zeppelin als Sandbox-Anwendung. Weitere Informationen finden Sie unter Amazon EMR 4.x-Release-Versionen im Amazon EMR Release Guide.

Standardmäßig ist Zeppelin mit einem Standard-Login und Passwort konfiguriert, was in einer Umgebung mit mehreren Mandanten nicht sicher ist.

Führen Sie für die Konfiguration von Zeppelin die folgenden Schritte aus.

1. Verändern Sie den Authentifizierungsmechanismus.

   Ändern Sie die `shiro.ini`-Datei, um Ihren bevorzugten Authentifizierungsmechanismus zu implementieren. Zeppelin unterstützt Active Directory, LDAPPAM, und Knox. SSO Weitere Informationen finden Sie unter Apache-Shiro-Authentifizierung für Apache Zeppelin.
2. Konfigurieren Sie Zeppelin so, dass es sich als Endbenutzer ausgibt

Wenn Sie Zeppelin erlauben, sich als Endbenutzer auszugeben, können von Zeppelin eingereichte Aufträge als dieser Endbenutzer ausgeführt werden. Fügen Sie die folgende Konfiguration zu `core-site.xml` hinzu:

```
[
  {
    "Classification": "core-site",
    "Properties": {
      "hadoop.proxyuser.zeppelin.hosts": "*",
      "hadoop.proxyuser.zeppelin.groups": "*"
    },
    "Configurations": [
    ]
  }
]
```

Fügen Sie als Nächstes die folgende Konfiguration zu `hadoop-kms-site.xml` in `/etc/hadoop/conf` hinzu:

```
[
  {
    "Classification": "hadoop-kms-site",
    "Properties": {
      "hadoop.kms.proxyuser.zeppelin.hosts": "*",
      "hadoop.kms.proxyuser.zeppelin.groups": "*"
    },
    "Configurations": [
    ]
  }
]
```

Sie können diese Konfigurationen auch mithilfe der Konsole zu Ihrem EMR Amazon-Cluster hinzufügen, indem Sie die Schritte unter Instanzgruppe neu konfigurieren in der Konsole befolgen.

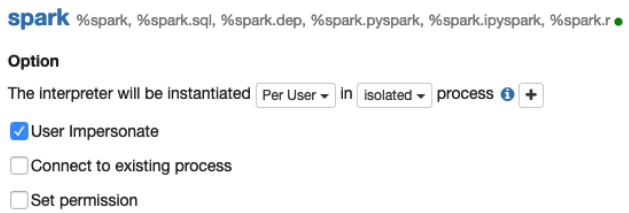
3. Erlauben Sie Zeppelin, als Endbenutzer sudo auszuführen

   Erstellen Sie eine Datei `/etc/sudoers.d/90-zeppelin-user`, die folgendes enthält:

```
zeppelin ALL=(ALL) NOPASSWD:ALL
```

4. Ändern Sie die Einstellungen der Interpreter, um Benutzeraufträge in ihren eigenen Prozessen auszuführen.

   Konfigurieren Sie alle Interpreter so, dass sie die Interpreter „pro Benutzer“ in „isolierten“ Prozessen instanziieren.



5. Modifizieren Sie **`zeppelin-env.sh`**

   Fügen Sie Folgendes zu `zeppelin-env.sh` hinzu, damit Zeppelin die Interpreter als Endbenutzer startet:

```
ZEPPELIN_IMPERSONATE_USER=`echo ${ZEPPELIN_IMPERSONATE_USER} | cut -d @ -f1`
export ZEPPELIN_IMPERSONATE_CMD='sudo -H -u ${ZEPPELIN_IMPERSONATE_USER} bash -c'
```

   Fügen Sie Folgendes zu `zeppelin-env.sh` hinzu, um die standardmäßigen Notebookberechtigungen für den Ersteller auf Schreibgeschützt zu ändern:

```
export ZEPPELIN_NOTEBOOK_PUBLIC="false"
```

   Fügen Sie abschließend Folgendes hinzu, `zeppelin-env.sh` um den EMR RecordServer Klassenpfad nach der ersten `CLASSPATH` Anweisung einzubeziehen:

```
export CLASSPATH="$CLASSPATH:/usr/share/aws/emr/record-server/lib/aws-emr-record-
server-connector-common.jar:/usr/share/aws/emr/record-server/lib/aws-emr-record-
server-spark-connector.jar:/usr/share/aws/emr/record-server/lib/aws-emr-record-
server-client.jar:/usr/share/aws/emr/record-server/lib/aws-emr-record-server-
common.jar:/usr/share/aws/emr/record-server/lib/jars/secret-agent-interface.jar"
```

6. Starten Sie Zeppelin neu.

   Führen Sie für den folgenden Befehle aus, um Zeppelin neu zu starten:

```
sudo systemctl restart zeppelin
```

Bekannte Probleme

Bekannte Probleme

Es gibt ein bekanntes Problem in EMR Amazon-Version 5.32, bei dem die Berechtigungen für geändert `hive-site.xml` wurden, sodass nur privilegierte Benutzer es lesen können, da darin möglicherweise Anmeldeinformationen gespeichert sind. Dies könnte Hue am Lesen von `hive-site.xml` hindern und dazu führen, dass Webseiten ständig neu geladen werden. Wenn dieses Problem auftritt, fügen Sie die folgende Konfiguration hinzu, um das Problem zu beheben:

```
[
  {
    "Classification": "hue-ini",
    "Properties": {},
    "Configurations": [
      {
        "Classification": "desktop",
        "Properties": {
          "server_group":"hive_site_reader"
         },
        "Configurations":[
        ]
      }
    ]
  }
]
```

Es gibt ein bekanntes Problem, dass das EMRFS S3-Plugin für Apache Ranger derzeit die Security Zone-Funktion von Apache Ranger nicht unterstützt. Einschränkungen der Zugriffskontrolle, die mit der Sicherheitszone-Funktion definiert wurden, gelten nicht für Ihre EMR Amazon-Cluster.

Anwendung UIs

Standardmäßig führen Benutzeroberflächen von Anwendungen keine Authentifizierung durch. Dazu gehören unter anderem die ResourceManager NodeManager Benutzeroberfläche, die Benutzeroberfläche und die Livy-Benutzeroberfläche. Darüber hinaus kann jeder Benutzer, der auf die UIs zugreifen kann, Informationen zu den Jobs aller anderen Benutzer einsehen.

Wenn dieses Verhalten nicht erwünscht ist, sollten Sie sicherstellen, dass eine Sicherheitsgruppe verwendet wird, um den Zugriff der Benutzer auf die Anwendung UIs einzuschränken.

HDFSStandardberechtigungen

Standardmäßig erhalten die Objekte, in HDFS denen Benutzer Objekte erstellen, weltweit lesbare Berechtigungen. Dies kann möglicherweise dazu führen, dass Daten von Benutzern gelesen werden, die keinen Zugriff darauf haben sollten. Gehen Sie wie folgt vor, um dieses Verhalten so zu ändern, dass die standardmäßigen Dateiberechtigungen nur vom Ersteller des Auftrags auf Lese- und Schreibzugriff festgelegt werden.

Geben Sie bei der Erstellung Ihres EMR Clusters die folgende Konfiguration an:

```
[
  {
    "Classification": "hdfs-site",
    "Properties": {
      "dfs.namenode.acls.enabled": "true",
      "fs.permissions.umask-mode": "077",
      "dfs.permissions.superusergroup": "hdfsadmingroup"
    }
  }
]
```

Führen Sie außerdem die folgende Bootstrap-Aktion aus:

```
--bootstrap-actions Name='HDFS UMask Setup',Path=s3://elasticmapreduce/hdfs/umask/
umask-main.sh
```

Apache-Ranger-Plugins

Apache-Ranger-Plugin – Dieses Plugin validiert den Zugriff eines Benutzers anhand der Autorisierungsrichtlinien, die im Apache-Ranger-Richtlinien-Admin-Server definiert sind.

Themen

- Apache- Hive-Plugin
- Apache Spark Plugin
- EMRFSS3-Plugin
- Trino-Plugin

Apache- Hive-Plugin

Apache Hive ist eine beliebte Ausführungs-Engine innerhalb des Hadoop-Ökosystems. Amazon EMR bietet ein Apache Ranger-Plugin, um detaillierte Zugriffskontrollen für Hive bereitstellen zu

können. Das Plugin ist mit Admin-Server-Version von Open-Source-Apache-Ranger 2.0 und höher kompatibel.

Themen

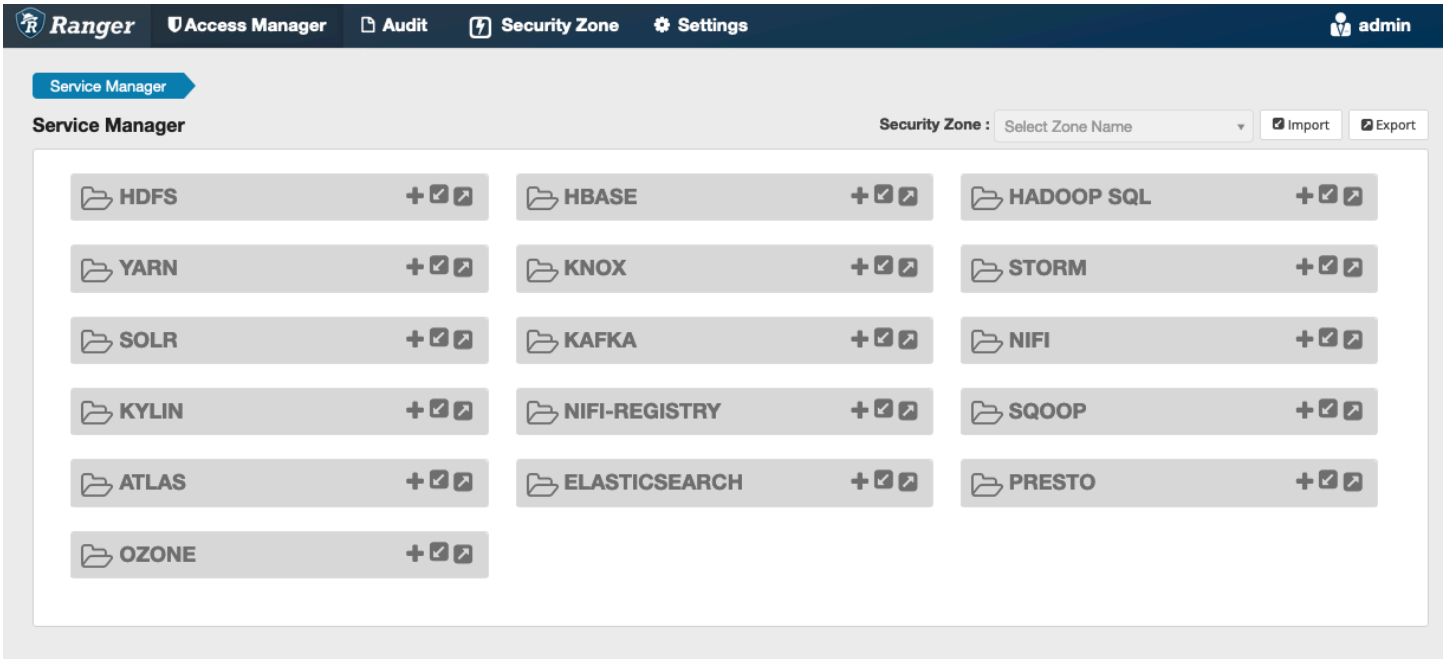
- Unterstützte Features
- Installation der Servicekonfiguration
- Überlegungen
- Einschränkungen

Unterstützte Features

Das Apache Ranger-Plugin für Hive on EMR unterstützt alle Funktionen des Open-Source-Plug-ins, einschließlich Zugriffskontrollen auf Datenbank-, Tabellen- und Spaltenebene sowie Zeilenfilterung und Datenmaskierung. Eine Tabelle mit Hive-Befehlen und den zugehörigen Ranger-Berechtigungen finden Sie unter Zuordnung von Hive-Befehlen zu Ranger-Berechtigungen.

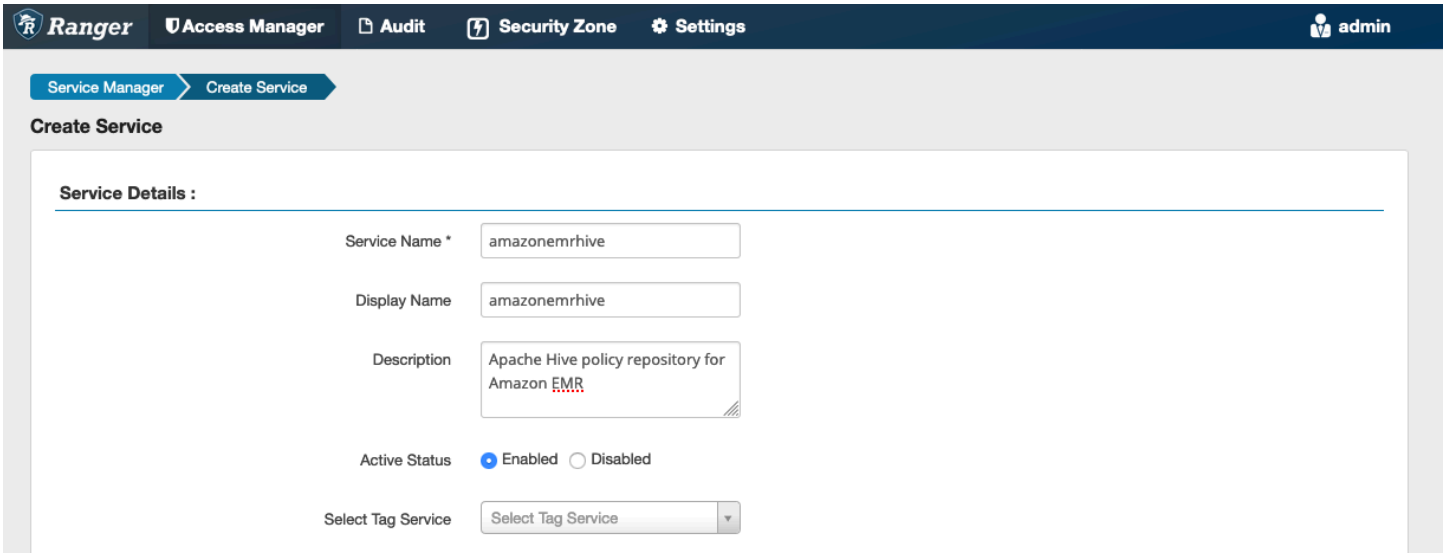
Installation der Servicekonfiguration

Das Apache Hive-Plug-in ist mit der bestehenden Hive-Dienstdefinition in Apache Hive Hadoop kompatibel. SQL



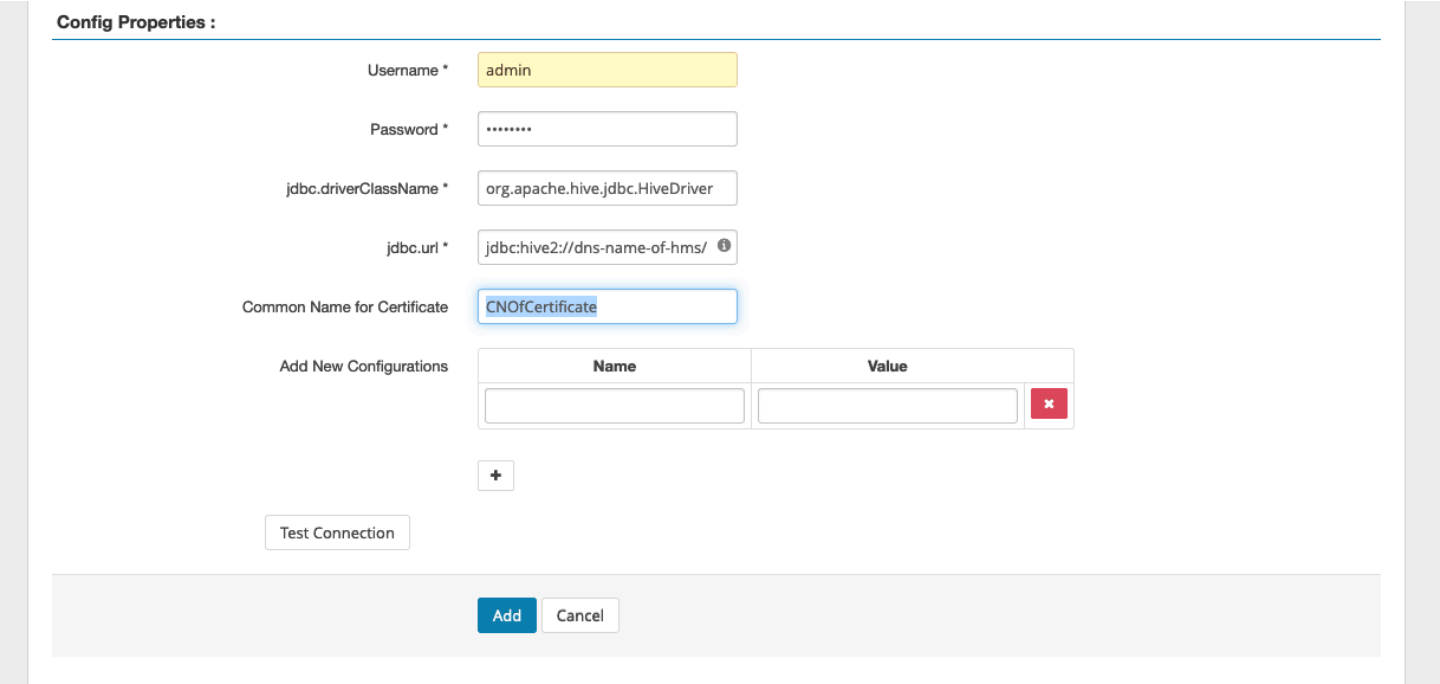
Wenn Sie keine Instanz des Dienstes unter Hadoop habenSQL, wie oben gezeigt, können Sie eine erstellen. Klicken Sie auf das + neben SQL Hadoop.

1. Servicename (falls angezeigt): Geben Sie den Servicenamen ein. Der vorgeschlagene Wert ist **`amazonemrhive`**. Notieren Sie sich diesen Dienstnamen. Er wird benötigt, wenn Sie eine EMR Sicherheitskonfiguration erstellen.
2. Anzeigename: Der Name, der für diesen Service angezeigt wird. Der vorgeschlagene Wert ist **`amazonemrhive`**.

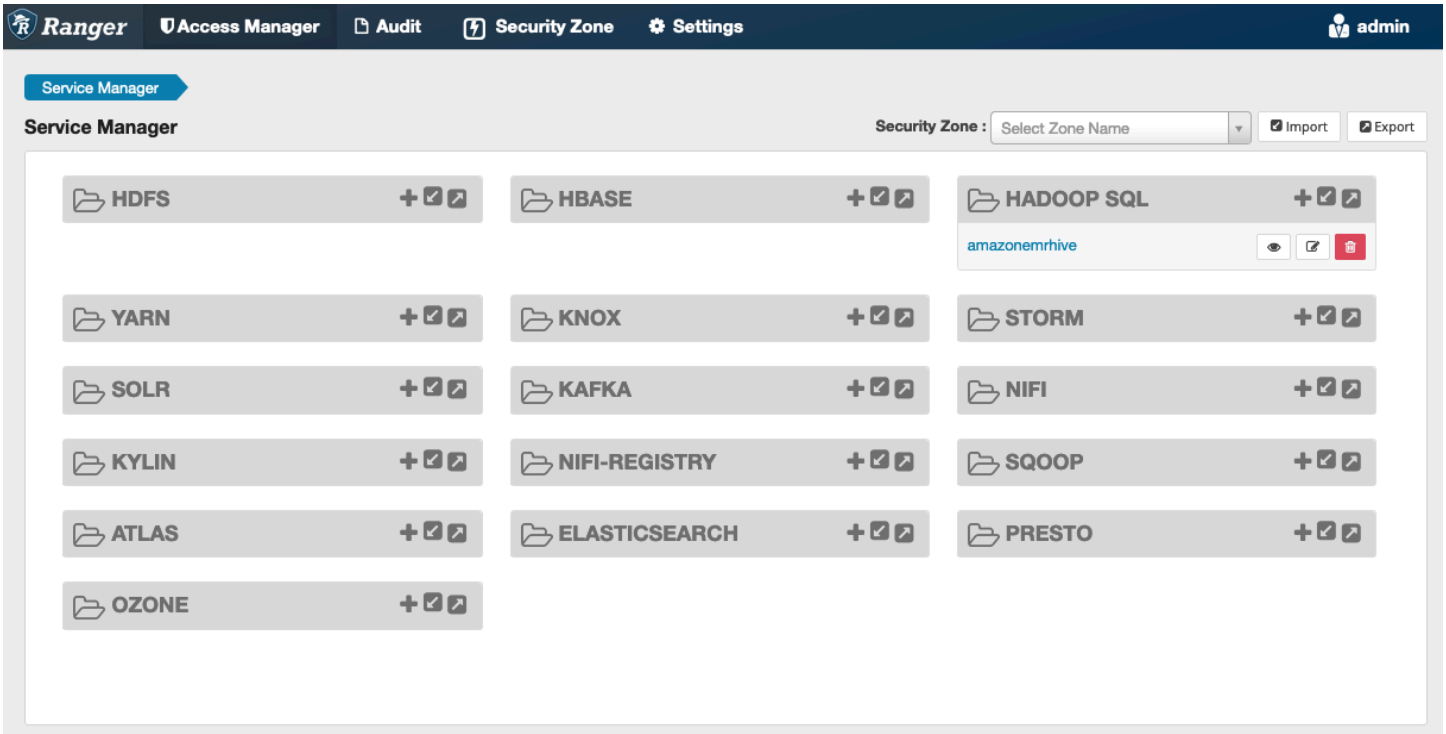


Die Apache Hive Config Properties werden verwendet, um eine Verbindung zu Ihrem Apache Ranger Admin-Server mit einer HiveServer 2 herzustellen, um die auto Vervollständigung bei der Erstellung von Richtlinien zu implementieren. Die folgenden Eigenschaften müssen nicht korrekt sein, wenn Sie nicht über einen persistenten HiveServer 2-Prozess verfügen, und sie können mit beliebigen Informationen gefüllt werden.

- Benutzername: Geben Sie einen Benutzernamen für die JDBC Verbindung zu einer Instanz einer HiveServer 2-Instanz ein.
- Passwort: Geben Sie das Passwort für den obigen Benutzernamen ein.
- jdbc.driver. ClassName: Geben Sie den Klassennamen der JDBC Klasse für die Apache Hive-Konnektivität ein. Sie können den Standardwert verwenden.
- jdbc.url: Geben Sie die JDBC Verbindungszeichenfolge ein, die beim Herstellen einer Verbindung zu 2 verwendet werden soll. HiveServer
- Allgemeiner Name für das Zertifikat: Das CN-Feld innerhalb des Zertifikats, das verwendet wird, um von einem Client-Plugin aus eine Verbindung zum Admin-Server herzustellen. Dieser Wert muss mit dem CN-Feld in Ihrem TLS Zertifikat übereinstimmen, das für das Plugin erstellt wurde.



Mit der Schaltfläche Verbindung testen wird getestet, ob die obigen Werte verwendet werden können, um erfolgreich eine Verbindung mit der HiveServer 2-Instanz herzustellen. Sobald der Service erfolgreich erstellt wurde, sollte der Service Manager wie folgt aussehen:



Überlegungen

Hive-Metadatenserver

Zum Schutz vor unbefugtem Zugriff können nur vertrauenswürdige Engines, insbesondere Hive und `emr_record_server`, auf den Hive-Metadatenserver zugreifen. Auf den Hive-Metadatenserver greifen auch alle Knoten im Cluster zu. Der erforderliche Port 9 083 ermöglicht allen Knoten den Zugriff auf den Hauptknoten.

Authentifizierung

Standardmäßig ist Apache Hive für die Authentifizierung mithilfe von Kerberos konfiguriert, wie in der Sicherheitskonfiguration konfiguriert. EMR HiveServer2 kann so konfiguriert werden, dass Benutzer auch mithilfe von authentifiziert werden. LDAP Weitere Informationen finden Sie unter Implementierung der LDAP Authentifizierung für Hive auf einem EMR Amazon-Cluster mit mehreren Mandanten.

Einschränkungen

Die folgenden Einschränkungen gelten derzeit für das Apache Hive-Plug-in auf Amazon EMR 5.x:

- Hive-Rollen werden derzeit nicht unterstützt. Die Anweisungen „Grant“ und „Revoke“ werden nicht unterstützt.
- Hive CLI wird nicht unterstützt. JDBC/Beeline ist die einzige autorisierte Methode, Hive zu verbinden.
- `hive.server2.builtin.udf.blacklist`Die Konfiguration sollte mit Daten gefüllt seinUDFs, die Sie für unsicher halten.

Apache Spark Plugin

Amazon EMR hat integriert EMR RecordServer , um eine differenzierte Zugriffskontrolle für Spark bereitzustellen. SQL EMR's RecordServer ist ein privilegierter Prozess, der auf allen Knoten eines Apache Ranger-fähigen Clusters ausgeführt wird. Wenn ein Spark-Treiber oder -Executor eine SQL Spark-Anweisung ausführt, durchlaufen alle Metadaten und Datenanfragen den. RecordServer Weitere Informationen EMR RecordServer dazu finden Sie auf der EMRAmazon-Komponenten Seite.

Themen

- Unterstützte Features

- Stellen Sie die Servicedefinition erneut bereit, um Anweisungen INSERTALTER, oder DDL zu verwenden
- Installation der Servicedefinition
- SQLSpark-Richtlinien erstellen
- Überlegungen
- Einschränkungen

Unterstützte Features

SQLAuskunft/Aktion des Rangers	STATUS	Unterstützte Version EMR
SELECT	Unterstützt	Ab 5.32
SHOW DATABASES	Unterstützt	Ab 5.32
SHOW COLUMNS	Unterstützt	Ab 5.32
SHOW TABLES	Unterstützt	Ab 5.32
SHOW TABLE PROPERTIES	Unterstützt	Ab 5.32
DESCRIBE TABLE	Unterstützt	Ab 5.32
INSERT OVERWRITE	Unterstützt	Ab 5.34 und 6.4
INSERT INTO	Unterstützt	Ab 5.34 und 6.4
ALTER TABLE	Unterstützt	Ab 6.4
CREATE TABLE	Unterstützt	Ab 5.35 und 6.7

SQLAuskunft/Aktion des Rangers	STATUS	Unterstützte Version EMR
CREATE DATABASE	Unterstützt	Ab 5.35 und 6.7
DROP TABLE	Unterstützt	Ab 5.35 und 6.7
DROP DATABASE	Unterstützt	Ab 5.35 und 6.7
DROP VIEW	Unterstützt	Ab 5.35 und 6.7
CREATE VIEW	Nicht unterstützt	

Die folgenden Funktionen werden bei der Verwendung von Spark unterstütztSQL:

- Eine detaillierte Zugriffskontrolle für Tabellen im Hive-Metastore und Richtlinien können auf Datenbank-, Tabellen- und Spaltenebene erstellt werden.
- Die Richtlinien von Apache Ranger können Richtlinien für die Gewährung und die Ablehnung von Benutzern und Gruppen beinhalten.
- Prüfereignisse werden an CloudWatch Logs übermittelt.

Stellen Sie die Servicedefinition erneut bereit, um Anweisungen INSERTALTER, oder DDL zu verwenden

> **Note**
>
> Ab Amazon EMR 6.4 können Sie Spark SQL mit den Anweisungen: INSERT INTO INSERTOVERWRITE, oder verwenden ALTERTABLE. Ab Amazon EMR 6.7 können Sie Spark verwenden, SQL um Datenbanken und Tabellen zu erstellen oder zu löschen. Wenn Sie bereits über eine Installation auf dem Apache Ranger-Server mit bereitgestellten Apache Spark-Servicedefinitionen verfügen, verwenden Sie den folgenden Code, um die Servicedefinitionen erneut bereitzustellen.

```
# Get existing Spark service definition id calling Ranger REST API and JSON
 processor
curl --silent -f -u <admin_user_login>:<password_for_ranger_admin_user> \
-H "Accept: application/json" \
-H "Content-Type: application/json" \
-k 'https://*<RANGER SERVER ADDRESS>*:6182/service/public/v2/api/servicedef/
name/amazon-emr-spark' | jq .id

# Download the latest Service definition
wget https://s3.amazonaws.com/elasticmapreduce/ranger/service-definitions/
version-2.0/ranger-servicedef-amazon-emr-spark.json

# Update the service definition using the Ranger REST API
curl -u <admin_user_login>:<password_for_ranger_admin_user> -X PUT -d @ranger-
servicedef-amazon-emr-spark.json \
-H "Accept: application/json" \
-H "Content-Type: application/json" \
-k 'https://*<RANGER SERVER ADDRESS>*:6182/service/public/v2/api/
servicedef/<Spark service definition id from step 1>'
```

Installation der Servicedefinition

Für die Installation EMR der Apache Spark-Servicedefinition muss der Ranger Admin-Server eingerichtet werden. Siehe Richten Sie den Ranger-Admin-Server ein.

Gehen Sie wie folgt vor, um die Apache-Spark-Servicedefinition zu installieren:

Schritt 1: SSH In den Apache Ranger Admin-Server

Beispielsweise:

```
ssh ec2-user@ip-xxx-xxx-xxx-xxx.ec2.internal
```

Schritt 2: Die Servicedefinition und das Apache-Ranger-Admin-Server-Plugin herunterladen

Laden Sie die Servicedefinition in einem temporären Verzeichnis herunter. Diese Servicedefinition wird von Ranger-2.x-Versionen unterstützt.

```
mkdir /tmp/emr-spark-plugin/
```

```
cd /tmp/emr-spark-plugin/

wget https://s3.amazonaws.com/elasticmapreduce/ranger/service-definitions/version-2.0/
ranger-spark-plugin-2.x.jar
wget https://s3.amazonaws.com/elasticmapreduce/ranger/service-definitions/version-2.0/
ranger-servicedef-amazon-emr-spark.json
```

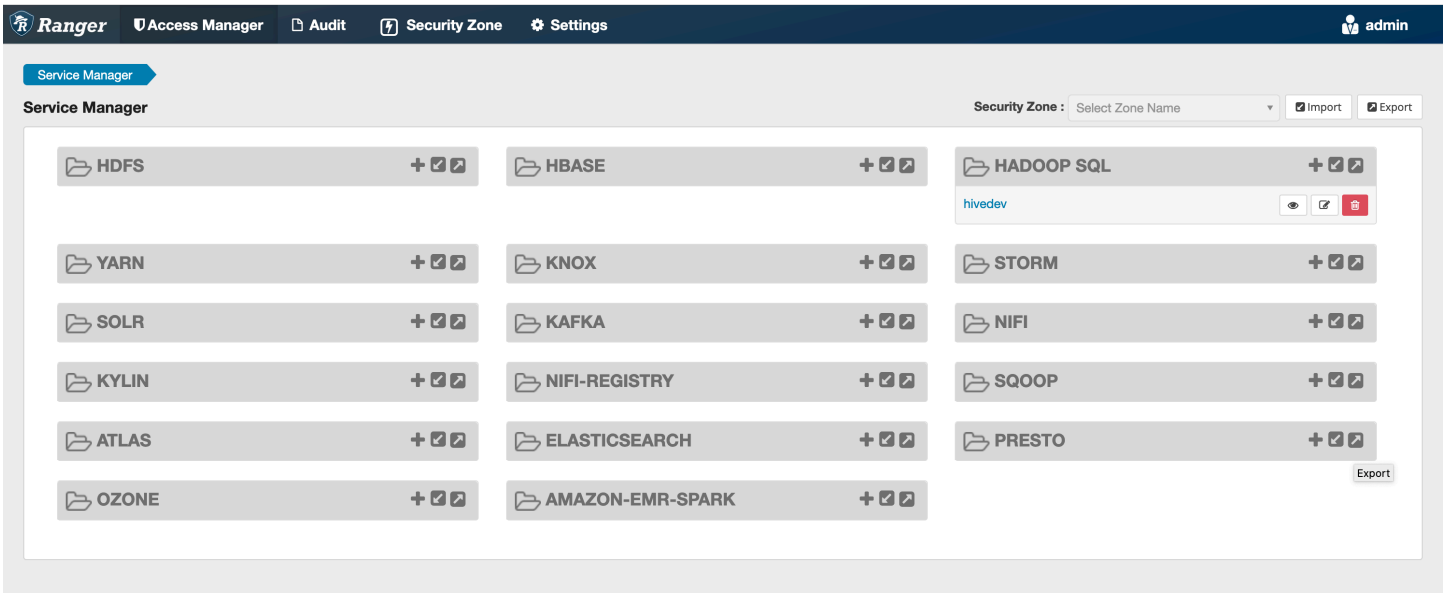
Schritt 3: Installieren Sie das Apache Spark-Plugin für Amazon EMR

```
export RANGER_HOME=.. # Replace this Ranger Admin's home directory eg /usr/lib/ranger/
ranger-2.0.0-admin
mkdir $RANGER_HOME/ews/webapp/WEB-INF/classes/ranger-plugins/amazon-emr-spark
mv ranger-spark-plugin-2.x.jar $RANGER_HOME/ews/webapp/WEB-INF/classes/ranger-plugins/
amazon-emr-spark
```

Schritt 4: Registrieren Sie die Apache Spark-Servicedefinition für Amazon EMR

```
curl -u *<admin users login>*:*_<_**_password_ **_for_** _ranger admin user_**_>_* -X
 POST -d @ranger-servicedef-amazon-emr-spark.json \
-H "Accept: application/json" \
-H "Content-Type: application/json" \
-k 'https://*<RANGER SERVER ADDRESS>*:6182/service/public/v2/api/servicedef'
```

Wenn dieser Befehl erfolgreich ausgeführt wird, sehen Sie in Ihrer Ranger-Admin-Benutzeroberfläche einen neuen Dienst namens "AMAZON- EMR - SPARK „, wie in der folgenden Abbildung gezeigt (Ranger-Version 2.0 wird gezeigt).

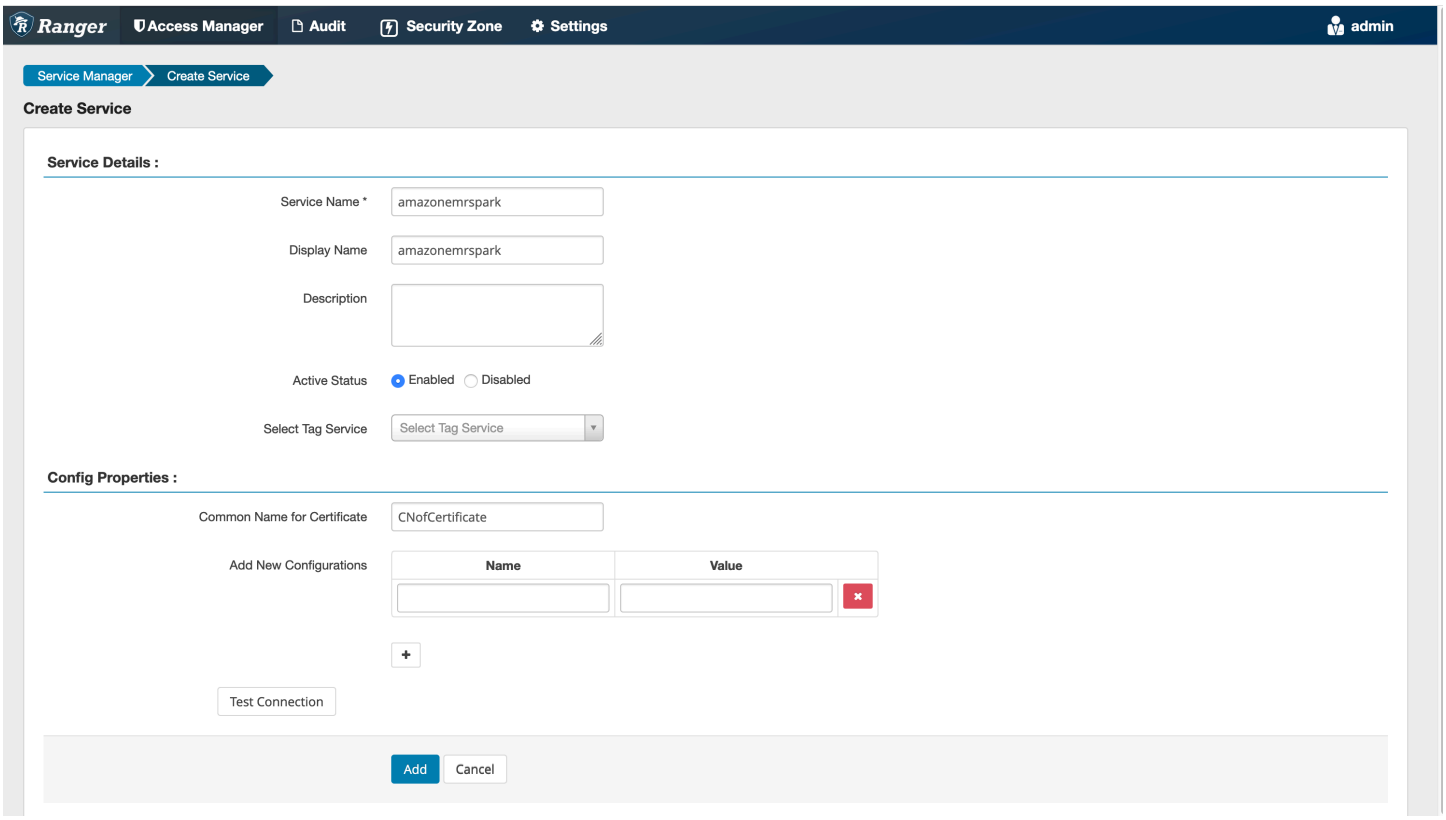


Schritt 5: Erstellen Sie eine Instanz der AMAZON - EMR - SPARK -Anwendung

Servicename (falls angezeigt): Der Servicename, der verwendet wird. Der vorgeschlagene Wert ist **`amazonemrspark`**. Notieren Sie sich diesen Dienstnamen, da er bei der Erstellung einer EMR Sicherheitskonfiguration benötigt wird.

Anzeigename: Der Name, der für diese Instance angezeigt werden soll. Der vorgeschlagene Wert ist **`amazonemrspark`**.

Allgemeiner Name für das Zertifikat: Das CN-Feld innerhalb des Zertifikats, das verwendet wird, um von einem Client-Plugin aus eine Verbindung zum Admin-Server herzustellen. Dieser Wert muss mit dem CN-Feld in Ihrem TLS Zertifikat übereinstimmen, das für das Plugin erstellt wurde.



> **Note**
>
> Das TLS Zertifikat für dieses Plugin sollte im Trust Store auf dem Ranger Admin-Server registriert worden sein. Weitere Details finden Sie unter TLSZertifikate.

SQLSpark-Richtlinien erstellen

Beim Erstellen einer neuen Richtlinie müssen folgende Felder ausgefüllt werden:

Richtlinienname: Der Name dieser Richtlinie.

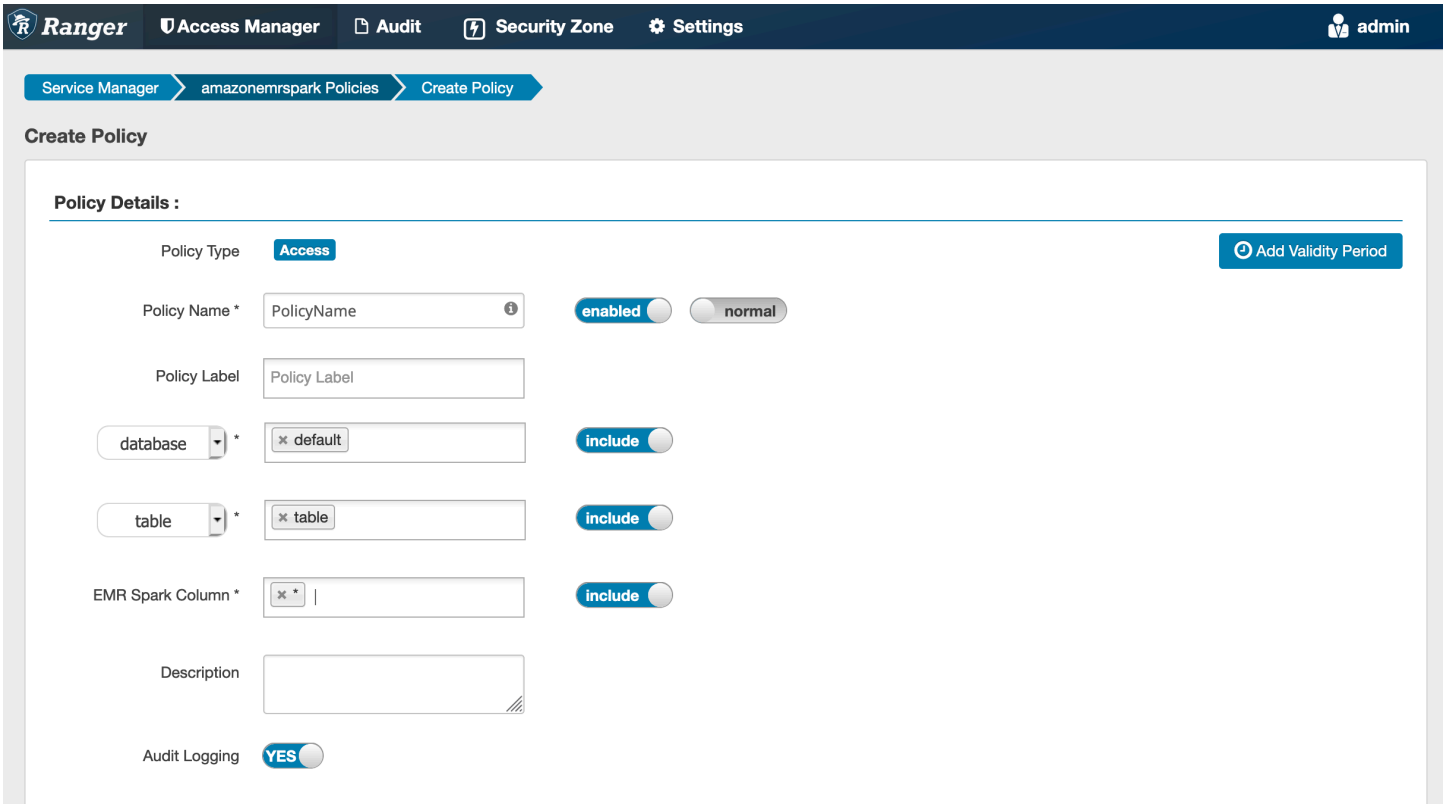
Richtlinienbezeichnung: Eine Bezeichnung, die Sie dieser Richtlinie hinzufügen können.

Datenbank: Die Datenbank, für die diese Richtlinie gilt. Der Platzhalter „*“ steht für alle Tabellen.

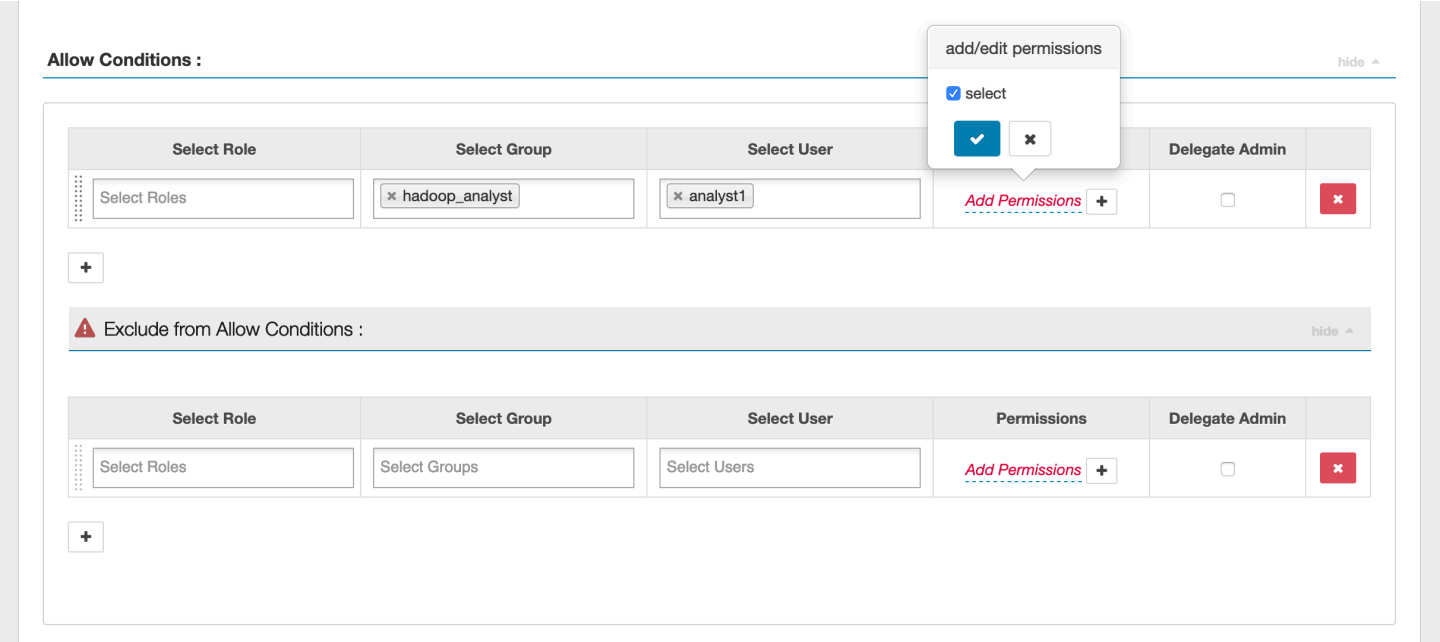
Tabelle: Die Tabellen, für die diese Richtlinie gilt. Der Platzhalter „*“ steht für alle Tabellen.

EMRSpark-Spalte: Die Spalten, für die diese Richtlinie gilt. Der Platzhalter „*“ steht für alle Spalten.

Beschreibung: Eine Beschreibung dieser Richtlinie.



Um die Benutzer und Gruppen anzugeben, geben Sie die Benutzer und Gruppen unten ein, um Berechtigungen zu erteilen. Sie können auch Ausnahmen für die Bedingungen Zulassen und Verweigern angeben.



Nachdem Sie die Bedingungen für das Zulassen und Verweigern angegeben haben, klicken Sie auf Speichern.

Überlegungen

Jeder Knoten innerhalb des EMR Clusters muss in der Lage sein, eine Verbindung zum Hauptknoten auf Port 9083 herzustellen.

Einschränkungen

Die folgenden Einschränkungen gelten derzeit für das Apache-Spark-Plugin:

- Der Record Server stellt immer eine Verbindung zur HMS Ausführung auf einem EMR Amazon-Cluster her. Konfigurieren HMS Sie bei Bedarf die Verbindung zum Remote-Modus. Sie sollten keine Konfigurationswerte in die Apache-Spark-Konfigurationsdatei Hive-site.xml einfügen.
- Tabellen, die mit Spark-Datenquellen auf CSV oder Avro erstellt wurden, können mit nicht gelesen werden. EMR RecordServer Verwenden Sie Hive, um Daten zu erstellen und zu schreiben, und lesen Sie sie mit Record.
- Delta Lake- und Hudi-Tabellen werden nicht unterstützt.
- Benutzer müssen Zugriff auf die Standarddatenbank haben. Dies ist eine Voraussetzung für Apache Spark.
- Der Ranger-Admin-Server unterstützt die automatische Vervollständigung nicht.
- Das SQL Spark-Plugin für Amazon unterstützt EMR keine Zeilenfilter oder Datenmaskierung.

- Bei der Verwendung ALTER TABLE mit Spark SQL muss ein Partitionsspeicherort das untergeordnete Verzeichnis einer Tabellenposition sein. Das Einfügen von Daten in eine Partition, deren Partitionsspeicherort sich von der Tabellenposition unterscheidet, wird nicht unterstützt.

EMRFSS3-Plugin

Um die Bereitstellung von Zugriffskontrollen für Objekte in S3 auf einem Multi-Tenant-Cluster zu vereinfachen, bietet das EMRFS S3-Plugin Zugriffskontrollen für die Daten in S3, wenn über EMRFS diese zugegriffen wird. Sie können den Zugriff auf S3-Ressourcen auf Benutzer- und Gruppenebene zulassen.

Um dies zu erreichen, EMRFS sendet Ihre Anwendung, wenn sie versucht, auf Daten innerhalb von S3 zuzugreifen, eine Anfrage nach Anmeldeinformationen an den Secret Agent-Prozess, wo die Anfrage anhand eines Apache Ranger-Plug-ins authentifiziert und autorisiert wird. Wenn die Anfrage autorisiert ist, übernimmt der Secret Agent die IAM Rolle der Apache Ranger Engines mit einer eingeschränkten Richtlinie zur Generierung von Anmeldeinformationen, die nur Zugriff auf die Ranger-Richtlinie haben, die den Zugriff gewährt hat. Die Anmeldeinformationen werden dann an den Zugriff EMRFS auf S3 zurückgegeben.

Themen

Unterstützte Features

EMRFSDas S3-Plugin ermöglicht die Autorisierung auf Speicherebene. Richtlinien können erstellt werden, um Benutzern und Gruppen Zugriff auf S3-Buckets und -Präfixe zu gewähren. Die Autorisierung erfolgt nur gegenEMRFS.

Installation der Servicekonfiguration

Um die EMRFS Dienstdefinition zu installieren, müssen Sie den Ranger Admin-Server einrichten. Informationen zum Einrichten des Servers finden Sie unterRichten Sie den Ranger-Admin-Server ein.

Gehen Sie wie folgt vor, um die EMRFS Dienstdefinition zu installieren.

Schritt 1: SSH in den Apache Ranger Admin-Server.

Beispielsweise:

```
ssh ec2-user@ip-xxx-xxx-xxx-xxx.ec2.internal
```

Schritt 2: Laden Sie die EMRFS Dienstdefinition herunter.

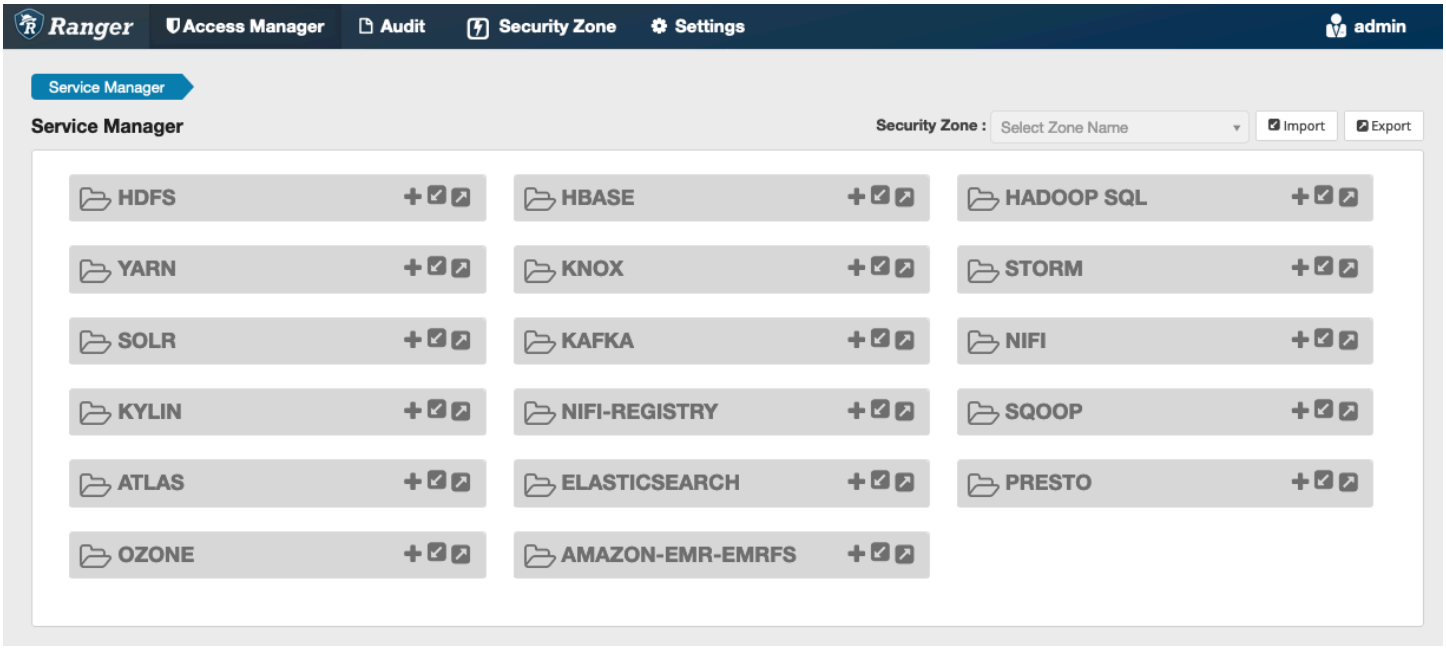
Laden Sie in einem temporären Verzeichnis die EMR Amazon-Servicedefinition herunter. Diese Servicedefinition wird von Ranger-2.x-Versionen unterstützt.

```
wget https://s3.amazonaws.com/elasticmapreduce/ranger/service-definitions/version-2.0/
ranger-servicedef-amazon-emr-emrfs.json
```

Schritt 3: Registrieren Sie die EMRFS S3-Servicedefinition.

```
curl -u *<admin users login>*:*_<_**_password_ **_for_** _ranger admin user_**_>_* -X
 POST -d @ranger-servicedef-amazon-emr-emrfs.json \
-H "Accept: application/json" \
-H "Content-Type: application/json" \
-k 'https://*<RANGER SERVER ADDRESS>*:6182/service/public/v2/api/servicedef'
```

Wenn dieser Befehl erfolgreich ausgeführt wird, wird in der Ranger-Admin-Benutzeroberfläche ein neuer Dienst mit dem Namen "AMAZON- EMR -S3" angezeigt, wie in der folgenden Abbildung gezeigt (Ranger-Version 2.0 wird gezeigt).



Schritt 4: Erstellen Sie eine Instanz der AMAZON - EMR - EMRFS Anwendung.

Erstellen Sie eine Instance der Servicedefinition.

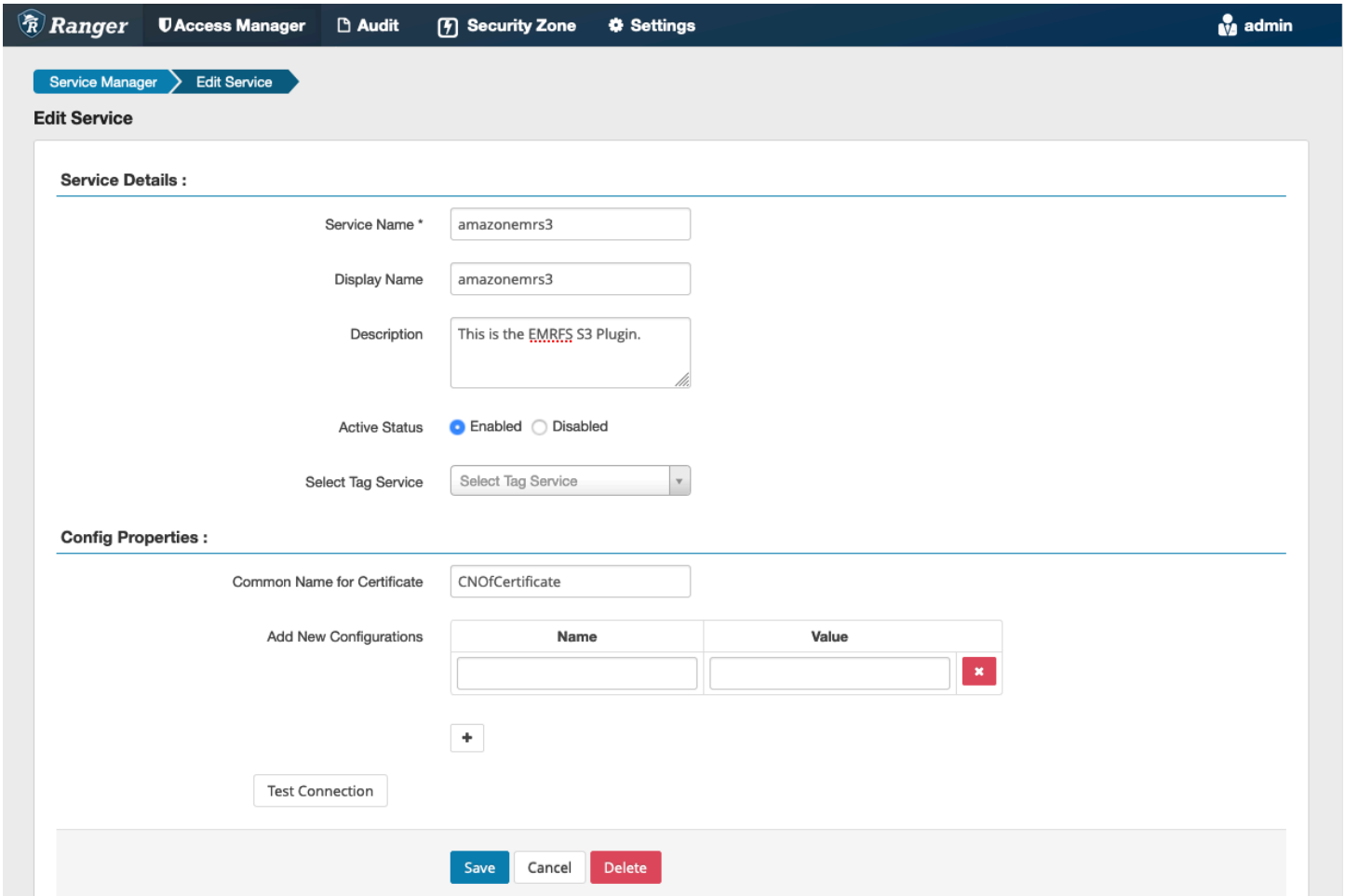
- Klicken Sie auf das + neben AMAZON - EMR -EMRFS.

Füllen Sie die folgenden Felder aus:

Servicename (falls angezeigt): Der vorgeschlagene Wert ist **`amazonemrspark`**. Notieren Sie sich diesen Dienstnamen, da er bei der Erstellung einer EMR Sicherheitskonfiguration benötigt wird.

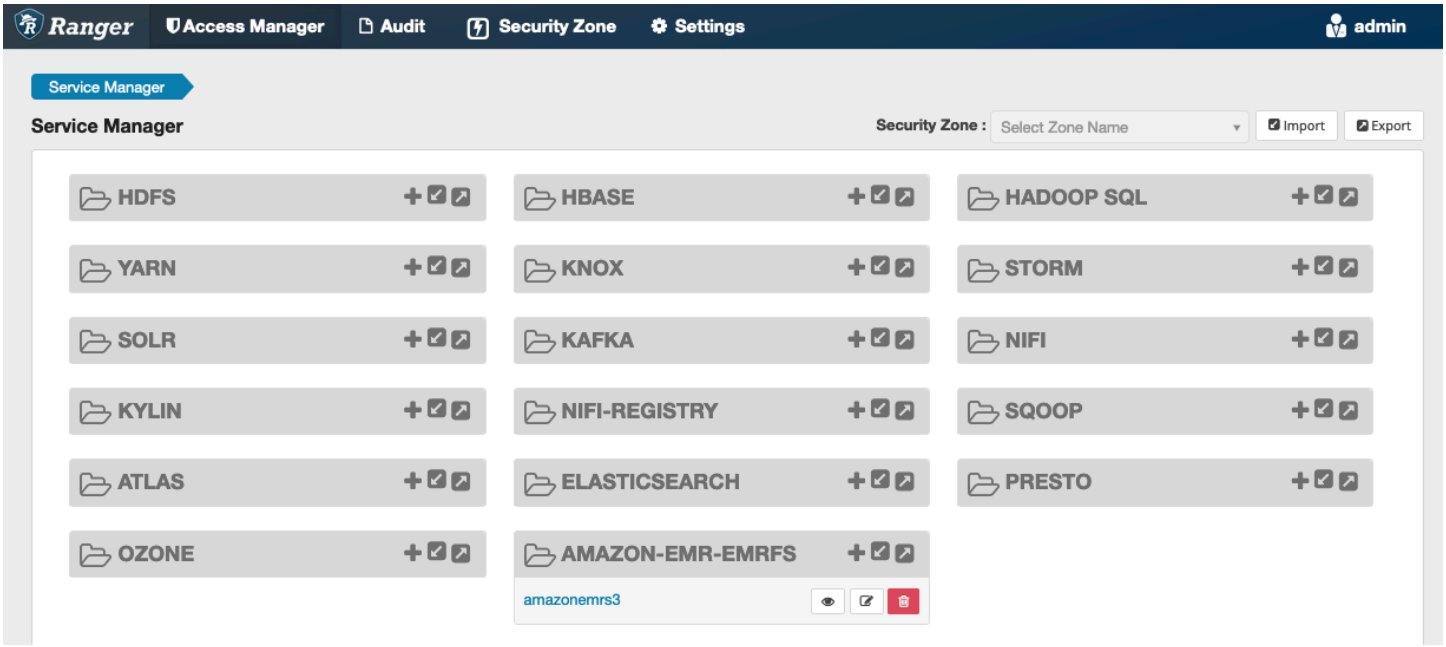
Anzeigename: Der Name, der für diesen Service angezeigt wird. Der vorgeschlagene Wert ist **`amazonemrspark`**.

Allgemeiner Name für das Zertifikat: Das CN-Feld innerhalb des Zertifikats, das verwendet wird, um von einem Client-Plugin aus eine Verbindung zum Admin-Server herzustellen. Dieser Wert muss mit dem CN-Feld in dem TLS Zertifikat übereinstimmen, das für das Plugin erstellt wurde.



> **Note**
>
> Das TLS Zertifikat für dieses Plugin sollte im Trust Store auf dem Ranger Admin-Server registriert worden sein. Weitere Details finden Sie unter TLSZertifikate.

Wenn der Dienst erstellt wird, schließt der Service Manager "AMAZON- EMR - EMRFS „ein, wie in der folgenden Abbildung dargestellt.



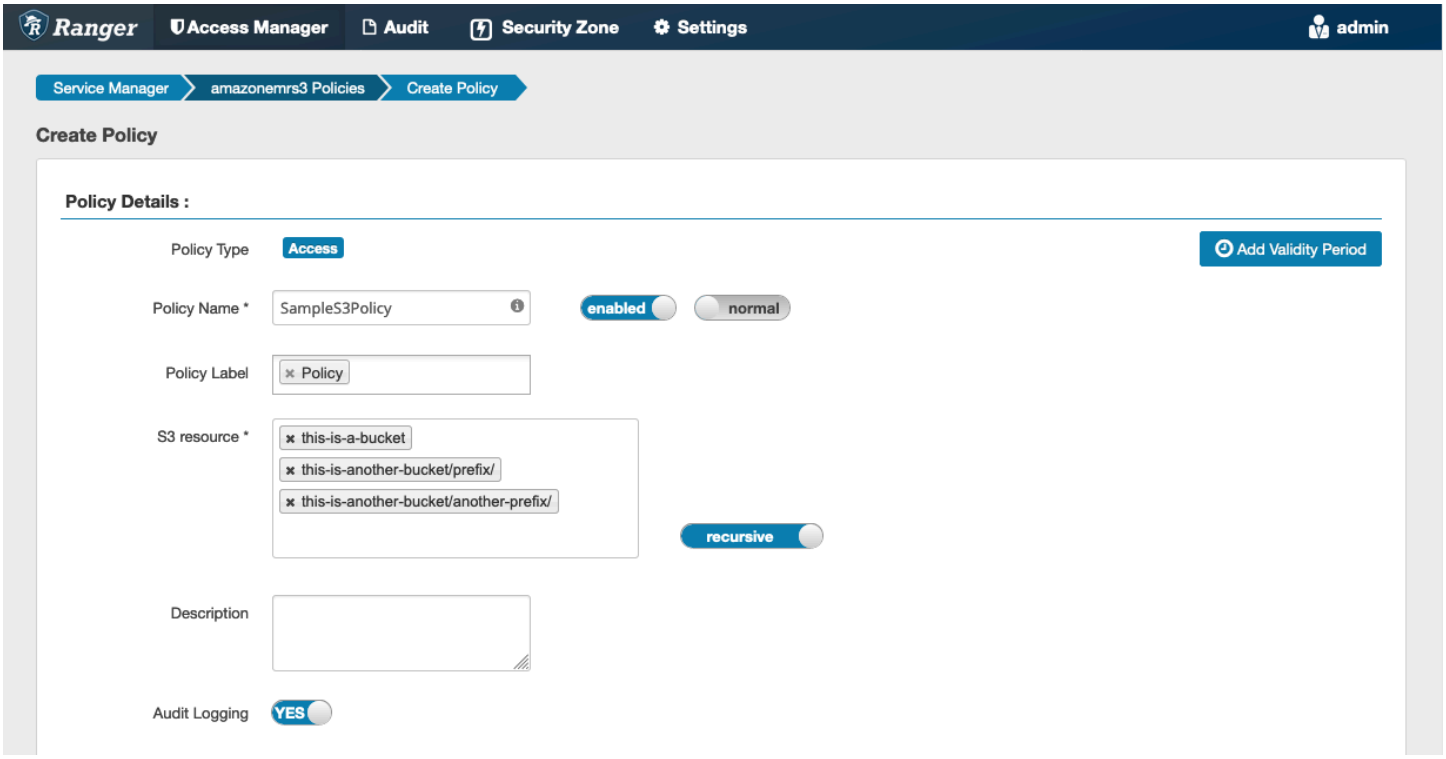
EMRFSS3-Richtlinien erstellen

Füllen Sie die folgenden Felder aus, um auf der Seite Richtlinie erstellen des Service Managers eine neue Richtlinie zu erstellen.

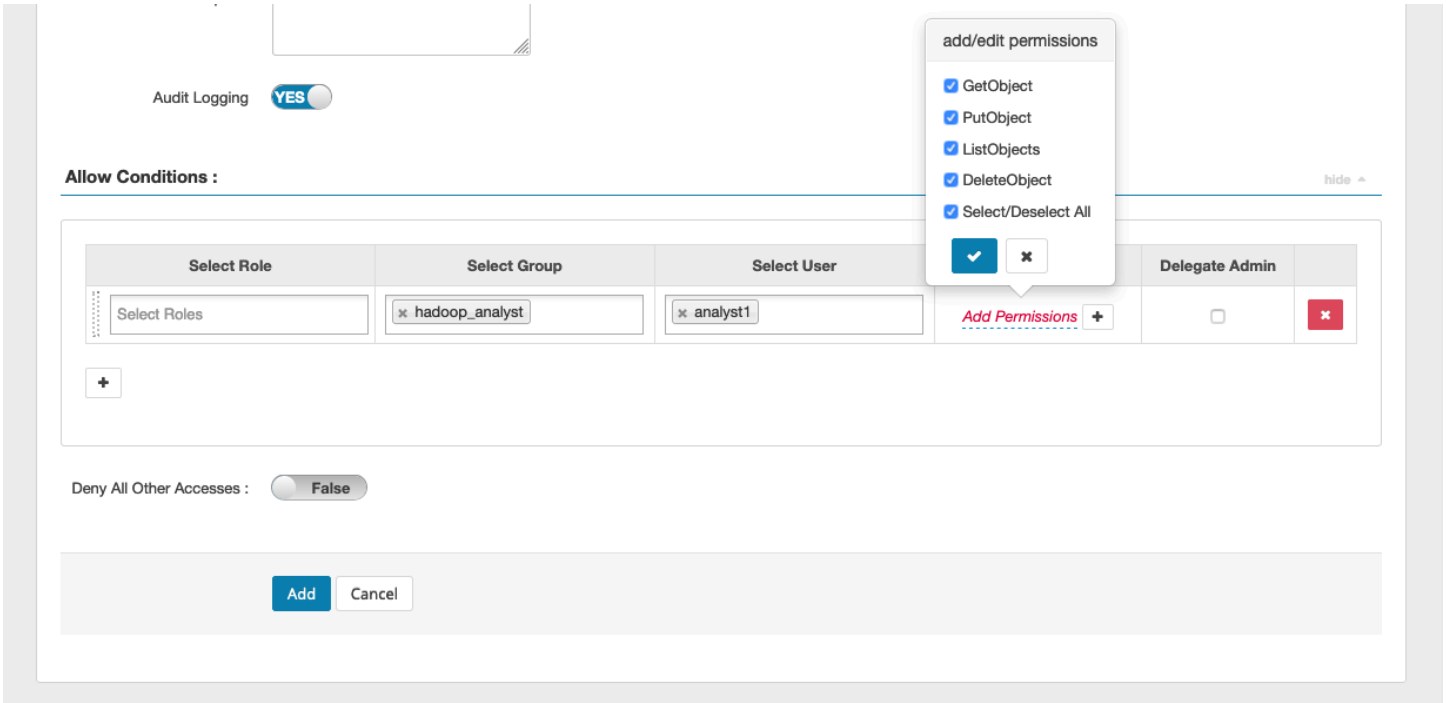
Richtlinienname: Der Name dieser Richtlinie.

Richtlinienbezeichnung: Eine Bezeichnung, die Sie dieser Richtlinie hinzufügen können.

S3-Ressource: Eine Ressource, die mit dem Bucket und dem optionalen Präfix beginnt. Weitere Informationen finden Sie unter Bewährte Methoden für EMRFSHinweise zur Verwendung von S3-Richtlinien. Ressourcen auf dem Ranger-Admin-Server sollten nicht **`s3://`**, **`s3a://`** oder **`s3n://`** enthalten.



Sie können Benutzer und Gruppen angeben, denen Berechtigungen erteilt werden sollen. Sie können auch Ausnahmen für Zulassungsbedingungen und Verweigerungsbedingungen angeben.



> **Note**
>
> Für jede Richtlinie sind maximal drei Ressourcen zulässig. Das Hinzufügen von mehr als drei Ressourcen kann zu einem Fehler führen, wenn diese Richtlinie auf einem Cluster verwendet wird. EMR Beim Hinzufügen von mehr als drei Richtlinien wird eine Erinnerung an das Richtlinienlimit angezeigt.

EMRFSHinweise zur Verwendung von S3-Richtlinien

Bei der Erstellung von S3-Richtlinien in Apache Ranger sind einige Nutzungsaspekte zu beachten.

Berechtigungen für mehrere S3-Objekte

Sie können rekursive Richtlinien und Platzhalterausdrücke verwenden, um mehreren S3-Objekten mit gemeinsamen Präfixen Berechtigungen zu erteilen. Rekursive Richtlinien gewähren allen Objekten mit einem gemeinsamen Präfix Berechtigungen. Platzhalterausdrücke wählen mehrere Präfixe aus. Zusammen gewähren sie allen Objekten mit mehreren gemeinsamen Präfixen, wie in den folgenden Beispielen gezeigt.

Example Verwenden einer rekursiven Richtlinie

Angenommen, Sie benötigen Berechtigungen, um alle Parquet-Dateien in einem S3-Bucket aufzulisten, der wie folgt organisiert ist.

```
s3://sales-reports/americas/
    +- year=2000
    |      +- data-q1.parquet
    |      +- data-q2.parquet
    +- year=2019
    |      +- data-q1.json
    |      +- data-q2.json
    |      +- data-q3.json
    |      +- data-q4.json
    |
    +- year=2020
    |      +- data-q1.parquet
    |      +- data-q2.parquet
    |      +- data-q3.parquet
    |      +- data-q4.parquet
    |      +- annual-summary.parquet
```

```
    +- year=2021
```

Betrachten Sie zunächst die Parquet-Dateien mit dem Präfix `s3://sales-reports/americas/year=2000`. Sie können allen auf zwei Arten GetObject Berechtigungen gewähren:

Verwenden von nichtrekursiven Richtlinien: Eine Option besteht darin, zwei separate nichtrekursive Richtlinien zu verwenden, eine für das Verzeichnis und die andere für die Dateien.

Die erste Richtlinie erteilt die Erlaubnis für das Präfix `s3://sales-reports/americas/year=2020` (es gibt keinen Trailing-/).

```
- S3 resource = "sales-reports/americas/year=2000"
- permission = "GetObject"
- user = "analyst"
```

Die zweite Richtlinie verwendet einen Platzhalterausdruck, um allen Dateien mit Präfix Berechtigungen zu erteilen `sales-reports/americas/year=2020/` (beachten Sie das Trailing-/).

```
- S3 resource = "sales-reports/americas/year=2020/*"
- permission = "GetObject"
- user = "analyst"
```

Verwendung einer rekursiven Richtlinie: Eine bequemere Alternative besteht darin, eine einzige rekursive Richtlinie zu verwenden und dem Präfix rekursive Berechtigungen zu erteilen.

```
 - S3 resource = "sales-reports/americas/year=2020"
 - permission = "GetObject"
 - user = "analyst"
 - is recursive = "True"
```

Bisher waren nur die Parquet-Dateien mit dem Präfix `s3://sales-reports/americas/year=2000` enthalten. Sie können jetzt auch die Parquet-Dateien mit einem anderen Präfix, `s3://sales-reports/americas/year=2020`, in dieselben rekursive Richtlinie aufnehmen, indem Sie einen Platzhalterausdruck wie folgt einfügen.

```
 - S3 resource = "sales-reports/americas/year=20?0"
 - permission = "GetObject"
 - user = "analyst"
 - is recursive = "True"
```

Richtlinien für PutObject und DeleteObject Berechtigungen

Das Schreiben von Richtlinien `PutObject` und `DeleteObject` Berechtigungen für Dateien auf EMRFS erfordert besondere Sorgfalt, da sie im Gegensatz zu GetObject Berechtigungen zusätzliche rekursive Berechtigungen erfordern, die dem Präfix gewährt werden.

Example Richtlinien für PutObject und Berechtigungen DeleteObject

Zum Löschen der Datei ist beispielsweise nicht nur eine DeleteObject Berechtigung für die eigentliche Datei `annual-summary.parquet` erforderlich.

```
- S3 resource = "sales-reports/americas/year=2020/annual-summary.parquet"
- permission = "DeleteObject"
- user = "analyst"
```

Außerdem ist eine Richtlinie erforderlich, die rekursive Rechte `GetObject` und `PutObject` Berechtigungen für das zugehörige Präfix gewährt.

In ähnlicher Weise erfordert das Ändern der Datei `annual-summary.parquet` nicht nur eine `PutObject`-Berechtigung für die eigentliche Datei.

```
- S3 resource = "sales-reports/americas/year=2020/annual-summary.parquet"
- permission = "PutObject"
- user = "analyst"
```

Außerdem ist eine Richtlinie erforderlich, die eine rekursive `GetObject`-Erlaubnis für ihr Präfix erteilt.

```
- S3 resource = "sales-reports/americas/year=2020"
- permission = "GetObject"
- user = "analyst"
- is recursive = "True"
```

Platzhalter in Richtlinien

Es gibt zwei Bereiche, in denen Platzhalter angegeben werden können. Bei der Angabe einer S3-Ressource können „*“ und „?“ verwendet werden. Das „*“ ermöglicht einen Abgleich mit einem S3-Pfad und entspricht allem, was hinter dem Präfix steht. Zum Beispiel die folgende Richtlinie.

```
S3 resource = "sales-reports/americas/*"
```

Dies entspricht den folgenden S3-Pfaden.

```
sales-reports/americas/year=2020/
sales-reports/americas/year=2019/
sales-reports/americas/year=2019/month=12/day=1/afile.parquet
sales-reports/americas/year=2018/month=6/day=1/afile.parquet
sales-reports/americas/year=2017/afile.parquet
```

Das Platzhalterzeichen „?“ entspricht nur einem einzelnen Zeichen. Beispielsweise für die Richtlinie.

```
S3 resource = "sales-reports/americas/year=201?/"
```

Dies entspricht den folgenden S3-Pfaden.

```
sales-reports/americas/year=2019/
sales-reports/americas/year=2018/
sales-reports/americas/year=2017/
```

Platzhalter bei Benutzern

Bei der Zuweisung von Benutzern, die Benutzern Zugriff gewähren sollen, gibt es zwei integrierte Platzhalter. Der erste ist der Platzhalter „{USER}“, der allen Benutzern Zugriff gewährt. Der zweite Platzhalter ist „{OWNER}“, der dem Besitzer eines bestimmten Objekts oder direkt Zugriff gewährt. Der Platzhalter „{USER}“ wird derzeit jedoch nicht unterstützt.

Einschränkungen

Im Folgenden sind die aktuellen Einschränkungen des EMRFS S3-Plug-ins aufgeführt:

- Apache-Ranger-Richtlinien können maximal drei Richtlinien haben.
- Der Zugriff auf S3 muss über Hadoop-bezogene Anwendungen erfolgen EMRFS und kann mit diesen verwendet werden. Folgendes wird nicht unterstützt:

  – Boto3-Bibliotheken

  - und AWS SDK AWK CLI

  – S3A-Open-Source-Konnektor
- Die Ablehnungs-Richtlinien von Apache Ranger werden nicht unterstützt.
- Operationen auf S3 mit Schlüsseln mit CSE - KMS Verschlüsselung werden derzeit nicht unterstützt.

- Die Freigabe über Regionsgrenzen hinweg wird nicht unterstützt.
- Das Sicherheitszone-Feature von Apache Ranger wird nicht unterstützt. Einschränkungen der Zugriffskontrolle, die mit der Sicherheitszone-Funktion definiert wurden, gelten nicht für Ihre EMR Amazon-Cluster.
- Der Hadoop-Benutzer generiert keine Audit-Ereignisse, da Hadoop immer auf das EC2 Instance-Profil zugreift.
- Es wird empfohlen, Amazon EMR Consistency View zu deaktivieren. S3 ist stark konsistent und wird daher nicht mehr benötigt. Weitere Informationen finden Sie unter Starke Konsistenz von Amazon-S3.
- Das EMRFS S3-Plugin führt zahlreiche STS Aufrufe durch. Es wird empfohlen, Belastungstests auf einem Entwicklungskonto durchzuführen und das STS Anrufvolumen zu überwachen. Es wird außerdem empfohlen, eine STS Anfrage zur Erhöhung der AssumeRole Servicelimits zu stellen.
- Der Ranger Admin-Server unterstützt die automatische Vervollständigung nicht.

Trino-Plugin

Trino (früher PrestoSQL) ist eine SQL Abfrage-Engine, mit der Sie Abfragen für Datenquellen wie ObjektspeicherHDFS, relationale Datenbanken und Nox-Datenbanken ausführen können. SQL Es macht die Migration von Daten an einen zentralen Ort überflüssig und ermöglicht es Ihnen, die Daten von jedem Ort aus abzufragen. Amazon EMR stellt ein Apache Ranger-Plugin zur Verfügung, um detaillierte Zugriffskontrollen für Trino bereitzustellen. Das Plugin ist mit Admin-Server-Version von Open-Source-Apache-Ranger 2.0 und höher kompatibel.

Themen

Unterstützte Features

Das Apache Ranger-Plugin für Trino auf Amazon EMR unterstützt alle Funktionen der Trino-Abfrage-Engine, die durch eine detaillierte Zugriffskontrolle geschützt ist. Dazu gehören Zugriffskontrollen auf Datenbank-, Tabellen- und Spaltenebene sowie Zeilenfilterung und Datenmaskierung. Die Richtlinien

von Apache Ranger können Richtlinien für die Gewährung und die Ablehnung von Benutzern und Gruppen beinhalten. Prüfereignisse werden auch in Protokolle aufgenommen. CloudWatch

Installation der Servicekonfiguration

Die Installation der Trino-Servicedefinition erfordert die Einrichtung des Ranger-Admin-Servers. Informationen zum Einrichten des Ranger-Admin-Servers finden Sie unter Richten Sie den Ranger-Admin-Server ein.

Zum Installieren der Trino-Servicedefinition führen Sie die folgenden Schritte aus.

1. SSHin den Apache Ranger Admin-Server.

```
ssh ec2-user@ip-xxx-xxx-xxx-xxx.ec2.internal
```

2. Deinstallieren Sie das Presto-Server-Plugin, falls es existiert. Führen Sie den folgenden Befehl aus. Wenn dieser Fehler mit der Fehlermeldung „Service nicht gefunden“ angezeigt wird, bedeutet dies, dass das Presto-Server-Plugin nicht auf Ihrem Server installiert wurde. Fahren Sie mit dem nächsten Schritt fort.

```
curl -f -u *<admin users login>*:*_<_**_password_ **_for_** _ranger admin
 user_**_>_* -X DELETE -k 'https://*<RANGER SERVER ADDRESS>*:6182/service/public/
v2/api/servicedef/name/presto'
```

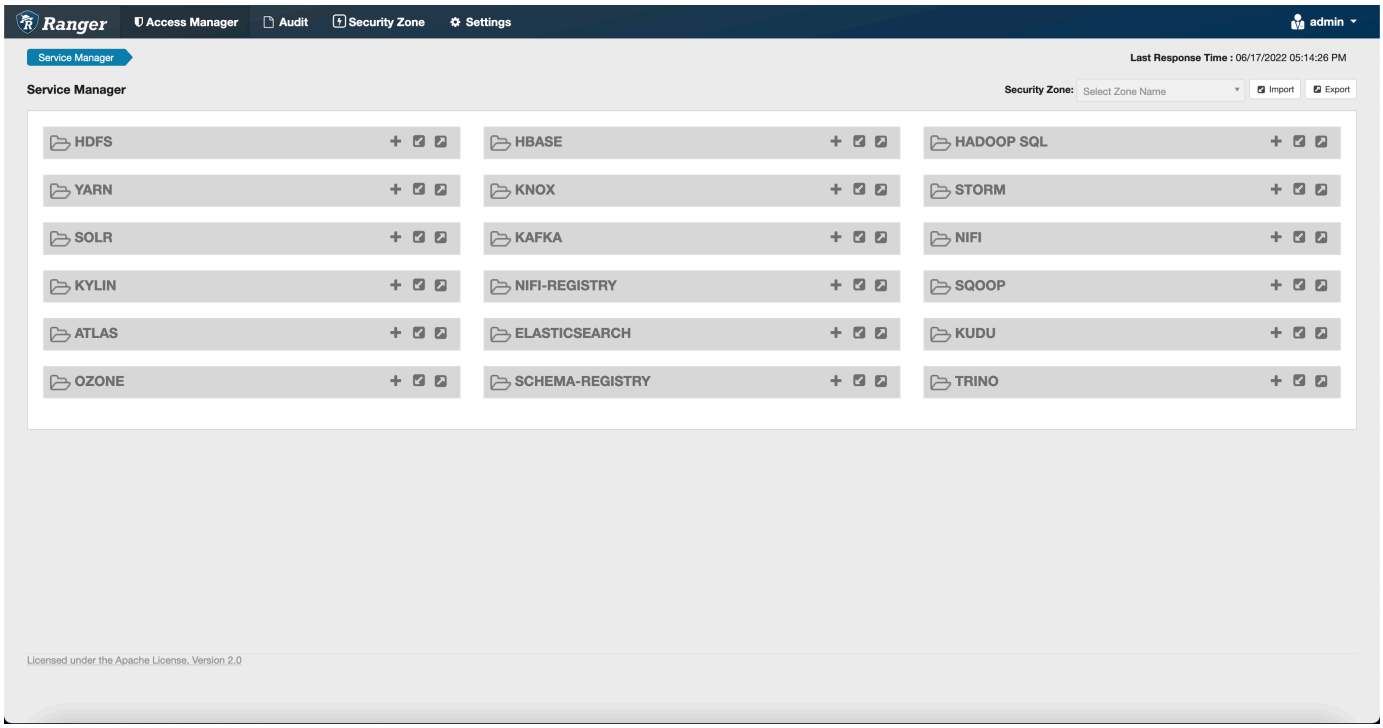
3. Laden Sie die Servicedefinition und das Apache-Ranger-Admin-Server-Plugin herunter. Laden Sie die Servicedefinition in einem temporären Verzeichnis herunter. Diese Servicedefinition wird von Ranger-2.x-Versionen unterstützt.

```
wget https://s3.amazonaws.com/elasticmapreduce/ranger/service-definitions/
version-2.0/ranger-servicedef-amazon-emr-trino.json
```

4. Registrieren Sie die Apache Trino-Servicedefinition für AmazonEMR.

```
curl -u *<admin users login>*:*_<_**_password_ **_for_** _ranger admin user_**_>_*
 -X POST -d @ranger-servicedef-amazon-emr-trino.json \
-H "Accept: application/json" \
-H "Content-Type: application/json" \
-k 'https://*<RANGER SERVER ADDRESS>*:6182/service/public/v2/api/servicedef'
```

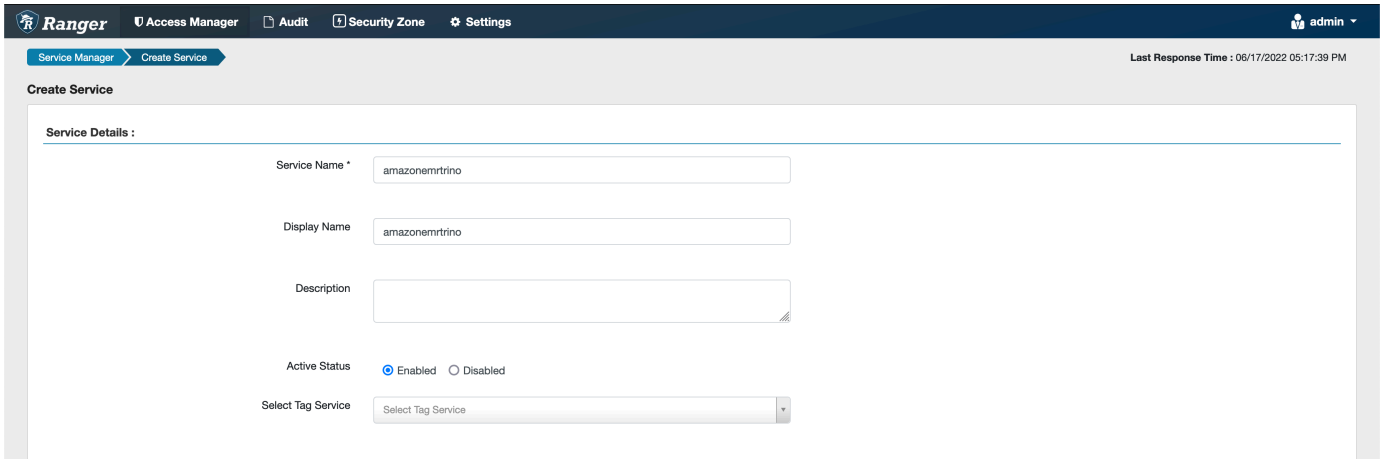
Wenn dieser Befehl erfolgreich ausgeführt wird, wird in Ihrer Ranger-Admin-Benutzeroberfläche ein neuer Service mit dem Namen `TRINO` angezeigt, wie in der folgenden Abbildung dargestellt.



5. Erstellen Sie eine Instance der `TRINO`-Anwendung und geben Sie die folgenden Informationen ein.

   Servicename: Der Servicename, den Sie verwenden werden. Der vorgeschlagene Wert ist `amazonemrtrino`. Notieren Sie sich diesen Servicenamen, da er bei der Erstellung einer EMR Amazon-Sicherheitskonfiguration benötigt wird.

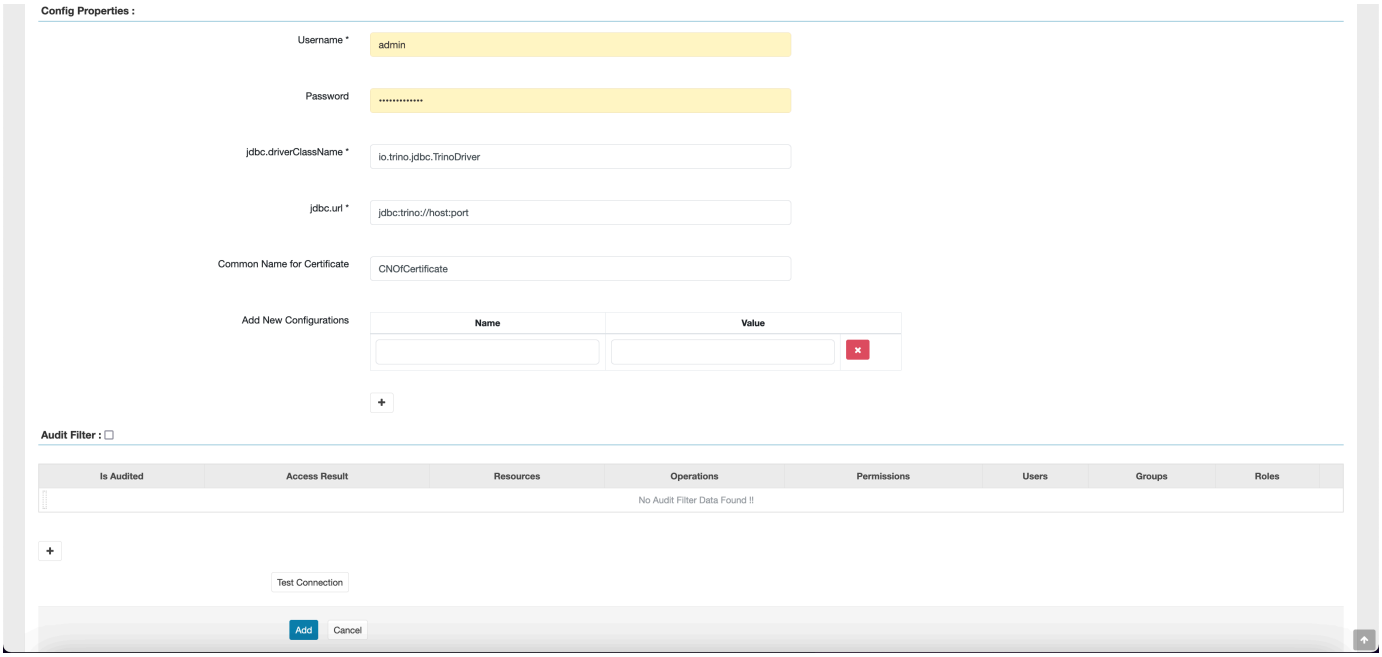
   Anzeigename: Der Name, der für diese Instance angezeigt werden soll. Der vorgeschlagene Wert ist `amazonemrtrino`.



jdbc.driver. ClassName: Der Klassenname der JDBC Klasse für Trino-Konnektivität. Sie können den Standardwert verwenden.

jdbc.url: Die JDBC Verbindungszeichenfolge, die verwendet werden soll, wenn eine Verbindung zum Trino-Koordinator hergestellt wird.

Allgemeiner Name für das Zertifikat: Das CN-Feld innerhalb des Zertifikats, das verwendet wird, um von einem Client-Plugin aus eine Verbindung zum Admin-Server herzustellen. Dieser Wert muss mit dem CN-Feld in Ihrem TLS Zertifikat übereinstimmen, das für das Plugin erstellt wurde.



Beachten Sie, dass das TLS Zertifikat für dieses Plugin im Trust Store auf dem Ranger Admin-Server registriert sein sollte. Weitere Informationen finden Sie unter TLSZertifikate.

Erstellen von Trino-Richtlinien

Wenn Sie eine neue Richtlinie erstellen, füllen Sie die folgenden Felder aus.

Richtlinienname: Der Name dieser Richtlinie.

Richtlinienbezeichnung: Eine Bezeichnung, die Sie dieser Richtlinie hinzufügen können.

Tabelle: Die Tabellen, für die diese Richtlinie gilt. Der Platzhalter „*“ steht für alle Tabellen.

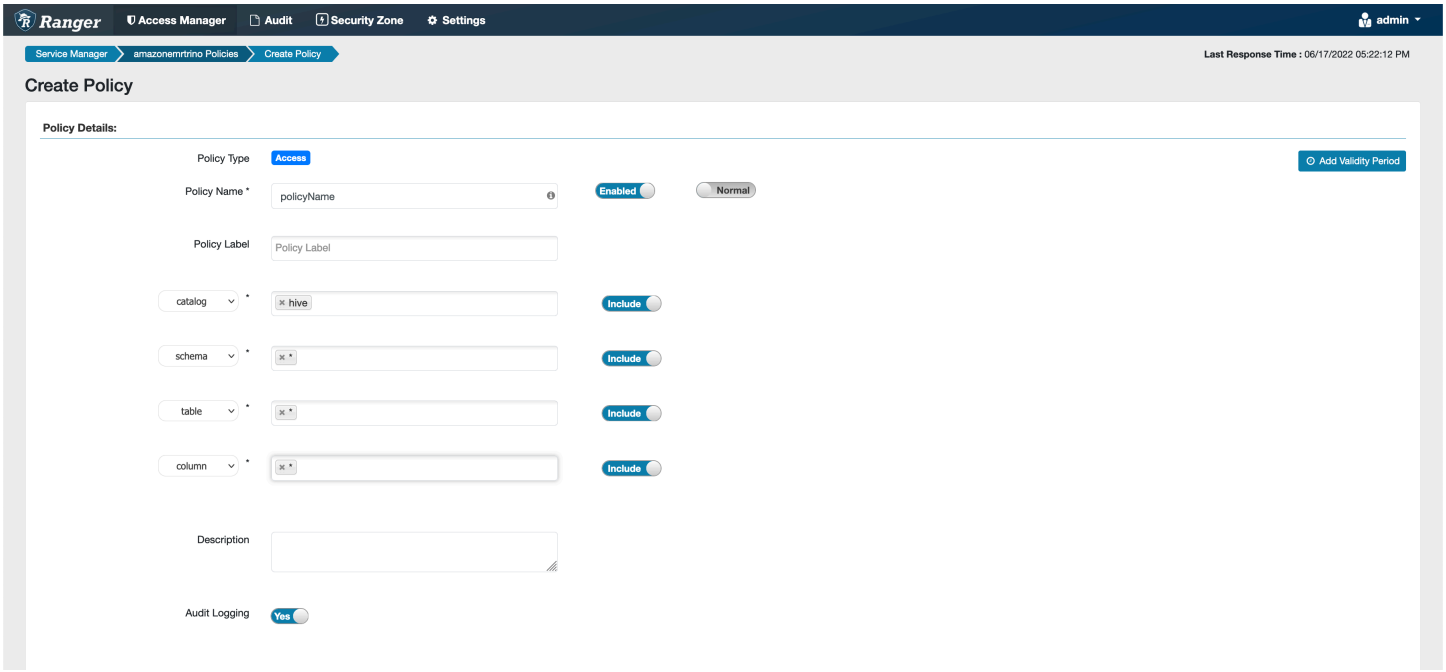
Schema: Die Schemas, für die diese Richtlinie gilt. Der Platzhalter „*“ steht für alle Schemas.

Tabelle: Die Tabellen, für die diese Richtlinie gilt. Der Platzhalter „*“ steht für alle Tabellen.

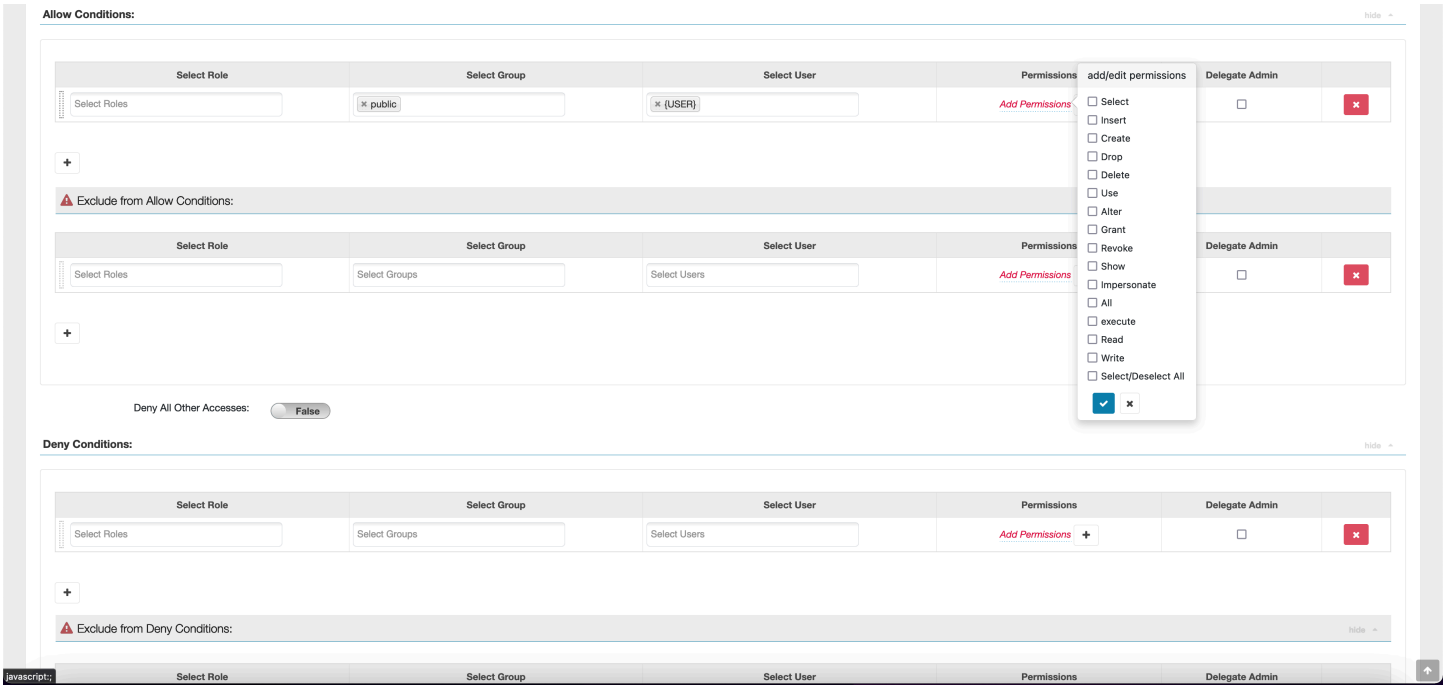
Spalte: Die Spalten, für die diese Richtlinie gilt. Der Platzhalter „*“ steht für alle Spalten.

Beschreibung: Eine Beschreibung dieser Richtlinie.

Andere Arten von Richtlinien gibt es für den Trino-Benutzer (für den Zugriff auf Benutzer, der sich als Benutzer ausgibt), die Trino-System-/Sitzungseigenschaft (zur Änderung der System- oder Sitzungseigenschaften der Engine), für Funktionen/Prozeduren (für das Zulassen von Funktions- oder Prozeduraufrufen) und die URL(zur Gewährung von Lese-/Schreibzugriff auf die Engine an Datenspeicherorten).



Um die Benutzer und Gruppen anzugeben, geben Sie die Benutzer und Gruppen unten ein, um Berechtigungen zu erteilen. Sie können auch Ausnahmen für Zulassungsbedingungen und Verweigerungsbedingungen angeben.



Nachdem Sie die Bedingungen für das Zulassen und Verweigern angegeben haben, klicken Sie auf Speichern.

Überlegungen

Bei der Erstellung von Trino-Richtlinien in Apache Ranger sind einige Nutzungsaspekte zu beachten.

Hive-Metadatenserver

Auf den Hive-Metadatenserver können nur vertrauenswürdige Engines zugreifen, insbesondere die Trino-Engine, um sich vor unbefugtem Zugriff zu schützen. Auf den Hive-Metadatenserver greifen auch alle Knoten im Cluster zu. Der erforderliche Port 9 083 ermöglicht allen Knoten den Zugriff auf den Hauptknoten.

Authentifizierung

Standardmäßig ist Trino so konfiguriert, dass es sich mithilfe von Kerberos authentifiziert, wie in der Amazon-Sicherheitskonfiguration konfiguriert. EMR

Verschlüsselung während der Übertragung erforderlich

Für das Trino-Plugin müssen Sie die Verschlüsselung während der Übertragung in der EMR Amazon-Sicherheitskonfiguration aktiviert haben. Informationen zum Aktivieren der Verschlüsselung finden Sie unter Verschlüsselung während der Übertragung.

Einschränkungen

Im Folgenden sind die aktuellen Einschränkungen des Trino-Plugins aufgeführt:

- Der Ranger-Admin-Server unterstützt die automatische Vervollständigung nicht.

Fehlerbehebung für Apache Ranger

Im Folgenden finden Sie einige häufig diagnostizierte Probleme im Zusammenhang mit der Verwendung von Apache Ranger.

Empfehlungen

- Testen Sie mit einem einzigen Hauptknotencluster: Master-Cluster mit einem Knoten können schneller bereitgestellt werden als ein Cluster mit mehreren Knoten, wodurch die Zeit für jede Test-Iteration verkürzt werden kann.
- Stellen Sie den Entwicklungsmodus auf dem Cluster ein. Wenn Sie Ihren EMR Cluster starten, setzen Sie den `--additional-info"` Parameter auf:

  ```
  '{"clusterType":"development"}'
  ```

  Dieser Parameter kann nur über AWS CLI oder festgelegt werden AWS SDK und ist nicht über die EMR Amazon-Konsole verfügbar. Wenn dieses Flag gesetzt ist und der Master nicht bereitstellen kann, hält der EMR Amazon-Service den Cluster für einige Zeit am Leben, bevor er ihn außer Betrieb nimmt. Diese Zeit ist sehr nützlich, um verschiedene Protokolldateien zu überprüfen, bevor der Cluster beendet wird.

EMRDer Cluster konnte nicht bereitgestellt werden

Es gibt mehrere Gründe, warum ein EMR Amazon-Cluster möglicherweise nicht gestartet werden kann. Im Folgenden finden Sie einige Möglichkeiten, das Problem zu diagnostizieren.

Überprüfen Sie die EMR Bereitstellungsprotokolle

Amazon EMR verwendet Puppet, um Anwendungen auf einem Cluster zu installieren und zu konfigurieren. Anhand der Protokolle erhalten Sie Informationen darüber, ob während der Bereitstellungsphase eines Clusters Fehler aufgetreten sind. Auf die Protokolle kann auf dem Cluster oder in S3 zugegriffen werden, wenn die Protokolle so konfiguriert sind, dass sie an S3 übertragen werden.

Die Protokolle werden auf der Festplatte `/var/log/provision-node/apps-phase/0/{UUID}/puppet.log` und `s3://<LOG LOCATION>/<CLUSTER ID>/node/<EC2 INSTANCE ID>/provision-node/apps-phase/0/{UUID}/puppet.log.gz.` gespeichert.

Allgemeine Fehlermeldungen

Fehlermeldung	Ursache
Puppet (err): Der Systemd-Start für ist fehlgeschlagen! emr-record-server journalctl-Protokoll für: emr-record-server	EMRDer Record Server konnte nicht gestartet werden. Weitere Informationen finden Sie weiter unten in den EMR Serverprotokollen.
Puppet (err): Der Systemd-Start für ist fehlgeschlagen! emr-record-server journalctl-Protokoll für emrsecretagent:	EMRSecret Agent konnte nicht gestartet werden. Weitere Informationen finden Sie weiter unten unter Secret-Agent-Protokolle überprüfen.
/Stage [main] /Ranger_Plugins: :Ranger_Hive_Plugin/Ranger_Plugins: :Prepare_Two_Way_TLS [2-Wege TLS im Hive-Plugin konfigurieren] /Exec [Keystore und Truststore für das Ranger Hive-Plugin erstellen] /returns (Hinweis): 140408606197664:error:0906d06C: Routinen: _read_bio:no start line:pem_lib.c:707 :Ich erwarte: PEM PEM ANY PRIVATE KEY	Das private TLS Zertifikat in Secret Manager für das Apache Ranger-Plug-in-Zertifikat hat nicht das richtige Format oder ist kein privates Zertifikat. Informationen zu TLSZertifikate den Zertifikatsformaten finden Sie unter.
/Stage [main] /Ranger_Plugins: :Ranger_S3_Plugin/Ranger_Plugins: :Prepare_Two_Way_TLS [2-Wege TLS im Ranger s3-Plugin konfigurieren] /Exec [Keystore und Truststore für Ranger amazon-emr-s 3-Plugin erstellen] /returns (Hinweis): Beim Aufrufen der GetSecretValue Operation ist ein Fehler aufgetreten (AccessDeniedException): User: arn:aws:sts: ::assumed-role/ XXXXXXXXX	Die EC2 Instanzprofilrolle hat nicht die richtigen Berechtigungen, um die TLS Zertifikate von Secrets Agent abzurufen.

Fehlermeldung	Ursache
XX _ _ EMR /i- ist nicht berechtigt,: secretsmanager: auf der Ressource: arn:aws:secretsmanager:us-east-1 ::secret: - auszuführen EC2DefaultRole XXXXXXXXXXXX GetSecretValue XXXXXXXXXX AdminServer XXXXX	

Überprüfen Sie die SecretAgent Protokolle

Secret Agent-Protokolle befinden sich `/emr/secretagent/log/` auf einem EMR Knoten oder im `s3://<LOG LOCATION>/<CLUSTER ID>/node/<EC2 INSTANCE ID>/daemons/secretagent/` Verzeichnis in S3.

Allgemeine Fehlermeldungen

Fehlermeldung	Ursache
Ausnahme im Thread „main“ com.amazonaws.services.securitytoken.model. AWSSecurityTokenServiceException: Der Benutzer: arn:aws:sts: :assumed-role/ XXXXXXXXXXXX _ EMR _ EC2 /i- DefaultRole ist XXXXXXXXX XXXXXX nicht berechtigt, Folgendes auszuführen: sts: on AssumeRole resource: arn:aws:iam: ::role/* (Service:; Statuscode: 403; Fehlercode:; Anforderungs-ID: - - - -; Proxy: null) XXXXXXXXXXXX RangerPluginDataAccessRole AWSSecurityTokenService AccessDenied XXXXXXXX XXXX XXXX XXXX XXXXXXXXXXXX	Die obige Ausnahme bedeutet, dass die Instanzprofilrolle nicht berechtigt ist, die Rolle anzunehmen EMREC2. RangerPluginDataAccessRole Siehe IAMRollen für die native Integration mit Apache Ranger.
ERRORqtp54617902-149: Eine Web-App-Ausnahme ist aufgetreten	Diese Fehler können ignoriert werden.

Fehlermeldung	Ursache
javax.ws.rs. NotAllowedException: 405 Methode nicht erlaubt HTTP	

Überprüfen Sie die Option Serverprotokolle aufzeichnen (für SparkSQL)

EMRServerprotokolle sind auf einem EMR Knoten unter /var/log/emr-record-server/oder im Verzeichnis s3: //< LOG LOCATION >/< ID>/< ID>/node/< CLUSTER ID>/daemons///in S3 verfügbar. EC2 INSTANCE emr-record-server

Allgemeine Fehlermeldungen

Fehlermeldung	Ursache
InstanceMetadataServiceResourceFetcherServerprotokolle sind unter /var/log//auf einem Knoten verfügbar, oder sie befinden sich im Verzeichnis s3: //< >/< ID>/node/< ID>/Daemons///in S3. ----sep----:105 - [] Token com.amazonaws konnte nicht abgerufen werden. SdkClientException: Es konnte keine Verbindung zum Service-Endpunkt hergestellt werden	Der EMR SecretAgent konnte nicht aufgerufen werden oder hat ein Problem. Untersuchen Sie die SecretAgent Protokolle auf Fehler und das Puppet-Skript, um festzustellen, ob Bereitstellungsfehler aufgetreten sind.

Abfragen schlagen unerwartet fehl

Überprüfen Sie die Protokolle des Apache Ranger-Plug-ins (Apache Hive EMRRecordServer, usw. EMR SecretAgent, Protokolle)

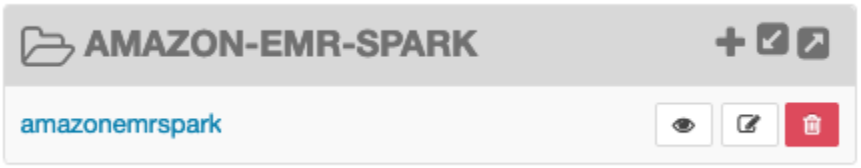
Dieser Abschnitt gilt für alle Anwendungen, die in das Ranger-Plugin integriert sind, wie Apache Hive, EMR Record Server und. EMR SecretAgent

Allgemeine Fehlermeldungen

Fehlermeldung	Ursache

Fehlermeldung	Ursache
ERROR PolicyRefresherDieser Abschnitt gilt für serviceName alle Anwendungen, die PolicyRefresher in das Ranger-Plugin integriert sind, wie Apache Hive, Record Server und. ----sep----:272 - [] (=policy-repository): failed to find service. Löscht den lokalen Richtliniencache (-1)	Diese Fehlermeldungen bedeuten, dass der Dienstname, den Sie in der EMR Sicherhei tskonfiguration angegeben haben, nicht mit einem Dienstrichtlinien-Repository auf dem Ranger Admin-Server übereinstimmt.

Wenn auf dem Ranger Admin-Server Ihr SPARK Dienst AMAZON EMR - - wie folgt aussieht, sollten Sie ihn **`amazonemrspark`** als Dienstnamen eingeben.



Arbeiten mit AWS Glue-Datenkatalogansichten (Vorschau)

> **Note**
>
> AWS Die Ansichten des Glue-Datenkatalogs in Amazon EMR befinden sich in der Vorschauversion und können sich ändern. Das Feature wird als Vorschau-Service gemäß der Definition in den AWS -Servicebedingungen bereitgestellt.

Sie können einzelne gemeinsame Ansichten im AWS Glue-Datenkatalog erstellen und verwalten. Einzelne allgemeine Ansichten sind nützlich, da sie mehrere SQL Abfrage-Engines unterstützen, sodass Sie auf dieselbe Ansicht in verschiedenen Ansichten zugreifen können AWS -Services, z. B. in Amazon EMR Amazon Athena und Amazon Redshift.

Indem Sie eine Ansicht im Datenkatalog erstellen, können Sie Ressourcenzuweisungen und tagbasierte Zugriffskontrollen verwenden, AWS Lake Formation um Zugriff auf eine Datenkatalogansicht zu gewähren. Mit dieser Methode der Zugriffskontrolle müssen Sie keinen zusätzlichen Zugriff auf die Tabellen konfigurieren, auf die Sie beim Erstellen der Ansicht verwiesen haben. Diese Methode zur Erteilung von Berechtigungen wird Definer-Semantik genannt, und diese

Ansichten werden Defineransichten genannt. Weitere Informationen zur Zugriffskontrolle in Lake Formation finden Sie unter Erteilen und Widerrufen von Berechtigungen für Datenkatalogressourcen. im AWS Lake Formation Entwicklerhandbuch.

Datenkatalogansichten sind für die folgenden Anwendungsfälle nützlich:

- Granulare Zugriffskontrolle — Erstellen Sie eine Ansicht, die den Datenzugriff auf der Grundlage der vom Benutzer benötigten Berechtigungen einschränkt. Sie können beispielsweise Ansichten im Datenkatalog verwenden, um zu verhindern, dass Mitarbeiter, die nicht in der Personalabteilung arbeiten, personenbezogene Daten sehen ()PII.
- Vollständige Definition der Ansicht — Indem Sie bestimmte Filter auf Ihre Ansicht im Datenkatalog anwenden, stellen Sie sicher, dass die Datensätze in einer Ansicht im Datenkatalog immer vollständig sind.
- Verbesserte Sicherheit — Die zur Erstellung der Ansicht verwendete Abfragedefinition muss vollständig sein. Dieser Vorteil bedeutet, dass Ansichten im Datenkatalog weniger anfällig für SQL Befehle von böswilligen Spielern sind.
- Einfaches Teilen von Daten — teilen Sie Daten mit anderen, AWS-Konten ohne Daten zu verschieben. Weitere Informationen finden Sie unter Kontoübergreifender Datenaustausch in Lake Formation.

Erstellen einer Data-Catalog-Ansicht

> **Important**
>
> In dieser Vorschauversion validiert Amazon den Spark-CodeSQL, den Sie bei der Erstellung der Ansicht verwenden, EMR nicht. Um das Risiko zu verringern, empfehlen wir, die Anzahl der Benutzer zu beschränken, denen Sie Berechtigungen zum Erstellen von Ansichten gewähren.

Um eine Datenkatalogsicht zu erstellen, müssen Sie eine IAM Rolle verwenden, die über die vollen `SELECT` Berechtigungen mit `Grantable` Optionen für alle Tabellen verfügt, auf die Sie beim Erstellen der Ansicht verweisen möchten. Diese Rolle wird als Definiererrolle bezeichnet. Eine vollständige Liste der Berechtigungen und Voraussetzungen, die zum Erstellen einer Datenkatalogansicht erforderlich sind, finden Sie unter Arbeiten mit Ansichten im AWS Lake Formation Entwicklerhandbuch. Sie müssen die verwenden AWS CLI , um Ihre IAM Rolle zu konfigurieren. Weitere Informationen finden Sie unter Verwenden einer IAM Rolle in der AWS CLI.

Gehen Sie wie folgt vor, um eine Datenkatalogansicht zu erstellen.

> **Note**
>
> Um von Apache Spark auf Amazon auf eine Datenkatalogansicht zuzugreifenEMR, müssen Sie den Dialekt auf SPARK und auf einstellen. `DialectVersion` `3.4.1-amzn-2`

1. Laden Sie zuerst das Vorschaumodell herunter.

```
aws s3 cp s3://emr-data-access-control-us-east-1/beta/glue-views/model/
service-2.json
```

2. Konfigurieren Sie das AWS CLI , um das Vorschaumodell zu verwenden.

```
aws configure add-model --service-model file:///<path-to-preview-model>/
service-2.json --service-name glue-views
```

3. Erstellen Sie die Ansicht.

```
aws glue-views create-table --cli-input-json '{
  "DatabaseName": "<database>",
  "TableInput": {
    "Name": "<view>",
    "StorageDescriptor": {
      "Columns": [
        {
          "Name": "<col1>",
          "Type": "<data-type>"
        },
        ...
        {
          "Name": "<colN>",
          "Type": "<data-type>"
        }
      ]
    },
    "ViewDefinition": {
      "SubObjects": [
        "arn:aws:glue:<aws-region;>:<aws-account-id>:table/<database>/<referenced-
table1>",
        ...
```

```
        "arn:aws:glue:<aws-region>:<aws-account-id>:table/<database>/<referenced-
tableN>",
       ],
       "IsProtected": true,
      "Representations": [
        {
          "Dialect": "SPARK",
          "DialectVersion": "3.4.1-amzn-2",
          "ViewOriginalText": "<Spark-SQL>",
          "ViewExpandedText": "<Spark-SQL>"
        }
      ]
    }
  }
}'
```

Zugriff auf eine Datenkatalogansicht aktivieren

> **Important**
>
> Es wird empfohlen, den Zugriff auf Datenkatalogansichten nur für EMR Cluster in Testumgebungen und nicht für Produktionsumgebungen zu aktivieren.

Um von Apache Spark auf Amazon auf die Datenkatalogansicht zuzugreifenEMR, müssen Sie zuerst die Unterstützung für Lake Formation aktivieren und das folgende Skript verwenden, um die Unterstützung für Ansichten mit Spark auf Amazon zu aktivierenEMR. Weitere Informationen zur Aktivierung des Supports finden Sie unter Aktivieren von Lake Formation mit Amazon EMR und Verwenden benutzerdefinierter Bootstrap-Aktionen.

```
# Download the script and upload it to Amazon S3
wget https://emr-data-access-control-us-east-1.s3.amazonaws.com/beta/glue-views/ba/
enable-mdv.sh /Users/$USER/enable-mdv.sh
aws s3 cp /Users/$USER/enable-views.sh s3://<bucket>/<prefix>/enable-views.sh

# EMR Security Configuration
cat <<EOT > /Users/$USER/lakeformation-protection.json
{
```

```
    "AuthorizationConfiguration":{
        "IAMConfiguration":{
            "EnableApplicationScopedIAMRole":true
        },
        "LakeFormationConfiguration":{
            "AuthorizedSessionTagValue":"Amazon EMR"
        }
    },
     "EncryptionConfiguration": {
            "EnableInTransitEncryption": true,
            "InTransitEncryptionConfiguration": {
            "TLSCertificateConfiguration": {
                 "CertificateProviderType": "PEM",
                 "S3Object": "s3://<BUCKET>/<PREFIX>/certificates.zip"
            }
        }
    }
}
EOT

SECURITY_CONFIG="RuntimeRolesWithAWSLakeFormation"

aws emr create-security-configuration \
--name $SECURITY_CONFIG \
--security-configuration file:///Users/$USER/lakeformation-protection.json

# EMR Cluster version
RELEASE_LABEL="emr-6.15.0"
```

Verwenden Sie dann den folgenden AWS CLI Befehl, der die Bootstrap-Aktion verwendet, um einen EMR Cluster zu erstellen, der Datenkatalogansichten unterstützt.

```
aws emr create-cluster \
...
--release-label $RELEASE_LABEL \
--security-configuration $SECURITY_CONFIG \
--bootstrap-actions \
 Name='Enable Views',Path="s3://<bucket>/<prefix>/enable-views.sh"
```

Abfrage einer Data-Catalog-Ansicht

> **Important**
>
> In dieser Vorschauversion empfehlen wir, dass Sie nur auf Ansichten aus vertrauenswürdigen Quellen zugreifen. In der Vorschauversion EMR bietet Amazon eine begrenzte Anzahl von Validierungen, die Ihren EMR Cluster schützen.

Nachdem Sie eine Datenkatalogansicht erstellt haben, können Sie nun eine IAM Rolle verwenden, um die Ansicht abzufragen. Die IAM Rolle muss über die `SELECT` Berechtigung für die Datenkatalogansicht verfügen. Sie müssen keinen Zugriff auf die zugrunde liegenden Tabellen gewähren, auf die in der Ansicht verwiesen wird. Sie müssen diese IAM Rolle als Runtime-Rolle verwenden. Sie können von einem EMR Cluster aus mithilfe einer Runtime-Rolle von Amazon EMR Steps, EMR Studio und SageMaker Studio aus auf die Ansicht zugreifen. Weitere Informationen zu Runtime-Rollen finden Sie unter EMRSchritte zu Runtime-Rollen für Amazon.

Sobald Sie alles eingerichtet haben, können Sie Ihre Ansicht abfragen. Nachdem Sie den EMR Cluster beispielsweise an Ihren Workspace in EMR Studio angehängt haben, können Sie die folgende Abfrage ausführen, um auf eine Ansicht zuzugreifen.

```
SELECT * from <database>.<glue-data-catalog-view> LIMIT 10
```

Einschränkungen

Beachten Sie bei der Verwendung von Datenkatalogansichten die folgenden Einschränkungen.

- Sie können Datenkatalogansichten nur mit Amazon EMR 6.15.0 erstellen.
- Sie können in der Ansichtsdefinition nur auf bis zu 10 Tabellen verweisen.
- Sie können nur `PROTECTED` Datenkatalogsichten erstellen. `UNPROTECTED`Ansichten werden nicht unterstützt.
- In Datenkatalogansichten können Sie nicht auf Tabellen AWS-Konto in anderen Tabellen verweisen.
- Benutzerdefinierte Funktionen (UDFs) werden nicht unterstützt.
- In Datenkatalogansichten können Sie nicht auf offene Tabellenformate wie Apache Hudi oder Apache Iceberg verweisen.
- Sie können in Datenkatalogansichten nicht auf andere Ansichten verweisen.

Steuerung des Netzwerkverkehrs mit Sicherheitsgruppen

Sicherheitsgruppen dienen als virtuelle Firewalls für EC2 Instances in Ihrem Cluster, um den eingehenden und ausgehenden Datenverkehr zu kontrollieren. Für jede Sicherheitsgruppe gibt es einen Satz von Regeln zur Kontrolle des eingehenden Datenverkehrs und einen Satz von Regeln zur Kontrolle des ausgehenden Datenverkehrs. Weitere Informationen finden Sie unter EC2Amazon-Sicherheitsgruppen für Linux-Instances im EC2Amazon-Benutzerhandbuch.

Bei Amazon verwenden Sie zwei Klassen von SicherheitsgruppenEMR: von Amazon EMR verwaltete Sicherheitsgruppen und zusätzliche Sicherheitsgruppen.

Mit jedem Cluster sind verwaltete Sicherheitsgruppen verknüpft. Sie können die von Amazon EMR erstellten standardmäßigen verwalteten Sicherheitsgruppen verwenden oder benutzerdefinierte verwaltete Sicherheitsgruppen angeben. In beiden Fällen fügt Amazon EMR automatisch Regeln zu verwalteten Sicherheitsgruppen hinzu, die ein Cluster für die Kommunikation zwischen Cluster-Instances und AWS Services benötigt.

Zusätzliche Sicherheitsgruppen sind optional. Sie können sie zusätzlich zu den verwalteten Sicherheitsgruppen angeben, um den Zugriff auf Cluster-Instances anzupassen. Zusätzliche Sicherheitsgruppen enthalten nur von Ihnen definierte Regeln. Amazon EMR ändert sie nicht.

Die Regeln, die Amazon in verwalteten Sicherheitsgruppen EMR erstellt, ermöglichen dem Cluster die Kommunikation zwischen internen Komponenten. Um Benutzern und Anwendungen den Zugriff auf einen Cluster von außerhalb des Clusters zu ermöglichen, können Sie Regeln in verwalteten Sicherheitsgruppen bearbeiten, zusätzliche Sicherheitsgruppen mit zusätzlichen Regeln erstellen oder beides ausführen.

> **Important**
>
> Das Bearbeiten von Regeln in verwalteten Sicherheitsgruppen kann unbeabsichtigte Folgen haben. Möglicherweise blockieren Sie versehentlich den Datenverkehr, der für die ordnungsgemäße Funktion der Cluster erforderlich ist, und verursachen Fehler, da die Knoten nicht erreichbar sind. Planen und testen Sie Sicherheitsgruppenkonfigurationen sorgfältig, bevor Sie diese implementieren.

Sie können Sicherheitsgruppen nur während der Erstellung eines Clusters angeben. Sie können keine Sicherheitsgruppen zu Clustern oder Cluster-Instances hinzufügen, während ein Cluster

ausgeführt wird. Sie können jedoch Regeln in vorhandenen Sicherheitsgruppen bearbeiten, hinzufügen und entfernen. Die Regeln treten in Kraft, sobald Sie sie speichern.

Sicherheitsgruppen sind standardmäßig einschränkend. Wenn keine Regel hinzugefügt wird, die Datenverkehr zulässt, wird der Datenverkehr zurückgewiesen. Wenn es mehr als eine Regel für denselben Datenverkehr und dieselbe Quelle gibt, wird die toleranteste Regel angewendet. Wenn Sie beispielsweise eine Regel haben, die SSH von der IP-Adresse 192.0.2.12/32 ausgeht, und eine weitere Regel, die den Zugriff auf den gesamten TCP Datenverkehr aus dem Bereich 192.0.2.0/24 zulässt, hat die Regel, die den gesamten Verkehr aus dem Bereich zulässt, der 192.0.2.12 umfasst, TCP Vorrang. In diesem Fall könnte der Client unter 192.0.2.12 mehr Zugriff erhalten als beabsichtigt.

> **Important**
>
> Seien Sie vorsichtig, wenn Sie Regeln für offene Ports für Sicherheitsgruppen bearbeiten. Stellen Sie sicher, dass Sie Regeln hinzufügen, die nur Datenverkehr von vertrauenswürdigen und authentifizierten Clients für die Protokolle und Ports zulassen, die zum Ausführen Ihrer Workloads erforderlich sind.

Sie können Amazon EMR Block Public Access in jeder Region, die Sie verwenden, konfigurieren, um die Cluster-Erstellung zu verhindern, wenn eine Regel öffentlichen Zugriff auf jedem Port erlaubt, den Sie nicht zu einer Ausnahmeliste hinzufügen. Für AWS Konten, die nach Juli 2019 erstellt wurden, ist Amazon EMR Block Public Access standardmäßig aktiviert. Für AWS Konten, die vor Juli 2019 einen Cluster erstellt haben, ist Amazon EMR Block Public Access standardmäßig deaktiviert. Weitere Informationen finden Sie unter Verwenden Sie Amazon, um EMR den öffentlichen Zugriff zu blockieren.

Themen

> **Note**
>
> Amazon EMR ist bestrebt, integrative Alternativen für potenziell anstößige oder nicht inklusive Branchenbegriffe wie „Master“ und „Slave“ zu verwenden. Wir haben auf eine neue Terminologie umgestellt, um ein umfassenderes Erlebnis zu bieten und Ihnen das Verständnis der Servicekomponenten zu erleichtern.
> Wir beschreiben „Knoten“ jetzt als Instances und EMR Amazon-Instance-Typen als Primär -, Kern - und Task-Instances. Während der Umstellung finden Sie möglicherweise immer noch ältere Verweise auf die veralteten Begriffe, z. B. solche, die sich auf Sicherheitsgruppen für Amazon EMR beziehen.

Arbeiten mit von Amazon EMR verwalteten Sicherheitsgruppen

> **Note**
>
> Amazon EMR ist bestrebt, integrative Alternativen für potenziell anstößige oder nicht inklusive Branchenbegriffe wie „Master“ und „Slave“ zu verwenden. Wir haben auf eine neue Terminologie umgestellt, um ein umfassenderes Erlebnis zu bieten und Ihnen das Verständnis der Servicekomponenten zu erleichtern.
> Wir beschreiben „Knoten“ jetzt als Instances und EMR Amazon-Instance-Typen als Primär -, Kern - und Task-Instances. Während der Umstellung finden Sie möglicherweise immer noch ältere Verweise auf die veralteten Begriffe, z. B. solche, die sich auf Sicherheitsgruppen für Amazon EMR beziehen.

Mit der Primär-Instance und den Core- und Aufgaben-Instances in einem Cluster sind verschiedene verwaltete Sicherheitsgruppen verknüpft. Sie benötigen eine zusätzliche verwaltete Sicherheitsgruppe für den Servicezugriff, wenn Sie einen Cluster in einem privaten Subnetz erstellen. Weitere Informationen zur Rolle von verwalteten Sicherheitsgruppen in Bezug auf Ihre Netzwerkkonfiguration finden Sie unter VPCAmazon-Optionen.

Wenn Sie verwaltete Sicherheitsgruppen für einen Cluster angeben, müssen Sie für alle verwalteten Sicherheitsgruppen denselben Typ von Sicherheitsgruppe (Standard oder benutzerdefiniert) verwenden. Sie können beispielsweise nicht eine benutzerdefinierte Sicherheitsgruppe für die Primär-Instance angeben und dann keine benutzerdefinierte Sicherheitsgruppe für die Core- und Aufgaben-Instances angeben.

Wenn Sie standardmäßige verwaltete Sicherheitsgruppen verwenden, müssen Sie diese beim Erstellen eines Clusters nicht angeben. Amazon verwendet EMR automatisch die Standardeinstellungen. Wenn die Standardeinstellungen in den Clustern noch nicht vorhanden sind, VPC EMR erstellt Amazon sie außerdem. Amazon erstellt sie EMR auch, wenn Sie sie explizit angeben und sie noch nicht existieren.

Sie können die Regeln in verwalteten Sicherheitsgruppen nach der Erstellung der Cluster bearbeiten. Wenn Sie einen neuen Cluster erstellen, EMR überprüft Amazon die Regeln in den von Ihnen angegebenen verwalteten Sicherheitsgruppen und erstellt dann alle fehlenden Regeln für eingehenden Datenverkehr, die der neue Cluster benötigt, zusätzlich zu den Regeln, die möglicherweise zuvor hinzugefügt wurden. Sofern nicht ausdrücklich anders angegeben, wird jede Regel für von Amazon EMR verwaltete Standardsicherheitsgruppen auch den von Ihnen angegebenen benutzerdefinierten, von EMR Amazon verwalteten Sicherheitsgruppen hinzugefügt.

Die standardmäßigen verwalteten Sicherheitsgruppen sind:

- ElasticMapReduce-primär

  Informationen zu den Regeln in dieser Sicherheitsgruppe finden Sie unter Von Amazon EMR verwaltete Sicherheitsgruppe für die primäre Instance (öffentliche Subnetze).
- ElasticMapReduce-Kern

  Informationen zu den Regeln in dieser Sicherheitsgruppe finden Sie unter Von Amazon EMR verwaltete Sicherheitsgruppe für Core- und Task-Instances (öffentliche Subnetze).
- ElasticMapReduce-Primär-Privat

  Informationen zu den Regeln in dieser Sicherheitsgruppe finden Sie unter Von Amazon EMR verwaltete Sicherheitsgruppe für die primäre Instance (private Subnetze).
- ElasticMapReduce-Kernprivat

  Informationen zu den Regeln in dieser Sicherheitsgruppe finden Sie unter Von Amazon EMR verwaltete Sicherheitsgruppe für Core- und Task-Instances (private Subnetze).
- ElasticMapReduce-ServiceAccess

  Informationen zu den Regeln in dieser Sicherheitsgruppe finden Sie unter Von Amazon EMR verwaltete Sicherheitsgruppe für den Servicezugriff (private Subnetze).

Von Amazon EMR verwaltete Sicherheitsgruppe für die primäre Instance (öffentliche Subnetze)

Die standardmäßige verwaltete Sicherheitsgruppe für die primäre Instance in öffentlichen Subnetzen hat den Gruppennamen -primary. ElasticMapReduce Sie hat die folgenden Regeln. Wenn Sie eine benutzerdefinierte verwaltete Sicherheitsgruppe angeben, EMR fügt Amazon Ihrer benutzerdefinierten Sicherheitsgruppe dieselben Regeln hinzu.

Typ	Protoc (Protok l)	Port-Bere ich	Quelle	Details
Regeln für eingehenden Datenverkehr				
Alles ICMP - IPv4	Alle	N/A	Die Gruppen-ID der verwalteten Sicherheitsgruppe für die Primär-In stance. Mit anderen Worten, dieselbe Sicherheitsgruppe, in der die Regel angezeigt wird.	Diese reflexiven Regeln ermöglichen eingehenden Datenverkehr aus allen mit der angegebenen Sicherheitsgruppe verknüpft en Instances. Wenn die Standardeinstellun g `ElasticMapReduce-primary` für mehrere Cluster verwendet wird, können die Kern- und Taskknoten dieser Cluster über einen ICMP TCP oder einen beliebigen UDP O-Port miteinander kommunizieren. Sie geben benutzerdefinierte verwaltete Sicherhei tsgruppen an, um den Cluster-übergreifenden Zugriff einzuschränken.
Alle TCP	TCP	Alle		
Alle UDP	UDP	Alle		
Alles ICMP - IPV4	Alle	N/A	Die Gruppen-ID der verwalteten Sicherheitsgruppe, die für Core- und Aufgabenknoten angegeben wurde.	Diese Regeln lassen den gesamten eingehenden ICMP Datenverkehr und den Datenverkehr über einen beliebigen TCP UDP Port von allen Core- und Task-Inst ances zu, die der angegebenen Sicherhei tsgruppe zugeordnet sind, auch wenn sich die Instances in unterschiedlichen Clustern befinden.
Alle TCP	TCP	Alle		
Alle UDP	UDP	Alle		

Typ	Protoc (Protok l)	Port-Bere ich	Quelle	Details
Benutzerd efiniert	TCP	8443	Verschiedene Amazon-IP-Adressbereiche	Diese Regeln ermöglichen dem Cluster-Manager die Kommunikation mit dem Primärknoten.

Um vertrauenswürdigen Quellen mit der Konsole SSH Zugriff auf die primäre Sicherheitsgruppe zu gewähren

Um Ihre Sicherheitsgruppen bearbeiten zu können, benötigen Sie die Berechtigung, Sicherheitsgruppen für die Gruppe zu verwaltenVPC, in der sich der Cluster befindet. Weitere Informationen finden Sie unter Ändern der Berechtigungen für einen Benutzer und unter der Beispielrichtlinie, die die Verwaltung von EC2 Sicherheitsgruppen ermöglicht, im IAMBenutzerhandbuch.

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie Clusters (Cluster) aus. Wählen Sie die ID des Clusters aus, den Sie ändern möchten.
3. Erweitern Sie im Bereich Netzwerk und Sicherheit die Dropdownliste EC2Sicherheitsgruppen (Firewall).
4. Wählen Sie unter Primärer Knoten Ihre Sicherheitsgruppe aus.
5. Wählen Sie Edit inbound rules (Regeln für eingehenden Datenverkehr bearbeiten) aus.
6. Suchen Sie mit den folgenden Einstellungen nach einer Regel für eingehenden Datenverkehr, die öffentlichen Zugriff ermöglicht. Falls sie existiert, wählen Sie Löschen, um sie zu entfernen.

   - Typ

     SSH
   - Port

     22
   - Quelle

Benutzerdefiniert 0.0.0.0/0

> **Warning**
>
> Vor Dezember 2020 gab es eine vorkonfigurierte Regel, die eingehenden Datenverkehr auf Port 22 aus allen Quellen zuließ. Diese Regel wurde erstellt, um die ersten SSH Verbindungen zum Primärknoten zu vereinfachen. Wir empfehlen Ihnen dringend, diese Eingangsregel zu entfernen und den Datenverkehr auf vertrauenswürdige Quellen zu beschränken.

7. Scrollen Sie zum Ende der Regelliste und wählen Sie Regel hinzufügen.
8. Wählen Sie als Typ die Option aus SSH.

   Bei der Auswahl SSH werden automatisch Protokoll und 22 als Portbereich eingegeben TCP.
9. Wählen Sie als Quelle Meine IP aus, um Ihre IP-Adresse automatisch als Quelladresse hinzuzufügen. Sie können auch einen Bereich benutzerdefinierter vertrauenswürdiger Client-IP-Adressen hinzufügen oder zusätzliche Regeln für andere Clients erstellen. In vielen Netzwerkumgebungen werden IP-Adressen dynamisch zugewiesen, sodass Sie in Zukunft möglicherweise Ihre IP-Adressen für vertrauenswürdige Clients aktualisieren müssen.
10. Wählen Sie Save (Speichern) aus.
11. Wählen Sie optional im Bereich Netzwerk und Sicherheit unter Kern- und Taskknoten die andere Sicherheitsgruppe aus und wiederholen Sie die obigen Schritte, um dem SSH Client Zugriff auf Core- und Taskknoten zu gewähren.

Von Amazon EMR verwaltete Sicherheitsgruppe für Core- und Task-Instances (öffentliche Subnetze)

Die standardmäßig verwaltete Sicherheitsgruppe für Core- und Task-Instances in öffentlichen Subnetzen hat den Gruppennamen -core. ElasticMapReduce Die standardmäßige verwaltete Sicherheitsgruppe hat die folgenden Regeln, und Amazon EMR fügt dieselben Regeln hinzu, wenn Sie eine benutzerdefinierte verwaltete Sicherheitsgruppe angeben.

Typ	Protoc (Protok l)	Port-Bere ich	Quelle	Details
Regeln für eingehenden Datenverkehr				
Alles ICMP - IPV4	Alle	N/A	Die Gruppen-ID der verwalteten Sicherheitsgruppe für Core- und Aufgaben-Instances. Mit anderen Worten, dieselbe Sicherheitsgruppe, in der die Regel angezeigt wird.	Diese reflexiven Regeln ermöglichen eingehenden Datenverkehr aus allen mit der angegebenen Sicherheitsgruppe verknüpften Instances. Wenn die Standardeinstellung `ElasticMapReduce-core` für mehrere Cluster verwendet wird, können die Core- und Task-Instances dieser Cluster über einen ICMP TCP oder einen beliebigen UDP O-Port miteinander kommunizieren. Sie geben benutzerdefinierte verwaltete Sicherheitsgruppen an, um den Cluster-übergreifenden Zugriff einzuschränken.
Alle TCP	TCP	Alle		
Alle UDP	UDP	Alle		
Alles ICMP - IPV4	Alle	N/A	Die Gruppen-ID der verwalteten Sicherheitsgruppe für die Primär-Instance.	Diese Regeln lassen den gesamten eingehenden ICMP Datenverkehr und den Datenverkehr über einen beliebigen TCP UDP Port von allen primären Instances zu, die der angegebenen Sicherheitsgruppe zugeordnet sind, auch wenn sich die Instances in unterschiedlichen Clustern befinden.
Alle TCP	TCP	Alle		
Alle UDP	UDP	Alle		

Von Amazon EMR verwaltete Sicherheitsgruppe für die primäre Instance (private Subnetze)

Die standardmäßig verwaltete Sicherheitsgruppe für die primäre Instanz in privaten Subnetzen hat den Gruppennamen -Primary-Private. ElasticMapReduce Die standardmäßige verwaltete Sicherheitsgruppe hat die folgenden Regeln, und Amazon EMR fügt dieselben Regeln hinzu, wenn Sie eine benutzerdefinierte verwaltete Sicherheitsgruppe angeben.

Typ	Protoc (Protok l)	Port-Bere ich	Quelle	Details
Regeln für eingehenden Datenverkehr				
Alles ICMP - IPv4	Alle	N/A	Die Gruppen-ID der verwalteten Sicherheitsgruppe für die Primär-In stance. Mit anderen Worten, dieselbe Sicherheitsgruppe, in der die Regel angezeigt wird.	Diese reflexiven Regeln ermöglichen eingehenden Datenverkehr aus allen mit der angegebenen Sicherheitsgruppe verknüpft en Instances, die aus dem privaten Subnetz erreichbar sind. Wenn die Standardeinstellun g `ElasticMapReduce-Primary-Pr ivate` für mehrere Cluster verwendet wird, können die Kern- und Taskknoten dieser Cluster über einen ICMP TCP oder einen beliebigen UDP O-Port miteinander kommunizieren. Sie geben benutzerdefinierte verwaltete Sicherheitsgruppen an, um den Cluster-übergreifenden Zugriff einzuschr änken.
Alle TCP	TCP	Alle		
Alle UDP	UDP	Alle		
Alles ICMP - IPV4	Alle	N/A	Die Gruppen-ID der verwalteten Sicherheitsgruppe für Core- und Aufgabenknoten.	Diese Regeln lassen den gesamten eingehenden ICMP Datenverkehr und den Datenverkehr über einen beliebigen TCP UDP Port von allen Core- und Task-Instances zu, die der angegebenen Sicherheitsgruppe zugeordnet sind und über das private Subnetz erreichbar sind, auch wenn sich die Instances in unterschiedlichen Clustern befinden.
Alle TCP	TCP	Alle		
Alle UDP	UDP	Alle		
HTTPS(843)	TCP	8443	Die Gruppen-ID der verwalteten Sicherheitsgruppe für den Servicezu griff in einem privaten Subnetz.	Diese Regel ermöglicht dem Cluster-Manager die Kommunikation mit dem Primärknoten.

Typ	Protoc (Protok l)	Port-Bere ich	Quelle	Details
Regeln für ausgehenden Datenverkehr				
Gesamter Datenverk ehr	Alle	Alle	0.0.0.0/0	Stellt ausgehenden Zugriff auf das Internet zu.
Benutzerd efiniert TCP	TCP	9443	Die Gruppen-ID der verwalteten Sicherheitsgruppe für den Servicezu griff in einem privaten Subnetz.	Wenn die obige Standardregel „Gesamter Datenverkehr“ für ausgehenden Datenverkehr entfernt wird, ist diese Regel eine Mindestan forderung für Amazon EMR 5.30.0 und höher. **Note** Amazon EMR fügt diese Regel nicht hinzu, wenn Sie eine benutzerd efinierte verwaltete Sicherheitsgruppe verwenden.
Benutzerd efiniert TCP	TCP	80 (http) oder 443 (https)	Die Gruppen-ID der verwalteten Sicherheitsgruppe für den Servicezu griff in einem privaten Subnetz.	Wenn die obige Standardregel „Gesamter Datenverkehr“ für ausgehenden Datenverkehr entfernt wird, ist diese Regel eine Mindestan forderung für Amazon EMR 5.30.0 und höher, um über https eine Verbindung zu Amazon S3 herzustellen. **Note** Amazon EMR fügt diese Regel nicht hinzu, wenn Sie eine benutzerd efinierte verwaltete Sicherheitsgruppe verwenden.

Von Amazon EMR verwaltete Sicherheitsgruppe für Core- und Task-Instances (private Subnetze)

Die standardmäßig verwaltete Sicherheitsgruppe für Core- und Task-Instances in privaten Subnetzen hat den Gruppennamen -Core-Private. ElasticMapReduce Die standardmäßige verwaltete Sicherheitsgruppe hat die folgenden Regeln, und Amazon EMR fügt dieselben Regeln hinzu, wenn Sie eine benutzerdefinierte verwaltete Sicherheitsgruppe angeben.

Typ	Protoc (Protok l)	Port-Bere ich	Quelle	Details
Regeln für eingehenden Datenverkehr				
Alles ICMP - IPV4	Alle	N/A	Die Gruppen-I D der verwaltet en Sicherhei tsgruppe für Core- und Aufgaben-Instances. Mit anderen Worten, dieselbe Sicherhei tsgruppe, in der die Regel angezeigt wird.	Diese reflexiven Regeln ermöglichen eingehenden Datenverkehr aus allen mit der angegebenen Sicherheitsgruppe verknüpft en Instances. Wenn die Standardeinstellung `ElasticMapReduce-core` für mehrere Cluster verwendet wird, können die Core- und Task-Instances dieser Cluster über einen ICMP TCP oder einen beliebigen UDP O-Port miteinander kommunizieren. Sie geben benutzerdefinierte verwaltete Sicherhei tsgruppen an, um den Cluster-übergreifenden Zugriff einzuschränken.
Alle TCP	TCP	Alle		
Alle UDP	UDP	Alle		
Alles ICMP - IPV4	Alle	N/A	Die Gruppen-ID der verwalteten Sicherheitsgruppe für die Primär-In stance.	Diese Regeln lassen den gesamten eingehenden ICMP Datenverkehr und den Datenverkehr über einen beliebigen TCP UDP Port von allen primären Instances zu, die der angegebenen Sicherheitsgruppe zugeordnet sind, auch wenn sich die Instances in unterschiedlichen Clustern befinden.
Alle TCP	TCP	Alle		
Alle UDP	UDP	Alle		

Typ	Protoc (Protok l)	Port-Bere ich	Quelle	Details
HTTPS(843)	TCP	8443	Die Gruppen-ID der verwalteten Sicherheitsgruppe für den Servicezu griff in einem privaten Subnetz.	Diese Regel ermöglicht dem Cluster-Manager die Kommunikation mit Core- und Aufgabenk noten.
Regeln für ausgehenden Datenverkehr				
Gesamter Datenverk ehr	Alle	Alle	0.0.0.0/0	Weitere Informationen finden Sie unter Regeln für ausgehenden Datenverkehr bearbeiten weiter unten in diesem Dokument.
Benutzerd efiniert TCP	TCP	80 (http) oder 443 (https)	Die Gruppen-ID der verwalteten Sicherheitsgruppe für den Servicezu griff in einem privaten Subnetz.	Wenn die obige Standardregel „Gesamter Datenverkehr“ für ausgehenden Datenverkehr entfernt wird, ist diese Regel eine Mindestan forderung für Amazon EMR 5.30.0 und höher, um über https eine Verbindung zu Amazon S3 herzustellen. **Note** Amazon EMR fügt diese Regel nicht hinzu, wenn Sie eine benutzerd efinierte verwaltete Sicherheitsgruppe verwenden.

Regeln für ausgehenden Datenverkehr bearbeiten

Standardmäßig EMR erstellt Amazon diese Sicherheitsgruppe mit ausgehenden Regeln, die den gesamten ausgehenden Verkehr auf allen Protokollen und Ports zulassen. Das Zulassen des gesamten ausgehenden Datenverkehrs ist ausgewählt, da für verschiedene Amazon EMR - und Kundenanwendungen, die auf EMR Amazon-Clustern ausgeführt werden können, möglicherweise

unterschiedliche Ausgangsregeln erforderlich sind. Amazon EMR kann diese spezifischen Einstellungen bei der Erstellung von Standardsicherheitsgruppen nicht antizipieren. Sie können den ausgehenden Datenverkehr in Ihren Sicherheitsgruppen einschränken, sodass nur die Regeln berücksichtigt werden, die Ihren Anwendungsfällen und Sicherheitsrichtlinien entsprechen. Für diese Sicherheitsgruppe sind mindestens die folgenden Regeln für ausgehenden Datenverkehr erforderlich, für einige Anwendungen sind jedoch möglicherweise zusätzliche Regeln für ausgehenden Datenverkehr erforderlich.

Typ	Protoc (Protok l)	Port-Bere ich	Bestimmungsort	Details
Alle TCP	TCP	Alle	pl-*xxxxxxxx*	Verwaltete Präfixliste von Amazon S3 `com.amazonaws.` *MyRegion*`.s3`.
Gesamter Datenverkehr	Alle	Alle	sg-*xxxxxxxxx xxxxxxxx*	Die ID der Sicherheitsgruppe `ElasticMapReduce-Core-Private` .
Gesamter Datenverkehr	Alle	Alle	sg-*xxxxxxxxx xxxxxxxx*	Die ID der Sicherheitsgruppe `ElasticMapReduce-Primary-Private` .
Benutzerdefiniert TCP	TCP	9443	sg-*xxxxxxxxx xxxxxxxx*	Die ID der Sicherheitsgruppe `ElasticMapReduce-ServiceAccess` .

Von Amazon EMR verwaltete Sicherheitsgruppe für den Servicezugriff (private Subnetze)

Die standardmäßig verwaltete Sicherheitsgruppe für den Servicezugriff in privaten Subnetzen hat den Gruppennamen -. ElasticMapReduce ServiceAccess Sie verfügt über Regeln für eingehenden und ausgehenden Datenverkehr, die den Datenverkehr HTTPS (Port 8443, Port 9443) zu den anderen verwalteten Sicherheitsgruppen in privaten Subnetzen zulassen. Diese Regeln ermöglichen dem Cluster-Manager die Kommunikation mit dem Primärknoten und mit Kern- und Aufgabenknoten. Dieselben Regeln sind erforderlich, wenn Sie benutzerdefinierte Sicherheitsgruppen verwenden.

Typ	Protoc (Protok l)	Port-Bere ich	Quelle	Details
Regeln für eingehende Nachrichten Erforderlich für EMR Amazon-Cluster mit EMR Amazon-Version 5.30.0 und höher.				
Benutzerdefiniert TCP	TCP	9443	Die Gruppen-ID der verwalteten Sicherheitsgruppe für die Primär-Instance.	Diese Regel ermöglicht die Kommunikation zwischen der Sicherheitsgruppe der Primär-Instance und der Sicherheitsgruppe des Servicezugriffs.
Ausgehende Regeln für alle EMR Amazon-Cluster erforderlich				
Benutzerdefiniert TCP	TCP	8443	Die Gruppen-ID der verwalteten Sicherheitsgruppe für die Primär-Instance.	Diese Regeln ermöglichen dem Cluster-Manager die Kommunikation mit dem Primärknoten und mit Kern- und Aufgabenknoten.
Benutzerdefiniert TCP	TCP	8443	Die Gruppen-ID der verwalteten Sicherheitsgruppe für Core- und Aufgaben-Instances.	Diese Regeln ermöglichen dem Cluster-Manager die Kommunikation mit dem Primärknoten und mit Kern- und Aufgabenknoten.

Arbeiten mit zusätzlichen Sicherheitsgruppen

Sie können zusätzliche Sicherheitsgruppen unabhängig davon verwenden, ob Sie die standardmäßigen verwalteten Sicherheitsgruppen verwenden oder benutzerdefinierte verwaltete Sicherheitsgruppen angeben. Mit zusätzlichen Sicherheitsgruppen können Sie den Zugriff auf die einzelnen Cluster und von externen Clients, Ressourcen und Anwendungen anpassen.

Betrachten Sie beispielsweise das folgende Szenario. Sie haben mehrere Cluster, die Sie für die Kommunikation untereinander benötigen, aber Sie möchten eingehenden SSH Zugriff auf die primäre

Instance nur für eine bestimmte Teilmenge von Clustern zulassen. Hierzu können Sie für die Cluster den gleichen Satz von verwalteten Sicherheitsgruppen verwenden. Anschließend erstellen Sie zusätzliche Sicherheitsgruppen, die eingehenden SSH Zugriff von vertrauenswürdigen Clients ermöglichen, und geben die zusätzlichen Sicherheitsgruppen für die primäre Instanz für jeden Cluster in der Teilmenge an.

Sie können bis zu 15 zusätzliche Sicherheitsgruppen für die primäre Instance, 15 für Core- und Task-Instances und 15 für den Dienstzugriff (in privaten Subnetzen) anwenden. Wenn notwendig, können Sie dieselben zusätzlichen Sicherheitsgruppen für Primär-Instances, Core- und Aufgaben-Instances und den Servicezugriff angeben. Die maximale Anzahl von Sicherheitsgruppen und -regeln in Ihrem Konto unterliegt Kontolimits. Weitere Informationen finden Sie unter Grenzwerte für Sicherheitsgruppen im VPCAmazon-Benutzerhandbuch.

Angabe von von EMR Amazon verwalteten und zusätzlichen Sicherheitsgruppen

Sie können Sicherheitsgruppen mit dem AWS Management Console AWS CLI, dem oder Amazon angeben EMRAPI. Wenn Sie keine Sicherheitsgruppen angeben, EMR erstellt Amazon Standardsicherheitsgruppen. Die Angabe zusätzlicher Sicherheitsgruppen ist optional. Sie können Primär-Instances, Core- und Aufgaben-Instances und dem Servicezugriff (nur private Subnetze) zusätzliche Sicherheitsgruppen zuweisen.

Console

Um Sicherheitsgruppen mit der Konsole anzugeben

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und anschließend Create cluster aus.
3. Wählen Sie unter Netzwerk den Pfeil neben EC2Sicherheitsgruppen (Firewall) aus, um diesen Abschnitt zu erweitern. Unter Primärer Knoten und Kern- und Aufgabenknoten sind standardmäßig die von Amazon EMR verwalteten Standardsicherheitsgruppen ausgewählt. Wenn Sie ein privates Subnetz verwenden, haben Sie auch die Möglichkeit, eine Sicherheitsgruppe für den Servicezugriff auszuwählen.
4. Um Ihre von Amazon EMR verwaltete Sicherheitsgruppe zu ändern, verwenden Sie das Dropdownmenü Sicherheitsgruppen auswählen, um eine andere Option aus der EMROptionsliste der von Amazon verwalteten Sicherheitsgruppen auszuwählen. Sie haben

eine von Amazon EMR verwaltete Sicherheitsgruppe sowohl für den Primärknoten als auch für den Kern- und Aufgabenknoten.

5. Um benutzerdefinierte Sicherheitsgruppen hinzuzufügen, verwenden Sie dasselbe Dropdownmenü Sicherheitsgruppen auswählen, um bis zu vier benutzerdefinierte Sicherheitsgruppen aus der Optionsliste Benutzerdefinierte Sicherheitsgruppen auszuwählen. Sie können bis zu vier benutzerdefinierte Sicherheitsgruppen sowohl für den Primärknoten als auch für den Kern- und Aufgabenknoten einrichten.
6. Wählen Sie alle anderen Optionen aus, die für Ihren Cluster gelten.
7. Um Ihren Cluster jetzt zu starten, wählen Sie Cluster erstellen aus.

Angeben von Sicherheitsgruppen mit der AWS CLI

Um Sicherheitsgruppen mithilfe von anzugeben, verwenden AWS CLI Sie den `create-cluster` Befehl mit den folgenden `--ec2-attributes` Optionsparametern:

Parameter	Beschreibung
`EmrManagedPrimarySecurityGroup`	Verwenden Sie diesen Parameter, um eine benutzerdefinierte verwaltete Sicherheitsgruppe für die Primär-Instance anzugeben. Wenn dieser Parameter angegeben wird, muss auch `EmrManagedCoreSecurityGroup` angegeben werden. Für Cluster in privaten Subnetzen muss auch `ServiceAccessSecurityGroup` angegeben werden.
`EmrManagedCoreSecurityGroup`	Verwenden Sie diesen Parameter, um eine benutzerdefinierte verwaltete Sicherheitsgruppe für Core- und Aufgaben-Instances anzugeben. Wenn dieser Parameter angegeben wird, muss auch `EmrManagedPrimarySecurityGroup` angegeben werden. Für Cluster in privaten Subnetzen muss auch `ServiceAccessSecurityGroup` angegeben werden.

Parameter	Beschreibung
`ServiceAccessSecurityGroup`	Verwenden Sie diesen Parameter, um eine benutzerdefinierte verwaltete Sicherheitsgruppe für den Servicezugriff anzugeben. Dies gilt nur für Cluster in privaten Subnetzen. Die Sicherheitsgruppe, als die Sie angeben, `ServiceAccessSecurityGroup` sollte nicht für andere Zwecke verwendet werden und sollte auch Amazon vorbehalten seinEMR. Wenn dieser Parameter angegeben wird, muss auch `EmrManagedPrimarySecurityGroup` angegeben werden.
`AdditionalPrimarySecurityGroups`	Verwenden Sie diesen Parameter, um bis zu vier zusätzliche verwaltete Sicherheitsgruppen für die Primär-Instance anzugeben.
`AdditionalCoreSecurityGroups`	Verwenden Sie diesen Parameter, um bis zu vier zusätzliche verwaltete Sicherheitsgruppen für die Core- und Aufgaben-Instances anzugeben.

Example — spezifizieren Sie benutzerdefinierte, von Amazon EMR verwaltete Sicherheitsgruppen und zusätzliche Sicherheitsgruppen

Das folgende Beispiel spezifiziert benutzerdefinierte, von Amazon EMR verwaltete Sicherheitsgruppen für einen Cluster in einem privaten Subnetz, mehrere zusätzliche Sicherheitsgruppen für die primäre Instance und eine einzelne zusätzliche Sicherheitsgruppe für Core- und Task-Instances.

> **Note**
>
> Linux-Zeilenfortsetzungszeichen (\) sind aus Gründen der Lesbarkeit enthalten. Sie können entfernt oder in Linux-Befehlen verwendet werden. Entfernen Sie sie unter Windows oder ersetzen Sie sie durch ein Caret-Zeichen (^).

```
aws emr create-cluster --name "ClusterCustomManagedAndAdditionalSGs" \
--release-label emr-emr-7.2.0 --applications Name=Hue Name=Hive \
Name=Pig --use-default-roles --ec2-attributes \
SubnetIds=subnet-xxxxxxxxxxxx,KeyName=myKey,\
ServiceAccessSecurityGroup=sg-xxxxxxxxxxxx,\
EmrManagedPrimarySecurityGroup=sg-xxxxxxxxxxxx,\
EmrManagedCoreSecurityGroup=sg-xxxxxxxxxxx,\
AdditionalPrimarySecurityGroups=['sg-xxxxxxxxxxx',\
'sg-xxxxxxxxxxx','sg-xxxxxxxxxx'],\
AdditionalCoreSecurityGroups=sg-xxxxxxxxxxx \
--instance-type m5.xlarge
```

Weitere Informationen finden Sie unter create-cluster in der AWS CLI -Befehlsreferenz.

EC2Sicherheitsgruppen für EMR Notebooks angeben

Wenn Sie ein EMR Notizbuch erstellen, werden zwei Sicherheitsgruppen verwendet, um den Netzwerkverkehr zwischen dem EMR Notebook und dem EMR Amazon-Cluster zu steuern, wenn Sie den Notebook-Editor verwenden. Die Standardsicherheitsgruppen haben minimale Regeln, die nur Netzwerkverkehr zwischen dem EMR Notebooks-Service und den Clustern zulassen, an die Notebooks angeschlossen sind.

Ein EMR Notebook verwendet Apache Livy, um über einen Proxy über TCP Port 18888 mit dem Cluster zu kommunizieren. Indem Sie benutzerdefinierte Sicherheitsgruppen mit an Ihre Umgebung angepassten Regeln erstellen, können Sie den Netzwerkdatenverkehr so einschränken, dass nur ein Teil der Notebooks Code innerhalb des Notebook-Editors auf bestimmten Clustern ausführen kann. Der Cluster verwendet Ihre benutzerdefinierte Sicherheit zusätzlich zu den Standardsicherheitsgruppen für den Cluster. Weitere Informationen finden Sie unter Steuern des Netzwerkverkehrs mit Sicherheitsgruppen im Amazon EMR Management Guide undEC2Sicherheitsgruppen für EMR Notebooks angeben.

EC2Standard-Sicherheitsgruppe für die primäre Instance

Die EC2 Standardsicherheitsgruppe für die primäre Instance ist zusätzlich zu den Sicherheitsgruppen des Clusters für die primäre Instance mit der primären Instance verknüpft.

Gruppenname: ElasticMapReduceEditors-Livy

Regeln

- Eingehend

Erlaube TCP Port 18888 von allen Ressourcen in der EC2 Standardsicherheitsgruppe für Notebooks EMR

- Ausgehend

  None

EC2Standardsicherheitsgruppe für Notebooks EMR

Die EC2 Standardsicherheitsgruppe für das EMR Notizbuch ist dem Notizbuch-Editor für jedes EMR Notizbuch zugeordnet, dem es zugewiesen ist.

Gruppenname: ElasticMapReduceEditors-Editor

Regeln

- Eingehend

  None

- Ausgehend

  Erlauben Sie TCP Port 18888 für alle Ressourcen in der EC2 Standardsicherheitsgruppe für EMR Notebooks.

Benutzerdefinierte EC2 Sicherheitsgruppe für EMR Notebooks bei der Verknüpfung von Notebooks mit Git-Repositorys

Um ein Git-Repository mit Ihrem Notebook zu verknüpfen, muss die Sicherheitsgruppe für das EMR Notebook eine ausgehende Regel enthalten, damit das Notebook den Datenverkehr ins Internet weiterleiten kann. Es wird empfohlen, zu diesem Zweck eine neue Sicherheitsgruppe zu erstellen. Durch die Aktualisierung der Standard-Sicherheitsgruppe ElasticMapReduceEditors-Editor gelten möglicherweise dieselben Regeln für ausgehende Nachrichten auch für andere Notebooks, die zu dieser Sicherheitsgruppe gehören.

Regeln

- Eingehend

  None

- Ausgehend

  Erlauben Sie dem Notebook, Datenverkehr über den Cluster an das Internet zu leiten, wie im folgenden Beispiel veranschaulicht. Der Wert 0.0.0.0/0 wird für Beispielzwecke verwendet. Sie können diese Regel ändern, um die IP-Adressen für Ihre Git-basierten Repositorys anzugeben.

Typ	Protocol (Protokoll)	Port-Bereich	Bestimmungsort
Benutzerdefinierte Regel TCP	TCP	18888	SG-
HTTPS	TCP	443	0.0.0.0/0

Verwenden Sie Amazon, um EMR den öffentlichen Zugriff zu blockieren

Amazon EMR Block Public Access (BPA) verhindert, dass Sie einen Cluster in einem öffentlichen Subnetz starten, wenn der Cluster über eine Sicherheitskonfiguration verfügt, die eingehenden Datenverkehr von öffentlichen IP-Adressen an einem Port zulässt.

> **Important**
>
> Den öffentlichen Zugriff blockieren ist standardmäßig aktiviert. Um den Kontoschutz zu erhöhen, empfehlen wir, ihn aktiviert zu lassen.

Grundlegendes zum Blockieren des öffentlichen Zugriffs

Sie können die Konfiguration Block Public Access auf Kontoebene verwenden, um den öffentlichen Netzwerkzugriff auf EMR Amazon-Cluster zentral zu verwalten.

Wenn ein Benutzer von Ihnen einen Cluster AWS-Konto startet, EMR überprüft Amazon die Portregeln in der Sicherheitsgruppe für den Cluster und vergleicht sie mit Ihren Regeln für eingehenden Datenverkehr. Wenn die Sicherheitsgruppe über eine Regel für eingehenden Datenverkehr verfügt, die Ports zu den öffentlichen IP-Adressen IPv4 0.0.0.0/0 oderIPv6:: /0 öffnet, und diese Ports nicht als Ausnahmen für Ihr Konto angegeben sind, lässt Amazon den Benutzer den EMR Cluster nicht erstellen.

Wenn ein Benutzer die Sicherheitsgruppenregeln für einen laufenden Cluster in einem öffentlichen Subnetz so ändert, dass er über eine Regel für den öffentlichen Zugriff verfügt, die gegen die BPA Konfiguration Ihres Kontos verstößt, EMR widerruft Amazon die neue Regel, sofern Amazon dazu berechtigt ist. Wenn Amazon EMR nicht berechtigt ist, die Regel zu widerrufen, wird im AWS Health Dashboard ein Ereignis erstellt, das den Verstoß beschreibt. Informationen dazu, wie Sie Amazon die Berechtigung zum Widerrufen der Regel erteilenEMR, finden Sie unterAmazon so konfigurierenEMR, dass Sicherheitsgruppenregeln aufgehoben werden.

Den öffentlichen Zugriff blockieren ist standardmäßig für alle Cluster in jedem AWS-Region für Ihr AWS-Konto aktiviert. BPAgilt für den gesamten Lebenszyklus eines Clusters, gilt jedoch nicht für Cluster, die Sie in privaten Subnetzen erstellen. Sie können Ausnahmen von der BPA Regel konfigurieren; Port 22 ist standardmäßig eine Ausnahme. Weitere Informationen zur Einstellung finden Sie unter Konfigurieren von Block Public Access.

Konfigurieren von Block Public Access

Sie können die Sicherheitsgruppen und die Konfiguration zum Sperren des öffentlichen Zugriffs in Ihren Konten jederzeit aktualisieren.

Sie können die Einstellungen für den öffentlichen Zugriff blockieren (BPA) mit dem AWS Management Console, dem AWS Command Line Interface (AWS CLI) und dem Amazon ein- und ausschalten EMRAPI. Die Einstellungen gelten für Ihr Konto je nach Region. Um die Clustersicherheit aufrechtzuerhalten, empfehlen wir die Verwendung vonBPA.

Console

Um den öffentlichen Zugriff blockieren mit der Konsole zu konfigurieren

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie dann die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie in der oberen Navigationsleiste die Region aus, die Sie konfigurieren möchten, sofern sie nicht bereits ausgewählt ist.
3. Wählen Sie EMREC2im linken Navigationsbereich unter Ein die Option Öffentlichen Zugriff blockieren aus.
4. Führen Sie unter Block public access settings (Einstellungen für die Sperrung des öffentlichen Zugriffs) die folgenden Schritte aus.

Zu …	Vorgehensweise
Block Public Access aktivieren oder deaktivieren	Wählen Sie Bearbeiten, wählen Sie je nach Bedarf Einschalten oder Ausschalten und wählen Sie dann Speichern.
Ports in der Liste der Ausnahmen bearbeiten	1. Wählen Sie Bearbeiten und suchen Sie den Abschnitt Ausnahmen für den Portbereich. 2. Um der Liste der Ausnahmen Ports hinzuzufügen, wählen Sie Add a port range (Port-Bereich hinzufügen) aus und geben Sie einen neuen Port oder Port-Bereich ein. Wiederholen Sie den Vorgang für jeden Port oder Port-Bereich, der hinzugefügt werden soll. 3. Um einen Port oder Portbereich zu entfernen, wählen Sie das Entfernen neben dem Eintrag in der Liste Portbereiche. 4. Wählen Sie Save (Speichern) aus.

AWS CLI

Um den öffentlichen Zugriff blockieren zu konfigurieren, verwenden Sie AWS CLI

- Verwenden Sie den `aws emr put-block-public-access-configuration`-Befehl, um Block Public Access zu konfigurieren, wie in den folgenden Beispielen gezeigt.

Zu …	Vorgehensweise
Block Public Access aktivieren	Legen Sie `BlockPublicSecurityGroupRules` wie im folgenden Beispiel gezeigt auf `true` fest. Damit der Cluster gestartet werden kann, darf keine Sicherheitsgruppe, die einem Cluster zugeordnet ist, über eine Regel für eingehenden Datenverkehr verfügen, die den öffentlichen Zugriff zulässt. `aws emr put-block-public-access-configuration --block-public-access-configuration BlockPublicSecurityGroupRules=true`
Block Public Access deaktivieren	Legen Sie `BlockPublicSecurityGroupRules` wie im folgenden Beispiel gezeigt auf `false` fest. Sicherheitsgruppen, die einem Cluster zugeordnet sind, können Regeln für eingehenden Datenverkehr aufweisen, die öffentlichen Zugriff auf beliebige Ports zulassen. Wir empfehlen diese Konfiguration nicht. `aws emr put-block-public-access-configuration --block-public-access-configuration BlockPublicSecurityGroupRules=false`

Zu …	Vorgehensweise
Block Public Access aktivieren und Ports als Ausnahmen angeben	Im folgenden Beispiel wird Block Public Access aktiviert und Port 22 sowie die Ports 100-101 werden als Ausnahmen angegeben. Auf diese Weise können Cluster erstellt werden, wenn eine zugeordnete Sicherheitsgruppe über eine Regel für eingehenden Datenverkehr verfügt, die öffentlichen Zugriff auf die Ports 22, 100 oder 101 zulässt. `aws emr put-block-public-access-configuration --block-public-access-configuration '{ "BlockPublicSecurityGroupRules": true, "PermittedPublicSecurityGroupRuleRanges": [ { "MinRange": 22, "MaxRange": 22 }, { "MinRange": 100, "MaxRange": 101 } ] }'`

Amazon so konfigurierenEMR, dass Sicherheitsgruppenregeln aufgehoben werden

Amazon EMR benötigt die Erlaubnis, Sicherheitsgruppenregeln zu widerrufen und Ihre Konfiguration für die Sperrung des öffentlichen Zugriffs einzuhalten. Sie können einen der folgenden Ansätze verwenden, um Amazon EMR die erforderliche Genehmigung zu erteilen:

- (Empfohlen) Fügen Sie der Servicerolle die `AmazonEMRServicePolicy_v2`-verwaltete Richtlinie hinzu. Weitere Informationen finden Sie unter Servicerolle für Amazon EMR (EMRRolle).
- Erstellen Sie eine neue Inline-Richtlinie, die die `ec2:RevokeSecurityGroupIngress`-Aktion für Sicherheitsgruppen ermöglicht. Weitere Informationen zum Ändern einer Rollenberechtigungsrichtlinie finden Sie unter Ändern einer Rollenberechtigungsrichtlinie mit der IAMKonsole und AWS CLIim IAMBenutzerhandbuch. AWS API

Beheben von Verletzungen des Blockieren des öffentlichen Zugriffs

Wenn ein Verstoß gegen die Sperrung des öffentlichen Zugriffs auftritt, können Sie ihn mit einer der folgenden Maßnahmen beheben:

- Wenn Sie auf eine Weboberfläche in Ihrem Cluster zugreifen möchten, verwenden Sie eine der unter Auf EMR Amazon-Clustern gehostete Weboberflächen anzeigen So greifen Sie über SSH (Port 22) auf die Schnittstelle zu.
- Um den Datenverkehr zum Cluster von bestimmten IP-Adressen statt von der öffentlichen IP-Adresse aus zuzulassen, fügen Sie eine Sicherheitsgruppenregel hinzu. Weitere Informationen finden Sie unter Regeln zu einer Sicherheitsgruppe hinzufügen im Amazon-Handbuch EC2 Erste Schritte.
- (Nicht empfohlen) Sie können EMR BPA Amazon-Ausnahmen so konfigurieren, dass sie den gewünschten Port oder Portbereich enthalten. Wenn Sie eine BPA Ausnahme angeben, gehen Sie mit einem ungeschützten Port ein Risiko ein. Wenn Sie beabsichtigen, eine Ausnahme anzugeben, sollten Sie die Ausnahme entfernen, sobald sie nicht mehr benötigt wird. Weitere Informationen finden Sie unter Konfigurieren von Block Public Access.

Identifizieren Sie Cluster, die Sicherheitsgruppenregeln zugeordnet sind

Möglicherweise müssen Sie alle Cluster identifizieren, die einer bestimmten Sicherheitsgruppenregel zugeordnet sind, oder die Sicherheitsgruppenregel für einen bestimmten Cluster finden.

- Wenn Sie die Sicherheitsgruppe kennen, können Sie die zugehörigen Cluster identifizieren, wenn Sie die Netzwerkschnittstellen für die Sicherheitsgruppe finden. Weitere Informationen finden Sie unter Wie finde ich die Ressourcen, die einer EC2 Amazon-Sicherheitsgruppe zugeordnet sind? nein AWS re:Post. Die EC2 Amazon-Instances, die an diese Netzwerkschnittstellen angeschlossen sind, werden mit der ID des Clusters gekennzeichnet, zu dem sie gehören.
- Wenn Sie die Sicherheitsgruppen für einen bekannten Cluster suchen möchten, folgen Sie den Schritten unter Cluster-Status und -Details anzeigen. Sie finden die Sicherheitsgruppen für den Cluster im Bereich Netzwerk und Sicherheit in der Konsole oder im `Ec2InstanceAttributes`-Feld unter AWS CLI.

Konformitätsvalidierung für Amazon EMR

Externe Prüfer bewerten die Sicherheit und Konformität von Amazon im EMR Rahmen mehrerer AWS Compliance-Programme. Dazu gehören SOC PCI RAMPHIPAA, Fed und andere.

Eine Liste der AWS Dienstleistungen im Rahmen bestimmter Compliance-Programme finden Sie unter AWS Dienstleistungen im Umfang der einzelnen Compliance-Programme. Allgemeine Informationen finden Sie unter AWS -Compliance-Programme.

Sie können Prüfberichte von Drittanbietern unter herunterladen AWS Artifact. Weitere Informationen finden Sie unter Berichte herunterladen in AWS Artifact.

Ihre Compliance-Verantwortung bei der Nutzung von Amazon EMR hängt von der Sensibilität Ihrer Daten, den Compliance-Zielen Ihres Unternehmens und den geltenden Gesetzen und Vorschriften ab. Wenn Ihre Nutzung von Amazon der Einhaltung von Standards wieHIPAA, oder Fed EMR unterliegtPCI, AWS bietet Fed RessourcenRAMP, die Ihnen helfen:

- Schnellstartanleitungen zu Sicherheit und Compliance — In diesen Bereitstellungsleitfäden werden architektonische Überlegungen erörtert und Schritte für die Implementierung von Basisumgebungen beschrieben, auf denen Sicherheit und Compliance im Vordergrund stehen. AWS
- Whitepaper Architecting for HIPAA Security and Compliance — In diesem Whitepaper wird beschrieben, wie Unternehmen damit -konforme Anwendungen erstellen können AWS . HIPAA
- AWS Ressourcen zur Einhaltung von Vorschriften — Diese Sammlung von Arbeitsmappen und Leitfäden kann auf Ihre Branche und Ihren Standort zutreffen.
- AWS Config— Dieser AWS Service bewertet, wie gut Ihre Ressourcenkonfigurationen den internen Praktiken, Branchenrichtlinien und Vorschriften entsprechen.
- AWS Security Hub— Dieser AWS Service bietet einen umfassenden Überblick über Ihren Sicherheitsstatus und hilft Ihnen AWS , die Einhaltung der Sicherheitsstandards und bewährten Verfahren der Sicherheitsbranche zu überprüfen.

Resilienz bei Amazon EMR

Die AWS globale Infrastruktur basiert auf AWS Regionen und Availability Zones. AWS Regionen bieten mehrere physisch getrennte und isolierte Availability Zones, die über Netzwerke mit niedriger Latenz, hohem Durchsatz und hoher Redundanz miteinander verbunden sind. Mithilfe von Availability

Zones können Sie Anwendungen und Datenbanken erstellen und ausführen, die automatisch Failover zwischen Availability Zones ausführen, ohne dass es zu Unterbrechungen kommt. Availability Zones sind besser hoch verfügbar, fehlertoleranter und skalierbarer als herkömmliche Infrastrukturen mit einem oder mehreren Rechenzentren.

Weitere Informationen zu AWS Regionen und Availability Zones finden Sie unter AWS Globale Infrastruktur.

Zusätzlich zur AWS globalen Infrastruktur EMR bietet Amazon mehrere Funktionen, um Ihre Datenstabilität und Backup-Anforderungen zu erfüllen.

- Integration mit Amazon S3 über EMRFS
- Support für mehrere Master-Knoten

Infrastruktursicherheit bei Amazon EMR

Als verwalteter Service EMR ist Amazon durch AWS globale Netzwerksicherheit geschützt. Informationen zu AWS Sicherheitsdiensten und zum AWS Schutz der Infrastruktur finden Sie unter AWS Cloud-Sicherheit. Informationen zum Entwerfen Ihrer AWS Umgebung unter Verwendung der bewährten Methoden für die Infrastruktursicherheit finden Sie unter Infrastructure Protection in Security Pillar AWS Well-Architected Framework.

Sie verwenden AWS veröffentlichte API Anrufe, um EMR über das Netzwerk auf Amazon zuzugreifen. Kunden müssen Folgendes unterstützen:

- Sicherheit auf Transportschicht (TLS). Wir benötigen TLS 1.2 und empfehlen TLS 1.3.
- Cipher-Suites mit perfekter Vorwärtsgeheimhaltung (PFS) wie (Ephemeral Diffie-Hellman) oder DHE (Elliptic Curve Ephemeral Diffie-Hellman). ECDHE Die meisten modernen Systeme wie Java 7 und höher unterstützen diese Modi.

Darüber hinaus müssen Anfragen mithilfe einer Zugriffsschlüssel-ID und eines geheimen Zugriffsschlüssels, der einem Prinzipal zugeordnet ist, signiert werden. IAM Alternativ können Sie mit AWS Security Token Service (AWS STS) temporäre Sicherheitsanmeldeinformationen erstellen, um die Anforderungen zu signieren.

Themen

- Stellen Sie EMR über einen VPC Schnittstellenendpunkt eine Connect zu Amazon her

Stellen Sie EMR über einen VPC Schnittstellenendpunkt eine Connect zu Amazon her

Sie können EMR über einen VPCSchnittstellenendpunkt (AWS PrivateLink) in Ihrer Virtual Private Cloud (VPC) eine direkte Verbindung zu Amazon herstellen, anstatt eine Verbindung über das Internet herzustellen. Wenn Sie einen VPC Schnittstellenendpunkt verwenden, EMR erfolgt die Kommunikation zwischen Ihnen VPC und Amazon ausschließlich innerhalb des AWS Netzwerks. Jeder VPC Endpunkt wird durch eine oder mehrere Elastic Network-Schnittstellen (ENIs) mit privaten IP-Adressen in Ihren VPC Subnetzen repräsentiert.

Der VPC Schnittstellenendpunkt verbindet Sie EMR ohne Internet-Gateway, NAT Gerät, VPN Verbindung oder AWS Direct Connect Verbindung VPC direkt mit Amazon. Die Instances in Ihrem System benötigen VPC keine öffentlichen IP-Adressen, um mit Amazon zu kommunizieren EMRAPI.

Um Amazon EMR über Ihren nutzen zu könnenVPC, müssen Sie eine Verbindung von einer Instance herstellen, die sich innerhalb des Netzwerks befindet, VPC oder über ein Amazon Virtual Private Network (VPN) oder Ihr VPC privates Netzwerk mit Ihrem verbinden AWS Direct Connect. Informationen zu Amazon VPN finden Sie unter VPNVerbindungen im Amazon Virtual Private Cloud Cloud-Benutzerhandbuch. Weitere Informationen AWS Direct Connect dazu finden Sie unter Verbindung erstellen im AWS Direct Connect Benutzerhandbuch.

Sie können einen VPC Schnittstellenendpunkt erstellen, um EMR mithilfe der AWS Konsole oder der Befehle AWS Command Line Interface (AWS CLI) eine Verbindung zu Amazon herzustellen. Weitere Informationen finden Sie unter Erstellen eines Schnittstellenendpunkts.

Wenn Sie nach dem Erstellen eines VPC Schnittstellenendpunkts private DNS Hostnamen für den Endpunkt aktivieren, wird der EMR Standard-Amazon-Endpunkt zu Ihrem VPC Endpunkt aufgelöst. Der Standardendpunkt für Servicenamen für Amazon EMR hat das folgende Format.

```
elasticmapreduce.Region.amazonaws.com
```

Wenn Sie private DNS Hostnamen nicht aktivieren, VPC stellt Amazon einen DNS Endpunktnamen bereit, den Sie im folgenden Format verwenden können.

```
VPC_Endpoint_ID.elasticmapreduce.Region.vpce.amazonaws.com
```

Weitere Informationen finden Sie unter Interface VPC Endpoints (AWS PrivateLink) im VPCAmazon-Benutzerhandbuch.

Amazon EMR unterstützt das Aufrufen all seiner APIAktionen in IhremVPC.

Sie können VPC Endpunktrichtlinien an einen VPC Endpunkt anhängen, um den Zugriff für IAM Prinzipale zu kontrollieren. Sie können einem VPC Endpunkt auch Sicherheitsgruppen zuordnen, um den eingehenden und ausgehenden Zugriff auf Grundlage des Ursprungs und Ziels des Netzwerkverkehrs zu steuern, z. B. anhand eines Bereichs von IP-Adressen. Weitere Informationen finden Sie unter Steuern des Zugriffs auf Dienste mit VPC Endpunkten.

Erstellen Sie eine VPC Endpunktrichtlinie für Amazon EMR

Sie können eine Richtlinie für VPC Amazon-Endgeräte für Amazon erstellenEMR, um Folgendes festzulegen:

- Prinzipal, der Aktionen ausführen/nicht ausführen kann
- Aktionen, die ausgeführt werden können
- Ressourcen, für die Aktionen ausgeführt werden können

Weitere Informationen finden Sie unter Steuern des Zugriffs auf Dienste mit VPC Endpunkten im VPCAmazon-Benutzerhandbuch.

Example — VPC Endpunktrichtlinie, um jeglichen Zugriff von einem bestimmten AWS Konto aus zu verweigern

Die folgende VPC Endpunktrichtlinie verweigert AWS das Konto *123456789012* der gesamte Zugriff auf Ressourcen über den Endpunkt.

```
{
    "Statement": [
        {
            "Action": "*",
            "Effect": "Allow",
            "Resource": "*",
            "Principal": "*"
        },
        {
            "Action": "*",
            "Effect": "Deny",
            "Resource": "*",
            "Principal": {
                "AWS": [
```

```
                    "123456789012"
                ]
            }
        }
    ]
}
```

Example — VPC Endpunktrichtlinie, die VPC den Zugriff nur einem bestimmten IAM Prinzipal (Benutzer) erlaubt

Die folgende VPC Endpunktrichtlinie erlaubt vollen Zugriff nur einem Benutzer *lijuan* im AWS Konto *123456789012*. Allen anderen IAM Prinzipalen wird der Zugriff über den Endpunkt verweigert.

```
{
    "Statement": [
        {
            "Action": "*",
            "Effect": "Allow",
            "Resource": "*",
            "Principal": {
                "AWS": [
                    "arn:aws:iam::123456789012:user/lijuan"
                ]
            }
        }]
}
```

Example — VPC Endpunktrichtlinie zur Zulassung von schreibgeschützten Vorgängen EMR

Die folgende VPC Endpunktrichtlinie erlaubt nur Konten AWS *123456789012* um die angegebenen EMR Amazon-Aktionen durchzuführen.

Die angegebenen Aktionen entsprechen dem Nur-Lese-Zugriff für Amazon. EMR Alle anderen Aktionen auf dem VPC werden für das angegebene Konto verweigert. Allen anderen Konten wird der Zugriff verweigert. Eine Liste der EMR Amazon-Aktionen finden Sie unter Aktionen, Ressourcen und Bedingungsschlüssel für Amazon EMR.

```
{
    "Statement": [
        {
            "Action": [
                "elasticmapreduce:DescribeSecurityConfiguration",
```

```
                "elasticmapreduce:GetBlockPublicAccessConfiguration",
                "elasticmapreduce:ListBootstrapActions",
                "elasticmapreduce:ViewEventsFromAllClustersInConsole",
                "elasticmapreduce:ListSteps",
                "elasticmapreduce:ListInstanceFleets",
                "elasticmapreduce:DescribeCluster",
                "elasticmapreduce:ListInstanceGroups",
                "elasticmapreduce:DescribeStep",
                "elasticmapreduce:ListInstances",
                "elasticmapreduce:ListSecurityConfigurations",
                "elasticmapreduce:DescribeEditor",
                "elasticmapreduce:ListClusters",
                "elasticmapreduce:ListEditors"
            ],
            "Effect": "Allow",
            "Resource": "*",
            "Principal": {
                "AWS": [
                    "123456789012"
                ]
            }
        }
    ]
}
```

Example — VPC Endpunktrichtlinie, die den Zugriff auf einen bestimmten Cluster verweigert

Die folgende VPC Endpunktrichtlinie ermöglicht vollen Zugriff für alle Konten und Prinzipale, verweigert jedoch jeglichen Zugriff für Konten AWS *123456789012* zu Aktionen, die auf dem EMR Amazon-Cluster mit Cluster-ID ausgeführt wurden *j-A1B2CD34EF5G*. Andere EMR Amazon-Aktionen, die keine Berechtigungen auf Ressourcenebene für Cluster unterstützen, sind weiterhin zulässig. Eine Liste der EMR Amazon-Aktionen und des entsprechenden Ressourcentyps finden Sie unter Aktionen, Ressourcen und Bedingungsschlüssel für Amazon EMR.

```
{
    "Statement": [
        {
            "Action": "*",
            "Effect": "Allow",
            "Resource": "*",
            "Principal": "*"
        },
        {
```

```
            "Action": "*",
            "Effect": "Deny",
            "Resource": "arn:aws:elasticmapreduce:us-west-2:123456789012:cluster/j-
A1B2CD34EF5G",
            "Principal": {
                "AWS": [
                    "123456789012"
                ]
            }
        }
    ]
}
```

Verwalten von Clustern

Nach dem Starten Ihres Clusters können Sie ihn überwachen und verwalten. Amazon EMR bietet mehrere Tools, mit denen Sie eine Verbindung zu Ihrem Cluster herstellen und ihn steuern können.

Themen

Verbinden mit einem Cluster

Wenn Sie einen EMR Amazon-Cluster ausführen, müssen Sie oft nur eine Anwendung ausführen, um Ihre Daten zu analysieren und dann die Ausgabe aus einem Amazon S3-Bucket zu sammeln. In anderen Fällen möchten Sie vielleicht mit dem Primärknoten interagieren, während der Cluster ausgeführt wird. Sie möchten z. B. eine Verbindung mit dem Primärknoten herstellen, um interaktive Abfragen auszuführen, Protokolldateien zu prüfen, ein Problem mit dem Cluster zu debuggen, Leistung mithilfe einer Anwendung wie Ganglia zu überwachen, die im Primärknoten ausgeführt wird, und so weiter. In den folgenden Abschnitten werden Techniken beschrieben, mit denen Sie eine Verbindung mit dem Primärknoten herstellen können.

In einem EMR Cluster ist der primäre Knoten eine EC2 Amazon-Instance, die die EC2 Instances koordiniert, die als Task- und Core-Knoten ausgeführt werden. Der primäre Knoten stellt einen öffentlichen DNS Namen zur Verfügung, mit dem Sie eine Verbindung zu ihm herstellen können. Standardmäßig EMR erstellt Amazon Sicherheitsgruppenregeln für den primären Knoten sowie für Kern- und Taskknoten, die bestimmen, wie Sie auf die Knoten zugreifen.

> **Note**
>
> Eine Verbindung mit dem Primärknoten ist nur möglich, während der Cluster ausgeführt wird. Wenn der Cluster beendet wird, wird die EC2 Instance, die als primärer Knoten

fungiert, beendet und ist nicht mehr verfügbar. Um eine Verbindung mit dem Primärknoten einzurichten, müssen Sie sich auch beim Cluster authentifizieren. Sie können entweder Kerberos für die Authentifizierung verwenden oder beim Starten des Clusters einen privaten EC2 Schlüssel für das Amazon-Schlüsselpaar angeben. Weitere Informationen zur Konfiguration von Kerberos und die Einrichtung einer Verbindung finden Sie unter Verwenden Sie Kerberos für die Authentifizierung bei Amazon EMR. Wenn Sie einen Cluster von der Konsole aus starten, wird der private EC2 Schlüssel des Amazon-Schlüsselpaars im Abschnitt Sicherheit und Zugriff auf der Seite Cluster erstellen angegeben.

Standardmäßig erlaubt die Sicherheitsgruppe ElasticMapReduce -master keinen eingehenden ZugriffSSH. Möglicherweise müssen Sie eine Regel für eingehenden Datenverkehr hinzufügen, die den SSH Zugriff (TCPPort 22) von den Quellen aus ermöglicht, auf die Sie zugreifen möchten. Weitere Informationen zum Ändern von Sicherheitsgruppenregeln finden Sie unter Regeln zu einer Sicherheitsgruppe hinzufügen im EC2Amazon-Benutzerhandbuch.

Important

Ändern Sie die verbleibenden Regeln in der Sicherheitsgruppe ElasticMapReduce -master nicht. Das Ändern dieser Regeln kann die Ausführung des Clusters beeinträchtigen.

Themen

Bevor Sie eine Verbindung herstellen: Autorisieren des eingehenden Datenverkehrs

Bevor Sie eine Verbindung zu einem EMR Amazon-Cluster herstellen, müssen Sie eingehenden SSH Datenverkehr (Port 22) von vertrauenswürdigen Clients autorisieren, z. B. die IP-Adresse Ihres Computers. Bearbeiten Sie dazu die verwalteten Sicherheitsgruppenregeln für die Knoten, zu denen Sie eine Verbindung herstellen möchten. Die folgenden Anweisungen zeigen Ihnen beispielsweise, wie Sie eine Regel für eingehenden Datenverkehr für den SSH Zugriff auf die Standard-Sicherheitsgruppe ElasticMapReduce -Master hinzufügen.

Weitere Informationen zur Verwendung von Sicherheitsgruppen mit Amazon EMR finden Sie unterSteuerung des Netzwerkverkehrs mit Sicherheitsgruppen.

Console

So gewähren Sie vertrauenswürdigen Quellen mit der Konsole SSH Zugriff auf die primäre Sicherheitsgruppe

Um Ihre Sicherheitsgruppen bearbeiten zu können, benötigen Sie die Berechtigung, Sicherheitsgruppen für die Gruppe zu verwaltenVPC, in der sich der Cluster befindet. Weitere Informationen finden Sie unter Ändern der Berechtigungen für einen Benutzer und unter der Beispielrichtlinie, die die Verwaltung von EC2 Sicherheitsgruppen ermöglicht, im IAMBenutzerhandbuch.

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und dann den Cluster aus, den Sie aktualisieren möchten. Dadurch wird die Cluster-Detailseite geöffnet. Die Registerkarte Eigenschaften auf dieser Seite sollte vorausgewählt sein.
3. Wählen Sie auf der Registerkarte Eigenschaften unter Netzwerk den Pfeil neben EC2Sicherheitsgruppen (Firewall) aus, um diesen Abschnitt zu erweitern. Wählen Sie unter Primärknoten den Link zur Sicherheitsgruppe aus. Daraufhin wird die EC2-Konsole geöffnet.
4. Wählen Sie die Registerkarte Eingehende Regeln und anschließend Eingehende Regeln bearbeiten aus.
5. Suchen Sie mit den folgenden Einstellungen nach einer Regel für eingehenden Datenverkehr, die öffentlichen Zugriff ermöglicht. Falls sie existiert, wählen Sie Löschen, um sie zu entfernen.

   - Typ

     SSH
   - Port

     22
   - Quelle

     Benutzerdefiniert 0.0.0.0/0

> **Warning**
>
> Vor Dezember 2020 verfügte die Sicherheitsgruppe ElasticMapReduce -master über eine vorkonfigurierte Regel, die eingehenden Datenverkehr auf Port 22 aus allen Quellen zuließ. Diese Regel wurde erstellt, um die ersten SSH Verbindungen zum Primärknoten zu vereinfachen. Wir empfehlen Ihnen dringend, diese Eingangsregel zu entfernen und den Datenverkehr auf vertrauenswürdige Quellen zu beschränken.

6. Scrollen Sie zum Ende der Regelliste und wählen Sie Regel hinzufügen.
7. Wählen Sie als Typ die Option aus SSH. Diese Auswahl wird automatisch TCPfür Protokoll und 22 für Portbereich eingegeben.
8. Wählen Sie als Quelle Meine IP aus, um Ihre IP-Adresse automatisch als Quelladresse hinzuzufügen. Sie können auch einen Bereich benutzerdefinierter vertrauenswürdiger Client-IP-Adressen hinzufügen oder zusätzliche Regeln für andere Clients erstellen. In vielen Netzwerkumgebungen werden IP-Adressen dynamisch zugewiesen, sodass Sie in Zukunft möglicherweise Ihre IP-Adressen für vertrauenswürdige Clients aktualisieren müssen.
9. Wählen Sie Save (Speichern) aus.
10. Kehren Sie optional zu Schritt 3 zurück, wählen Sie Core- und Aufgabenknoten aus, und wiederholen Sie die Schritte 4 bis 8. Dadurch wird den Kern- und Taskknoten SSH Client-Zugriff gewährt.

Connect zum Primärknoten her mit SSH

Secure Shell (SSH) ist ein Netzwerkprotokoll, mit dem Sie eine sichere Verbindung zu einem Remote-Computer herstellen können. Nach dem Verbinden verhält sich das Terminal auf Ihrem lokalen Computer so, als würde es auf dem Remote-Computer ausgeführt. Lokal erstellte Befehle werden auf dem Remote-Computer ausgeführt und die Befehlsausgabe vom Remote-Computer wird im Terminal-Fenster angezeigt.

Wenn Sie SSH with verwenden AWS, stellen Sie eine Verbindung zu einer EC2 Instanz her, bei der es sich um einen virtuellen Server handelt, der in der Cloud ausgeführt wird. Bei der Arbeit mit Amazon EMR SSH wird am häufigsten eine Verbindung zu der EC2 Instance hergestellt, die als primärer Knoten des Clusters fungiert.

Wenn Sie die Verbindung SSH zum primären Knoten herstellen, können Sie den Cluster überwachen und mit ihm interagieren. Sie können Linux-Befehle auf dem Primärknoten absetzen, Anwendungen wie Hive und Pig interaktiv ausführen, Verzeichnisse durchsuchen, Protokolldateien lesen usw. Sie können in Ihrer SSH Verbindung auch einen Tunnel erstellen, um die auf dem Primärknoten gehosteten Webschnittstellen anzuzeigen. Weitere Informationen finden Sie unter Auf EMR Amazon-Clustern gehostete Weboberflächen anzeigen.

Um eine Verbindung zum Primärknoten herzustellenSSH, benötigen Sie den öffentlichen DNS Namen des Primärknotens. Darüber hinaus muss die Sicherheitsgruppe, die dem Primärknoten zugeordnet ist, über eine Regel für eingehenden Datenverkehr SSH (TCPPort 22) von einer Quelle verfügen, zu der auch der Client gehört, von dem die SSH Verbindung stammt. Möglicherweise müssen Sie eine Regel hinzufügen, um eine SSH Verbindung von Ihrem Client aus zuzulassen. Weitere Informationen zum Ändern von Sicherheitsgruppenregeln finden Sie unter Steuerung des Netzwerkverkehrs mit Sicherheitsgruppen Regeln zu einer Sicherheitsgruppe hinzufügen im EC2Amazon-Benutzerhandbuch.

Rufen Sie den öffentlichen DNS Namen des primären Knotens ab

Sie können den primären öffentlichen DNS Namen über die EMR Amazon-Konsole und die abrufen AWS CLI.

Console

Um den öffentlichen DNS Namen des primären Knotens mit der neuen Konsole abzurufen

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters aus und wählen Sie dann den Cluster aus, für den Sie den öffentlichen DNS Namen abrufen möchten.
3. Notieren Sie sich den öffentlichen DNS Wert für den primären Knoten im Abschnitt Zusammenfassung der Cluster-Detailseite.

CLI

Um den öffentlichen DNS Namen des primären Knotens mit dem abzurufen AWS CLI

1. Geben Sie den folgenden Befehl ein, um die Cluster-Kennung abzurufen:

```
aws emr list-clusters
```

In der Ausgabe werden Ihre Cluster einschließlich des Clusters aufgeführtIDs. Notieren Sie die Cluster-ID für den Cluster, mit dem Sie eine Verbindung herstellen.

```
"Status": {
    "Timeline": {
        "ReadyDateTime": 1408040782.374,
        "CreationDateTime": 1408040501.213
    },
    "State": "WAITING",
    "StateChangeReason": {
        "Message": "Waiting after step completed"
    }
},
"NormalizedInstanceHours": 4,
"Id": "j-2AL4XXXXXX5T9",
"Name": "My cluster"
```

2. Um die Cluster-Instances einschließlich des öffentlichen DNS Namens für den Cluster aufzulisten, geben Sie einen der folgenden Befehle ein. Ersetzen *j-2AL4XXXXXX5T9* mit der Cluster-ID, die vom vorherigen Befehl zurückgegeben wurde.

```
aws emr list-instances --cluster-id j-2AL4XXXXXX5T9
```

Oder:

```
aws emr describe-cluster --cluster-id j-2AL4XXXXXX5T9
```

In der Ausgabe werden die Cluster-Instanzen einschließlich DNS Namen und IP-Adressen aufgeführt. Notieren Sie den Wert für `PublicDnsName`.

```
"Status": {
    "Timeline": {
        "ReadyDateTime": 1408040779.263,
        "CreationDateTime": 1408040515.535
    },
    "State": "RUNNING",
    "StateChangeReason": {}
```

```
},
"Ec2InstanceId": "i-e89b45e7",
"PublicDnsName": "ec2-###-##-##-###.us-west-2.compute.amazonaws.com"

"PrivateDnsName": "ip-###-##-##-###.us-west-2.compute.internal",
"PublicIpAddress": "##.###.###.##",
"Id": "ci-12XXXXXXXXFMH",
"PrivateIpAddress": "###.##.#.###"
```

Weitere Informationen finden Sie unter EMRAmazon-Befehle in der AWS CLI.

Stellen Sie unter Linux, Unix SSH und Mac OS X mithilfe eines EC2 privaten Amazon-Schlüssels eine Connect zum Primärknoten her

Um eine mit einer privaten Schlüsseldatei authentifizierte SSH Verbindung herzustellen, müssen Sie den privaten Schlüssel des EC2 Amazon-Schlüsselpaars angeben, wenn Sie einen Cluster starten. Weitere Informationen zum Zugriff auf Ihr key pair finden Sie unter EC2Amazon-Schlüsselpaare im EC2Amazon-Benutzerhandbuch.

Ihr Linux-Computer enthält höchstwahrscheinlich standardmäßig einen SSH Client. Open SSH ist beispielsweise auf den meisten Linux-, Unix- und MacOS-Betriebssystemen installiert. Sie können nach einem SSH Client suchen, indem Sie ihn ssh in der Befehlszeile eingeben. Wenn Ihr Computer den Befehl nicht erkennt, installieren Sie einen SSH Client, um eine Verbindung zum Primärknoten herzustellen. Das SSH Open-Projekt bietet eine kostenlose Implementierung der gesamten SSH Toolsuite. Weitere Informationen finden Sie auf der SSHOpen-Website.

Die folgenden Anweisungen zeigen das Öffnen einer SSH Verbindung zum EMR Amazon-Primärknoten unter Linux, Unix und Mac OS X.

So konfigurieren Sie Berechtigungen für die Datei mit dem privaten Schlüssel Ihres Schlüsselpaares

Bevor Sie den privaten Schlüssel Ihres EC2 Amazon-Schlüsselpaars verwenden können, um eine SSH Verbindung herzustellen, müssen Sie die Berechtigungen für die `.pem` Datei so einrichten, dass nur der Schlüsselinhaber Zugriff auf die Datei hat. Dies ist erforderlich, um eine SSH Verbindung mit dem Terminal oder dem herzustellen AWS CLI.

1. Stellen Sie sicher, dass Sie eingehenden SSH Verkehr zugelassen haben. Detaillierte Anweisungen finden Sie unter Bevor Sie eine Verbindung herstellen: Autorisieren des eingehenden Datenverkehrs.

2. Suchen Sie Ihre `.pem`-Datei. In dieser Anleitung wird davon ausgegangen, dass die Datei `mykeypair.pem` heißt und im Stammverzeichnis des aktuellen Benutzers gespeichert ist.
3. Geben Sie den folgenden Befehl ein, um die Berechtigungen festzulegen. Ersetzen *~/mykeypair.pem* mit dem vollständigen Pfad und Dateinamen der privaten Schlüsseldatei Ihres Schlüsselpaars. Zum Beispiel `C:/Users/<username>/.ssh/mykeypair.pem`.

   ```
   chmod 400 ~/mykeypair.pem
   ```

   Wenn Sie keine Berechtigungen für die `.pem`-Datei festlegen, erhalten Sie die Fehlermeldung, dass Ihre Schlüsseldatei nicht geschützt ist und der Schlüssel abgelehnt wird. Zum Verbinden müssen Sie die Berechtigungen für die Datei mit dem privaten Schlüssel Ihres Schlüsselpaares nur bei der ersten Verwendung festlegen.

So stellen Sie eine Verbindung mit dem Primärknoten mithilfe des Terminals her

1. Öffnen Sie ein Terminal-Fenster. Wählen Sie unter Mac OS X Applications > Utilities > Terminal (Anwendungen > Dienstprogramme > Terminal) aus. In anderen Linux-Distributionen befindet sich „Terminal“ in der Regel unter Applications > Accessories > Terminal (Anwendungen > Zubehör > Terminal).
2. Geben Sie den folgenden Befehl ein, um eine Verbindung mit dem Primärknoten herzustellen. Ersetzen *ec2-###-##-##-###.compute-1.amazonaws.com* mit dem primären öffentlichen DNS Namen Ihres Clusters und ersetzen *~/mykeypair.pem* mit dem vollständigen Pfad und Dateinamen Ihrer `.pem` Datei. Zum Beispiel `C:/Users/<username>/.ssh/mykeypair.pem`.

   ```
   ssh hadoop@ec2-###-##-##-###.compute-1.amazonaws.com -i ~/mykeypair.pem
   ```

   > **Important**
   >
   > Sie müssen den Anmeldenamen verwendenhadoop, wenn Sie sich mit dem EMR primären Amazon-Node verbinden. Andernfalls wird möglicherweise ein Fehler ähnlich dem folgenden angezeigt`Server refused our key`.

3. Es wird die Warnung angezeigt, dass die Authentizität des Hosts, mit dem Sie eine Verbindung herstellen, nicht überprüft werden konnte. Geben Sie `yes` ein, um fortzufahren.
4. Wenn Sie mit der Arbeit am Primärknoten fertig sind, geben Sie den folgenden Befehl ein, um die SSH Verbindung zu schließen.

```
exit
```

Wenn Sie Probleme bei der Verbindung mit SSH Ihrem primären Knoten haben, finden Sie weitere Informationen unter Problembehandlung beim Herstellen einer Verbindung zu Ihrer Instance.

Stellen Sie unter Windows eine Connect SSH zum Primärknoten her

Windows-Benutzer können einen SSH Client wie Pu verwendenTTY, um eine Verbindung zum Primärknoten herzustellen. Bevor Sie eine Verbindung zum EMR Amazon-Primärknoten herstellen, sollten Sie Pu und P herunterladen TTY und installierenuTTYgen. Sie können diese Tools von der TTYPu-Download-Seite herunterladen.

Pu TTY unterstützt das von Amazon EC2 generierte Schlüsselpaar-Dateiformat für private Schlüssel (`.pem`) nicht nativ. Sie verwenden PuTTYgen , um Ihre Schlüsseldatei in das erforderliche TTY Pu-Format (`.ppk`) zu konvertieren. Sie müssen Ihren Schlüssel in dieses Format (`.ppk`) konvertieren, bevor Sie versuchen, mit Pu eine Verbindung zum Primärknoten herzustellenTTY.

Weitere Informationen zur Konvertierung Ihres Schlüssels finden Sie unter Konvertieren Ihres privaten Schlüssels mit P uTTYgen im EC2Amazon-Benutzerhandbuch.

So stellen Sie mit Pu eine Verbindung zum Primärknoten her TTY

1. Stellen Sie sicher, dass Sie eingehenden SSH Datenverkehr zugelassen haben. Detaillierte Anweisungen finden Sie unter Bevor Sie eine Verbindung herstellen: Autorisieren des eingehenden Datenverkehrs.
2. Öffnen Sie `putty.exe`. Sie können Pu auch TTY von der Windows-Programmliste aus starten.
3. Falls erforderlich, wählen Sie in der Category (Kategorie)-Liste Session (Sitzung) aus.
4. Geben Sie als Hostname (oder IP-Adresse) Folgendes ein `hadoop@`*`MasterPublicDNS`*. Zum Beispiel: `hadoop@`*`ec2-###-##-##-###.compute-1.amazonaws.com`*.
5. Wählen Sie in der Kategorienliste die Option Verbindung > SSH, Auth aus.
6. Klicken Sie bei Private key file for authentication (Private Schlüsseldatei für Authentifizierung) auf Browse (Durchsuchen), und wählen Sie die `.ppk`-Datei aus, die Sie generiert haben.
7. Wählen Sie Öffnen und dann Ja, um die TTY Pu-Sicherheitswarnung zu schließen.

> **Important**
>
> Wenn Sie sich beim Primärknoten anmelden und zur Angabe eines Benutzernamens aufgefordert werden, geben Sie `hadoop` ein.

8. Wenn Sie mit der Arbeit am Primärknoten fertig sind, können Sie die SSH Verbindung schließen, indem Sie Pu TTY schließen.

> **Note**
>
> Um zu verhindern, dass bei der SSH Verbindung ein Timeout auftritt, können Sie in der Kategorienliste die Option Verbindung auswählen und die Option Enable TCP _keepalives auswählen. Wenn Sie eine aktive SSH Sitzung in Pu habenTTY, können Sie Ihre Einstellungen ändern, indem Sie den Kontext (Rechtsklick) für die TTY Pu-Titelleiste öffnen und Einstellungen ändern wählen.

Wenn Sie Probleme bei der Verbindung mit SSH Ihrem primären Knoten haben, finden Sie weitere Informationen unter Problembehandlung beim Herstellen einer Verbindung zu Ihrer Instance.

Mit dem Primärknoten über die AWS CLI verbinden

Sie können unter Windows sowie unter Linux, Unix und Mac OS X eine SSH Verbindung mit dem Primärknoten herstellen. Unabhängig von der Plattform benötigen Sie den öffentlichen DNS Namen des Primärknotens und Ihren privaten Schlüssel für das EC2 Amazon-Schlüsselpaar. AWS CLI Wenn Sie den AWS CLI unter Linux, Unix oder Mac OS X verwenden, müssen Sie auch die Berechtigungen für die Datei mit dem privaten Schlüssel (`.pem`oder der Datei`.ppk`) festlegen, wie unter beschriebenSo konfigurieren Sie Berechtigungen für die Datei mit dem privaten Schlüssel Ihres Schlüsselpaares.

Um eine Verbindung zum Primärknoten herzustellen, verwenden Sie AWS CLI

1. Stellen Sie sicher, dass Sie eingehenden SSH Datenverkehr zugelassen haben. Detaillierte Anweisungen finden Sie unter Bevor Sie eine Verbindung herstellen: Autorisieren des eingehenden Datenverkehrs.
2. Geben Sie Folgendes ein, um die Cluster-Kennung abzurufen:

```
aws emr list-clusters
```

In der Ausgabe werden Ihre Cluster einschließlich des Clusters IDs aufgeführt. Notieren Sie die Cluster-ID für den Cluster, mit dem Sie eine Verbindung herstellen.

```
"Status": {
    "Timeline": {
        "ReadyDateTime": 1408040782.374,
        "CreationDateTime": 1408040501.213
    },
    "State": "WAITING",
    "StateChangeReason": {
        "Message": "Waiting after step completed"
    }
},
"NormalizedInstanceHours": 4,
"Id": "j-2AL4XXXXXX5T9",
"Name": "AWS CLI cluster"
```

3. Geben Sie den folgenden Befehl ein, um eine SSH Verbindung zum primären Knoten herzustellen. Ersetzen Sie im folgenden Beispiel *j-2AL4XXXXXX5T9* durch die Cluster-ID und ersetzen *~/mykeypair.key* mit dem vollständigen Pfad und Dateinamen Ihrer `.pem` Datei (für Linux, Unix und Mac OS X) oder `.ppk` Datei (für Windows). Zum Beispiel `C:\Users\<username>\.ssh\mykeypair.pem`.

   ```
   aws emr ssh --cluster-id j-2AL4XXXXXX5T9 --key-pair-file ~/mykeypair.key
   ```

4. Wenn Sie mit der Arbeit am Primärknoten fertig sind, schließen Sie das AWS CLI Fenster.

   Weitere Informationen finden Sie unter EMRAmazon-Befehle in der AWS CLI. Wenn Sie Probleme bei der Verbindung mit SSH Ihrem primären Knoten haben, finden Sie weitere Informationen unter Problembehandlung beim Herstellen einer Verbindung zu Ihrer Instance.

EMRAmazon-Serviceanschlüsse

> **Note**
>
> Im Folgenden finden Sie Schnittstellen und Serviceports für Komponenten bei AmazonEMR. Dies ist keine vollständige Liste der Serviceports. Nicht standardmäßige Dienste, wie SSL Ports und verschiedene Arten von Protokollen, sind nicht aufgeführt.

> **Important**
>
> Seien Sie vorsichtig, wenn Sie Regeln für offene Ports für Sicherheitsgruppen bearbeiten. Stellen Sie sicher, dass Sie Regeln hinzufügen, die nur Datenverkehr von vertrauenswürdigen und authentifizierten Clients für die Protokolle und Ports zulassen, die zum Ausführen Ihrer Workloads erforderlich sind.

Komponente	Service description (Service-Beschreibung)	Der Service wird standardmäßig ausgeführt	Port	Schlüssel zur Konfiguration
Hadoop	HTTP KMS REST API	Ja	9600	hadoop.kms.http.port
HDFS	Namenode-Web-Benutzeroberfläche	Ja	9870	dfs.namenode.http-address
	Namensknoten RPC	Ja	8020	dfs.namenode.rpc-address
	DataNode Web-Benutzeroberfläche	Ja	9864	dfs.datanode.http.address
	Datanode HTTP für die Datenübertragung	Ja	9866	dfs.datanode.address

Komponente	Service description (Service-Beschreibung)	Der Service wird standardmäßig ausgeführt	Port	Schlüssel zur Konfiguration
	Datanode RPC für die Datenübertragung	Ja	9867	dfs.datanode.ipc.address
Hive	HiveServer2 Sparsamkeit	Ja	10000	hive.server2.thrift.port
	HiveServer2 HTTP	Nein	10001	hive.server2.thrift.http.port
	HiveServer2 Webbenutzeroberfläche	Ja	10002	hive.server2.webui.port
	Hive Metastore	Ja	9083	hive.metastore.port / metastore.thrift.port
	W ebHCat	Nein	50111	templeton.port
	LLAPDaemon-Verwaltungsdienst () RPC	Nein	15004	hive.llap.management.rpc.port
	YARNShuffle-Port für LLAP auf -daemon gehostetes Shuffle	Nein	15551	hive.llap.daemon.yarn.shuffle.port
	Der LLAP Dämon RPC	Nein	Dynamisch	hive.llap.daemon.rpc.port
	LLAPDaemon-Webbenutzeroberfläche	Nein	15002	hive.llap.daemon.web.port

Komponente	Service description (Service-Beschreibung)	Der Service wird standardmäßig ausgeführt	Port	Schlüssel zur Konfiguration
	LLAPDienst für die Ausgabe eines Daemons	Nein	15003	hive.llap.daemon.output.service.port
Oozie		Ja	11000	
Tez	Tez UI	Ja	8080	
YARN	Shuffle	Ja	13562	mapreduce.shuffle.port
	Lokalisierer RPC	Ja	8040	yarn.nodemanager.localizer.address
		Ja	8041	
	NM-Webapp-Adresse	Ja	8042	yarn.nodemanager.webapp.address
	RM-Webanwendung	Ja	8088	yarn.resourcemanager.webapp.address
		Ja	8025	
	Scheduler	Ja	8030	yarn.resourcemanager.scheduler.address

Komponente	Service description (Service-Beschreibung)	Der Service wird standardmäßig ausgeführt	Port	Schlüssel zur Konfiguration
	Schnittstelle für den Anwendungsmanager	Ja	8032	yarn.resourcemanager.address
	RM-Admin-Oberfläche	Ja	8033	yarn.resourcemanager.admin.address
	JobHistory Web-Benutzeroberfläche des Servers	Ja	1988	mapreduce.jobhistory.webapp.address
	JobHistory Webbenutzeroberfläche für Serveradministratoren	Ja	10033	mapreduce.jobhistory.admin.adresse
	JobHistory Server () RPC	Ja	10020	mapreduce.jobhistory.address
	Timeline-Server für Anwendungen () RPC	Ja	10200	garn.timeline-service.adresse
	HTTPWeb-Benutzeroberfläche für den Anwendungszeitplanserver	Ja	8188	garn.timeline-service.webapp.adresse

Komponente	Service description (Service-Beschreibung)	Der Service wird standardmäßig ausgeführt	Port	Schlüssel zur Konfiguration
	HTTPSWeb-UI für den Anwendungszeitleistenserver	Nein	8190	yarn.timeline-service.webapp.https.address
		Ja	20888	
Zookeeper	Client-Port	Ja	2181	
		Ja	37301	
		Ja	8341	

Auf EMR Amazon-Clustern gehostete Weboberflächen anzeigen

> **Important**
>
> Sie können eine benutzerdefinierte Sicherheitsgruppe konfigurieren, um den eingehenden Zugriff auf diese Webschnittstellen zu ermöglichen. Beachten Sie, dass jeder Port, an dem Sie eingehenden Datenverkehr zulassen, eine potenzielle Sicherheitslücke darstellt. Überprüfen Sie sorgfältig die benutzerdefinierten Sicherheitsgruppen, um Schwachstellen zu minimieren. Weitere Informationen finden Sie unter Steuerung des Netzwerkverkehrs mit Sicherheitsgruppen.

Hadoop und andere Anwendungen, die Sie auf Ihrem EMR Cluster installieren, veröffentlichen Benutzeroberflächen als Websites, die auf dem primären Knoten gehostet werden. Aus Sicherheitsgründen sind diese Websites bei der Verwendung von Amazon EMR Managed Security Groups nur auf dem lokalen Webserver des primären Knotens verfügbar. Aus diesem Grund müssen Sie eine Verbindung zum Primärknoten herstellen, um die Weboberflächen anzeigen zu können. Weitere Informationen finden Sie unter Connect zum Primärknoten her mit SSH. Hadoop veröffentlicht Benutzeroberflächen auch als Websites, die auf Core- und Aufgabenknoten gehostet werden. Diese Websites sind ebenfalls nur auf dem lokalen Webserver auf dem Knoten verfügbar.

Die folgende Tabelle enthält die Webschnittstellen, die Sie auf Cluster-Instances anzeigen lassen können: Diese Hadoop-Schnittstellen sind in allen Clustern verfügbar. Ersetzen Sie für die Master-Instance-Schnittstellen *master-public-dns-name* wobei der Master Public auf der Registerkarte Cluster-Zusammenfassung in der EMR Amazon-Konsole DNS aufgeführt ist. Für Core- und Task-Instance-Schnittstellen ersetzen Sie *coretask-public-dns-name* durch den für die Instanz aufgeführten öffentlichen DNS Namen. Um den öffentlichen DNS Namen einer Instance zu finden, wählen Sie in der EMR Amazon-Konsole Ihren Cluster aus der Liste aus, wählen Sie den Tab Hardware, wählen Sie die ID der Instance-Gruppe, die die Instance enthält, zu der Sie eine Verbindung herstellen möchten, und notieren Sie sich dann den öffentlichen DNS Namen, der für die Instance aufgeführt ist.

Name der Schnittstelle	URI
Flink History Server (EMRVersion 5.33 und höher)	http://*master-public-dns-name* :8082/
Ganglia	http://*master-public-dns-name* /Ganglien/
Hadoop HDFS NameNode (Version vor 6.x) EMR	https://*master-public-dns-name* :50470/
Hadoop HDFS NameNode	http://*master-public-dns-name* :50070/
Hadoop HDFS DataNode	http://*coretask-public-dns-name* :50075/
Hadoop HDFS NameNode (Version 6.x) EMR	https://*master-public-dns-name* :9870/
Hadoop HDFS DataNode (Version vor 6.x) EMR	https://*coretask-public-dns-name* :50475/
Hadoop HDFS DataNode (Version 6.x) EMR	https://*coretask-public-dns-name* :9865/
HBase	http://*master-public-dns-name* :16010/
Hue	http://*master-public-dns-name* :8888/
JupyterHub	https://*master-public-dns-name* :9443/

Name der Schnittstelle	URI
Livy	http://*master-public-dns-name* :8998/
Funke HistoryServer	http://*master-public-dns-name* :18080/
Tez	http://*master-public-dns-name* :8080/tez-ui
YARN NodeManager	http://*coretask-public-dns-name* :8042/
YARN ResourceManager	http://*master-public-dns-name* :8088/
Zeppelin	http://*master-public-dns-name* :8890/

Da auf dem Primärknoten mehrere anwendungsspezifische Schnittstellen verfügbar sind, die auf den Core- und Task-Knoten nicht verfügbar sind, beziehen sich die Anweisungen in diesem Dokument speziell auf den EMR Amazon-Primärknoten. Auf die Webschnittstellen im Core- und Aufgabenknoten kann auf die gleiche Weise zugegriffen werden wie auf die Webschnittstellen im Primärknoten.

Es gibt mehrere Möglichkeiten, auf die Webschnittstellen im Primärknoten zuzugreifen. Die einfachste und schnellste Methode besteht darin, eine Verbindung SSH zum Primärknoten herzustellen und den textbasierten Browser Lynx zu verwenden, um die Websites in Ihrem Client anzuzeigen. SSH Lynx ist jedoch ein textbasierter Browser mit einer eingeschränkten Benutzeroberfläche, die keine Grafiken anzeigen kann. Das folgende Beispiel zeigt, wie Sie die ResourceManager Hadoop-Schnittstelle mit Lynx öffnen (Lynx URLs werden auch bereitgestellt, wenn Sie sich mit) am Primärknoten anmelden. SSH

```
lynx http://ip-###-##-##-###.us-west-2.compute.internal:8088/
```

Es gibt zwei verbleibende Optionen für den Zugriff auf Webschnittstellen im Primärknoten, die vollständige Browserfunktionalität bieten. Wählen Sie eine der folgenden Optionen aus:

- Option 1 (empfohlen für technisch versierte Benutzer): Verwenden Sie einen SSH Client, um eine Verbindung zum primären Knoten herzustellen, konfigurieren Sie SSH Tunneling mit lokaler Portweiterleitung und verwenden Sie einen Internetbrowser, um Webschnittstellen zu öffnen, die auf dem primären Knoten gehostet werden. Mit dieser Methode können Sie den Zugriff auf die Weboberfläche konfigurieren, ohne einen SOCKS Proxy zu verwenden.

- Option 2 (für neue Benutzer empfohlen): Verwenden Sie einen SSH Client, um eine Verbindung zum primären Knoten herzustellen, konfigurieren Sie SSH Tunneling mit dynamischer Portweiterleitung und konfigurieren Sie Ihren Internetbrowser so, dass er ein Add-on wie FoxyProxy für Firefox oder SwitchyOmega Chrome verwendet, um Ihre SOCKS Proxyeinstellungen zu verwalten. Mit dieser Methode können Sie automatisch URLs anhand von Textmustern filtern und die Proxyeinstellungen auf Domänen beschränken, die der Form des Namens des primären Knotens DNS entsprechen. Weitere Informationen zur Konfiguration FoxyProxy für Firefox und Google Chrome finden Sie unterOption 2, Teil 2: Proxy-Einstellungen konfigurieren, um auf dem Primärknoten gehostete Websites anzeigen zu lassen.

> **Note**
>
> Wenn Sie den Port, auf dem eine Anwendung ausgeführt wird, über die Cluster-Konfiguration ändern, wird der Hyperlink zum Port in der EMR Amazon-Konsole nicht aktualisiert. Das liegt daran, dass die Konsole nicht über die Funktionalität verfügt, die Konfiguration `server.port` zu lesen.

Mit EMR Amazon-Version 5.25.0 oder höher können Sie von der Konsole aus auf die Benutzeroberfläche des Spark-History-Servers zugreifen, ohne einen Web-Proxy über eine SSH Verbindung einrichten zu müssen. Weitere Informationen finden Sie unter Zugriff auf den persistenten Spark History Server mit nur einem Klick.

Themen

Option 1: Richten Sie mithilfe der lokalen SSH Portweiterleitung einen Tunnel zum primären Knoten ein

Um eine Verbindung zum lokalen Webserver auf dem Primärknoten herzustellen, erstellen Sie einen SSH Tunnel zwischen Ihrem Computer und dem Primärknoten. Dies wird auch als Port-Weiterleitung

bezeichnet. Wenn Sie keinen SOCKS Proxy verwenden möchten, können Sie mithilfe der lokalen Portweiterleitung einen SSH Tunnel zum Primärknoten einrichten. Bei der lokalen Port-Weiterleitung geben Sie ungenutzte lokale Ports an, die zum Weiterleiten von Datenverkehr zu bestimmten Remote-Ports auf dem lokalen Webserver des Primärknotens verwendet werden.

Für die Einrichtung eines SSH Tunnels mithilfe der lokalen Portweiterleitung sind der öffentliche DNS Name des primären Knotens und die private Schlüsseldatei Ihres Schlüsselpaars erforderlich. Hinweise zum Auffinden des öffentlichen DNS Hauptnamens finden Sie unterRufen Sie den öffentlichen DNS Namen des primären Knotens ab. Weitere Informationen zum Zugriff auf Ihr key pair finden Sie unter EC2Amazon-Schlüsselpaare im EC2Amazon-Benutzerhandbuch. Weitere Informationen zu den Websites, die Sie sich auf dem Primärknoten ansehen können, finden Sie unter Auf EMR Amazon-Clustern gehostete Weboberflächen anzeigen.

Richten Sie mithilfe der lokalen Portweiterleitung mit Open einen SSH Tunnel zum primären Knoten ein SSH

Um einen SSH Tunnel mithilfe der lokalen Portweiterleitung im Terminal einzurichten

1. Stellen Sie sicher, dass Sie eingehenden SSH Verkehr zugelassen haben. Detaillierte Anweisungen finden Sie unter Bevor Sie eine Verbindung herstellen: Autorisieren des eingehenden Datenverkehrs.
2. Öffnen Sie ein Terminal-Fenster. Wählen Sie unter Mac OS X Applications > Utilities > Terminal (Anwendungen > Dienstprogramme > Terminal) aus. In anderen Linux-Distributionen befindet sich „Terminal“ in der Regel unter Applications > Accessories > Terminal (Anwendungen > Zubehör > Terminal).
3. Geben Sie den folgenden Befehl ein, um einen SSH Tunnel auf Ihrem lokalen Computer zu öffnen. Dieser Beispielbefehl greift auf die ResourceManager Weboberfläche zu, indem er den Datenverkehr auf dem lokalen Port 8157 (einem zufällig ausgewählten ungenutzten lokalen Port) an den Port 8088 auf dem lokalen Webserver des Master-Knotens weiterleitet.

   Ersetzen Sie im Befehl *~/mykeypair.pem* mit dem Speicherort und dem Dateinamen Ihrer `.pem` Datei und ersetzen *ec2-###-##-##-###.compute-1.amazonaws.com* mit dem öffentlichen DNS Master-Namen Ihres Clusters. Um auf eine andere Weboberfläche zuzugreifen, `8088` ersetzen Sie diese durch die entsprechende Portnummer. Ersetzen Sie beispielsweise `8088` durch `8890` für die Zeppelin-Schnittstelle.

```
ssh -i ~/mykeypair.pem -N -L 8157:ec2-###-##-##-
###.compute-1.amazonaws.com:8088 hadoop@ec2-###-##-##-###.compute-1.amazonaws.com
```

`-L` bezeichnet die Verwendung der lokalen Port-Weiterleitung. Damit können Sie einen lokalen Port für die Weiterleitung von Datenverkehr zu einem bestimmten Remote-Port auf dem lokalen Webserver des Hauptknotens angeben.

Nachdem Sie diesen Befehl ausgeführt haben, bleibt das Terminal geöffnet und gibt keine Antwort zurück.

4. Um die ResourceManager Weboberfläche in Ihrem Browser zu öffnen, geben Sie `http://localhost:8157/` in die Adressleiste ein.
5. Wenn Sie die Arbeit mit den Webschnittstellen im Primärknoten beendet haben, schließen Sie die Terminal-Fenster.

Option 2, Teil 1: Richten Sie mithilfe dynamischer Portweiterleitung einen SSH Tunnel zum primären Knoten ein

Um eine Verbindung zum lokalen Webserver auf dem Primärknoten herzustellen, erstellen Sie einen SSH Tunnel zwischen Ihrem Computer und dem Primärknoten. Dies wird auch als Port-Weiterleitung bezeichnet. Wenn Sie Ihren SSH Tunnel mithilfe der dynamischen Portweiterleitung erstellen, wird der gesamte Datenverkehr, der an einen angegebenen ungenutzten lokalen Port geleitet wird, an den lokalen Webserver auf dem Primärknoten weitergeleitet. Dadurch wird ein SOCKS Proxy erstellt. Anschließend können Sie Ihren Internetbrowser so konfigurieren, dass er ein Add-on verwendet FoxyProxy oder SwitchyOmega Ihre SOCKS Proxyeinstellungen verwaltet.

Mithilfe eines Proxy-Management-Add-ons können Sie automatisch URLs anhand von Textmustern filtern und die Proxyeinstellungen auf Domänen beschränken, die der Form des öffentlichen DNS Namens des primären Knotens entsprechen. Das Browser-Add-On aktiviert und deaktiviert den Proxy automatisch, wenn Sie zwischen den auf dem Primärknoten gehosteten Websites und solchen im Internet wechseln.

Bevor Sie beginnen, benötigen Sie den öffentlichen DNS Namen des primären Knotens und Ihre private Schlüsseldatei für das key pair. Hinweise zum Auffinden des primären öffentlichen DNS Namens finden Sie unterRufen Sie den öffentlichen DNS Namen des primären Knotens ab. Weitere Informationen zum Zugriff auf Ihr key pair finden Sie unter EC2Amazon-Schlüsselpaare im EC2Amazon-Benutzerhandbuch. Weitere Informationen zu den Websites, die Sie sich auf dem Primärknoten ansehen können, finden Sie unter Auf EMR Amazon-Clustern gehostete Weboberflächen anzeigen.

Richten Sie mithilfe der dynamischen Portweiterleitung mit Open einen SSH Tunnel zum primären Knoten ein SSH

So richten Sie einen SSH Tunnel mit dynamischer Portweiterleitung mit Open ein SSH

1. Stellen Sie sicher, dass Sie eingehenden SSH Datenverkehr zugelassen haben. Detaillierte Anweisungen finden Sie unter Bevor Sie eine Verbindung herstellen: Autorisieren des eingehenden Datenverkehrs.
2. Öffnen Sie ein Terminal-Fenster. Wählen Sie unter Mac OS X Applications > Utilities > Terminal (Anwendungen > Dienstprogramme > Terminal) aus. In anderen Linux-Distributionen befindet sich „Terminal“ in der Regel unter Applications > Accessories > Terminal (Anwendungen > Zubehör > Terminal).
3. Geben Sie den folgenden Befehl ein, um einen SSH Tunnel auf Ihrem lokalen Computer zu öffnen. Ersetzen *~/mykeypair.pem* durch den Speicherort und den Dateinamen Ihrer `.pem` Datei ersetzen *8157* durch eine unbenutzte, lokale Portnummer und ersetzen *ec2-###-##-##-###.compute-1.amazonaws.com* mit dem primären öffentlichen DNS Namen Ihres Clusters.

   ```
   ssh -i ~/mykeypair.pem -N -D 8157 hadoop@ec2-###-##-##-###.compute-1.amazonaws.com
   ```

   Nachdem Sie diesen Befehl ausgeführt haben, bleibt das Terminal geöffnet und gibt keine Antwort zurück.

   > **Note**
   >
   > `-D` bezeichnet die Verwendung der dynamischen Port-Weiterleitung. Damit können Sie einen lokalen Port für die Weiterleitung von Datenverkehr zu allen Remote-Ports auf dem lokalen Webserver des Primärknotens angeben. Durch die dynamische Portweiterleitung wird ein lokaler SOCKS Proxy erstellt, der den im Befehl angegebenen Port überwacht.

4. Nachdem der Tunnel aktiv ist, konfigurieren Sie einen SOCKS Proxy für Ihren Browser. Weitere Informationen finden Sie unter Option 2, Teil 2: Proxy-Einstellungen konfigurieren, um auf dem Primärknoten gehostete Websites anzeigen zu lassen.
5. Wenn Sie die Arbeit mit den Webschnittstellen im Primärknoten beendet haben, schließen Sie das Terminal-Fenster.

Richten Sie einen SSH Tunnel mit dynamischer Portweiterleitung mit dem ein AWS CLI

Sie können unter Windows sowie unter Linux, Unix und Mac OS X eine SSH Verbindung mit dem Primärknoten herstellen. Wenn Sie den AWS CLI unter Linux, Unix oder Mac OS X verwenden, müssen Sie die Berechtigungen für die `.pem` Datei festlegen, wie unter beschriebenSo konfigurieren Sie Berechtigungen für die Datei mit dem privaten Schlüssel Ihres Schlüsselpaares. AWS CLI Wenn Sie den AWS CLI unter Windows verwenden, TTY muss Pu in der Umgebungsvariablen path erscheinen. Andernfalls wird möglicherweise eine Fehlermeldung wie Öffnen SSH oder Pu TTY nicht verfügbar angezeigt.

Um einen SSH Tunnel mit dynamischer Portweiterleitung einzurichten, verwenden Sie AWS CLI

1. Stellen Sie sicher, dass Sie eingehenden SSH Datenverkehr zugelassen haben. Detaillierte Anweisungen finden Sie unter Bevor Sie eine Verbindung herstellen: Autorisieren des eingehenden Datenverkehrs.
2. Stellen Sie eine SSH Verbindung mit dem primären Knoten her, wie unter gezeigt. Mit dem Primärknoten über die AWS CLI verbinden
3. Geben Sie Folgendes ein, um die Cluster-Kennung abzurufen:

   ```
   aws emr list-clusters
   ```

   In der Ausgabe werden Ihre Cluster einschließlich des Clusters aufgeführtIDs. Notieren Sie die Cluster-ID für den Cluster, mit dem Sie eine Verbindung herstellen.

   ```
   "Status": {
       "Timeline": {
           "ReadyDateTime": 1408040782.374,
           "CreationDateTime": 1408040501.213
       },
       "State": "WAITING",
       "StateChangeReason": {
           "Message": "Waiting after step completed"
       }
   },
   "NormalizedInstanceHours": 4,
   "Id": "j-2AL4XXXXXX5T9",
   "Name": "AWS CLI cluster"
   ```

4. Geben Sie den folgenden Befehl ein, um mithilfe der dynamischen Portweiterleitung einen SSH Tunnel zum primären Knoten zu öffnen. Ersetzen Sie im folgenden Beispiel *j-2AL4XXXXXX5T9*

durch die Cluster-ID und ersetzen *~/mykeypair.key* mit dem Speicherort und dem Dateinamen Ihrer `.pem` Datei (für Linux, Unix und Mac OS X) oder `.ppk` Datei (für Windows).

```
aws emr socks --cluster-id j-2AL4XXXXXX5T9 --key-pair-file ~/mykeypair.key
```

> **Note**
>
> Mit dem Socks-Befehl wird die dynamische Port-Weiterleitung am lokalen Port 8157 automatisch konfiguriert. Derzeit kann diese Einstellung nicht geändert werden.

5. Nachdem der Tunnel aktiv ist, konfigurieren Sie einen SOCKS Proxy für Ihren Browser. Weitere Informationen finden Sie unter Option 2, Teil 2: Proxy-Einstellungen konfigurieren, um auf dem Primärknoten gehostete Websites anzeigen zu lassen.
6. Wenn Sie mit der Arbeit mit den Webschnittstellen auf dem primären Knoten fertig sind, schließen Sie das AWS CLI Fenster.

   Weitere Informationen zur Verwendung von EMR Amazon-Befehlen in der AWS CLI finden Sie unterhttps://docs.aws.amazon.com/cli/latest/reference/emr.

Richten Sie mit Pu einen SSH Tunnel zum Primärknoten ein TTY

Windows-Benutzer können einen SSH Client wie Pu verwendenTTY, um einen SSH Tunnel zum Primärknoten zu erstellen. Bevor Sie eine Verbindung zum EMR Amazon-Primärknoten herstellen, sollten Sie Pu und P herunterladen TTY und installierenuTTYgen. Sie können diese Tools von der TTYPu-Download-Seite herunterladen.

Pu TTY unterstützt das von Amazon EC2 generierte Schlüsselpaar-Dateiformat für private Schlüssel (`.pem`) nicht nativ. Sie verwenden PuTTYgen , um Ihre Schlüsseldatei in das erforderliche TTY Pu-Format (`.ppk`) zu konvertieren. Sie müssen Ihren Schlüssel in dieses Format (`.ppk`) konvertieren, bevor Sie versuchen, mit Pu eine Verbindung zum Primärknoten herzustellenTTY.

Weitere Informationen zur Konvertierung Ihres Schlüssels finden Sie unter Konvertieren Ihres privaten Schlüssels mit P uTTYgen im EC2Amazon-Benutzerhandbuch.

So richten Sie einen SSH Tunnel mit dynamischer Portweiterleitung mit Pu ein TTY

1. Stellen Sie sicher, dass Sie eingehenden SSH Verkehr zugelassen haben. Detaillierte Anweisungen finden Sie unter Bevor Sie eine Verbindung herstellen: Autorisieren des eingehenden Datenverkehrs.
2. Doppelklicken Sie`putty.exe`, um Pu TTY zu starten. Sie können Pu auch TTY von der Windows-Programmliste aus starten.

   > **Note**
   >
   > Wenn Sie bereits eine aktive SSH Sitzung mit dem primären Knoten haben, können Sie einen Tunnel hinzufügen, indem Sie mit der rechten Maustaste auf die TTY Pu-Titelleiste klicken und Einstellungen ändern wählen.

3. Falls erforderlich, wählen Sie in der Category (Kategorie)-Liste Session (Sitzung) aus.
4. Geben Sie im Feld Hostname Folgendes ein **hadoop@***MasterPublicDNS*. Zum Beispiel: **hadoop@***ec2-###-##-##-###.compute-1.amazonaws.com*.
5. Erweitern Sie in der Kategorienliste den Eintrag Verbindung > SSH und wählen Sie dann Auth aus.
6. Klicken Sie bei Private key file for authentication (Private Schlüsseldatei für Authentifizierung) auf Browse (Durchsuchen), und wählen Sie die `.ppk`-Datei aus, die Sie generiert haben.

   > **Note**
   >
   > Pu TTY unterstützt das von Amazon EC2 generierte Schlüsselpaar-Dateiformat für private Schlüssel (`.pem`) nicht nativ. Sie verwenden PuTTYgen , um Ihre Schlüsseldatei in das erforderliche TTY Pu-Format (`.ppk`) zu konvertieren. Sie müssen Ihren Schlüssel in dieses Format (`.ppk`) konvertieren, bevor Sie versuchen, mit Pu eine Verbindung zum Primärknoten herzustellenTTY.

7. Erweitern Sie in der Kategorienliste den Eintrag Verbindung > SSH und wählen Sie dann Tunnel aus.
8. Geben Sie im Feld Quellport `8157` (einen nicht verwendeten lokalen Port) ein und wählen Sie dann Hinzufügen aus.
9. Lassen Sie das Feld Destination (Zieladresse) leer.
10. Wählen Sie die Optionen Dynamic (Dynamisch) und Auto.

11. Klicken Sie auf Open.
12. Wählen Sie Ja, um die TTY Pu-Sicherheitswarnung zu schließen.

> **Important**
>
> Wenn Sie sich beim Primärknoten anmelden, geben Sie `hadoop` ein, wenn Sie zur Angabe eines Benutzernamens aufgefordert werden.

13. Nachdem der Tunnel aktiv ist, konfigurieren Sie einen SOCKS Proxy für Ihren Browser. Weitere Informationen finden Sie unter Option 2, Teil 2: Proxy-Einstellungen konfigurieren, um auf dem Primärknoten gehostete Websites anzeigen zu lassen.
14. Wenn Sie mit der Arbeit mit den Webschnittstellen auf dem Primärknoten fertig sind, schließen Sie das TTY Pu-Fenster.

Option 2, Teil 2: Proxy-Einstellungen konfigurieren, um auf dem Primärknoten gehostete Websites anzeigen zu lassen

Wenn Sie einen SSH Tunnel mit dynamischer Portweiterleitung verwenden, müssen Sie ein SOCKS Proxy-Management-Add-on verwenden, um die Proxy-Einstellungen in Ihrem Browser zu steuern. Mithilfe eines SOCKS Proxy-Management-Tools können Sie automatisch URLs anhand von Textmustern filtern und die Proxyeinstellungen auf Domänen beschränken, die der Form des öffentlichen DNS Namens des primären Knotens entsprechen. Das Browser-Add-On aktiviert und deaktiviert den Proxy automatisch, wenn Sie zwischen den auf dem Primärknoten gehosteten Websites und solchen im Internet wechseln. Um Ihre Proxyeinstellungen zu verwalten, konfigurieren Sie Ihren Browser so, dass er ein Add-on wie FoxyProxy oder verwendet SwitchyOmega.

Weitere Informationen zum Erstellen eines SSH Tunnels finden Sie unterOption 2, Teil 1: Richten Sie mithilfe dynamischer Portweiterleitung einen SSH Tunnel zum primären Knoten ein. Weitere Informationen zu den verfügbaren Webschnittstellen finden Sie unter Auf EMR Amazon-Clustern gehostete Weboberflächen anzeigen.

Geben Sie bei der Einrichtung Ihres Proxy-Add-ons die folgenden Einstellungen an:

- Verwenden Sie localhost als Hostadresse.
- Verwenden Sie dieselbe lokale Portnummer, die Sie für die Einrichtung des SSH Tunnels mit dem Primärknoten ausgewählt habenOption 2, Teil 1: Richten Sie mithilfe dynamischer Portweiterleitung einen SSH Tunnel zum primären Knoten ein. Zum Beispiel Port *8157*. Dieser Port muss auch

mit der Portnummer übereinstimmen, die Sie in Pu TTY oder einem anderen Terminalemulator verwenden, den Sie für die Verbindung verwenden.

- Geben Sie das SOCKSv5-Protokoll an. SOCKSMit v5 können Sie optional die Benutzerautorisierung einrichten.
- URLMuster

  Die folgenden URL Muster sollten auf der Zulassungsliste stehen und mit einem Platzhaltermustertyp angegeben werden:

  - Das *ec2*. *berechnen*.amazonaws.com*- und*10*.amazonaws.com*-Muster, sodass sie dem öffentlichen Namen von Clustern in US-Regionen entsprechen. DNS
  - Die Muster *ec2*.compute* und *10*.compute* entsprechen dem öffentlichen Namen von Clustern in allen anderen Regionen. DNS
  - Eine 10. * Muster für den Zugriff auf die JobTracker Protokolldateien in Hadoop. Ändern Sie diesen Filter bei Konflikten mit Ihrem Netzwerkzugriffsplan.
  - Die Muster *.ec2.internal* und *.compute.internal* müssen den privaten (internen) DNS Namen von Clustern in der Region bzw. allen anderen Regionen entsprechen. `us-east-1`

Beispiel FoxyProxy: Für Firefox konfigurieren

Das folgende Beispiel zeigt eine FoxyProxy Standardkonfiguration (Version 7.5.1) für Mozilla Firefox.

FoxyProxy stellt eine Reihe von Tools zur Proxyverwaltung bereit. Damit können Sie einen Proxy-Server verwenden, um Muster abzugleichenURLs, die den Domänen entsprechen, die von den EC2 Amazon-Instances in Ihrem EMR Amazon-Cluster verwendet werden.

Zur Installation und Konfiguration FoxyProxy mit Mozilla Firefox

1. Gehen Sie in Firefox zu https://addons.mozilla.org/, suchen Sie nach FoxyProxy Standard und folgen Sie den Anweisungen zum Hinzufügen FoxyProxy zu Firefox.
2. Erstellen Sie mit einem Texteditor eine JSON Datei mit dem Namen `foxyproxy-settings.json` aus der folgenden Beispielkonfiguration.

```
{
  "k20d21508277536715": {
    "active": true,
    "address": "localhost",
    "port": 8157,
    "username": "",
```

```
    "password": "",
    "type": 3,
    "proxyDNS": true,
    "title": "emr-socks-proxy",
    "color": "#0055E5",
    "index": 9007199254740991,
    "whitePatterns": [
      {
        "title": "*ec2*.*compute*.amazonaws.com*",
        "active": true,
        "pattern": "*ec2*.*compute*.amazonaws.com*",
        "importedPattern": "*ec2*.*compute*.amazonaws.com*",
        "type": 1,
        "protocols": 1
      },
      {
        "title": "*ec2*.compute*",
        "active": true,
        "pattern": "*ec2*.compute*",
        "importedPattern": "*ec2*.compute*",
        "type": 1,
        "protocols": 1
      },
      {
        "title": "10.*",
        "active": true,
        "pattern": "10.*",
        "importedPattern": "http://10.*",
        "type": 1,
        "protocols": 2
      },
      {
        "title": "*10*.amazonaws.com*",
        "active": true,
        "pattern": "*10*.amazonaws.com*",
        "importedPattern": "*10*.amazonaws.com*",
        "type": 1,
        "protocols": 1
      },
      {
        "title": "*10*.compute*",
        "active": true,
        "pattern": "*10*.compute*",
        "importedPattern": "*10*.compute*",
```

```
        "type": 1,
        "protocols": 1
      },
      {
        "title": "*.compute.internal*",
        "active": true,
        "pattern": "*.compute.internal*",
        "importedPattern": "*.compute.internal*",
        "type": 1,
        "protocols": 1
      },
      {
        "title": "*.ec2.internal* ",
        "active": true,
        "pattern": "*.ec2.internal*",
        "importedPattern": "*.ec2.internal*",
        "type": 1,
        "protocols": 1
      }
    ],
    "blackPatterns": []
  },
  "logging": {
    "size": 100,
    "active": false
  },
  "mode": "patterns",
  "browserVersion": "68.12.0",
  "foxyProxyVersion": "7.5.1",
  "foxyProxyEdition": "standard"
}
```

3. Öffnen Sie die Firefox-Seite Ihre Erweiterungen verwalten (gehen Sie zu about:addons und wählen Sie dann Erweiterungen).
4. Wählen Sie FoxyProxy Standard und dann die Schaltfläche für weitere Optionen (die Schaltfläche, die wie eine Ellipse aussieht).
5. Wählen Sie Optionen aus der Dropdown-Liste aus.
6. Wählen Sie im linken Menü die Option Einstellungen importieren.
7. Wählen Sie auf der Seite „Importeinstellungen“ unter „Importeinstellungen aus Version FoxyProxy 6.0+“ die Option „Importeinstellungen“, navigieren Sie zum Speicherort der von

Ihnen erstellten **`foxyproxy-settings.json`** Datei, wählen Sie die Datei aus und wählen Sie „Öffnen“.

8. Wählen Sie OK, wenn Sie aufgefordert werden, die vorhandenen Einstellungen zu überschreiben und Ihre neue Konfiguration zu speichern.

Beispiel: Für Chrome konfigurieren SwitchyOmega

Das folgende Beispiel zeigt, wie Sie die SwitchyOmega Erweiterung für Google Chrome einrichten. SwitchyOmega ermöglicht es Ihnen, mehrere Proxys zu konfigurieren, zu verwalten und zwischen ihnen zu wechseln.

Zur Installation und Konfiguration SwitchyOmega mit Google Chrome

1. Gehen Sie zu https://chrome.google.com/webstore/Kategorie/Erweiterungen, suchen Sie nach Proxy SwitchyOmega und fügen Sie ihn zu Chrome hinzu.
2. Wählen Sie Neues Profil und geben Sie `emr-socks-proxy` als Profilnamen ein.
3. Wählen Sie PACProfil und dann Erstellen. Mithilfe von Dateien zur automatischen Proxykonfiguration (PAC) können Sie eine Zulassungsliste für Browseranfragen definieren, die an einen Web-Proxyserver weitergeleitet werden sollen.
4. Ersetzen Sie im Feld PACSkript den Inhalt durch das folgende Skript, das definiert, was über Ihren Web-Proxyserver weitergeleitet werden URLs soll. Wenn Sie bei der Einrichtung Ihres SSH Tunnels eine andere Portnummer angegeben haben, ersetzen Sie *8157* mit Ihrer Portnummer.

```
function FindProxyForURL(url, host) {
    if (shExpMatch(url, "*ec2*.*compute*.amazonaws.com*")) return 'SOCKS5
 localhost:8157';
    if (shExpMatch(url, "*ec2*.compute*")) return 'SOCKS5 localhost:8157';
    if (shExpMatch(url, "http://10.*")) return 'SOCKS5 localhost:8157';
    if (shExpMatch(url, "*10*.compute*")) return 'SOCKS5 localhost:8157';
    if (shExpMatch(url, "*10*.amazonaws.com*")) return 'SOCKS5 localhost:8157';
    if (shExpMatch(url, "*.compute.internal*")) return 'SOCKS5 localhost:8157';
    if (shExpMatch(url, "*ec2.internal*")) return 'SOCKS5 localhost:8157';
    return 'DIRECT';
}
```

5. Wählen Sie unter Aktionen die Option Änderungen übernehmen aus, um Ihre Proxyeinstellungen zu speichern.

6. Wählen Sie in der Chrome-Symbolleiste das `emr-socks-proxy` Profil aus SwitchyOmega und wählen Sie es aus.

Im Browser auf eine Weboberfläche zugreifen

Um eine Weboberfläche zu öffnen, geben Sie den öffentlichen DNS Namen Ihres Primär- oder Core-Nodes gefolgt von der Portnummer für die von Ihnen gewählte Schnittstelle in die Adressleiste Ihres Browsers ein. Das folgende Beispiel zeigt den, den URL Sie eingeben würden, um eine Verbindung zum Spark herzustellen HistoryServer.

```
http://master-public-dns-name:18080/
```

Anweisungen zum Abrufen des öffentlichen DNS Namens eines Knotens finden Sie unterRufen Sie den öffentlichen DNS Namen des primären Knotens ab. Eine vollständige Liste der Webschnittstellen finden Sie URLs unterAuf EMR Amazon-Clustern gehostete Weboberflächen anzeigen.

Übermitteln von Arbeit an einen Cluster

In diesem Abschnitt werden die Methoden beschrieben, mit denen Sie Arbeiten an einen EMR Amazon-Cluster einreichen können. Um Arbeit einzureichen, können Sie Schritte hinzufügen oder interaktiv Hadoop-Jobs an den Primärknoten senden.

Beachten Sie beim Einreichen von Schritten an einen Cluster die folgenden Verhaltensregeln:

- Eine Schritt-ID kann bis zu 256 Zeichen enthalten.
- Sie können bis zu 256 PENDING RUNNING Schritte in einem Cluster haben.
- Sie können Aufträge interaktiv an den Primärknoten übermitteln, auch dann, wenn auf dem Cluster 256 aktive Schritte ausgeführt werden. Sie können während der Laufzeit eines Clusters mit langer Laufzeit eine unbegrenzte Anzahl von Schritten einreichen, aber es können jeweils nur 256 Schritte RUNNING oder PENDING gleichzeitig eingereicht werden.
- Mit EMR Amazon-Versionen 4.8.0 und höher, außer Version 5.0.0, können Sie ausstehende Schritte stornieren. Weitere Informationen finden Sie unter Abbrechen von Schritten.
- Mit EMR Amazon-Versionen 5.28.0 und höher können Sie sowohl ausstehende als auch laufende Schritte stornieren. Sie können auch mehrere Schritte parallel ausführen, um die Clusterauslastung zu verbessern und Kosten zu sparen. Weitere Informationen finden Sie unter Überlegungen zum parallelen Ausführen mehrerer Schritte.

> **Note**
>
> Für eine optimale Leistung empfehlen wir, benutzerdefinierte Bootstrap-Aktionen, -Skripts und andere Dateien, die Sie mit Amazon verwenden möchten, EMR in einem Amazon S3 S3-Bucket zu speichern, der sich in demselben AWS-Region Cluster befindet.

Themen

Hinzufügen von Schritten zu einem Cluster mit der Amazon EMR Management Console

Verwenden Sie die folgenden Verfahren, um Schritte zu einem Cluster mit dem AWS Management Console hinzuzufügen. Detaillierte Informationen zum Einreichen von Schritten für bestimmte Big-Data-Anwendungen finden Sie in den folgenden Abschnitten des EMRAmazon-Versionshandbuchs:

- Reichen Sie einen benutzerdefinierten JAR Schritt ein
- Einen Hadoop-Streaming-Schritt senden
- Einen Spark-Schritt senden
- Einen Pig-Schritt senden
- Einen Befehl oder ein Skript als Schritt ausführen
- Werte in Schritte übergeben, um Hive-Skripte auszuführen

So fügen Sie Schritte während der Clustererstellung hinzu

Über den können Sie Schritte hinzufügen AWS Management Console, wenn Sie einen Cluster erstellen.

Console

Um Schritte hinzuzufügen, wenn Sie einen Cluster mit der Konsole erstellen

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und anschließend Create cluster aus.
3. Wählen Sie unter Schritte die Option Schritt hinzufügen aus. Geben Sie die entsprechenden Werte in die Felder im Dialogfeld Schritt hinzufügen ein. Informationen zur Formatierung Ihrer Schrittargumente finden Sie unter Schritt-Argumente hinzufügen. Die Optionen unterscheiden sich je nach Schritttyp. Um Ihren Schritt hinzuzufügen und das Dialogfeld zu verlassen, wählen Sie Schritt hinzufügen.
4. Wählen Sie alle anderen Optionen aus, die für Ihren Cluster gelten.
5. Um Ihren Cluster jetzt zu starten, wählen Sie Cluster erstellen aus.

So fügen Sie einem ausgeführten Cluster Schritte hinzu

Mit dem können Sie einem Cluster Schritte hinzufügen AWS Management Console, bei denen die Option zum automatischen Beenden deaktiviert ist.

Console

Um einem laufenden Cluster mit der Konsole Schritte hinzuzufügen

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und dann den Cluster aus, den Sie aktualisieren möchten.
3. Wählen Sie auf der Seite „Clusterdetails“ auf der Registerkarte Schritte die Option Schritt hinzufügen aus. Um einen vorhandenen Schritt zu klonen, wählen Sie das Dropdownmenü Aktionen und dann Schritt klonen aus.
4. Geben Sie die entsprechenden Werte in die Felder im Dialogfeld Schritt hinzufügen ein. Die Optionen unterscheiden sich je nach Schritttyp. Um Ihren Schritt hinzuzufügen und das Dialogfeld zu verlassen, wählen Sie Schritt hinzufügen.

So ändern Sie die Nebenläufigkeitsstufe für Schritte in einem ausgeführten Cluster

Mit dem AWS Management Console können Sie die Stufe der Schrittparallelität in einem laufenden Cluster ändern.

> **Note**
>
> Sie können nur mehrere Schritte parallel mit EMR Amazon-Version 5.28.0 und höher ausführen.

Console

Um die Schrittparallelität in einem laufenden Cluster mit der Konsole zu ändern

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und dann den Cluster aus, den Sie aktualisieren möchten. Der Cluster muss ausgeführt werden, um sein Parallelitätsattribut zu ändern.
3. Suchen Sie auf der Seite mit den Cluster-Details auf der Registerkarte Schritte den Abschnitt Attribute. Wählen Sie Bearbeiten aus, um die Parallelität zu ändern. Geben Sie einen Wert zwischen 1 und 256 ein.

Schritt-Argumente hinzufügen

Wenn Sie Ihrem Cluster einen Schritt AWS Management Console hinzufügen, können Sie Argumente für diesen Schritt im Feld Argumente angeben. Sie müssen Argumente durch Leerzeichen trennen und Zeichenkettenargumente einschließen, die aus Zeichen und Leerzeichen in Anführungszeichen bestehen.

Example : Richtige Argumente

Die folgenden Beispielargumente sind für das korrekt formatiert AWS Management Console, wobei das letzte Zeichenkettenargument in Anführungszeichen gesetzt ist.

```
bash -c "aws s3 cp s3://DOC-EXAMPLE-BUCKET/my-script.sh ."
```

Sie können auch jedes Argument aus Gründen der besseren Lesbarkeit in eine separate Zeile einfügen, wie im folgenden Beispiel gezeigt.

```
bash
-c
"aws s3 cp s3://DOC-EXAMPLE-BUCKET/my-script.sh ."
```

Example : Falsche Argumente

Die folgenden Beispielargumente sind falsch formatiert für AWS Management Console. Beachten Sie, dass das letzte Zeichenkettenargument, `aws s3 cp s3://DOC-EXAMPLE-BUCKET/my-script.sh .`, Leerzeichen enthält und nicht von Anführungszeichen umgeben ist.

```
bash -c aws s3 cp s3://DOC-EXAMPLE-BUCKET/my-script.sh .
```

Hinzufügen von Schritten zu einem Cluster mit dem AWS CLI

Die folgenden Verfahren zeigen, wie Sie Schritte zu einem neu erstellten Cluster und zu einem aktiven Cluster mit der AWS CLI hinzufügen. In beiden Beispielen wird der Unterbefehl `--steps` verwendet, um Schritte zum Cluster hinzuzufügen.

So fügen Sie Schritte während der Clustererstellung hinzu

- Geben Sie den folgenden Befehl ein, um einen Cluster zu erstellen und einen Apache Pig-Schritt hinzuzufügen. Stellen Sie sicher, dass Sie es ersetzen *myKey* mit dem Namen Ihres EC2 Amazon-Schlüsselpaars.

  ```
  aws emr create-cluster --name "Test cluster" \
  --applications Name=Spark \
  --use-default-roles \
  --ec2-attributes KeyName=myKey \
  --instance-groups InstanceGroupType=PRIMARY,InstanceCount=1,InstanceType=m5.xlarge
   InstanceGroupType=CORE,InstanceCount=2,InstanceType=m5.xlarge \
  --steps '[{"Args":["spark-submit","--deploy-mode","cluster","--
  class","org.apache.spark.examples.SparkPi","/usr/lib/spark/examples/jars/spark-
  examples.jar","5"],"Type":"CUSTOM_JAR","ActionOnFailure":"CONTINUE","Jar":"command-
  runner.jar","Properties":"","Name":"Spark application"}]'
  ```

> **Note**
>
> Die Liste der Argumente ändert sich je nach Art des Schritts.

Standardmäßig ist Nebenläufigkeitsstufe für Schritte 1. Sie können die Nebenläufigkeitsstufe für Schritte festlegen, indem Sie den `StepConcurrencyLevel`-Parameter beim Erstellen eines Clusters verwenden.

Die Ausgabe ist eine Cluster-Kennung ähnlich der folgenden.

```
{
    "ClusterId": "j-2AXXXXXXGAPLF"
}
```

So fügen Sie einen Schritt einem aktiven Cluster hinzu

- Geben Sie den folgenden Befehl ein, um einen Schritt zu einem aktiven Cluster hinzuzufügen. Ersetzen Sie *`j-2AXXXXXXGAPLF`* durch die ID Ihres eigenen Clusters.

```
aws emr add-steps --cluster-id j-2AXXXXXXGAPLF \
--steps '[{"Args":["spark-submit","--deploy-mode","cluster","--
class","org.apache.spark.examples.SparkPi","/usr/lib/spark/examples/jars/spark-
examples.jar","5"],"Type":"CUSTOM_JAR","ActionOnFailure":"CONTINUE","Jar":"command-
runner.jar","Properties":"","Name":"Spark application"}]'
```

Die Ausgabe ist eine Schrittkennung ähnlich der folgenden.

```
{
    "StepIds": [
 "s-Y9XXXXXXAPMD"
    ]
}
```

Um das StepConcurrencyLevel in einem laufenden Cluster zu ändern

1. In einem laufenden Cluster können Sie den `StepConcurrencyLevel` mit dem ändern `ModifyClusterAPI`. Geben Sie beispielsweise den folgenden Befehl ein, um die`StepConcurrencyLevel` für Schritte auf `10` zu erhöhen. Ersetzen Sie *`j-2AXXXXXXGAPLF`* durch die ID Ihres Clusters.

   ```
   aws emr modify-cluster --cluster-id j-2AXXXXXXGAPLF --step-concurrency-level 10
   ```

2. Die Ausgabe sieht folgendermaßen oder ähnlich aus.

   ```
   {
   "StepConcurrencyLevel": 10
   }
   ```

Weitere Informationen zur Verwendung von EMR Amazon-Befehlen in der AWS CLI finden Sie in der AWS CLI Befehlsreferenz.

Überlegungen zum parallelen Ausführen mehrerer Schritte

- Parallel laufende Schritte können in beliebiger Reihenfolge abgeschlossen werden, aber ausstehende Schritte in der Warteschlange gehen in der Reihenfolge in den laufenden Zustand über, in der sie eingereicht wurden.
- Wenn Sie eine Nebenläufigkeitsstufe für Schritte für den Cluster auswählen, müssen Sie überlegen, ob der Primärknoten-Instance-Typ die Speicheranforderungen von Benutzer-Workloads erfüllt. Der Hauptschrittausführungsprozess wird für jeden Schritt auf dem Primärknoten ausgeführt. Die parallel Ausführung mehrerer Schritte erfordert mehr Speicher und CPU Auslastung vom Primärknoten als die Ausführung eines Schritts nach dem anderen.
- Um eine komplexe Planung und Ressourcenverwaltung gleichzeitiger Schritte zu erreichen, können Sie YARN Planungsfunktionen wie `FairScheduler` oder `CapacityScheduler` verwenden. Beispielsweise können Sie `FairScheduler` mit einem `queueMaxAppsDefault`-Satz verwenden, um zu verhindern, dass mehr als eine bestimmte Anzahl von Aufgaben gleichzeitig ausgeführt werden.
- Die Nebenläufigkeitsstufe für Schritte unterliegt den Konfigurationen von Ressourcenmanagern. Wenn beispielsweise nur mit einer Parallelität von konfiguriert YARN ist5, können nur fünf YARN Anwendungen parallel ausgeführt werden, auch wenn die auf eingestellt

`StepConcurrencyLevel` ist. `10` Weitere Informationen zur Konfiguration von Resource Managern finden Sie unter Configure applications im Amazon EMR Release Guide.

- Sie können nur dann einen Schritt hinzufügen, CONTINUE wenn die Schrittparallelitätsstufe des Clusters größer als 1 ist. `ActionOnFailure`
- Wenn die Step-Parallelitätsstufe eines Clusters größer als eins ist, wird das `ActionOnFailure`-Step-Feature nicht aktiviert.
- Wenn ein Cluster über die Schritt-Parallelitätsstufe 1, aber über mehrere laufende Schritte verfügt, wird `TERMINATE_CLUSTER ActionOnFailure` möglicherweise aktiviert, `CANCEL_AND_WAIT ActionOnFailure` jedoch nicht. Dieser Grenzfall tritt auf, wenn die Parallelitätsstufe für Clusterschritte höher als eins war, aber während der Ausführung mehrerer Schritte niedriger war.
- Sie können die EMR automatische Skalierung verwenden, um je nach Ressourcen nach oben oder unten zu skalieren, um YARN Ressourcenkonflikte zu vermeiden. Weitere Informationen finden Sie unter Verwenden der automatischen Skalierung mit einer benutzerdefinierten Richtlinie für Instanzgruppen im Amazon EMR Management Guide.
- Wenn Sie die Stufe der gleichzeitigen Schritte verringern, EMR können alle laufenden Schritte abgeschlossen werden, bevor die Anzahl der Schritte reduziert wird. Wenn die Ressourcen ausgeschöpft sind, weil der Cluster zu viele gleichzeitige Schritte ausführt, empfehlen wir, alle laufenden Schritte manuell abzubrechen, um Ressourcen freizumachen.

Anzeigen von Schritten

Sie können bis zu 10.000 Schritte sehen, die Amazon in den letzten sieben Tagen EMR abgeschlossen hat. Sie können sich auch jederzeit 1.000 Schritte ansehen, die Amazon EMR abgeschlossen hat. Diese Gesamtzahl umfasst sowohl vom Benutzer übermittelte Schritte als auch Systemschritte.

Wenn Sie neue Schritte einreichen, sobald der Cluster das Datensatzlimit von 1.000 Schritten erreicht hat, EMR löscht Amazon die inaktiven, von Benutzern eingereichten Schritte COMPLETEDCANCELLED, deren Status länger als sieben Tage oder FAILED länger ist. Wenn Sie Schritte einreichen, die das Limit von 10.000 Schrittdatensätzen überschreiten, EMR löscht Amazon die inaktiven, von Benutzern eingereichten Schrittdatensätze unabhängig von ihrer inaktiven Dauer. Amazon entfernt diese Datensätze EMR nicht aus den Protokolldateien. Amazon EMR entfernt sie aus der AWS Konsole und sie werden nicht zurückgegeben, wenn Sie das AWS CLI oder API zum Abrufen von Cluster-Informationen verwenden. Systemschrittdatensätze werden niemals entfernt.

Welche Schrittinformationen Sie anzeigen können, hängt vom Mechanismus ab, der zum Abrufen der Cluster-Informationen verwendet wurde. Die folgende Tabelle enthält die Schrittinformationen, die von den verfügbaren Optionen jeweils zurückgegeben werden.

Option	DescribeJobFlow oder --describe --jobflow	ListSteps oder liste-steps
SDK	256 Schritte	Bis zu 10.000 Schritte
Amazon EMR CLI	256 Schritte	N/A
AWS CLI	N/A	Bis zu 10.000 Schritte
API	256 Schritte	Bis zu 10.000 Schritte

Abbrechen von Schritten

Sie können ausstehende und laufende Schritte von Amazon AWS Management Console AWS CLI, The oder Amazon stornieren EMRAPI.

Console

Um Schritte mit der Konsole abzubrechen

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMROn die Option Clusters aus und wählen Sie dann den Cluster aus, den Sie aktualisieren möchten.
3. Klicken Sie auf der Seite Cluster-Details auf der Registerkarte Schritte auf das Kontrollkästchen neben dem Schritt, den Sie abbrechen möchten. Wählen Sie das Dropdownmenü Aktionen und dann Schritte abbrechen aus.
4. Wählen Sie im Dialogfeld Schritt abbrechen entweder den Schritt abbrechen und warten, bis er beendet ist, oder ob Sie den Schritt abbrechen und das Beenden erzwingen möchten. Wählen Sie dann Confirm (Bestätigen) aus.
5. Der Status der Schritte in der Tabelle Schritte ändert sich in `CANCELLED`.

CLI

Um abzubrechen mit dem AWS CLI

- Verwenden Sie den Befehl `aws emr cancel-steps` unter Angabe des Clusters und der abzubrechenden Schritte. Das folgende Beispiel zeigt einen AWS CLI -Befehl für den Abbruch von zwei Schritten.

```
aws emr cancel-steps --cluster-id j-2QUAXXXXXXXXX \
--step-ids s-3M8DXXXXXXXXX s-3M8DXXXXXXXXX \
--step-cancellation-option SEND_INTERRUPT
```

Mit EMR Amazon-Version 5.28.0 können Sie beim Stornieren von Schritten eine der beiden folgenden Stornierungsoptionen als `StepCancellationOption` Parameter wählen.

- `SEND_INTERRUPT` – Dies ist die Standardoption. Wenn eine Anforderung zum Abbruch eines Schritts eingeht, wird ein `SIGTERM` Signal an den Schritt EMR gesendet. Fügen Sie Ihrer Schrittlogik einen `SIGTERM` Signal-Handler hinzu, um dieses Signal abzufangen und die Prozesse der untergeordneten Schritte zu beenden oder zu warten, bis sie abgeschlossen sind.
- `TERMINATE_PROCESS`— Wenn diese Option ausgewählt ist, wird ein EMR `SIGKILL` Signal an den Schritt und alle seine untergeordneten Prozesse gesendet, wodurch sie sofort beendet werden.

Was es bei der Stornierung von Schritten zu berücksichtigen gibt

- Wenn Sie einen laufenden oder ausstehenden Schritt abbrechen, wird dieser Schritt aus der aktiven Schrittanzahl entfernt.
- Wenn Sie einen laufenden Schritt abbrechen, kann ein ausstehender Schritt nicht ausgeführt werden, vorausgesetzt, dass keine Änderung an `stepConcurrencyLevel` vorgenommen wurde.
- Durch das Abbrechen eines laufenden Schritts wird der Schritt `ActionOnFailure` nicht ausgelöst.
- Sendet für EMR 5.32.0 und höher ein `SEND_INTERRUPT StepCancellationOption` `SIGTERM` Signal an den untergeordneten Schrittprozess. Sie sollten auf dieses Signal achten und eine Säuberung durchführen und das System ordnungsgemäß herunterfahren. `TERMINATE_PROCESS` `StepCancellationOption` sendet ein `SIGKILL`-Signal an den untergeordneten Schrittprozess und alle seine untergeordneten Prozesse. Asynchrone Prozesse sind jedoch nicht betroffen.

Einen Cluster anzeigen und überwachen

Amazon EMR bietet mehrere Tools, mit denen Sie Informationen über Ihren Cluster sammeln können. Sie können über die Konsole, die CLI oder programmgesteuert auf Informationen über den Cluster zugreifen. Die Standard-Hadoop-Webschnittstellen und Protokolldateien sind auf dem Primärknoten verfügbar. Sie können auch Überwachungsdienste wie CloudWatch Ganglia verwenden, um die Leistung Ihres Clusters zu verfolgen.

Der Anwendungsverlauf ist ab Amazon EMR 5.25.0 auch über die Konsole mit der „persistenten“ Anwendung UIs für Spark History Server verfügbar. Mit Amazon EMR 6.x sind auch persistente YARN Timeline-Server- und Tez-Benutzeroberflächen verfügbar. Diese Dienste werden außerhalb des Clusters gehostet, sodass Sie nach Beendigung des Clusters noch 30 Tage lang auf den Anwendungsverlauf zugreifen können, ohne dass eine SSH Verbindung oder ein Webproxy erforderlich ist. Weitere Information unter Anwendungsverlauf anzeigen

Themen

Cluster-Status und -Details anzeigen

Nach dem Erstellen eines Clusters können Sie seinen Status überwachen und detaillierte Informationen zu seiner Ausführung sowie eventuell aufgetretenen Fehlern erhalten – auch nach dem Beenden des Clusters. Amazon EMR speichert Metadaten zu beendeten Clustern zwei Monate lang als Referenz. Danach werden die Metadaten gelöscht. Sie können keine Cluster aus dem Cluster-Verlauf löschen. Sie können jedoch in der AWS Management Console die Option Filter (Filtern) und in der AWS CLI Optionen mit dem Befehl `list-clusters` verwenden, um sich auf für Sie relevante Cluster zu konzentrieren.

Sie können auf den innerhalb des Clusters gespeicherten Anwendungsverlauf eine Woche ab dem Zeitpunkt der Aufzeichnung zugreifen, unabhängig davon, ob der Cluster ausgeführt wird oder beendet wurde. Darüber hinaus speichern persistente Anwendungsbenutzeroberflächen den Anwendungsverlauf für 30 Tage nach Beendigung eines Clusters außerhalb des Clusters. Weitere Information unter Anwendungsverlauf anzeigen

Weitere Hinweise zu Clusterstatus, wie Wartend und Läuft, finden Sie unter Verstehen des Cluster-Lebenszyklus.

So lassen Sie Cluster-Details mithilfe der AWS Management Console anzeigen

In der Clusterliste unter https://console.aws.amazon.com/emr sind alle Cluster in Ihrem Konto und Ihrer AWS Region aufgeführt, einschließlich beendeter Cluster. Die Liste enthält für jeden Cluster Folgendes: den Namen und die ID, die Status- und Statusdetails, die Erstellungszeit, die abgelaufene Zeit, in der der Cluster ausgeführt wurde, und die Stunden der normalisierten Instanz, die für alle Instances im Cluster angefallen sind. EC2 Diese Liste ist der Ausgangspunkt für die Überwachung des Status von Clustern. Sie ist so konzipiert, dass Sie jeden Cluster zu Analyse- und Fehlerbehebungszwecken aufschlüsseln können.

Console

Um Clusterinformationen mit der Konsole anzuzeigen

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und dann den Cluster aus, den Sie anzeigen möchten.
3. Verwenden Sie den Bereich Zusammenfassung, um die Grundlagen Ihrer Cluster-Konfiguration anzuzeigen, z. B. den Cluster-Status, die Open-Source-Anwendungen, die Amazon auf dem Cluster EMR installiert hat, und die Version von AmazonEMR, mit der Sie den Cluster erstellt haben. Verwenden Sie jede Registerkarte unterhalb der Zusammenfassung, um die in der folgenden Tabelle beschriebenen Informationen anzuzeigen.

Sehen Sie sich Cluster-Details an, indem Sie AWS CLI

Die folgenden Beispiele zeigen, wie Sie Cluster-Details über die AWS CLI abrufen. Weitere Informationen zu verfügbaren Befehlen finden Sie in der AWS CLI Befehlsreferenz für Amazon

EMR. Sie können den Befehl describe-cluster verwenden, um Details auf Clusterebene wie Status, Hardware- und Softwarekonfiguration, VPC Einstellungen, Bootstrap-Aktionen, Instance-Gruppen usw. anzuzeigen. Weitere Informationen zu Cluster-Status finden Sie unter Verstehen des Cluster-Lebenszyklus. Das folgende Beispiel zeigt die Verwendung des Befehls `describe-cluster`, gefolgt von Beispielen für den Befehl list-clusters.

Example Anzeigen des Cluster-Status

Sie benötigen die Cluster-ID, um den Befehl `describe-cluster` zu verwenden. Dieses Beispiel zeigt, wie Sie eine Liste von Clustern abrufen, die innerhalb eines bestimmten Datumsbereichs erstellt wurden, und anschließend mithilfe eines IDs zurückgegebenen Clusters weitere Informationen zum Status eines einzelnen Clusters auflisten können.

Der folgende Befehl beschreibt den Cluster *`j-1K48XXXXXXHCB`*, die Sie durch Ihre Cluster-ID ersetzen.

```
aws emr describe-cluster --cluster-id j-1K48XXXXXXHCB
```

Die Ausgabe des Befehls ähnelt der folgenden:

```
{
    "Cluster": {
        "Status": {
            "Timeline": {
                "ReadyDateTime": 1438281058.061,
                "CreationDateTime": 1438280702.498
            },
            "State": "WAITING",
            "StateChangeReason": {
                "Message": "Waiting for steps to run"
            }
        },
        "Ec2InstanceAttributes": {
            "EmrManagedMasterSecurityGroup": "sg-cXXXXX0",
            "IamInstanceProfile": "EMR_EC2_DefaultRole",
            "Ec2KeyName": "myKey",
            "Ec2AvailabilityZone": "us-east-1c",
            "EmrManagedSlaveSecurityGroup": "sg-example"
        },
        "Name": "Development Cluster",
        "ServiceRole": "EMR_DefaultRole",
        "Tags": [],
```

```
        "TerminationProtected": false,
        "ReleaseLabel": "emr-4.0.0",
        "NormalizedInstanceHours": 16,
        "InstanceGroups": [
            {
                "RequestedInstanceCount": 1,
                "Status": {
                    "Timeline": {
                        "ReadyDateTime": 1438281058.101,
                        "CreationDateTime": 1438280702.499
                    },
                    "State": "RUNNING",
                    "StateChangeReason": {
                        "Message": ""
                    }
                },
                "Name": "CORE",
                "InstanceGroupType": "CORE",
                "Id": "ig-2EEXAMPLEXXP",
                "Configurations": [],
                "InstanceType": "m5.xlarge",
                "Market": "ON_DEMAND",
                "RunningInstanceCount": 1
            },
            {
                "RequestedInstanceCount": 1,
                "Status": {
                    "Timeline": {
                        "ReadyDateTime": 1438281023.879,
                        "CreationDateTime": 1438280702.499
                    },
                    "State": "RUNNING",
                    "StateChangeReason": {
                        "Message": ""
                    }
                },
                "Name": "MASTER",
                "InstanceGroupType": "MASTER",
                "Id": "ig-2A1234567XP",
                "Configurations": [],
                "InstanceType": "m5.xlarge",
                "Market": "ON_DEMAND",
                "RunningInstanceCount": 1
            }
```

```
        ],
        "Applications": [
            {
                "Version": "1.0.0",
                "Name": "Hive"
            },
            {
                "Version": "2.6.0",
                "Name": "Hadoop"
            },
            {
                "Version": "0.14.0",
                "Name": "Pig"
            },
            {
                "Version": "1.4.1",
                "Name": "Spark"
            }
        ],
        "BootstrapActions": [],
        "MasterPublicDnsName": "ec2-X-X-X-X.compute-1.amazonaws.com",
        "AutoTerminate": false,
        "Id": "j-jobFlowID",
        "Configurations": [
            {
                "Properties": {
                    "hadoop.security.groups.cache.secs": "250"
                },
                "Classification": "core-site"
            },
            {
                "Properties": {
                    "mapreduce.tasktracker.reduce.tasks.maximum": "5",
                    "mapred.tasktracker.map.tasks.maximum": "2",
                    "mapreduce.map.sort.spill.percent": "90"
                },
                "Classification": "mapred-site"
            },
            {
                "Properties": {
                    "hive.join.emit.interval": "1000",
                    "hive.merge.mapfiles": "true"
                },
                "Classification": "hive-site"
```

```
            }
        ]
    }
}
```

Example Auflisten von Clustern nach Erstellungsdatum

Zum Abrufen von in einem bestimmten Datenbereich erstellten Clustern verwenden Sie den Befehl `list-clusters` mit den Parametern `--created-after` und `--created-before`.

Mit dem folgenden Befehl werden alle Cluster aufgelistet, die zwischen dem 9. Oktober 2019 und dem 12. Oktober 2019 erstellt wurden.

```
aws emr list-clusters --created-after 2019-10-09T00:12:00 --created-
before 2019-10-12T00:12:00
```

Example Auflisten von Clustern nach Status

Verwenden Sie zum Auflisten von Clustern nach Status den Befehl `list-clusters` mit dem Parameter `--cluster-states`. Zu den gültigen Clusterstatus gehören: STARTING BOOTSTRAPPINGRUNNING,WAITING,TERMINATING,TERMINATED,, und TERMINATED _ WITH _ERRORS.

```
aws emr list-clusters --cluster-states TERMINATED
```

Sie können auch die folgenden Abkürzungsparameter verwenden, um alle Cluster in den angegebenen Zuständen aufzulisten:

- `--active`filtert Cluster in den TERMINATING Bundesstaaten STARTING BOOTSTRAPPINGRUNNING,WAITING,, oder.
- `--terminated`filtert Cluster im TERMINATED Bundesstaat.
- `--failed`Der Parameter filtert Cluster im ERRORS Status TERMINATED WITH _ _.

Die folgenden Befehle geben dasselbe Ergebnis zurück.

```
aws emr list-clusters --cluster-states TERMINATED
```

```
aws emr list-clusters --terminated
```

Weitere Informationen zu Cluster-Status finden Sie unter Verstehen des Cluster-Lebenszyklus.

Verbessertes Schritt-Debuggen

Wenn ein EMR Amazon-Schritt fehlschlägt und Sie Ihre Arbeit mithilfe der API Step-Operation mit einer Version AMI der Version 5.x oder höher eingereicht haben, EMR kann Amazon in einigen Fällen die Hauptursache des Schrittfehlers ermitteln und zurückgeben, zusammen mit dem Namen der entsprechenden Protokolldatei und einem Teil des Anwendungs-Stack-Trace überAPI. Die folgenden Fehler können identifiziert werden:

- Ein üblicher Hadoop-Fehler, wie z. B. das Ausgabeverzeichnis ist bereits vorhanden, das Eingabeverzeichnis ist nicht vorhanden oder für eine Anwendung ist nicht mehr genügend Speicherplatz vorhanden.
- Java-Fehler, wie z. B. eine Anwendung, die mit einer inkompatiblen Version von Java kompiliert und mit einer Hauptklasse ausgeführt wurde, die nicht gefunden wird.
- Ein Problem mit dem Zugriff auf Objekte, die in Amazon S3 gespeichert sind.

Diese Informationen sind mit den ListStepsAPIOperationen DescribeStepund verfügbar. Das FailureDetailsFeld, das von diesen Operationen StepSummaryzurückgegeben wurde. Um auf die FailureDetails Informationen zuzugreifen, verwenden Sie die Konsole AWS CLI, oder AWS SDK.

Console

Die neue EMR Amazon-Konsole bietet kein schrittweises Debugging. Mit den folgenden Schritten können Sie jedoch Details zur Clusterbeendigung anzeigen.

Um Fehlerdetails mit der Konsole anzuzeigen

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMROn die Option Clusters aus und wählen Sie dann den Cluster aus, den Sie anzeigen möchten.
3. Notieren Sie sich den Statuswert im Abschnitt Zusammenfassung der Cluster-Detailseite. Wenn der Status Mit Fehlern beendet lautet, bewegen Sie den Mauszeiger über den Text, um Details zum Clusterausfall anzuzeigen.

CLI

Um Fehlerdetails anzuzeigen, verwenden Sie AWS CLI

- Verwenden Sie den `describe-step` Befehl AWS CLI, um Fehlerdetails für einen Schritt mit dem abzurufen.

```
aws emr describe-step --cluster-id j-1K48XXXXXHCB --step-id s-3QM0XXXXXM1W
```

Die Ausgabe sieht etwa folgendermaßen aus:

```
{
  "Step": {
    "Status": {
      "FailureDetails": {
        "LogFile": "s3://myBucket/logs/j-1K48XXXXXHCB/steps/s-3QM0XXXXXM1W/
stderr.gz",
        "Message": "org.apache.hadoop.mapred.FileAlreadyExistsException: Output
 directory s3://myBucket/logs/beta already exists",
        "Reason": "Output directory already exists."
      },
      "Timeline": {
        "EndDateTime": 1469034209.143,
        "CreationDateTime": 1469033847.105,
        "StartDateTime": 1469034202.881
      },
      "State": "FAILED",
      "StateChangeReason": {}
    },
    "Config": {
      "Args": [
        "wordcount",
        "s3://myBucket/input/input.txt",
        "s3://myBucket/logs/beta"
      ],
      "Jar": "s3://myBucket/jars/hadoop-mapreduce-examples-2.7.2-amzn-1.jar",
      "Properties": {}
    },
    "Id": "s-3QM0XXXXXM1W",
    "ActionOnFailure": "CONTINUE",
    "Name": "ExampleJob"
  }
```

```
}
```

Anwendungsverlauf anzeigen

Sie können die Details der Spark History Server- und YARN Timeline-Dienstanwendungen auf der Detailseite des Clusters in der Konsole einsehen. Der EMR Amazon-Bewerbungsverlauf erleichtert Ihnen die Fehlerbehebung und Analyse aktiver Jobs und des Jobverlaufs.

> **Note**
>
> Um die Sicherheit der Anwendungen außerhalb der Konsole zu erhöhen, die Sie möglicherweise mit Amazon verwendenEMR, werden die Anwendungshosting-Domains in der Liste der öffentlichen Suffixe () registriert. PSL Zu diesen Hosting-Domains gehören beispielsweise die folgenden: `emrstudio-prod.us-east-1.amazonaws.com`, `emrnotebooks-prod.us-east-1.amazonaws.com`, `emrappui-prod.us-east-1.amazonaws.com`. Aus Sicherheitsgründen empfehlen wir Ihnen, Cookies mit einem `__Host-`-Präfix zu verwenden, falls Sie jemals sensible Cookies im Standard-Domainnamen einrichten müssen. Dies trägt dazu bei, Ihre Domain vor standortübergreifenden Anforderungsfälschungsversuchen zu schützen (). CSRF Weitere Informationen finden Sie auf der Set-Cookie-Seite im Mozilla Developer Network.

Der Abschnitt Anwendungsbenutzeroberflächen auf der Registerkarte Anwendungen bietet verschiedene Anzeigeoptionen, abhängig vom Clusterstatus und den Anwendungen, die Sie auf dem Cluster installiert haben.

- Zugriff außerhalb des Clusters auf persistente Anwendungsbenutzeroberflächen — Ab EMR Amazon-Version 5.25.0 sind persistente Links zur Anwendungsbenutzeroberfläche für Spark UI und Spark History Service verfügbar. Mit EMR Amazon-Version 5.30.1 und höher verfügen die Tez-Benutzeroberfläche und der YARN Timeline-Server auch über persistente Anwendungsbenutzeroberflächen. Der YARN Timeline-Server und die Tez-Benutzeroberfläche sind Open-Source-Anwendungen, die Metriken für aktive und beendete Cluster bereitstellen. Die Spark-Benutzeroberfläche bietet Details zu den Phasen und Aufgaben des Schedulers, zu RDD Größe und Speicherbelegung, Umgebungsinformationen und Informationen zu den ausgeführten Executoren. Persistente Anwendungen UIs werden außerhalb des Clusters ausgeführt, sodass Clusterinformationen und -protokolle nach dem Beenden einer Anwendung 30 Tage lang verfügbar

sind. Im Gegensatz zu Benutzeroberflächen für Cluster-Anwendungen müssen Sie bei persistenten Anwendungen UIs keinen Webproxy über eine Verbindung einrichten. SSH

- Anwendungsbenutzeroberflächen innerhalb des Clusters – Es gibt eine Vielzahl von Anwendungsverlauf-Benutzeroberflächen, die auf einem Cluster ausgeführt werden können. Benutzeroberflächen im Cluster werden auf dem Master-Knoten gehostet und erfordern, dass Sie eine SSH Verbindung zum Webserver einrichten. Anwendungsbenutzeroberflächen innerhalb eines Clusters speichern den Anwendungsverlauf für eine Woche nach dem Beenden einer Anwendung. Weitere Informationen und Anweisungen zum Einrichten eines SSH Tunnels finden Sie unterAuf EMR Amazon-Clustern gehostete Weboberflächen anzeigen.

  Mit Ausnahme der Anwendungen Spark History Server, YARN Timeline Server und Hive kann der Anwendungsverlauf auf dem Cluster nur angezeigt werden, während der Cluster läuft.

Persistente Anwendungsbenutzeroberflächen anzeigen

Ab EMR Amazon-Version 5.25.0 können Sie über die Cluster-Übersichtsseite oder die Registerkarte Anwendungsbenutzeroberflächen in der Konsole eine Verbindung zu den persistenten Spark History Server-Anwendungsdetails herstellen, die außerhalb des Clusters gehostet werden. Die persistenten Anwendungsschnittstellen für Tez UI und YARN Timeline Server sind ab EMR Amazon-Version 5.30.1 verfügbar. Der Zugriff auf den persistenten Anwendungsverlauf mit einem Klick bietet folgende Vorteile:

- Sie können aktive Jobs und den Jobverlauf schnell analysieren und Fehler beheben, ohne einen Web-Proxy über eine Verbindung einrichten zu müssen. SSH
- Sie können auf den Anwendungsverlauf und relevante Protokolldateien für aktive und beendete Cluster zugreifen. Die Protokolle stehen nach dem Ende der Anwendung 30 Tage lang zur Verfügung.

Navigieren Sie in der Konsole zu Ihren Clusterdetails und wählen Sie die Registerkarte Applications (Anwendungen) aus. Wählen Sie die Benutzeroberfläche der Anwendung aus, die Sie nach dem Start Ihres Clusters verwenden möchten. Die Benutzeroberfläche der Anwendung wird in einer neuen Browserregisterkarte geöffnet. Weitere Informationen finden Sie unter Überwachung und Instrumentierung.

Sie können YARN Container-Logs über die Links auf dem Spark-Verlaufsserver, dem YARN Timeline-Server und der Tez-Benutzeroberfläche einsehen.

> **Note**
>
> Um vom Spark-History-Server, YARN Timeline-Server und der Tez-Benutzeroberfläche auf YARN Container-Logs zuzugreifen, müssen Sie die Protokollierung bei Amazon S3 für Ihren Cluster aktivieren. Wenn Sie die Protokollierung nicht aktivieren, funktionieren die Links zu YARN Container-Logs nicht.

Protokollsammlung

Um den Zugriff auf persistente Anwendungsbenutzeroberflächen mit einem Klick zu ermöglichen, EMR sammelt Amazon zwei Arten von Protokollen:

- Anwendungsereignisprotokolle werden in einem EMR System-Bucket gesammelt. Die Ereignisprotokolle werden im Ruhezustand mit serverseitiger Verschlüsselung mit Amazon S3 Managed Keys (SSE-S3) verschlüsselt. Wenn Sie ein privates Subnetz für Ihren Cluster verwenden, stellen Sie sicher, dass Sie "`arn:aws:s3:::prod.MyRegion.appinfo.src/*`" in die Ressourcenliste der Amazon S3-Richtlinie für das private Subnetz aufnehmen. Weitere Informationen finden Sie unter Amazon-S3-Mindestrichtlinie für privates Subnetz.
- YARNContainer-Logs werden in einem Amazon S3 S3-Bucket gesammelt, den Sie besitzen. Sie müssen die Protokollierung für Ihren Cluster aktivieren, um auf YARN Container-Logs zugreifen zu können. Weitere Informationen finden Sie unter Konfigurieren der Cluster-Protokollierung und des Debuggings.

Wenn Sie diese Funktion aus Datenschutzgründen deaktivieren müssen, können Sie den Daemon mithilfe eines Bootstrap-Skripts beim Erstellen eines Clusters stoppen, wie im folgenden Beispiel gezeigt wird.

```
aws emr create-cluster --name "Stop Application UI Support" --release-label emr-7.2.0 \
--applications Name=Hadoop Name=Spark --ec2-attributes KeyName=<myEMRKeyPairName> \
--instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m3.xlarge
 InstanceGroupType=CORE,InstanceCount=1,InstanceType=m3.xlarge
 InstanceGroupType=TASK,InstanceCount=1,InstanceType=m3.xlarge \
--use-default-roles --bootstrap-actions Path=s3://region.elasticmapreduce/bootstrap-
actions/run-if,Args=["instance.isMaster=true","echo Stop Application UI | sudo tee /
etc/apppusher/run-apppusher; sudo systemctl stop apppusher || exit 0"]
```

Nachdem Sie dieses Bootstrap-Skript ausgeführt haben, EMR sammelt Amazon keine Spark History Server- oder YARN Timeline-Server-Ereignisprotokolle im EMR System-Bucket. Auf der Registerkarte Application user interfaces (Anwendungsbenutzeroberflächen) werden keine Informationen zum Anwendungsverlauf verfügbar sein und Sie verlieren den Zugriff auf alle Anwendungsbenutzeroberflächen über die Konsole.

Große Spark-Ereignisprotokolldateien

In einigen Fällen können Spark-Jobs mit langer Laufzeit, wie Spark-Streaming, und große Jobs, wie SQL Spark-Abfragen, umfangreiche Ereignisprotokolle generieren. Bei großen Ereignisprotokollen können Sie schnell Festplattenspeicher auf Recheninstanzen verbrauchen und beim Laden von Persistent `0utOfMemory` UIs Fehler auftreten. Um diese Probleme zu vermeiden, wird empfohlen, dass Sie das Feature zum Rollen und Verdichten von Spark-Ereignisprotokollen aktivieren. Diese Funktion ist in den EMR Amazon-Versionen emr-6.1.0 und höher verfügbar. Weitere Informationen zum Rollen und Verdichten finden Sie in der Spark-Dokumentation unter Anwenden der Komprimierung auf Protokolldateien für rollende Ereignisse.

Um das Feature zum Rollen und Verdichten des Spark-Ereignisprotokolls zu aktivieren, aktivieren Sie die folgenden Spark-Konfigurationseinstellungen.

- `spark.eventLog.rolling.enabled` – Aktiviert das Rolling des Ereignisprotokolls je nach Größe. Diese Einstellung ist standardmäßig deaktiviert.
- `spark.eventLog.rolling.maxFileSize` – Wenn das Rolling aktiviert ist, gibt dies die maximale Größe der Ereignisprotokolldatei an, bevor ein Rollover ausgeführt wird. Der Standardwert ist 128 MB.
- `spark.history.fs.eventLog.rolling.maxFilesToRetain`– Gibt die maximale Anzahl nicht komprimierter Ereignisprotokolldateien an, die aufbewahrt werden sollen. Standardmäßig werden alle Ereignisprotokolldateien aufbewahrt. Stellen Sie einen niedrigeren Wert ein, um ältere Ereignisprotokolle zu komprimieren. Der niedrigste Wert ist 1.

Beachten Sie, dass bei der Komprimierung versucht wird, Ereignisse mit veralteten Ereignisprotokolldateien auszuschließen, wie z. B. die folgenden. Wenn dabei Ereignisse verworfen werden, werden sie nicht mehr auf der Benutzeroberfläche des Spark History Servers angezeigt.

- Ereignisse für abgeschlossene Aufträge und zugehörige Phasen- oder Aufgabenereignisse.
- Ereignisse für beendete Exekutoren.

- Ereignisse für abgeschlossene SQL Anfragen und damit verbundene Aufgaben-, Phasen- und Aufgabenereignisse.

So starten Sie einen Cluster mit aktiviertem Rollen und Komprimieren

1. Erstellen Sie eine Konfigurationsdatei `spark-configuration.json` mit der folgenden Konfiguration.

```
[
   {
     "Classification": "spark-defaults",
        "Properties": {
           "spark.eventLog.rolling.enabled": true,
           "spark.history.fs.eventLog.rolling.maxFilesToRetain": 1
        }
   }
]
```

2. Erstellen Sie den Cluster mit der Spark-Rolling-Compaction-Konfiguration wie folgt.

```
aws emr create-cluster \
--release-label emr-6.6.0 \
--instance-type m4.large \
--instance-count 2 \
--use-default-roles \
--configurations file://spark-configuration.json
```

Überlegungen und Einschränkungen

Der Ein-Klick-Zugriff auf persistente Anwendungsbenutzeroberflächen hat derzeit folgende Einschränkungen.

- Es wird mindestens zwei Minuten dauern, bis die Anwendungsdetails auf der Benutzeroberfläche des Spark History Servers angezeigt werden.
- Diese Funktion funktioniert nur, wenn sich das Ereignisprotokollverzeichnis für die Anwendung im Verzeichnis befindetHDFS. Standardmäßig EMR speichert Amazon Ereignisprotokolle in einem Verzeichnis vonHDFS. Wenn Sie das Standardverzeichnis in ein anderes Dateisystem ändern, beispielsweise Amazon S3, funktioniert das Feature nicht.

- Diese Funktion ist derzeit nicht für EMR Cluster mit mehreren Master-Knoten oder für integrierte EMR Cluster verfügbar AWS Lake Formation.
- Um den Zugriff mit einem Klick auf persistente Anwendungsbenutzeroberflächen zu ermöglichen, müssen Sie über die Berechtigung für die `DescribeCluster` Aktion für Amazon EMR verfügen. Wenn Sie einem IAM Principal die Genehmigung für diese Aktion verweigern, dauert es ungefähr fünf Minuten, bis die Änderung der Zugriffsrechte wirksam wird.
- Wenn Sie Anwendungen in einem laufenden Cluster neu konfigurieren, ist der Anwendungsverlauf nicht über das Anwendungs-UI verfügbar.
- Für jede AWS-Konto Anwendung liegt das Standardlimit für aktive Anwendungen UIs bei 200.
- Im Folgenden AWS-Regionen können Sie mit Amazon EMR 6.14.0 und UIs höher von der Konsole aus auf die Anwendung zugreifen:
  - Asien-Pazifik (Jakarta) (ap-southeast-3)
  - Europa (Spanien) (eu-south-2)
  - Asien-Pazifik (Melbourne) (ap-southeast-4)
  - Israel (Tel Aviv) (il-central-1)
  - Naher Osten (UAE) (me-central-1)
- Im Folgenden AWS-Regionen können Sie mit Amazon EMR 5.25.0 und UIs höher von der Konsole aus auf die Anwendung zugreifen:
  - USA Ost (Nord-Virginia): (us-east-1)
  - USA West (Oregon): (us-west-2)
  - Asien-Pazifik (Mumbai): (ap-south-1)
  - Asien-Pazifik (Seoul): (ap-northeast-2)
  - Asien-Pazifik (Singapur): (ap-southeast-1)
  - Asien-Pazifik (Sydney): (ap-southeast-2)
  - Asien-Pazifik (Tokyo) (ap-northeast-1)
  - Kanada (Zentral): (ca-central-1)
  - Südamerika (São Paulo) (sa-east-1)
  - Europa (Frankfurt) (eu-central-1)
  - Europa (Irland) (eu-west-1)
  - Europa (London) (eu-west-2)
  - Europa (Paris) (eu-west-3)
  - Europa (Stockholm) (eu-north-1)

- China (Peking) (cn-north-1)
- China (Ningxia) (cn-northwest-1)

Einen übergeordneten Anwendungsverlauf anzeigen

> **Note**
>
> Wir empfehlen Ihnen, die persistente Anwendungsoberfläche zu verwenden, um eine bessere Benutzererfahrung zu erzielen. Dabei wird der Anwendungsverlauf bis zu 30 Tage lang gespeichert. Der auf dieser Seite beschriebene allgemeine Anwendungsverlauf ist in der neuen EMR Amazon-Konsole (https://console.aws.amazon.com/emr) nicht verfügbar. Weitere Informationen finden Sie unter Persistente Anwendungsbenutzeroberflächen anzeigen.

Mit den EMR Amazon-Versionen 5.8.0 bis 5.36.0 und 6.x-Versionen bis 6.8.0 können Sie auf der Registerkarte Anwendungsbenutzeroberflächen in der alten Amazon-Konsole einen allgemeinen Anwendungsverlauf anzeigen. EMR Eine EMR Amazon-Anwendungsbenutzeroberfläche speichert die Zusammenfassung des Anwendungsverlaufs für 7 Tage nach Abschluss eines Antrags.

Überlegungen und Einschränkungen

Beachten Sie die folgenden Einschränkungen, wenn Sie den Tab Anwendungsbenutzeroberflächen in der alten EMR Amazon-Konsole verwenden.

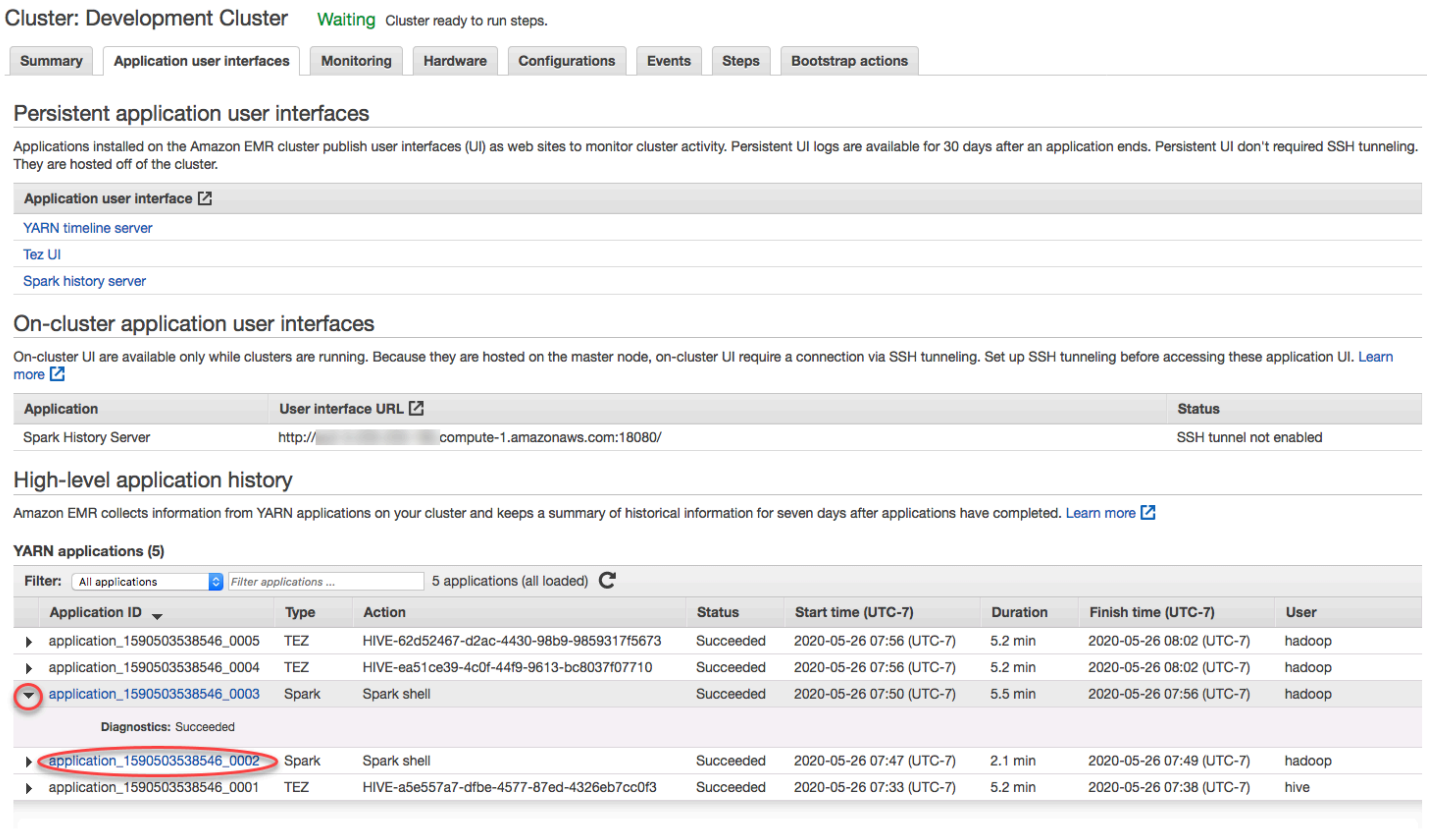
- Sie können nur auf die allgemeine Anwendungsverlaufsfunktion zugreifen, wenn Sie die EMR Amazon-Versionen 5.8.0 bis 5.36.0 und 6.x-Versionen bis 6.8.0 verwenden. Mit Wirkung zum 23. Januar 2023 EMR wird Amazon den High-Level-Anwendungsverlauf für alle Versionen einstellen. Wenn Sie EMR Amazon-Version 5.25.0 oder höher verwenden, empfehlen wir, stattdessen die persistente Anwendungsbenutzeroberfläche zu verwenden.
- Das Feature zum Anwendungsverlauf auf hoher Ebene unterstützt keine Spark-Streaming-Anwendungen.
- Der Ein-Klick-Zugriff auf persistente Anwendungsbenutzeroberflächen ist derzeit nicht für EMR Amazon-Cluster mit mehreren Master-Knoten oder für integrierte EMR Amazon-Cluster verfügbar. AWS Lake Formation

Beispiel: Einen übergeordneten Anwendungsverlauf anzeigen

Die folgende Sequenz zeigt einen Drilldown durch einen Spark oder eine YARN Anwendung zu Jobdetails mithilfe der Registerkarte Anwendungsbenutzeroberflächen auf der Cluster-Detailseite der alten Konsole.

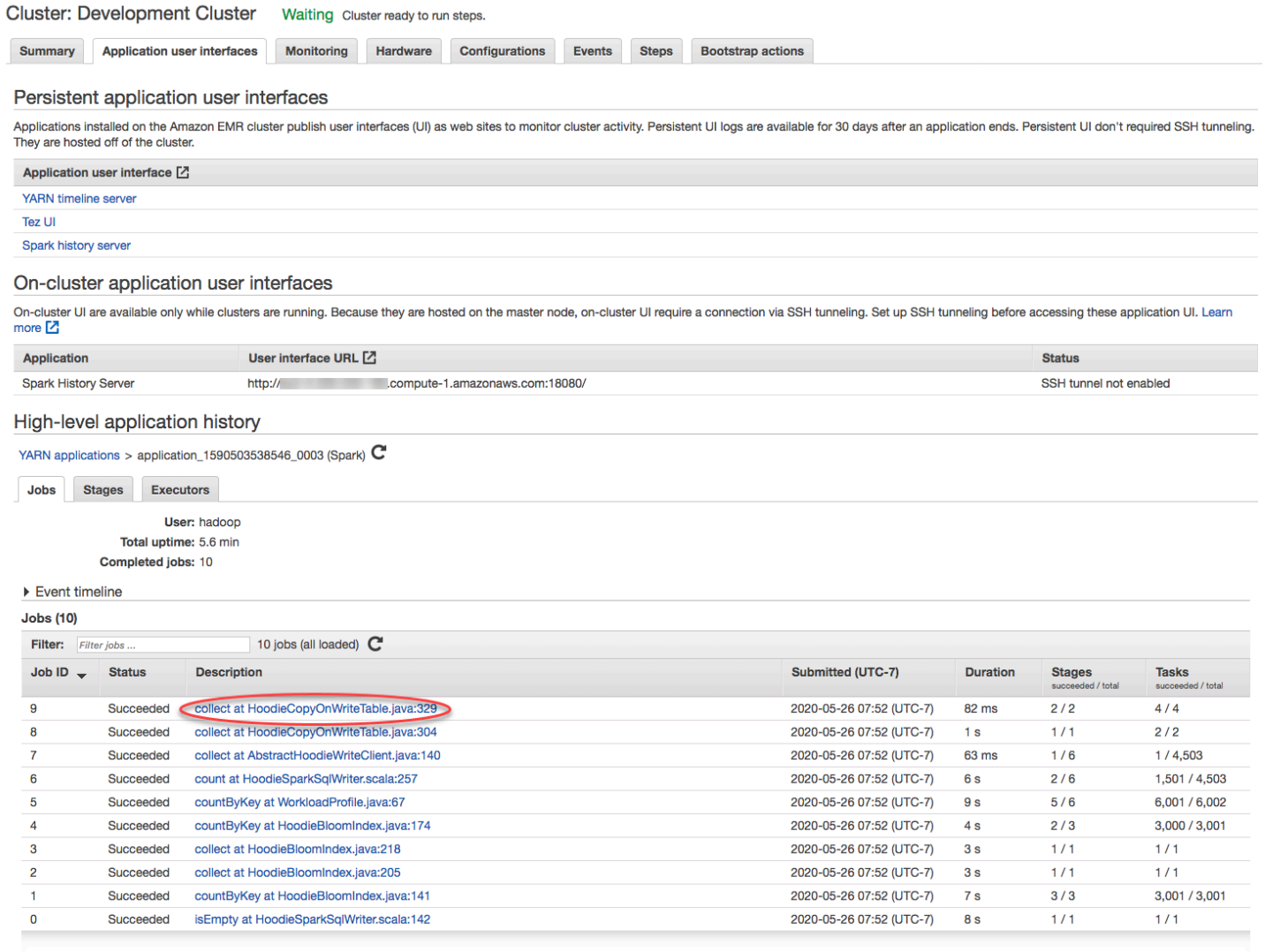
Sie können in der Liste Cluster die Option Name für einen Cluster auswählen, um Details zu diesem anzuzeigen. Um Informationen über YARN Container-Logs anzuzeigen, müssen Sie die Protokollierung für Ihren Cluster aktivieren. Weitere Informationen finden Sie unter Konfigurieren der Cluster-Protokollierung und des Debuggings. Für den Spark-Anwendungsverlauf sind die in der Zusammenfassungstabelle bereitgestellten Informationen nur eine Teilmenge der Informationen, die über die Benutzeroberfläche des Spark-History-Servers verfügbar sind.

Auf der Registerkarte Anwendungsbenutzeroberflächen unter Anwendungsverlauf auf hoher Ebene können Sie eine Zeile erweitern, um die Diagnoseübersicht für eine Spark-Anwendung anzuzeigen, oder einen Anwendungs-ID-Link auswählen, um Details zu einer anderen Anwendung anzuzeigen.

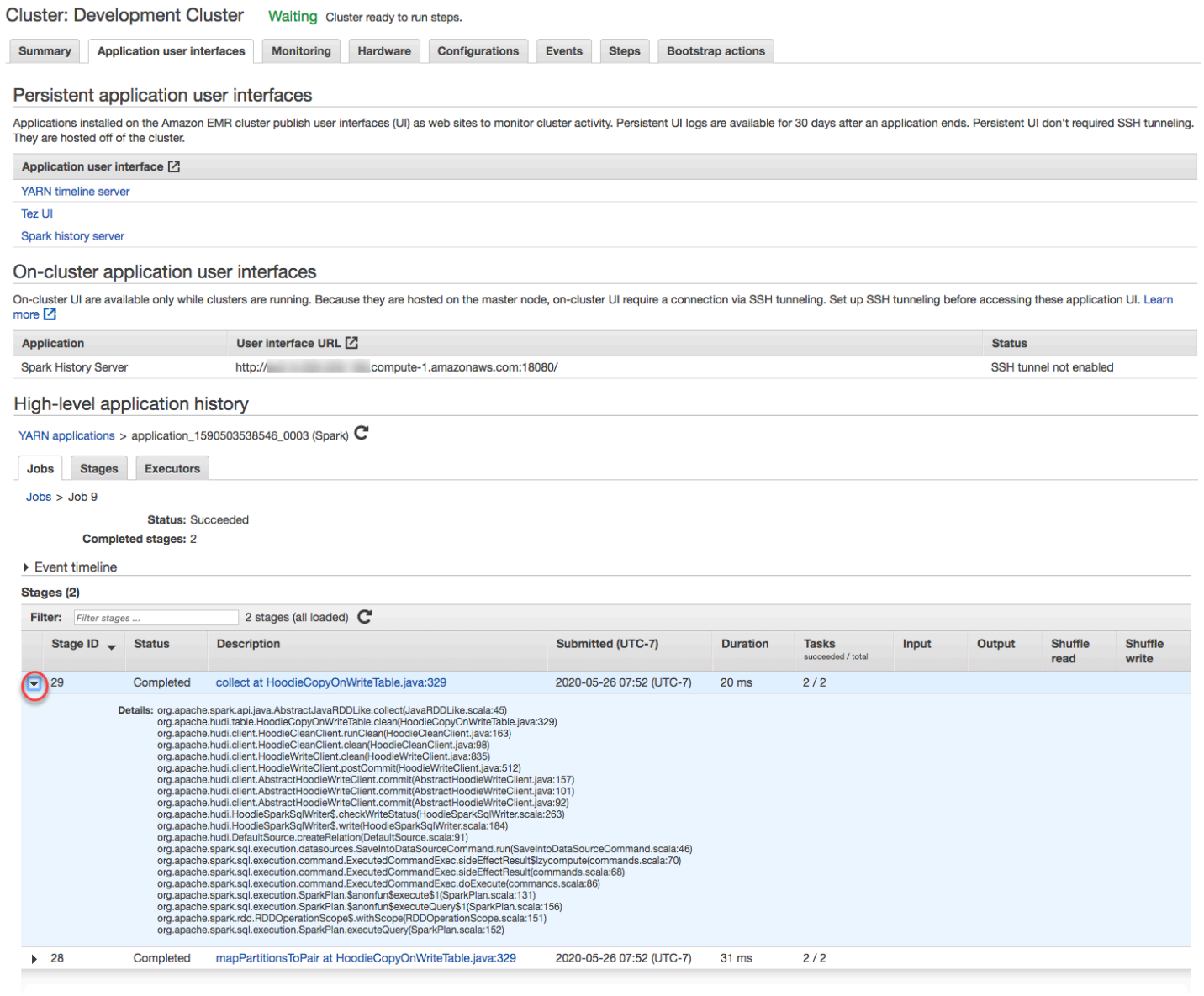


Wenn Sie einen Anwendungs-ID-Link auswählen, ändert sich die Benutzeroberfläche und zeigt die YARNAnwendungsdetails für diese Anwendung an. Auf der Registerkarte „Jobs“ der

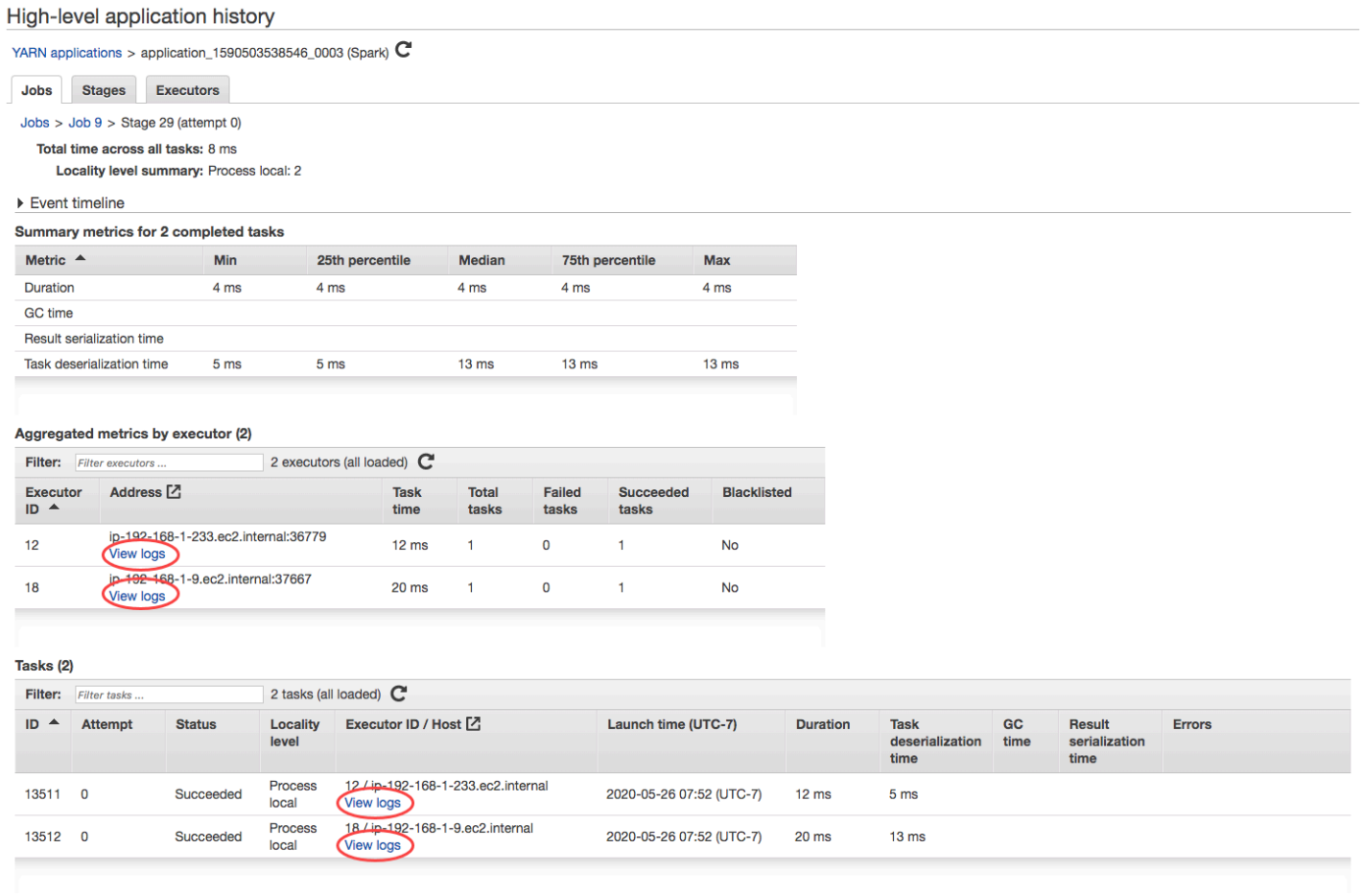
YARNBewerbungsdetails können Sie den Link Beschreibung für eine Stelle auswählen, um Details zu dieser Stelle anzuzeigen.



Auf der Seite der Auftragsdetails können Sie Informationen zu einzelnen Auftragsphasen erweitern und dann auf den Link Beschreibung klicken, um die Phasendetails anzuzeigen.



Auf der Seite mit den Phasendetails können Sie die wichtigsten Kennzahlen für die Aufgaben und Ausführenden der Phase einsehen. Sie können Aufgaben- und Ausführungsprotokolle auch über die Links Protokolle anzeigen ansehen.



Anzeige von -Protokolldateien

Amazon EMR und Hadoop erstellen beide Protokolldateien, die den Status des Clusters melden. Standardmäßig werden diese Dateien im Primärknoten im `/mnt/var/log/`-Verzeichnis gespeichert. Abhängig von der Konfiguration Ihres Clusters beim Start können diese Protokolle auch in Amazon S3 archiviert und über das grafische Debugging-Tool angezeigt werden.

Es gibt viele Arten von Protokollen, die auf dem Primärknoten gespeichert werden. Amazon EMR schreibt Step-, Bootstrap-Action- und Instance-Status-Logs. Apache Hadoop erstellt Protokolle mit Daten zur Verarbeitung von Aufträgen, Aufgaben und versuchten Aufgaben. Hadoop protokolliert außerdem Protokolle seiner Daemons. Weitere Informationen zu den von Hadoop geschriebenen Protokollen finden Sie unter http://hadoop.apache.org/docs/stable/ hadoop-project-dist ClusterSetup / hadoop-common/ .html.

Protokolldateien auf dem Primärknoten anzeigen

Die folgende Tabelle listet einige der Protokolldateien auf, die auf dem Primärknoten zu finden sind.

Ort	Beschreibung
/emr/instance-controller/log/bootstrap-actions	Protokolle, die bei der Verarbeitung von Bootstrap-Aktionen geschrieben werden.
/mnt/var/log/ hadoop-state-pusher	Protokolle, die vom Hadoop-Status-Push-Prozess geschrieben werden.
/emr/instance-controller/log	Instance-Controller-Protokolle.
/emr/instance-state	instance-Statusprotokolle. Diese enthalten Informationen über den Speicherstatus und die CPU Garbage-Collector-Threads des Knotens.
/emr/service-nanny	Protokolle, die vom Service-Nanny-Prozess geschrieben werden.
/mnt/var/log/*application*	Protokolle, die sich auf eine bestimmte Anwendung beziehen, wie z. B. Hadoop, Spark oder Hive.
/mnt/var/log/hadoop/steps/*N*	Schrittprotokolle, die Informationen über die Verarbeitung des Schritts enthalten. Der Wert von *N* gibt die von Amazon stepId zugewiese ne anEMR. Beispiel: Ein Cluster verfügt über zwei Schritte: `s-1234ABCDEFGH` und `s-5678IJKLMNOP` . Der erste Schritt befindet sich in `/mnt/var/log/hadoop/steps/s-1234ABCDEFGH/` und der zweite in `/mnt/var/log/hadoop/steps/s-5678IJKLMNOP/` . Die von Amazon geschriebenen Schrittpr otokolle EMR lauten wie folgt. • controller – Informationen zur Verarbeitung des Schritts. Wenn Ihr Schritt beim Laden fehlschlägt, finden Sie den Stack-Trace in diesem Protokoll.

Ort	Beschreibung
	• syslog – Beschreibt die Ausführung von Hadoop-Jobs in diesem Schritt. • stderr – Der Standardfehlerkanal von Hadoop bei der Verarbeitung des Schritts. • stdout – Der Standardausgabekanal von Hadoop während der Verarbeitung des Schritts.

So zeigen Sie Protokolldateien auf dem Primärknoten mit dem AWS CLI an.

1. Verwenden Sie diese Option, SSH um eine Verbindung zum primären Knoten herzustellen, wie unter beschriebenConnect zum Primärknoten her mit SSH.
2. Navigieren Sie zu dem Verzeichnis mit den Protokolldateiinformationen, die Sie anzeigen möchten. Die oben stehenden Tabelle gibt eine Liste der verfügbaren Protokolldateien mit dem entsprechenden Speicherort an. Das folgende Beispiel zeigt den Befehl für die Navigation zum Schrittprotokoll mit einer ID, `s-1234ABCDEFGH`.

   ```
   cd /mnt/var/log/hadoop/steps/s-1234ABCDEFGH/
   ```

3. Verwenden Sie einen Datei-Viewer Ihrer Wahl, um die Protokolldatei anzuzeigen. Im folgenden Beispiel wird der Linux-Befehl `less` verwendet, um die Protokolldatei `controller` anzuzeigen.

   ```
   less controller
   ```

In Amazon S3 archivierte Protokolldateien anzeigen

Standardmäßig archivieren EMR Amazon-Cluster, die über die Konsole gestartet werden, automatisch Protokolldateien in Amazon S3. Sie können einen eigenen Protokollpfad angeben, und zulassen, dass die Konsole automatisch einen Protokollpfad generiert. Für Cluster, die mit dem CLI oder gestartet wurdenAPI, müssen Sie die Amazon S3 S3-Protokollarchivierung manuell konfigurieren.

Wenn Amazon so konfiguriert EMR ist, dass es Protokolldateien in Amazon S3 archiviert, speichert es die Dateien an dem von Ihnen angegebenen S3-Speicherort, im/*cluster-id*/Ordner, wo *cluster-id* ist die Cluster-ID.

Die folgende Tabelle listet einige der Protokolldateien auf, die in Amazon S3 zu finden sind.

Ort	Beschreibung
/*cluster-id* /node/	Knotenprotokolle, einschließlich Bootstrap -Aktion, Instance-Status und Anwendung sprotokollen für den Knoten. Die Protokoll e für jeden Knoten werden in einem Ordner gespeichert, der mit der ID der EC2 Instanz dieses Knotens beschriftet ist.
/*cluster-id* /node/*instance-id* /*application*	Die Protokolle, die von einzelnen Anwendung en oder Daemons, die einer Anwendung zugeordnet sind, erstellt wurden. Das Hive-Server-Protokoll befindet sich beispielsweise im Verzeichnis *cluster-id* /node/*instance-id* /hive/hive-server.log .
/*cluster-id* /Schritte/*step-id*/	Schrittprotokolle, die Informationen über die Verarbeitung des Schritts enthalten. Der Wert von *step-id* gibt die von Amazon zugewiese ne Schritt-ID anEMR. Beispiel: Ein Cluster verfügt über zwei Schritte: s-1234ABCDEFGH und s-5678IJKLMNOP . Der erste Schritt befindet sich in /mnt/var/log/hadoop/steps/s-1234ABCDEFGH/ und der zweite in /mnt/var/log/hadoop/steps/s-5678IJKLMNOP/ . Die von Amazon geschriebenen Schrittpr otokolle EMR lauten wie folgt. • controller – Informationen zur Verarbeitung des Schritts. Wenn Ihr Schritt beim Laden

Ort	Beschreibung
	fehlschlägt, finden Sie den Stack-Trace in diesem Protokoll. • syslog – Beschreibt die Ausführung von Hadoop-Jobs in diesem Schritt. • stderr – Der Standardfehlerkanal von Hadoop bei der Verarbeitung des Schritts. • stdout – Der Standardausgabekanal von Hadoop während der Verarbeitung des Schritts.
/*cluster-id* containers/	Anwendungscontainerprotokolle. Die Protokolle für jede YARN Anwendung werden an diesen Orten gespeichert.
/*cluster-id* /hadoop-mapreduce/	Die Protokolle, die Informationen über Konfigurationsdetails und den Jobverlauf von Jobs enthalten. MapReduce

So zeigen Sie Protokolldateien an, die mit der Amazon-S3-Konsole in Amazon S3 archiviert wurden

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die Amazon S3 S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Öffnen Sie den S3-Bucket, den Sie angegeben haben, als Sie den Cluster für die Archivierung von Protokolldateien in Amazon S3 konfiguriert haben.
3. Navigieren Sie zu der Protokolldatei, die die Informationen enthält, die angezeigt werden sollen. Die oben stehenden Tabelle gibt eine Liste der verfügbaren Protokolldateien mit dem entsprechenden Speicherort an.
4. Laden Sie das Protokolldateiobjekt herunter, um es anzuzeigen. Anweisungen finden Sie unter Objekt herunterladen.

Cluster-Instances in Amazon anzeigen EC2

Um Ihnen bei der Verwaltung Ihrer Ressourcen zu helfen, EC2 ermöglicht Ihnen Amazon, Ressourcen Metadaten in Form von Tags zuzuweisen. Jedes EC2 Amazon-Tag besteht aus

einem Schlüssel und einem Wert. Mithilfe von Tags können Sie Ihre EC2 Amazon-Ressourcen auf unterschiedliche Weise kategorisieren: zum Beispiel nach Zweck, Eigentümer oder Umgebung.

Sie können die Ressourcen auf Grundlage der Tags suchen und filtern. Die Tags, die Sie Ressourcen über Ihr AWS Konto zuweisen, stehen nur Ihnen zur Verfügung. Andere Konten, die dieselbe Ressource nutzen, können Ihre Tags nicht sehen.

Amazon EMR kennzeichnet automatisch jede EC2 Instance, die es startet, mit Schlüssel-Wert-Paaren. Die Schlüssel identifizieren den Cluster und die Instance-Gruppe, zu der die Instance gehört. Dies macht es einfach, Ihre EC2 Instances zu filtern, um beispielsweise nur die Instances anzuzeigen, die zu einem bestimmten Cluster gehören, oder um alle aktuell laufenden Instances in der Instance-Gruppe für die Aufgabe anzuzeigen. Dies ist besonders nützlich, wenn Sie mehrere Cluster gleichzeitig ausführen oder eine große Anzahl von EC2 Instanzen verwalten.

Dies sind die vordefinierten Schlüssel-Wert-Paare, die Amazon EMR zuweist:

Schlüssel	Wert	Wert-Definition
aws:elasticmapreduce:job-flow-id	*job-flow-identifier*	Die ID des Clusters, für den die Instance bereitgestellt wird. Es wird im folgenden Format `j-XXXXXXXXXXXXX` angezeigt und kann bis zu 256 Zeichen lang sein.
aws:elasticmapreduce:instance-group-role	*group-role*	Der Typ der Instance-Gruppe, eingegeben als einer der folgenden Werte: `master`, `core` oder `task`.

Sie können die von Amazon hinzugefügten Tags anzeigen und nach ihnen EMR filtern. Weitere Informationen finden Sie unter Verwenden von Tags im EC2Amazon-Benutzerhandbuch. Da es sich bei den von Amazon festgelegten Tags um System-Tags EMR handelt, die weder bearbeitet noch gelöscht werden können, sind die Abschnitte zum Anzeigen und Filtern von Tags am relevantesten.

> **Note**
>
> Amazon EMR fügt der EC2 Instance Tags hinzu, wenn ihr Status auf Wird ausgeführt aktualisiert wird. Wenn zwischen dem Zeitpunkt, zu dem die EC2 Instance bereitgestellt wird, und dem Zeitpunkt, zu dem ihr Status auf Running gesetzt wird, Latenz auftritt, werden die

von Amazon EMR festgelegten Tags angezeigt, sobald die Instance gestartet wird. Wenn keine Tags angezeigt werden, warten Sie einige Minuten und aktualisieren Sie die Ansicht.

CloudWatch Ereignisse und Metriken

Verwenden Sie Ereignisse und Metriken, um die Aktivität und den Zustand eines EMR Amazon-Clusters zu verfolgen. Ereignisse sind nützlich zur Überwachung bestimmter Vorgänge in einem Cluster, beispielsweise wenn sich der Zustand eines Clusters vom Starten zum Ausführen ändert. Metriken sind nützlich, um einen bestimmten Wert zu überwachen, z. B. den Prozentsatz des verfügbaren Festplattenspeichers, der innerhalb eines Clusters genutzt HDFS wird.

Weitere Informationen zu CloudWatch Veranstaltungen finden Sie im Amazon CloudWatch Events-Benutzerhandbuch. Weitere Informationen zu CloudWatch Metriken finden Sie unter Verwenden von CloudWatch Amazon-Metriken und Erstellen von CloudWatch Amazon-Alarmen im CloudWatch Amazon-Benutzerhandbuch.

Themen

- Überwachung von EMR Amazon-Metriken mit CloudWatch
- Überwachung von EMR Amazon-Ereignissen mit CloudWatch
- Auf CloudWatch Ereignisse reagieren

Überwachung von EMR Amazon-Metriken mit CloudWatch

Die Metriken werden alle fünf Minuten aktualisiert und automatisch CloudWatch für jeden EMR Amazon-Cluster gesammelt und weitergeleitet. Dieses Intervall kann nicht konfiguriert werden. Für die unter angegebenen EMR Amazon-Metriken fallen keine Gebühren an CloudWatch. Diese fünfminütigen Datenpunktmetriken werden 63 Tage lang archiviert. Danach werden die Daten verworfen.

Wie verwende ich EMR Amazon-Metriken?

Die folgende Tabelle zeigt die häufigsten Verwendungszwecke für von Amazon gemeldete MetrikenEMR. Es handelt sich dabei um Vorschläge für den Einstieg und nicht um eine umfassende Liste. Eine vollständige Liste der von Amazon EMR gemeldeten Kennzahlen finden Sie unterVon Amazon gemeldete Metriken EMR in CloudWatch.

Wie gehe ich vor?	Relevante Metriken
Verfolgen des Cluster-Fortschritts	Sehen Sie sich die Metriken `RunningMapTasks` , `RemainingMapTasks` , `RunningReduceTasks` und `RemainingReduceTasks` an.
Erkennen von Clustern im Leerlauf	Die `IsIdle`-Metrik verfolgt, ob ein Cluster verfügbar ist, aber aktuell keine Aufgaben ausführt. Sie können einen Alarm einrichten, wenn sich der Cluster für einen bestimmten Zeitraum im Leerlauf befunden hat z. B. 30 Minuten.
Erkennen, wenn ein Knoten zu wenig Speicherplatz hat	Die `MRUnhealthyNodes` Metrik verfolgt, wann einem oder mehreren Kern- oder Taskknoten der lokale Festplattenspeicher ausgeht und sie in einen bestimmten `UNHEALTHY` YARN Status übergehen. Zum Beispiel haben Core- oder Aufgabenknoten nur noch wenig Speicherplatz zur Verfügung und sie können keine Aufgaben ausführen.
Erkennen, wenn ein Cluster zu wenig Speicherplatz hat	Die `HDFSUtilization` Metrik überwacht die kombinierte HDFS Kapazität des Clusters und kann eine Größenänderung des Clusters erfordern, um weitere Kernknoten hinzuzufügen. Beispielsweise ist die HDFS Auslastung hoch, was sich auf Jobs und den Zustand des Clusters auswirken kann.
Erkennt, wenn ein Cluster mit reduzierter Kapazität läuft	Die `MRLostNodes` -Metrik verfolgt, wann ein oder mehrere Core- oder Aufgabenknoten nicht mit dem Hauptknoten kommunizieren können. Beispielsweise ist der Core- oder Aufgabenknoten für den Hauptknoten nicht erreichbar.

Weitere Informationen finden Sie unter Der Cluster endet mit SLAVE NO_ _ und den Kernknoten _BY_ LEFT FAILED MASTER und AWSSupport-A nalyzeEMRLogs.

CloudWatch Zugriffsmetriken für Amazon EMR

Sie können die Metriken, über die Amazon EMR berichtet, CloudWatch über die EMR Amazon-Konsole oder die CloudWatch Konsole anzeigen. Sie können Metriken auch mit dem CloudWatch CLI Befehl `mon-get-stats` oder dem abrufen CloudWatch `GetMetricStatistics`API. Weitere Informationen zum Anzeigen oder Abrufen von Messwerten für Amazon EMR finden CloudWatch Sie im CloudWatch Amazon-Benutzerhandbuch.

Console

So zeigen Sie Metriken mit der Konsole an

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und dann den Cluster aus, für den Sie Metriken anzeigen möchten. Dadurch wird die Cluster-Detailseite geöffnet.
3. Wählen Sie auf der Cluster-Detailseite die Registerkarte Überwachung aus. Wählen Sie eine der Optionen Clusterstatus, Knotenstatus oder Ein- und Ausgaben aus, um die Berichte über den Fortschritt und den Zustand des Clusters zu laden.
4. Nachdem Sie eine Metrik zur Anzeige ausgewählt haben, können Sie jedes Diagramm vergrößern. Um den Zeitrahmen Ihres Diagramms zu filtern, wählen Sie eine vorausgefüllte Option oder wählen Sie Benutzerdefiniert.

Von Amazon gemeldete Metriken EMR in CloudWatch

In den folgenden Tabellen sind die Metriken aufgeführt, die Amazon in der Konsole EMR meldet und an CloudWatch die Amazon weiterleitet.

EMRAmazon-Metriken

Amazon EMR sendet Daten für verschiedene Metriken an CloudWatch. Alle EMR Amazon-Cluster senden automatisch Metriken in Intervallen von fünf Minuten. Die Metriken werden für zwei Wochen archiviert. Nach Ablauf dieses Zeitraums werden die Daten verworfen.

Der `AWS/ElasticMapReduce`-Namespace enthält die folgenden Metriken.

> **Note**
>
> Amazon EMR ruft Metriken aus einem Cluster ab. Wenn die Verbindung zu einem Cluster verloren geht, werden keine Metriken gemeldet, bis der Cluster wieder verfügbar ist.

Die folgenden Metriken sind für Cluster mit Hadoop 2.x -Versionen verfügbar.

Metrik	Beschreibung
Cluster-Status	
IsIdle	Gibt an, dass ein Cluster keine Arbeiten mehr ausführt, aber unverändert aktiv ist und Kosten verursacht. Der Wert beträgt 1, wenn weder Tasks noch Aufträge ausgeführt werden, andernfalls beträgt der Wert 0. Dieser Wert wird in 5-Minuten-Intervallen geprüft. Wenn der Wert 1 beträgt, bedeutet dies, dass der Cluster zum Zeitpunkt der Prüfung ungenutzt war, aber nicht die gesamten fünf Minuten. Um Falschmeldungen zu vermeiden, sollten Sie einen Alarm auslösen, wenn dieser Wert in mehreren aufeinand er folgenden 5-Minuten-Prüfungen 1 beträgt. Sie können zum Beispiel einen Alarm auslösen, wenn dieser Wert 30 Minuten oder länger 1 beträgt. Anwendungsfall: Cluster-Leistung überwachen Einheiten: boolescher Wert
ContainerAllocated	Die Anzahl der Ressourcencontainer, die von der ResourceManager zugewiesen wurden. Anwendungsfall: Cluster-Fortschritt überwachen Einheiten: Anzahl
ContainerReserved	Anzahl der reservierten Container. Anwendungsfall: Cluster-Fortschritt überwachen

Metrik	Beschreibung
	Einheiten: Anzahl
ContainerPending	Anzahl der Container in der Warteschlange, die noch nicht zugeordnet worden sind. Anwendungsfall: Cluster-Fortschritt überwachen Einheiten: Anzahl
ContainerPendingRatio	Das Verhältnis von ausstehenden Containern zu zugewiesenen Containern (ContainerPendingRatio = ContainerPending / ContainerAllocated). Wenn ContainerAllocated = 0, dann ContainerPendingRatio =ContainerPending. Der Wert von ContainerPendingRatio steht für eine Zahl, nicht für einen Prozentsatz. Dieser Wert ist zum Skalieren von Cluster-Ressourcen anhand des Zuordnungsverhaltens des Containers hilfreich. Einheiten: Anzahl
AppsCompleted	Die Anzahl der eingereichten Anträge, YARN die abgeschlossen wurden. Anwendungsfall: Cluster-Fortschritt überwachen Einheiten: Anzahl
AppsFailed	Die Anzahl der YARN eingereichten Anträge wurde nicht abgeschlossen. Anwendungsfall: Cluster-Fortschritt überwachen, Cluster-Zustand überwachen Einheiten: Anzahl

Metrik	Beschreibung
AppsKilled	Die Anzahl der YARN eingereichten Anträge wurde abgelehnt. Anwendungsfall: Cluster-Fortschritt überwachen, Cluster-Zustand überwachen Einheiten: Anzahl
AppsPending	Die Anzahl der bei YARN dieser Stelle eingereichten Anträge ist noch nicht abgeschlossen. Anwendungsfall: Cluster-Fortschritt überwachen Einheiten: Anzahl
AppsRunning	Die Anzahl der Bewerbungen, die bei YARN diesem Dienst eingereicht wurden, laufen. Anwendungsfall: Cluster-Fortschritt überwachen Einheiten: Anzahl
AppsSubmitted	Die Anzahl der Anträge, die bei eingereicht wurdenYARN. Anwendungsfall: Cluster-Fortschritt überwachen Einheiten: Anzahl
Knotenstatus	
CoreNodesRunning	Anzahl der arbeitenden Core-Knoten. Die Datenpunkte dieser Metrik werden nur dann angegeben, wenn die zugehörige Instance-Gruppe existiert. Anwendungsfall: Cluster-Zustand überwachen Einheiten: Anzahl

Metrik	Beschreibung
CoreNodesPending	Anzahl der Core-Knoten, die auf eine Zuordnung warten. Es müssen nicht alle angeforderten Core-Knoten sofort verfügbar sein. Diese Metrik gibt die ausstehenden Anforderungen an. Die Datenpunkte dieser Metrik werden nur dann angegeben, wenn die zugehörige Instance-Gruppe existiert. Anwendungsfall: Cluster-Zustand überwachen Einheiten: Anzahl
LiveDataNodes	Prozentsatz der Datenknoten, die Arbeit von Hadoop empfangen. Anwendungsfall: Cluster-Zustand überwachen Einheiten: Prozent
MRTotalNodes	Die Anzahl der Knoten, die derzeit für MapReduce Jobs verfügbar sind. Entspricht einer YARN Metrik`mapred.resourcemanager.TotalNodes` . Anwendungsfall: Cluster-Fortschritt überwachen Einheiten: Anzahl
MRActiveNodes	Die Anzahl der Knoten, auf denen derzeit MapReduce Aufgaben oder Jobs ausgeführt werden. Entspricht einer YARN Metrik`mapred.resourcemanager.NoOf ActiveNodes` . Anwendungsfall: Cluster-Fortschritt überwachen Einheiten: Anzahl

Metrik	Beschreibung
MRLostNodes	Die Anzahl der Knoten MapReduce , denen zugewiesen wurde, die in einem LOST Status markiert wurden. Entsprich t einer YARN Metrik`mapred.resourcemanager.NoOf LostNodes` . Anwendungsfall: Cluster-Fortschritt überwachen, Cluster-Z ustand überwachen Einheiten: Anzahl
MRUnhealthyNodes	Die Anzahl der Knoten, die für MapReduce Jobs verfügbar sind, die in einem UNHEALTHY Status markiert sind. Entspricht einer YARN Metrik`mapred.resourceman ager.NoOfUnhealthyNodes` . Anwendungsfall: Cluster-Fortschritt überwachen Einheiten: Anzahl
MRDecommissionedNodes	Die Anzahl der Knoten, die MapReduce Anwendungen zugewiesen sind, die als DECOMMISSIONED Status markiert wurden. Entspricht einer YARN Metrik`mapred.re sourcemanager.NoOfDecommissionedNodes` . Anwendungsfall: Cluster-Fortschritt überwachen, Cluster-Z ustand überwachen Einheiten: Anzahl
MRRebootedNodes	Die Anzahl der verfügbaren Knoten, MapReduce die neu gestartet und als Status markiert wurden. REBOOTED Entspricht einer Metrik. YARN `mapred.resourceman ager.NoOfRebootedNodes` Anwendungsfall: Cluster-Fortschritt überwachen, Cluster-Z ustand überwachen Einheiten: Anzahl

Metrik	Beschreibung
MultiMasterInstanceGroupNodesRunning	Die Anzahl der zurzeit ausgeführten Master-Knoten. Anwendungsfall: Überwachen von Ausfall und Ersetzung eines Master-Knotens Einheiten: Anzahl
MultiMasterInstanceGroupNodesRunningPercentage	Der Prozentsatz der zurzeit im Verhältnis zur angeforderten Instance-Zahl für Master-Knoten ausgeführten Master-Knoten. Anwendungsfall: Überwachen von Ausfall und Ersetzung eines Master-Knotens Einheiten: Prozent
MultiMasterInstanceGroupNodesRequested	Die Anzahl der angeforderten Master-Knoten. Anwendungsfall: Überwachen von Ausfall und Ersetzung eines Master-Knotens Einheiten: Anzahl
IO	
S3 BytesWritten	Anzahl der auf Amazon S3 geschriebenen Bytes. Diese Metrik aggregiert nur MapReduce Jobs und gilt nicht für andere Workloads bei Amazon. EMR Anwendungsfall: Cluster-Leistung analysieren, Cluster-Fortschritt überwachen Einheiten: Anzahl

Metrik	Beschreibung
S3 BytesRead	Anzahl der von Amazon S3 gelesenen Bytes. Diese Metrik aggregiert nur MapReduce Jobs und gilt nicht für andere Workloads bei Amazon. EMR Anwendungsfall: Cluster-Leistung analysieren, Cluster-F ortschritt überwachen Einheiten: Anzahl
HDFSUtilization	Der Prozentsatz des aktuell genutzten HDFS Speichers. Anwendungsfall: Cluster-Leistung analysieren Einheiten: Prozent
HDFSBytesRead	Die Anzahl der Byte, aus denen gelesen wurdeHDFS. Diese Metrik aggregiert nur MapReduce Jobs und gilt nicht für andere Workloads bei Amazon. EMR Anwendungsfall: Cluster-Leistung analysieren, Cluster-F ortschritt überwachen Einheiten: Anzahl
HDFSBytesWritten	Die Anzahl der Byte, in die geschrieben wurde. HDFS Diese Metrik aggregiert nur MapReduce Jobs und gilt nicht für andere Workloads bei Amazon. EMR Anwendungsfall: Cluster-Leistung analysieren, Cluster-F ortschritt überwachen Einheiten: Anzahl
MissingBlocks	Die Anzahl der Blöcke, in denen es keine HDFS Replikate gibt. Hierbei kann es sich um beschädigte Blöcke handeln. Anwendungsfall: Cluster-Zustand überwachen Einheiten: Anzahl

Metrik	Beschreibung
CorruptBlocks	Die Anzahl der Blöcke, die als beschädigt HDFS gemeldet werden. Anwendungsfall: Cluster-Zustand überwachen Einheiten: Anzahl
TotalLoad	Gesamtanzahl der gleichzeitigen Datenübertragungen. Anwendungsfall: Cluster-Zustand überwachen Einheiten: Anzahl
MemoryTotalMB	Gesamtgröße des Speichers im Cluster. Anwendungsfall: Cluster-Fortschritt überwachen Einheiten: Anzahl
MemoryReservedMB	Größe des reservierten Speichers. Anwendungsfall: Cluster-Fortschritt überwachen Einheiten: Anzahl
MemoryAvailableMB	Verfügbarer zuzuordnender Speicher. Anwendungsfall: Cluster-Fortschritt überwachen Einheiten: Anzahl
YARNMemoryAvailablePercentage	Der Prozentsatz des verbleibenden Speichers, der für verfügbar ist YARN (YARNMemoryAvailablePercentage= MemoryAvailable MB/MemoryTotalMB). Dieser Wert ist nützlich für die Skalierung von Clusterressourcen auf der Grundlage der YARN Speichernutzung. Einheiten: Prozent

Metrik	Beschreibung
MemoryAllocatedMB	Menge des dem Cluster zugeordneten Speichers. Anwendungsfall: Cluster-Fortschritt überwachen Einheiten: Anzahl
PendingDeletionBlocks	Anzahl der zum Löschen gekennzeichneten Blöcke. Anwendungsfall: Cluster-Fortschritt überwachen, Cluster-Zustand überwachen Einheiten: Anzahl
UnderReplicatedBlocks	Anzahl der Blöcke, die nochmals repliziert werden müssen. Anwendungsfall: Cluster-Fortschritt überwachen, Cluster-Zustand überwachen Einheiten: Anzahl
DfsPendingReplicationBlocks	Status der Blockreplikation: replizierte Blöcke, Alter der Replikationsanforderung und nicht erfolgreiche Replikationsanforderungen. Anwendungsfall: Cluster-Fortschritt überwachen, Cluster-Zustand überwachen Einheiten: Anzahl
CapacityRemainingGB	Die Menge der verbleibenden HDFS Festplattenkapazität. Anwendungsfall: Cluster-Fortschritt überwachen, Cluster-Zustand überwachen Einheiten: Anzahl

Nachfolgend sind die Hadoop 1-Metriken aufgeführt:

Metrik	Beschreibung
Cluster-Status	
IsIdle	Gibt an, dass ein Cluster keine Arbeiten mehr ausführt, aber unverändert aktiv ist und Kosten verursacht. Der Wert beträgt 1, wenn weder Tasks noch Aufträge ausgeführt werden, andernfalls beträgt der Wert 0. Dieser Wert wird in 5-Minuten-Intervallen geprüft. Wenn der Wert 1 beträgt, bedeutet dies, dass der Cluster zum Zeitpunkt der Prüfung ungenutzt war, aber nicht die gesamten fünf Minuten. Um Falschmeldungen zu vermeiden, sollten Sie einen Alarm auslösen, wenn dieser Wert in mehreren aufeinand er folgenden 5-Minuten-Prüfungen 1 beträgt. Sie können zum Beispiel einen Alarm auslösen, wenn dieser Wert 30 Minuten oder länger 1 beträgt. Anwendungsfall: Cluster-Leistung überwachen Einheiten: boolescher Wert
JobsRunning	Anzahl der Aufträge im Cluster, die gegenwärtig ausgeführt werden. Anwendungsfall: Cluster-Zustand überwachen Einheiten: Anzahl
JobsFailed	Anzahl der fehlgeschlagenen Aufträge im Cluster. Anwendungsfall: Cluster-Zustand überwachen Einheiten: Anzahl
Map/Reduce	
MapTasksRunning	Anzahl der Map-Tasks für jeden Auftrag. Wenn Sie einen Scheduler installiert haben und mehrere Aufträge ausführen, werden mehrere Grafiken erstellt.

Metrik	Beschreibung
	Anwendungsfall: Cluster-Fortschritt überwachen Einheiten: Anzahl
MapTasksRemaining	Anzahl der verbleibenden Map-Tasks für jeden Auftrag. Wenn Sie einen Scheduler installiert haben und mehrere Aufträge ausführen, werden mehrere Grafiken erstellt. Eine verbleibende Map-Task ist eine Task, die sich in keinem der folgenden Zustände befindet: Running, Killed oder Completed. Anwendungsfall: Cluster-Fortschritt überwachen Einheiten: Anzahl
MapSlotsOpen	Ungenutzte Kapazität für Map-Tasks. Dies wird als die maximale Anzahl von Map-Tasks für einen bestimmte n Cluster abzüglich der Gesamtanzahl der gegenwärtig ausgeführten Map-Tasks in diesem Cluster berechnet. Anwendungsfall: Cluster-Leistung analysieren Einheiten: Anzahl
RemainingMapTasksPerSlot	Das Verhältnis der insgesamt verbleibenden Map-Tasks , bezogen auf die insgesamt verfügbaren Map-Slots im Cluster. Anwendungsfall: Cluster-Leistung analysieren Einheiten: Verhältnis
ReduceTasksRunning	Anzahl der laufenden Reduce-Tasks für jeden Auftrag. Wenn Sie einen Scheduler installiert haben und mehrere Aufträge ausführen, werden mehrere Grafiken erstellt. Anwendungsfall: Cluster-Fortschritt überwachen Einheiten: Anzahl

Metrik	Beschreibung
ReduceTasksRemaining	Anzahl der verbleibenden Reduce-Tasks für jeden Auftrag. Wenn Sie einen Scheduler installiert haben und mehrere Aufträge ausführen, werden mehrere Grafiken erstellt. Anwendungsfall: Cluster-Fortschritt überwachen Einheiten: Anzahl
ReduceSlotsOpen	Ungenutzte Kapazität für Reduce-Tasks. Dies wird als die maximale Anzahl von Reduce-Tasks für einen bestimmte n Cluster abzüglich der Gesamtanzahl der gegenwärtig ausgeführten Reduce-Tasks in diesem Cluster berechnet. Anwendungsfall: Cluster-Leistung analysieren Einheiten: Anzahl
Knotenstatus	
CoreNodesRunning	Anzahl der arbeitenden Core-Knoten. Die Datenpunk te dieser Metrik werden nur dann angegeben, wenn die zugehörige Instance-Gruppe existiert. Anwendungsfall: Cluster-Zustand überwachen Einheiten: Anzahl
CoreNodesPending	Anzahl der Core-Knoten, die auf eine Zuordnung warten. Es müssen nicht alle angeforderten Core-Knoten sofort verfügbar sein. Diese Metrik gibt die ausstehenden Anforderungen an. Die Datenpunkte dieser Metrik werden nur dann angegeben, wenn die zugehörige Instance-Gruppe existiert. Anwendungsfall: Cluster-Zustand überwachen Einheiten: Anzahl

Metrik	Beschreibung
LiveDataNodes	Prozentsatz der Datenknoten, die Arbeit von Hadoop empfangen. Anwendungsfall: Cluster-Zustand überwachen Einheiten: Prozent
TaskNodesRunning	Anzahl der arbeitenden Aufgabenknoten. Die Datenpunkte dieser Metrik werden nur dann angegeben, wenn die zugehörige Instance-Gruppe existiert. Anwendungsfall: Cluster-Zustand überwachen Einheiten: Anzahl
TaskNodesPending	Anzahl der Aufgabenknoten, die auf eine Zuordnung warten. Es müssen nicht alle angeforderten Aufgabenknoten sofort verfügbar sein. Diese Metrik gibt die ausstehenden Anforderungen an. Die Datenpunkte dieser Metrik werden nur dann angegeben, wenn die zugehörige Instance-Gruppe existiert. Anwendungsfall: Cluster-Zustand überwachen Einheiten: Anzahl
LiveTaskTrackers	Prozentsatz der funktionierenden Task-Tracker. Anwendungsfall: Cluster-Zustand überwachen Einheiten: Prozent
IO	

Metrik	Beschreibung
S3 BytesWritten	Anzahl der auf Amazon S3 geschriebenen Bytes. Diese Metrik aggregiert nur MapReduce Jobs und gilt nicht für andere Workloads bei Amazon. EMR Anwendungsfall: Cluster-Leistung analysieren, Cluster-Fortschritt überwachen Einheiten: Anzahl
S3 BytesRead	Anzahl der von Amazon S3 gelesenen Bytes. Diese Metrik aggregiert nur MapReduce Jobs und gilt nicht für andere Workloads bei Amazon. EMR Anwendungsfall: Cluster-Leistung analysieren, Cluster-Fortschritt überwachen Einheiten: Anzahl
HDFSUtilization	Der Prozentsatz des aktuell genutzten HDFS Speichers. Anwendungsfall: Cluster-Leistung analysieren Einheiten: Prozent
HDFSBytesRead	Die Anzahl der Byte, aus denen gelesen wurdeHDFS. Anwendungsfall: Cluster-Leistung analysieren, Cluster-Fortschritt überwachen Einheiten: Anzahl
HDFSBytesWritten	Die Anzahl der Byte, in die geschrieben wurdeHDFS. Anwendungsfall: Cluster-Leistung analysieren, Cluster-Fortschritt überwachen Einheiten: Anzahl

Metrik	Beschreibung
MissingBlocks	Die Anzahl der Blöcke, in denen es HDFS keine Replikate gibt. Hierbei kann es sich um beschädigte Blöcke handeln. Anwendungsfall: Cluster-Zustand überwachen Einheiten: Anzahl
TotalLoad	Die aktuelle Gesamtzahl der Leser und Schreiber, die von allen DataNodes in einem Cluster gemeldet wurden. Anwendungsfall: Diagnose des Grads, in dem ein hoher E/A-Wert zu einer schlechten Leistung bei der Job-Ausführung beitragen könnte. Worker-Knoten, auf denen der DataNode Daemon ausgeführt wird, müssen auch Mapping- und Reduce-Aufgaben ausführen. Dauerhaft hohe TotalLoad Werte im Laufe der Zeit können darauf hindeuten, dass eine hohe I/O-Leistung möglicherweise zu einer schlechten Leistung beiträgt. Gelegentliche Spitzen in diesem Wert sind typisch und weisen in der Regel nicht auf ein Problem hin. Einheiten: Anzahl

Cluster-Kapazitätsmetriken

Die folgenden Metriken geben die aktuelle oder Zielkapazitäten eines Clusters an. Diese Metriken sind nur verfügbar, wenn verwaltete Skalierung oder automatische Beendigung aktiviert ist.

Bei Clustern, die aus Instance-Flotten bestehen, werden die Cluster-Kapazitätsmetriken in `Units` gemessen. Bei Clustern, die aus Instance-Gruppen bestehen, werden die Clusterkapazitätsmetriken in `Nodes` oder `VCPU` basierend auf dem Einheitentyp gemessen, der in der Richtlinie für verwaltete Skalierung verwendet wird. Weitere Informationen finden Sie unter Using EMR -managed scaling im Amazon EMR Management Guide.

Metrik	Beschreibung
• `TotalUnitsRequested` • `TotalNodesRequested` • `TotalVCPURequested`	Die angestrebte Gesamtzahl der Einheiten/Knoten/ vCPUs in einem Cluster, wie sie durch verwaltete Skalierung bestimmt wird. Einheiten: Anzahl
• `TotalUnitsRunning` • `TotalNodesRunning` • `TotalVCPURunning`	Die aktuelle Gesamtzahl der vCPUs Einheiten/Knoten/, die in einem laufenden Cluster verfügbar sind. Wenn eine Clustergrößenänderung angefordert wird, wird diese Metrik aktualisiert, nachdem die neuen Instances hinzugefügt oder aus dem Cluster entfernt wurden. Einheiten: Anzahl
• `CoreUnitsRequested` • `CoreNodesRequested` • `CoreVCPURequested`	Die Zielanzahl von CORE Einheiten/Knoten/ vCPUs in einem Cluster, wie sie durch verwaltete Skalierung bestimmt wird. Einheiten: Anzahl
• `CoreUnitsRunning` • `CoreNodesRunning` • `CoreVCPURunning`	Die aktuelle Anzahl von CORE vCPUs Einheiten/Knoten/, die in einem Cluster ausgeführt werden. Einheiten: Anzahl
• `TaskUnitsRequested` • `TaskNodesRequested` • `TaskVCPURequested`	Die Zielanzahl von TASK Einheiten/Knoten/ vCPUs in einem Cluster, wie sie durch verwaltete Skalierung bestimmt wird. Einheiten: Anzahl

Metrik	Beschreibung
• `TaskUnitsRunning` • `TaskNodesRunning` • `TaskVCPURunning`	Die aktuelle Anzahl von TASK vCPUs Einheiten/Knoten/, die in einem Cluster ausgeführt werden. Einheiten: Anzahl

Amazon EMR gibt die folgenden Metriken mit einer Granularität von einer Minute aus, wenn Sie die automatische Kündigung mithilfe einer automatischen Kündigungsrichtlinie aktivieren. Einige Metriken sind nur für EMR Amazon-Versionen 6.4.0 und höher verfügbar. Weitere Informationen zur automatischen Beendigung finden Sie unter Verwenden einer Richtlinie zur automatischen Beendigung.

Metrik	Beschreibung
`TotalNotebookKernels`	Die Gesamtzahl der laufenden und inaktiven Notebook-Kernel auf dem Cluster. Diese Metrik ist nur für EMR Amazon-Versionen 6.4.0 und höher verfügbar.
`AutoTerminationIsClusterIdle`	Gibt an, ob der Cluster verwendet wird. Der Wert 0 gibt an, dass der Cluster von einer der folgenden Komponenten aktiv verwendet wird: • Eine Anwendung YARN • HDFS • Ein Notebook •

Metrik	Beschreibung
	Eine Cluster-Benutzeroberfläche, z. B. der Spark History Server Ein Wert von 1 gibt an, dass sich der Cluster im Leerlauf befindet. Amazon EMR prüft, ob der Cluster kontinuierlich inaktiv ist (`AutoTerminationIsClusterIdle` = 1). Wenn die Leerlaufzeit eines Clusters dem `IdleTimeout` Wert in Ihrer Richtlinie zur automatischen Kündigung entspricht, EMR beendet Amazon den Cluster.

Dimensionen für EMR Amazon-Metriken

EMRAmazon-Daten können mit jeder der Dimensionen in der folgenden Tabelle gefiltert werden.

Dimension	Beschreibung
JobFlowId	Entspricht der Cluster-ID, der eindeutigen Kennung eines Clusters mit dem Format `j-XXXXXXXXXXXXX` . Finden Sie diesen Wert, indem Sie in der EMR Amazon-Konsole auf den Cluster klicken.

Überwachung von EMR Amazon-Ereignissen mit CloudWatch

Amazon EMR verfolgt Ereignisse und speichert Informationen über sie für bis zu sieben Tage in der EMR Amazon-Konsole. Amazon EMR zeichnet Ereignisse auf, wenn sich der Status von Clustern, Instance-Gruppen, Instance-Flotten, automatischen Skalierungsrichtlinien oder Schritten ändert. Ereignisse erfassen Datum und Uhrzeit des Ereignisses, Details zu den betroffenen Elementen und andere wichtige Datenpunkte.

In der folgenden Tabelle sind EMR Amazon-Ereignisse zusammen mit dem Status oder der Statusänderung, auf die das Ereignis hinweist, der Schweregrad des Ereignisses, der Ereignistyp,

der Ereigniscode und die Ereignismeldungen aufgeführt. Amazon EMR stellt Ereignisse als JSON Objekte dar und sendet sie automatisch an einen Event-Stream. Das JSON Objekt ist wichtig, wenn Sie Regeln für die Ereignisverarbeitung mithilfe von CloudWatch Ereignissen einrichten, da Regeln versuchen, Mustern im JSON Objekt zu entsprechen. Weitere Informationen finden Sie unter Ereignisse und Ereignismuster und EMRAmazon-Ereignisse im Amazon CloudWatch Events-Benutzerhandbuch.

> **Note**
>
> Um sicherzustellen, dass wir Ihnen die relevantesten Informationen zur Verfügung stellen, verfeinern wir unsere Fehlermeldungen kontinuierlich. Aus diesem Grund wird empfohlen, dass Sie den Text der Nachrichten nicht analysieren, um die nächsten Aktionen in Ihrem Workflow einzuleiten.

Cluster-Startereignisse

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
`CREATING`	`WARN`	EMRBereitstellung von Instance-Flotten	EC2Bereitstellung — Unzureichende Instanzkapazität	Wir können Ihren EMR Amazon-Cluster `ClusterId ( ClusterName)` für die Instance-Flotte nicht erstellen `. InstanceFleetID` Amazon EC2 hat nicht genügend Spot-Kapazität für den Instance-Typ `[Instance type1, Instancet`

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
				ype2] und nicht genügend On-Demand-Kapazität für den Instance-Typ [Instance type3, Ins tancetype 4] in der Availability Zone[Availabi lityZone1 , Avaliab ilityZone 2] . Weitere Informati onen darüber, wie Sie auf dieses Ereignis reagieren können, finden Sie in der Dokumentation.

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
CREATING	WARN	EMRBereitstellung von Instanzgruppen	EC2Bereitstellung — Unzureichende Instanzkapazität	Wir können Ihren EMR Amazon-Cluster ClusterId (ClusterName) für die Instance-Gruppe nicht erstellen . InstanceGroupID Amazon EC2 hat nicht genügend Spot-Kapazität für den Instance-Typ [Instancetype1, Instancetype2] und nicht genügend On-Demand-Kapazität für den Instance-Typ [Instancetype3, Instancetype4] in der Availability Zone[AvailabilityZone1, AvaliabilityZone2] . Weitere

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
				Informationen darüber, wie Sie auf dieses Ereignis reagieren können, finden Sie in der Dokumentation.

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
`CREATING`	`WARN`	EMRBereitstellung von Instance-Flotten	EC2Bereitstellung — Nicht genügend freie Adressen im Subnetz	Wir können den EMR Amazon-Cluster`ClusterId (ClusterName)` , den Sie für die Instance-Flotte angefordert haben, nicht erstellen, `InstanceFleetID` da das angegebene Subnetz `[Subnet1, Subnet2]` nicht genügend freie private IP-Adressen enthält, um Ihre Anfrage zu erfüllen. Verwenden Sie den `DescribeSubnets` Vorgang, um zu sehen, wie viele IP-Adressen in Ihrem Subnetz verfügbar (ungenutzt) sind. Informationen darüber, wie Sie auf dieses Ereignis

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
				reagieren können, finden Sie unter Fehlercodes für Amazon EC2 API

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
`CREATING`	`WARN`	EMRBereitstellung von Instanzgruppen	EC2Bereitstellung — Nicht genügend freie Adressen im Subnetz	Wir können den EMR Amazon-Cluster`ClusterId (ClusterName)` , den Sie für die Instance-Gruppe angefordert haben, nicht erstellen, `InstanceGroupID` da das angegebene Subnetz `[Subnet1, Subnet2]` nicht genügend freie private IP-Adressen enthält, um Ihre Anfrage zu erfüllen. Verwenden Sie den `DescribeSubnets` Vorgang, um zu sehen, wie viele IP-Adressen in Ihrem Subnetz verfügbar (ungenutzt) sind. Informationen darüber, wie Sie auf

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
				dieses Ereignis reagieren können, finden Sie unter Fehlercodes für Amazon EC2 API
`CREATING`	`WARN`	EMRBereitstellung von Instanzflotten	EC2Bereitstellung — CPU v-Limit überschritten	Die Bereitstellung von `InstanceFleetID` im EMR Amazon-Cluster verzögert `ClusterId (ClusterName)` sich, da Sie das Limit für die Anzahl der vCPUs (virtuellen Verarbeitungseinheiten) erreicht haben, die den laufenden Instances in Ihrem zugewiesen sind`account (accountId)` . Weitere Informationen finden Sie unter Fehlercodes für Amazon EC2 API

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
`CREATING`	`WARN`	EMRBereitstellung von Instanzgruppen	EC2Bereitstellung — CPU v-Limit überschritten	Die Bereitstellung der Instance-Gruppe `InstanceGroupID` im EMR Amazon-Cluster verzögert `ClusterId` sich, da Sie das Limit für die Anzahl der vCPUs (virtuellen Verarbeitungseinheiten) erreicht haben, die den laufenden Instances in Ihrem Konto zugewiesen sind(`accountId`). Weitere Informationen finden Sie unter Fehlercodes für Amazon EC2 API

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
`CREATING`	`WARN`	EMRBereitstellung von Instanzflotten	EC2Bereitstellung — Das Limit für die Anzahl der Spot-Instances wurde überschritten	Die Bereitstellung der Instance-Flotte `InstanceFleetID` im EMR Amazon-Cluster verzögert `ClusterID (ClusterName)` sich, da Sie das Limit für die Anzahl der Spot-Instances erreicht haben, die Sie in Ihrem starten können`account (accountId)` . Weitere Informationen finden Sie unter Fehlercodes für Amazon EC2 API.

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
`CREATING`	`WARN`	EMRBereitstellung von Instanzgruppen	EC2Bereitstellung — Das Limit für die Anzahl der Spot-Instances wurde überschritten	Die Bereitstellung der Instance-Gruppe `InstanceGroupID` im EMR Amazon-Cluster verzögert `ClusterID (ClusterName)` sich, da Sie das Limit für die Anzahl der Spot-Instances erreicht haben, die Sie in Ihrem starten können`account (accountId)` . Weitere Informationen finden Sie unter Fehlercodes für Amazon EC2 API.

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
`CREATING`	`WARN`	EMRBereitstellung von Instanzflotten	EC2Bereitstellung — Instanzlimit überschritten	Die Bereitstellung der Instance-Flotte `InstanceFleetID` im EMR Amazon-Cluster verzögert `ClusterId (ClusterName)` sich, da Sie das Limit für die Anzahl der Instances erreicht haben, die Sie gleichzeitig in Ihrem `account (accountID)` ausführen können. Weitere Informationen zu den EC2 Servicebeschränkungen von Amazon finden Sie unter Fehlercodes für Amazon EC2 API.

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
`CREATING`	`WARN`	EMRBereitstellung von Instanzgruppen	EC2Bereitstellung — Instanzlimit überschritten	Die Bereitstellung der Instanzgruppe `InstanceGroupID` im EMR Amazon-Cluster verzögert `ClusterId (ClusterName)` sich, da Sie das Limit für die Anzahl der Instances erreicht haben, die Sie gleichzeitig in Ihrem `account (accountID)` ausführen können. Weitere Informationen zu den EC2 Servicebeschränkungen von Amazon finden Sie unter Fehlercodes für Amazon EC2 API.

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
`CREATING`	`WARN`	EMRBereitstellung von Instanzgruppen	Keine	Der EMR Amazon-Cluster `ClusterId (ClusterName)` wurde am `Time` erstellt und ist einsatzbereit. – oder – Der EMR Amazon-Cluster hat die Ausführung aller ausstehenden Schritte unter `ClusterId (ClusterName)` abgeschlossen`Time`. **Note** Ein Cluster im `WAITING`-Status kann trotzdem Aufträge

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
				bearbeiten.
STARTING	INFO	EMRÄnderung des Cluster-Status	Keine	Der EMR Amazon-Cluster ClusterId (ClusterName) wurde am angefordert Time und wird gerade erstellt.

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
`STARTING`	`INFO`	EMRÄnderung des Cluster-Status	Keine	**Note** Gilt nur für Cluster mit der Instance-Flottenkonfiguration und mehreren innerhalb von Amazon EC2 ausgewählten Availability Zones. Der EMR Amazon-Cluster `ClusterId (ClusterName)` wird in Zone (`AvailabilityZoneID`) erstellt, die aus den angegebenen Availability

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
				Zone-Optionen ausgewählt wurde.
`STARTING`	`INFO`	EMRÄnderung des Cluster-Status	Keine	Der EMR Amazon-Cluster `ClusterId (ClusterName)` begann mit der Ausführung von Schritten am`Time`.

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
`WAITING`	`INFO`	EMRÄnderung des Cluster-Status	Keine	Der EMR Amazon-Cluster `ClusterId (ClusterName)` wurde am erstellt `Time` und ist einsatzbereit. – oder – Der EMR Amazon-Cluster hat die Ausführung aller ausstehenden Schritte unter `ClusterId (ClusterName)` abgeschlossen`Time`. **Note** Ein Cluster im `WAITING`-Status kann trotzdem Aufträge

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
				bearbeiten.

> **Note**
>
> Die Ereignisse mit dem Ereigniscode `EC2 provisioning - Insufficient Instance Capacity` werden regelmäßig ausgelöst, wenn Ihr EMR Cluster während der Clustererstellung oder Größenänderung auf einen Fehler von Amazon mit unzureichender Kapazität EC2 für Ihre Instance-Flotte oder Instance-Gruppe stößt. Weitere Informationen zum Umgang mit diesen Ereignissen finden Sie unter Reaktion auf Ereignisse mit unzureichender Instance-Kapazität im EMR Amazon-Cluster.

Cluster-Abbruchsereignisse

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
`TERMINATED`	Der Schweregrad ist abhängig vom Grund für die Statusänderung, wie nachfolgend dargestellt: • **`CRITICAL`** wenn der Cluster aufgrund einer der folgenden Statusänd	EMRÄnderung des Cluster-Status	Keine	Amazon EMR Cluster `ClusterId (ClusterName)` wurde am aus `Time` einem Grund von gekündigt `StateChangeReason: Code` .

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
	erungen beendet wurde: `INTERNAL_ERROR` , `VALIDATION_ERROR` , `INSTANCE_FAILURE` , `BOOTSTRAP_FAILURE` oder `STEP_FAILURE` . • **`INFO`** wenn der Cluster aufgrund einer der folgenden Statusänderungen beendet wurde: `USER_REQUEST` oder `ALL_STEPS_COMPLETED` .			

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
`TERMINATED_WITH_ERRORS`	`CRITICAL`	EMRÄnderung des Cluster-Status	Keine	Amazon EMR Cluster `ClusterId (ClusterName)` wurde mit Fehlern unter aus `Time` dem folgenden Grund beendet`StateChangeReason: Code` .
`TERMINATED_WITH_ERRORS`	`CRITICAL`	EMRÄnderung des Cluster-Status	Keine	Amazon EMR Cluster `ClusterId (ClusterName)` wurde mit Fehlern unter aus `Time` dem folgenden Grund beendet`StateChangeReason: Code` .

Ereignisse zur Änderung des Status der Instance-Flotte

> **Note**
>
> Die Konfiguration der Instance-Flotten ist nur in EMR Amazon-Versionen 4.8.0 und höher verfügbar, mit Ausnahme von 5.0.0 und 5.0.3.

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
Von PROVISIONING bis WAITING	INFO		Keine	Die Bereitstellung der Instance-Flotte InstanceFleetID im EMR Amazon-Cluster ClusterId (ClusterName) ist abgeschlossen. Die Bereitstellung startete um Time und dauerte Num Minuten. Die Instance-Flotte verfügt jetzt über eine On-Demand-Kapazität von Num und eine Spot-Kapazität von Num. Die anvisierte On-Demand-Kapazität betrug Num und die anvisierte Spot-Kapazität betrug Num.
Von WAITING bis RESIZING	INFO		Keine	Eine Größenänderung für

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
				beispielsweise eine Flotte `InstanceFleetID` im EMR Amazon-Cluster `ClusterId (ClusterName)` begann bei`Time`. Die Instance-Flotte verändert ihre Größe von einer On-Demand-Kapazität von `Num` auf eine Zielkapazität von `Num` und von einer Spot-Kapazität von `Num` auf eine Zielkapazität von `Num`.

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
Von `RESIZING` bis `WAITING`	`INFO`		Keine	Die Größenänderung für die Instance-Flotte `InstanceFleetID` im EMR Amazon-Cluster `ClusterId (ClusterName)` ist abgeschlossen. Die Größenänderung startete um `Time` und dauerte `Num` Minuten. Die Instance-Flotte verfügt jetzt über eine On-Demand-Kapazität von `Num` und eine Spot-Kapazität von `Num`. Die anvisierte On-Demand-Kapazität betrug `Num` und die anvisierte Spot-Kapazität betrug `Num`.

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
Von `RESIZING` bis `WAITING`	`INFO`		Keine	Der Vorgang zur Größenänderung für die Instance-Flotte `InstanceFleetID` im EMR Amazon-Cluster `ClusterId (ClusterName)` hat das Timeout erreicht und wurde beendet. Die Größenänderung startete um `Time` und wurde nach `Num` Minuten gestoppt. Die Instance-Flotte verfügt jetzt über eine On-Demand-Kapazität von `Num` und eine Spot-Kapazität von `Num`. Die anvisierte On-Demand-Kapazität betrug `Num` und die anvisierte Spot-

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
				Kapazität betrug Num.
SUSPENDED	ERROR		Keine	Die Instance-Flotte InstanceFleetID im EMR Amazon-Cluster ClusterId (ClusterName) wurde aus dem folgenden Grund festgenommen:ReasonDesc . Time
RESIZING	WARNING		Keine	Der Vorgang zur Größenänderung für die Instance-Flotte InstanceFleetID im EMR Amazon-Cluster ClusterId (ClusterName) ist aus dem folgenden Grund blockiert :ReasonDesc .

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
`WAITING` oder `Running`	`INFO`		Keine	Die Größenänderung für die Instance-Flotte `InstanceFleetID` im EMR Amazon-Cluster `ClusterId (ClusterName)` konnte nicht abgeschlossen werden, während Amazon Spot-Kapazität in der Availability Zone EMR hinzugefügt hat`AvailabilityZone` . Wir haben Ihre Anfrage zur Bereitstellung zusätzlicher Spot-Kapazität storniert. Die empfohlenen Maßnahmen finden Sie unter Bewährte Methoden für Instance- und Availability Zone-Flexibilität. Bitte

Status oder Statusänderung	Schweregrad	Ereignistyp	Ereigniscode	Fehlermeldung
				versuchen Sie es erneut.
`WAITING` oder `Running`	`INFO`		Keine	Ein Vorgang zur Größenänderung für die Instance-Flotte `InstanceFleetID` im EMR Amazon-Cluster `ClusterId` `(ClusterName)` wurde von `Entity` at initiiert`Time`.

Ereignisse zur Änderung der Größe der Instance-Flotte

Ereignistyp	Schweregrad	Ereigniscode	Fehlermeldung
EMRGröße der Instance-Flotte ändern	`ERROR`	Spot-Provisioning-Timeout	Der Vorgang zur Größenänderung für die Instanzflotte `InstanceFleetID` im EMR Amazon-Cluster `ClusterId` `(ClusterName)` konnte beim Erwerb von Spot-Kapazität in AZ `AvailabilityZone` nicht abgeschlossen

Ereignistyp	Schweregrad	Ereigniscode	Fehlermeldung
			werden. Wir haben jetzt Ihre Anfrage storniert und den Versuch beendet, zusätzliche Spot-Kapazität bereitzustellen, und die Instance-Flotte hat Spot-Kapazität von num bereitgestellt. Die Ziel-Spot-Kapazität war num. Weitere Informationen und Handlungsempfehlungen finden Sie auf der Dokumentationsseite hier. Bitte versuchen Sie es erneut.

Ereignistyp	Schweregrad	Ereigniscode	Fehlermeldung
EMRGröße der Instance-Flotte ändern	ERROR	Timeout für die On-Demand-Bereitstellung	Der Vorgang zur Größenänderung für die Instanzflotte `InstanceFleetID` im EMR Amazon-Cluster `ClusterId (Cluster Name)` konnte beim Erwerb von On-Demand-Kapazität in AZ `AvailabilityZone` nicht abgeschlossen werden. Wir haben jetzt Ihre Anfrage storniert und den Versuch beendet, zusätzliche On-Demand-Kapazität bereitzustellen, und die Instance-Flotte hat On-Demand-Kapazität von `num` bereitgestellt. Die gewünschte On-Demand-Kapazität war `num`. Weitere Informationen und Handlungsempfehlungen finden Sie auf der Dokumentationsseite hier. Bitte versuchen Sie es erneut.

Ereignistyp	Schweregrad	Ereigniscode	Fehlermeldung
EMRGröße der Instance-Flotte ändern	`WARNING`	EC2Bereitstellung — Unzureichende Instance-Kapazität	Wir können den Vorgang zur Größenänderung für die Instance-Flotte `InstanceFleetID` im EMR Cluster nicht abschließen, `ClusterId (ClusterName)` da Amazon nicht EC2 über ausreichende Spot-Kapazität für Instance-Typen `[Instancetype1, Instancetype2]` und über unzureichende On-Demand-Kapazität für Instance-Typen `[Instancetype3, Instancetype4]` in der Availability Zone `[AvailabilityZone1]` verfügt. Bisher hat die Instance-Flotte On-Demand-Kapazität von num bereitgestellt und die angestrebte On-Demand-Kapazität war num. Die bereitgestellte Spot-Kapazität ist num und die Ziel-Spot-

Ereignistyp	Schweregrad	Ereigniscode	Fehlermeldung
			Kapazität war num. Weitere Informati onen darüber, wie Sie auf dieses Ereignis reagieren können, finden Sie in der Dokumentation.

Ereignistyp	Schweregrad	Ereigniscode	Fehlermeldung
EMRGröße der Instance-Flotte ändern	`WARNING`	Zeitlimit für Spot-Bereitstellung – Fortsetzung der Größenänderung	Wir stellen immer noch Spot-Kapazität für den Vorgang zur Größenänderung der Instance-Flotte bereit, der beispielsweise mit der `time` Instance-Flotte-ID `InstanceFleetID` im EMR Amazon-Cluster `ClusterId (ClusterName)` für `[Instancetype1, Instancetype2]` in AZ gestartet wurde. `AvailabilityZone` Für den vorherigen Vorgang zur Größenänderung, der am gestartet wurde`time`, ist der Timeout-Zeitraum abgelaufen, sodass Amazon die Bereitstellung `num` von Spot-Kapazität EMR eingestellt hat, nachdem die angeforderten `num` Instances zu Ihrer Instance-Flotte hinzugefügt wurden. Weitere Informationen

Ereignistyp	Schweregrad	Ereigniscode	Fehlermeldung
			finden Sie auf der Dokumentationsseite hier.

Ereignistyp	Schweregrad	Ereigniscode	Fehlermeldung
EMRGröße der Instance-Flotte ändern	`WARNING`	Timeout für On-Demand-Bereitstellung – Fortsetzung der Größenänderung	Wir stellen weiterhin On-Demand-Kapazität für den Vorgang zur Größenänderung der Instance-Flotte bereit, der beispielsweise mit der `time` Instance-Flotte-ID `InstanceFleetID` im EMR Amazon-Cluster `ClusterId (ClusterName)` für `[Instancetype1, Instancetype2]` in AZ gestartet wurde. `AvailabilityZone` Für den vorherigen Vorgang zur Größenänderung, der am gestartet wurde`time`, ist der Timeout-Zeitraum abgelaufen, sodass Amazon die Bereitstellung `num` von On-Demand-Kapazität EMR eingestellt hat, nachdem die angeforderten `num` Instances zu Ihrer Instance-Flotte hinzugefügt wurden. Weitere Informationen

Ereignistyp	Schweregrad	Ereigniscode	Fehlermeldung
			finden Sie auf der Dokumentationsseite hier.
EMRGröße der Instance-Flotte ändern	`WARNING`	EC2Bereitstellung — Unzureichende freie Adresse im Subnetz	Wir können den Vorgang zur Größenänderung für die Instance-Flotte `InstanceFleetID` im EMR Amazon-Cluster nicht abschließen, `ClusterId (ClusterName)` da das angegebene Subnetz [Subnet1, Subnet2] nicht genügend freie private IP-Adressen enthält, um Ihre Anfrage zu bearbeiten. Verwenden Sie diesen `DescribeSubnets` Vorgang, um zu sehen, wie viele IP-Adressen in Ihrem Subnetz verfügbar (ungenutzt) sind. Informationen darüber, wie Sie auf dieses Ereignis reagieren können, finden Sie unter Fehlercodes für Amazon EC2 API.

Ereignistyp	Schweregrad	Ereigniscode	Fehlermeldung
EMRGröße der Instanzflotte ändern	`WARNING`	EC2Bereitstellung — CPU v-Limit überschri tten	Die Größenänderung der Instance-Flotte `InstanceFleetID` im EMR Amazon-Cluster verzögert `ClusterName` sich, da Sie das Limit für die Anzahl der vCPUs (virtuellen Verarbeitungseinhe iten) erreicht haben, die den laufenden Instances in Ihrem `account (accountId)` zugewiesen sind. Weitere Informati onen finden Sie unter Fehlercodes für Amazon EC2 API.

Ereignistyp	Schweregrad	Ereigniscode	Fehlermeldung
EMRGröße der Instanzflotte ändern	`WARNING`	EC2Bereitstellung — Das Limit für die Anzahl der Spot-Instances wurde überschritten	Die Bereitstellung der Instance-Flotte `InstanceFleetID` im EMR Amazon-Cluster verzögert `ClusterID (ClusterName)` sich, da Sie das Limit für die Anzahl der Spot-Instances erreicht haben, die Sie in Ihrem starten könnenaccount `(accountId)` . Weitere Informationen finden Sie unter Fehlercodes für Amazon EC2 API.

Ereignistyp	Schweregrad	Ereigniscode	Fehlermeldung
EMRGröße der Instanzflotte ändern	`WARNING`	EC2Bereitstellung — Instanzlimit überschri tten	Die Bereitstellung der Instance-Flotte `InstanceF leetID` im EMR Amazon-Cluster verzögert `ClusterID (ClusterName)` sich, da Sie das Limit für die Anzahl der On-Demand-Instances, die Sie in Ihrem ausführen können, erreicht habe`account (accountId)` . Weitere Informationen zu Fehlercodes für Amazon EC2 API.

> **Note**
>
> Die Timeout-Ereignisse für die Bereitstellung werden ausgelöst, wenn Amazon die Bereitstellung von Spot- oder On-Demand-Kapazität für die Flotte nach Ablauf des Timeouts EMR einstellt. Weitere Informationen zum Umgang mit diesen Ereignissen finden Sie unter Reaktion auf Timeout-Ereignisse zur Größenänderung der EMR Amazon-Cluster-Instance-Flotte.

Instance-Gruppen-Ereignisse

Ereignistyp	Schweregrad	Ereigniscode	Fehlermeldung
Von `RESIZING` bis `Running`	`INFO`	Keine	Die Größenänderung für die Instance-

Ereignistyp	Schweregrad	Ereigniscode	Fehlermeldung
			Gruppe `InstanceGroupID` im EMR Amazon-Cluster `ClusterId (ClusterName)` ist abgeschlossen. Sie verfügt jetzt über eine Instance-Anzahl von `Num`. Die Größenänderung startete um `Time` und dauerte `Num` Minuten bis zum Abschluss.
Von `RUNNING` bis `RESIZING`	`INFO`	Keine	Eine Größenänderung für die Instance-Gruppe `InstanceGroupID` im EMR Amazon-Cluster `ClusterId (ClusterName)` begann bei`Time`. Die Größenänderung erfolgt von einer Instance-Anzahl von `Num` auf `Num`.

Ereignistyp	Schweregrad	Ereigniscode	Fehlermeldung
SUSPENDED	ERROR	Keine	Die Instanzgruppe InstanceGroupID im EMR Amazon-Cluster ClusterId (ClusterName) wurde aus dem folgenden Grund festgenommen:ReasonDesc . Time
RESIZING	WARNING	Keine	Der Vorgang zur Größenänderung für die Instanzgruppe InstanceGroupID im EMR Amazon-Cluster ClusterId (ClusterName) ist aus dem folgenden Grund blockiert:ReasonDesc .

Ereignistyp	Schweregrad	Ereigniscode	Fehlermeldung
EMRGröße der Instanzgruppe ändern	`WARNING`	EC2Bereitstellung — Unzureichende Instanzkapazität	Wir können den Vorgang zur Größenänderung, der bei `time` für die Instanzgruppe `InstanceGroupID` im EMR Cluster gestartet wurde, nicht abschließen, `ClusterId (ClusterName)` da Amazon nicht genügend `Spot/On Demand` Kapazität für den Instance-Typ `[Instance type]` in der Availability Zone EC2 `[AvailabilityZone1]` hat. Bisher hatte die Instance-Gruppe eine Anzahl laufender Instances von `num` und die Anzahl der angeforderten Instances betrug `num`. Weitere Informationen darüber, wie Sie auf dieses Ereignis reagieren können, finden Sie in der Dokumentation.

Ereignistyp	Schweregrad	Ereigniscode	Fehlermeldung
EMRGröße der Instanzgruppe ändern	`WARNING`	EC2Bereitstellung — Unzureichende freie Adresse im Subnetz	Wir können die Größenänderung für die Instance-Gruppe `InstanceGroupID` im EMR Amazon-Cluster nicht abschließen, `ClusterId (ClusterName)` da das angegebene Subnetz [Subnet1, Subnet2] nicht genügend freie private IP-Adressen enthält, um Ihre Anfrage zu bearbeiten. Verwenden Sie den `DescribeSubnets` Vorgang, um zu sehen, wie viele IP-Adressen in Ihrem Subnetz verfügbar (ungenutzt) sind. Informationen darüber, wie Sie auf dieses Ereignis reagieren können, finden Sie unter Fehlercodes für Amazon EC2 API.

Ereignistyp	Schweregrad	Ereigniscode	Fehlermeldung
EMRGröße der Instanzgruppe ändern	`WARNING`	EC2Bereitstellung — CPU v-Limit überschri tten	Die Größenänderung der Instanzgruppe `InstanceGroupID` im EMR Amazon-Cluster verzögert `ClusterName` sich, da Sie das Limit für die Anzahl der vCPUs (virtuellen Verarbeitungseinhe iten) erreicht haben, die den laufenden Instances in Ihrem `account (accountId)` zugewiesen sind. Weitere Informati onen finden Sie unter Fehlercodes für Amazon EC2 API.

Ereignistyp	Schweregrad	Ereigniscode	Fehlermeldung
EMRGröße der Instanzgruppe ändern	`WARNING`	EC2Bereitstellung — Das Limit für die Anzahl der Spot-Instances wurde überschritten	Die Bereitstellung der Instance-Gruppe `InstanceGroupID` im EMR Amazon-Cluster verzögert `ClusterID (ClusterName)` sich, da Sie das Limit für die Anzahl der Spot-Instances erreicht haben, die Sie in Ihrem starten könnenaccount `(accountId)` . Weitere Informationen finden Sie unter Fehlercodes für Amazon EC2 API.

Ereignistyp	Schweregrad	Ereigniscode	Fehlermeldung
EMRGröße der Instanzgruppe ändern	`WARNING`	EC2Bereitstellung — Instanzlimit überschri tten	Die Bereitstellung der Instanzgr uppe `InstanceG roupID` im EMR Amazon-Cluster verzögert `ClusterID (ClusterName)` sich, da Sie das Limit für die Anzahl der On-Demand-Instances, die Sie in Ihrem ausführen können, erreicht habe`account (accountId)` . Weitere Informationen zu Fehlercodes für Amazon EC2 API.
Von `RUNNING` bis `RESIZING`	`INFO`	Keine	Eine Größenänd erung für die Instance-Gruppe `InstanceGroupID` im EMR Amazon-Cl uster `ClusterId (ClusterName)` wurde von `Entity` at `Time` initiiert.

> **Note**
>
> Mit EMR Amazon-Version 5.21.0 und höher können Sie Cluster-Konfigurationen überschreiben und zusätzliche Konfigurationsklassifizierungen für jede Instance-Gruppe in einem laufenden Cluster angeben. Sie tun dies, indem Sie die EMR Amazon-Konsole,

die AWS Command Line Interface (AWS CLI) oder die verwenden AWS SDK. Weitere Informationen finden Sie unter Angeben einer Konfiguration für eine Instance-Gruppe in einem aktiven Cluster.

In der folgenden Tabelle sind EMR Amazon-Ereignisse für den Rekonfigurationsvorgang aufgeführt, zusammen mit dem Status oder der Statusänderung, auf die das Ereignis hinweist, dem Schweregrad des Ereignisses und den Ereignismeldungen.

Status oder Statusänderung	Schweregrad	Fehlermeldung
RUNNING	INFO	Eine Neukonfiguration für die Instanzgruppe InstanceGroupID im EMR Amazon-Cluster ClusterId (ClusterName) wurde vom Benutzer unter Time initiiert. Die Version der angeforderten Konfiguration ist Num.
Von RECONFIGURING bis Running	INFO	Der Neukonfigurationsvorgang für die Instanzgruppe InstanceGroupID im EMR Amazon-Cluster ClusterId (ClusterName) ist abgeschlossen. Die Rekonfiguration startete um Time und benötigte Num Minuten bis zum Abschluss. Die aktuelle Konfigurationsversion ist Num.
Von RUNNING bis RECONFIGURING in	INFO	Eine Neukonfiguration für die Instanzgruppe InstanceGroupID im EMR Amazon-Cluster ClusterId (ClusterName) begann

Status oder Statusänderung	Schweregrad	Fehlermeldung
		amTime. Sie konfiguriert von Versionsnummer Num bis Versionsnummer Num.
RESIZING	INFO	Die Neukonfiguration des Vorgangs hin zur Konfigura tionsversion Num für die Instanzgruppe InstanceG roupID im EMR Amazon-Cluster ClusterId (ClusterName) ist vorübergehend blockiert, Time da sich die Instanzgruppe in State befindet.
RECONFIGURING	INFO	Die Größenänderung der Instance-Anzahl Num für die Instance-Gruppe InstanceGroupID im EMR Amazon-Cluster ClusterId (ClusterName) ist vorübergehend blockiertTime, da sich die Instance-Gruppe in befindetState.
RECONFIGURING	WARNING	Der Neukonfigurationsv organg für die Instanzgruppe InstanceGroupID im EMR Amazon-Cluster ClusterId (ClusterName) schlug fehl Time und dauerte Num Minuten, bis er fehlschlug. Die fehlgeschlagene Konfigura tionsversion ist Num.

Status oder Statusänderung	Schweregrad	Fehlermeldung
RECONFIGURING	INFO	Die Konfigurationen werden auf die vorherige erfolgrei che Versionsnummer Num für die Instance-Gruppe InstanceGroupID im EMR Amazon-Cluster ClusterId (ClusterName) unter Time zurückgesetzt. Die neue Konfigurationsversion ist Num.
Von RECONFIGURING bis Running	INFO	Konfigurationen wurden erfolgreich auf die vorherige erfolgreiche Version Num für die Instance-Gruppe InstanceGroupID im EMR Amazon-Cluster ClusterId (ClusterName) unter Time zurückgesetzt. Die neue Konfigurationsversion ist Num.
Von RECONFIGURING bis SUSPENDED	CRITICAL	Fehler beim Zurücksetzen auf die vorherige erfolgreiche Version Num für die Instance-Gruppe InstanceGroupID im EMR Amazon-Cluster ClusterId (ClusterN ame) unterTime.

Auto-Scaling-Richtlinienereignisse

Status oder Statusänderung	Schweregrad	Fehlermeldung
PENDING	INFO	Eine Auto Scaling Scaling-R ichtlinie wurde der Instanzgr

Status oder Statusänderung	Schweregrad	Fehlermeldung
		uppe `InstanceGroupID` im EMR Amazon-Cluster `ClusterId (Cluster Name)` unter hinzugefü gt`Time`. Die Ausrüstung der Richtlinie ist noch anhängig. – oder – Die Auto Scaling Scaling-Richtlinie für die Instance-Gruppe `InstanceGroupID` im EMR Amazon-Cluster `ClusterId (Cluster Name)` wurde am aktualisi ert`Time`. Die Ausrüstung der Richtlinie ist noch anhängig.
`ATTACHED`	`INFO`	Die Auto Scaling Scaling-Richtlinie für die Instance-Gruppe `InstanceGroupID` im EMR Amazon-Cluster `ClusterId (Cluster Name)` wurde unter angehängt`Time`.
`DETACHED`	`INFO`	Die Auto Scaling Scaling-Richtlinie für die Instance-Gruppe `InstanceGroupID` im EMR Amazon-Cluster `ClusterId (Cluster Name)` wurde unter getrennt`Time`.

Status oder Statusänderung	Schweregrad	Fehlermeldung
FAILED	ERROR	Die Auto Scaling Scaling-Richtlinie für die Instance-Gruppe InstanceGroupID im EMR Amazon-Cluster ClusterId (Cluster Name) konnte nicht angehängt werden und schlug bei fehlTime. – oder – Die Auto Scaling Scaling-Richtlinie für die Instance-Gruppe InstanceGroupID im EMR Amazon-Cluster ClusterId (Cluster Name) konnte nicht getrennt werden und schlug bei Time fehl.

Schritt-Ereignisse

Status oder Statusänderung	Schweregrad	Fehlermeldung
PENDING	INFO	Der Schritt StepID (StepName) wurde dem EMR Amazon-Cluster ClusterId (ClusterName) unter hinzugefügt Time und seine Ausführung steht noch aus.
CANCEL_PENDING	WARN	Der Schritt StepID (StepName) im EMR Amazon-Cluster ClusterId

Status oder Statusänderung	Schweregrad	Fehlermeldung
		(ClusterName) wurde am storniert Time und steht noch aus.
RUNNING	INFO	Schritt StepID (StepName) im EMR Amazon-Cluster ClusterId (ClusterName) wurde am gestartet Time.
COMPLETED	INFO	Die Ausführung des Schritts StepID (StepName) im EMR Amazon-Cluster ClusterId (ClusterName) wurde am abgeschlossenTime. Der Schritt begann um Time mit der Ausführung und dauerte Num Minuten bis zum Abschluss.
CANCELLED	WARN	Die Stornierungsanforderung für den Cluster-Schritt StepID (StepName) im EMR Amazon-Cluster ClusterId (ClusterName) unter war erfolgreichTime, und der Schritt ist jetzt storniert.
FAILED	ERROR	Der Schritt StepID (StepName) im EMR Amazon-Cluster ClusterId (ClusterName) ist am fehlgeschlagenTime.

Ungesunde Ereignisse beim Austausch von Knoten

Ereignistyp	Schweregrad	Ereigniscode	Fehlermeldung
Austausch eines EMR defekten Amazon-Knotens	INFO	Ein fehlerhafter Kernknoten wurde erkannt	Amazon EMR hat festgestellt, dass sich die Kerninstanz [instanceID (Instance Name)] InstanceGroup/Fleet im EMR Amazon-Cluster clusterID (ClusterName) befindetUNHEALTHY. Amazon EMR wird versuchen, die Instance wiederherzustellen oder ordnungsgemäß zu ersetzen. UNHEALTHY
Austausch eines EMR defekten Amazon-Knotens	INFO	Der Kernknoten ist defekt — der Austausch ist deaktiviert	Amazon EMR hat festgestellt, dass sich die Kerninstanz [instanceID (Instance Name)] InstanceGroup/Flee

Ereignistyp	Schweregrad	Ereigniscode	Fehlermeldung	
			t im EMR Amazon-Cluster (clusterID) (ClusterName) befindetUNHEALTHY . Aktivieren Sie den ordnungsgemäßen Austausch ungesunder Kernknoten in Ihrem Cluster, damit Amazon EMR die UNHEALTHY Instances ordnungsgemäß ersetzt, falls sie nicht wiederhergestellt werden können.	

Ereignistyp	Schweregrad	Ereigniscode	Fehlermeldung	
Austausch eines EMR defekten Amazon-Knotens	WARN	Ungesunder Kernknoten wurde nicht ersetzt	Amazon EMR kann Ihre UNHEALTHY Core-Instance [instanceID (InstanceName)] InstanceGroup/Fleet im EMR Amazon-Cluster clusterID (ClusterName) aus gutem Grund nicht ersetzen. **Note** Der Grund, warum Amazon Ihren Core-Node nicht ersetzen EMR kann, hängt von Ihrem Szenario ab. Ein	

Ereignistyp	Schweregrad	Ereigniscode	Fehlermeldung	
			Grund dafür, dass Amazon einen Knoten nicht löschen EMR kann, ist beispielsweise, dass ein Cluster keine verbleibenden Kernknoten haben würde.	

Ereignistyp	Schweregrad	Ereigniscode	Fehlermeldung	
Austausch eines EMR defekten Amazon-Knotens	INFO	Ein fehlerhafter Kernknoten wurde wiederhergestellt	Amazon EMR hat Ihre UNHEALTHY Core-Instances [instanceID (Instance Name)] InstanceGroup/Fleet im EMR Amazon-Cluster wiederhergestellt clusterID (ClusterName)	

Weitere Informationen zum Austausch fehlerhafter Knoten finden Sie unter Ersetzen fehlerhafter Knoten.

Ereignisse mit der EMR Amazon-Konsole anzeigen

Für jeden Cluster können Sie eine einfache Liste der Ereignisse im Detailbereich anzeigen, der die Ereignisse in der Reihenfolge ihres Auftretens auflistet. Sie können auch alle Ereignisse für alle Cluster in einer Region in absteigender Reihenfolge ihres Auftretens anzeigen.

Wenn Sie nicht möchten, dass ein Benutzer alle Cluster-Ereignisse für eine Region sehen kann, erstellen Sie eine Anweisung, die die Berechtigung (`"Effect": "Deny"`) für die Aktion `elasticmapreduce:ViewEventsFromAllClustersInConsole` ablehnt. Fügen Sie diese Anweisung einer Richtlinie hinzu, die dem Benutzer zugeordnet ist.

Um Ereignisse für alle Cluster in einer Region mit der Konsole anzuzeigen

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen EMRSie EC2 im linken Navigationsbereich unter on die Option Ereignisse aus.

Um Ereignisse für einen bestimmten Cluster mit der Konsole anzuzeigen

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und dann einen Cluster aus.
3. Um alle Ihre Ereignisse anzuzeigen, wählen Sie auf der Cluster-Detailseite die Registerkarte Ereignisse aus.

Auf CloudWatch Ereignisse reagieren

In diesem Abschnitt werden verschiedene Möglichkeiten beschrieben, wie Sie auf umsetzbare Ereignisse reagieren können, die Amazon als EMR CloudWatch Ereignisnachrichten ausgibt.

Themen

- Regeln für EMR Amazon-Events erstellen mit CloudWatch
- Alarme für CloudWatch Metriken einrichten
- Reaktion auf Ereignisse mit unzureichender Instance-Kapazität im EMR Amazon-Cluster
- Reaktion auf Timeout-Ereignisse zur Größenänderung der EMR Amazon-Cluster-Instance-Flotte

Regeln für EMR Amazon-Events erstellen mit CloudWatch

Amazon sendet Ereignisse EMR automatisch an einen CloudWatch Event-Stream. Sie können Regeln erstellen, die nach einem bestimmten Muster auf Ereignisse zutreffen, und Sie können die Ereignisse an Ziele weiterleiten, um entsprechende Maßnahmen zu ergreifen, z. B. E-Mail-Benachrichtigungen senden. Muster werden mit dem JSON Ereignisobjekt abgeglichen. Weitere Informationen zu EMR Amazon-Veranstaltungsdetails finden Sie unter EMRAmazon-Veranstaltungen im Amazon CloudWatch Events-Benutzerhandbuch.

Informationen zum Einrichten von CloudWatch Ereignisregeln finden Sie unter Eine CloudWatch Regel erstellen, die bei einem Ereignis ausgelöst wird.

Alarme für CloudWatch Metriken einrichten

Amazon EMR überträgt Kennzahlen an Amazon CloudWatch. Als Reaktion darauf können Sie Alarme CloudWatch für Ihre EMR Amazon-Metriken einrichten. Sie können beispielsweise einen

Alarm so konfigurieren, CloudWatch dass Sie jedes Mal eine E-Mail erhalten, wenn die HDFS Auslastung über 80% steigt. Eine ausführliche Anleitung finden Sie unter Einen CloudWatch Alarm erstellen oder bearbeiten im CloudWatch Amazon-Benutzerhandbuch.

Reaktion auf Ereignisse mit unzureichender Instance-Kapazität im EMR Amazon-Cluster

Übersicht

EMRAmazon-Cluster geben den Eventcode zurück`EC2 provisioning - Insufficient Instance Capacity`, wenn die ausgewählte Availability Zone nicht über genügend Kapazität verfügt, um Ihre Anfrage zum Cluster-Start oder zur Größenänderung zu erfüllen. Das Ereignis wird regelmäßig sowohl bei Instance-Gruppen als auch bei Instance-Flotten ausgelöst, wenn Amazon EMR wiederholt auf Ausnahmen mit unzureichender Kapazität stößt und Ihre Bereitstellungsanforderung für einen Cluster-Start oder eine Cluster-Größenänderung nicht erfüllen kann.

Auf dieser Seite wird beschrieben, wie Sie am besten auf diesen Ereignistyp reagieren können, wenn er für Ihren Cluster eintritt. EMR

Empfohlene Reaktion auf ein Ereignis mit unzureichender Kapazität

Es wird empfohlen, dass Sie auf ein Ereignis mit unzureichender Kapazität mit einer der folgenden Methoden reagieren:

- Warten Sie, bis die Kapazität wiederhergestellt ist. Die Kapazität ändert sich häufig, sodass sich eine Ausnahme mit unzureichender Kapazität von selbst erholen kann. Ihre Cluster beginnen oder beenden die Größenänderung, sobald EC2 Amazon-Kapazität verfügbar ist.
- Alternativ können Sie Ihren Cluster beenden, Ihre Instance-Typ-Konfigurationen ändern und einen neuen Cluster mit der aktualisierten Cluster-Konfigurationsanforderung erstellen. Weitere Informationen finden Sie unter Bewährte Methoden für Instance- und Availability Zone-Flexibilität.

Sie können auch Regeln oder automatische Reaktionen auf ein Ereignis mit unzureichender Kapazität einrichten, wie im nächsten Abschnitt beschrieben.

Automatisierte Wiederherstellung nach einem Ereignis mit unzureichender Kapazität

Sie können eine Automatisierung als Reaktion auf EMR Amazon-Ereignisse erstellen, z. B. solche mit Ereigniscode`EC2 provisioning - Insufficient Instance Capacity`. Die folgende AWS Lambda Funktion beendet beispielsweise einen EMR Cluster mit einer Instance-Gruppe, die

On-Demand-Instances verwendet, und erstellt dann einen neuen EMR Cluster mit einer Instance-Gruppe, die andere Instance-Typen als die ursprüngliche Anfrage enthält.

Die folgenden Bedingungen lösen den automatisierten Prozess aus:

- Das Ereignis „unzureichende Kapazität“ wird seit mehr als 20 Minuten für Primär- oder Core-Knoten ausgelöst.
- Der Cluster befindet sich nicht im WAITINGStatus READYOder. Weitere Hinweise zu den EMR Clusterstatus finden Sie unterVerstehen des Cluster-Lebenszyklus.

> **Note**
>
> Wenn Sie einen automatisierten Prozess für eine Ausnahme mit unzureichender Kapazität erstellen, sollten Sie berücksichtigen, dass das Ereignis „unzureichende Kapazität“ wiederherstellbar ist. Die Kapazität ändert sich häufig und Ihre Cluster setzen die Größenänderung fort oder beginnen mit dem Betrieb, sobald EC2 Amazon-Kapazität verfügbar ist.

Example Funktion zur Reaktion auf ein Ereignis mit unzureichender Kapazität

```
// Lambda code with Python 3.10 and handler is lambda_function.lambda_handler
// Note: related IAM role requires permission to use Amazon EMR

import json
import boto3
import datetime
from datetime import timezone

INSUFFICIENT_CAPACITY_EXCEPTION_DETAIL_TYPE = "EMR Instance Group Provisioning"
INSUFFICIENT_CAPACITY_EXCEPTION_EVENT_CODE = (
    "EC2 provisioning - Insufficient Instance Capacity"
)
ALLOWED_INSTANCE_TYPES_TO_USE = [
    "m5.xlarge",
    "c5.xlarge",
    "m5.4xlarge",
    "m5.2xlarge",
    "t3.xlarge",
]
```

```
CLUSTER_START_ACCEPTABLE_STATES = ["WAITING", "RUNNING"]
CLUSTER_START_SLA = 20

CLIENT = boto3.client("emr", region_name="us-east-1")

# checks if the incoming event is 'EMR Instance Fleet Provisioning' with eventCode 'EC2
 provisioning - Insufficient Instance Capacity'
def is_insufficient_capacity_event(event):
    if not event["detail"]:
        return False
    else:
        return (
            event["detail-type"] == INSUFFICIENT_CAPACITY_EXCEPTION_DETAIL_TYPE
            and event["detail"]["eventCode"]
            == INSUFFICIENT_CAPACITY_EXCEPTION_EVENT_CODE
        )


# checks if the cluster is eligible for termination
def is_cluster_eligible_for_termination(event, describeClusterResponse):
    # instanceGroupType could be CORE, MASTER OR TASK
    instanceGroupType = event["detail"]["instanceGroupType"]
    clusterCreationTime = describeClusterResponse["Cluster"]["Status"]["Timeline"][
        "CreationDateTime"
    ]
    clusterState = describeClusterResponse["Cluster"]["Status"]["State"]

    now = datetime.datetime.now()
    now = now.replace(tzinfo=timezone.utc)
    isClusterStartSlaBreached = clusterCreationTime < now - datetime.timedelta(
        minutes=CLUSTER_START_SLA
    )

    # Check if instance group receiving Insufficient capacity exception is CORE or
 PRIMARY (MASTER),
    # and it's been more than 20 minutes since cluster was created but the cluster
 state and the cluster state is not updated to RUNNING or WAITING
    if (
        (instanceGroupType == "CORE" or instanceGroupType == "MASTER")
        and isClusterStartSlaBreached
        and clusterState not in CLUSTER_START_ACCEPTABLE_STATES
    ):
        return True
    else:
```

```
        return False


# Choose item from the list except the exempt value
def choice_excluding(exempt):
    for i in ALLOWED_INSTANCE_TYPES_TO_USE:
        if i != exempt:
            return i


# Create a new cluster by choosing different InstanceType.
def create_cluster(event):
    # instanceGroupType cloud be CORE, MASTER OR TASK
    instanceGroupType = event["detail"]["instanceGroupType"]

    # Following two lines assumes that the customer that created the cluster already
 knows which instance types they use in original request
    instanceTypesFromOriginalRequestMaster = "m5.xlarge"
    instanceTypesFromOriginalRequestCore = "m5.xlarge"

    # Select new instance types to include in the new createCluster request
    instanceTypeForMaster = (
        instanceTypesFromOriginalRequestMaster
        if instanceGroupType != "MASTER"
        else choice_excluding(instanceTypesFromOriginalRequestMaster)
    )
    instanceTypeForCore = (
        instanceTypesFromOriginalRequestCore
        if instanceGroupType != "CORE"
        else choice_excluding(instanceTypesFromOriginalRequestCore)
    )

    print("Starting to create cluster...")
    instances = {
        "InstanceGroups": [
            {
                "InstanceRole": "MASTER",
                "InstanceCount": 1,
                "InstanceType": instanceTypeForMaster,
                "Market": "ON_DEMAND",
                "Name": "Master",
            },
            {
                "InstanceRole": "CORE",
```

```
                "InstanceCount": 1,
                "InstanceType": instanceTypeForCore,
                "Market": "ON_DEMAND",
                "Name": "Core",
            },
        ]
    }
    response = CLIENT.run_job_flow(
        Name="Test Cluster",
        Instances=instances,
        VisibleToAllUsers=True,
        JobFlowRole="EMR_EC2_DefaultRole",
        ServiceRole="EMR_DefaultRole",
        ReleaseLabel="emr-6.10.0",
    )

    return response["JobFlowId"]


# Terminated the cluster using clusterId received in an event
def terminate_cluster(event):
    print("Trying to terminate cluster, clusterId: " + event["detail"]["clusterId"])
    response = CLIENT.terminate_job_flows(JobFlowIds=[event["detail"]["clusterId"]])
    print(f"Terminate cluster response: {response}")


def describe_cluster(event):
    response = CLIENT.describe_cluster(ClusterId=event["detail"]["clusterId"])
    return response


def lambda_handler(event, context):
    if is_insufficient_capacity_event(event):
        print(
            "Received insufficient capacity event for instanceGroup, clusterId: "
            + event["detail"]["clusterId"]
        )

        describeClusterResponse = describe_cluster(event)

        shouldTerminateCluster = is_cluster_eligible_for_termination(
            event, describeClusterResponse
        )
        if shouldTerminateCluster:
```

```
            terminate_cluster(event)

            clusterId = create_cluster(event)
            print("Created a new cluster, clusterId: " + clusterId)
        else:
            print(
                "Cluster is not eligible for termination, clusterId: "
                + event["detail"]["clusterId"]
            )

    else:
        print("Received event is not insufficient capacity event, skipping")
```

Reaktion auf Timeout-Ereignisse zur Größenänderung der EMR Amazon-Cluster-Instance-Flotte

Übersicht

EMRAmazon-Cluster geben Ereignisse aus, während die Größenänderung für Instance-Flottencluster ausgeführt wird. Die Timeout-Ereignisse für die Bereitstellung werden ausgelöst, wenn Amazon die Bereitstellung von Spot- oder On-Demand-Kapazität für die Flotte nach Ablauf des Timeouts EMR einstellt. Die Dauer des Timeouts kann vom Benutzer im Rahmen der Größenänderungsspezifikationen für die Instance-Flotten konfiguriert werden. In Szenarien mit aufeinanderfolgenden Größenänderungen für dieselbe Instance-Flotte EMR gibt Amazon die `On-Demand provisioning timeout - continuing resize` Ereignisse `Spot provisioning timeout - continuing resize` oder aus, wenn das Timeout für den aktuellen Größenänderungsvorgang abläuft. Dann beginnt es mit der Bereitstellung von Kapazität für die nächste Größenänderung der Flotte.

Reagieren auf Timeout-Ereignisse zur Größenänderung der Instanceflotte

Es wird empfohlen, dass Sie auf ein Bereitstellungs-Timeout-Ereignis mit einer der folgenden Methoden reagieren:

- GreifenSie die Größenänderungsspezifikationen wieder auf und versuchen Sie erneut, die Größe zu ändern. Da sich die Kapazität häufig ändert, wird die Größe Ihrer Cluster erfolgreich angepasst, sobald EC2 Amazon-Kapazität verfügbar ist. Wir empfehlen unseren Kunden, niedrigere Werte für die Timeout-Dauer für Jobs zu konfigurieren, die strengere Anforderungen stellen. SLAs
- Alternativ können Sie entweder:
  - Einen neuen Cluster mit diversifizierten Instance-Typen auf der Grundlage von bewährten Methoden wie der Flexibilität von Instances und Availability Zones starten oder

  - Einen Cluster mit On-Demand-Kapazität starten
- Für das Ereignis „Timeout bei der Bereitstellung – Fortsetzung der Größenänderung“ können Sie zusätzlich warten, bis die Größenänderungsvorgänge verarbeitet sind. Amazon EMR wird die für die Flotte ausgelösten Größenänderungsvorgänge weiterhin sequentiell verarbeiten und dabei die konfigurierten Größenänderungsspezifikationen einhalten.

Sie können auch Regeln oder automatische Reaktionen auf dieses Ereignis einrichten, wie im nächsten Abschnitt beschrieben.

Automatisierte Wiederherstellung nach einem Bereitstellungs-Timeout-Ereignis

Mit dem `Spot Provisioning timeout` Eventcode können Sie als Reaktion auf EMR Amazon-Ereignisse eine Automatisierung erstellen. Die folgende AWS Lambda Funktion fährt beispielsweise einen EMR Cluster mit einer Instance-Flotte herunter, die Spot-Instances für Task-Knoten verwendet, und erstellt dann einen neuen EMR Cluster mit einer Instance-Flotte, die diversifiziertere Instance-Typen als die ursprüngliche Anfrage enthält. In diesem Beispiel löst das für Aufgabenknoten ausgegebene `Spot Provisioning timeout` Ereignis die Ausführung der Lambda-Funktion aus.

Example Beispielfunktion zur Reaktion auf ein **`Spot Provisioning timeout`**-Ereignis

```
// Lambda code with Python 3.10 and handler is lambda_function.lambda_handler
// Note: related IAM role requires permission to use Amazon EMR

import json
import boto3
import datetime
from datetime import timezone

SPOT_PROVISIONING_TIMEOUT_EXCEPTION_DETAIL_TYPE = "EMR Instance Fleet Resize"
SPOT_PROVISIONING_TIMEOUT_EXCEPTION_EVENT_CODE = (
    "Spot Provisioning timeout"
)

CLIENT = boto3.client("emr", region_name="us-east-1")

# checks if the incoming event is 'EMR Instance Fleet Resize' with eventCode 'Spot
 provisioning timeout'
def is_spot_provisioning_timeout_event(event):
    if not event["detail"]:
        return False
    else:
```

```
        return (
            event["detail-type"] == SPOT_PROVISIONING_TIMEOUT_EXCEPTION_DETAIL_TYPE
            and event["detail"]["eventCode"]
            == SPOT_PROVISIONING_TIMEOUT_EXCEPTION_EVENT_CODE
        )


# checks if the cluster is eligible for termination
def is_cluster_eligible_for_termination(event, describeClusterResponse):
    # instanceFleetType could be CORE, MASTER OR TASK
    instanceFleetType = event["detail"]["instanceFleetType"]

    # Check if instance fleet receiving Spot provisioning timeout event is TASK
    if (instanceFleetType == "TASK"):
        return True
    else:
        return False


# create a new cluster by choosing different InstanceType.
def create_cluster(event):
    # instanceFleetType cloud be CORE, MASTER OR TASK
    instanceFleetType = event["detail"]["instanceFleetType"]

    # the following two lines assumes that the customer that created the cluster
 already knows which instance types they use in original request
    instanceTypesFromOriginalRequestMaster = "m5.xlarge"
    instanceTypesFromOriginalRequestCore = "m5.xlarge"

    # select new instance types to include in the new createCluster request
    instanceTypesForTask = [
        "m5.xlarge",
        "m5.2xlarge",
        "m5.4xlarge",
        "m5.8xlarge",
        "m5.12xlarge"
    ]

    print("Starting to create cluster...")
    instances = {
        "InstanceFleets": [
            {
                "InstanceFleetType":"MASTER",
                "TargetOnDemandCapacity":1,
```

```
                "TargetSpotCapacity":0,
                "InstanceTypeConfigs":[
                    {
                        'InstanceType': instanceTypesFromOriginalRequestMaster,
                        "WeightedCapacity":1,
                    }
                ]
            },
            {
                "InstanceFleetType":"CORE",
                "TargetOnDemandCapacity":1,
                "TargetSpotCapacity":0,
                "InstanceTypeConfigs":[
                    {
                        'InstanceType': instanceTypesFromOriginalRequestCore,
                        "WeightedCapacity":1,
                    }
                ]
            },
            {
                "InstanceFleetType":"TASK",
                "TargetOnDemandCapacity":0,
                "TargetSpotCapacity":100,
                "LaunchSpecifications":{},
                "InstanceTypeConfigs":[
                    {
                        'InstanceType': instanceTypesForTask[0],
                        "WeightedCapacity":1,
                    },
                    {
                        'InstanceType': instanceTypesForTask[1],
                        "WeightedCapacity":2,
                    },
                    {
                        'InstanceType': instanceTypesForTask[2],
                        "WeightedCapacity":4,
                    },
                    {
                        'InstanceType': instanceTypesForTask[3],
                        "WeightedCapacity":8,
                    },
                    {
                        'InstanceType': instanceTypesForTask[4],
                        "WeightedCapacity":12,
```

```
                    }
                ],
                "ResizeSpecifications": {
                    "SpotResizeSpecification": {
                        "TimeoutDurationMinutes": 30
                    }
                }
            }
        ]
    }
    response = CLIENT.run_job_flow(
        Name="Test Cluster",
        Instances=instances,
        VisibleToAllUsers=True,
        JobFlowRole="EMR_EC2_DefaultRole",
        ServiceRole="EMR_DefaultRole",
        ReleaseLabel="emr-6.10.0",
    )

    return response["JobFlowId"]


# terminated the cluster using clusterId received in an event
def terminate_cluster(event):
    print("Trying to terminate cluster, clusterId: " + event["detail"]["clusterId"])
    response = CLIENT.terminate_job_flows(JobFlowIds=[event["detail"]["clusterId"]])
    print(f"Terminate cluster response: {response}")


def describe_cluster(event):
    response = CLIENT.describe_cluster(ClusterId=event["detail"]["clusterId"])
    return response


def lambda_handler(event, context):
    if is_spot_provisioning_timeout_event(event):
        print(
            "Received spot provisioning timeout event for instanceFleet, clusterId: "
            + event["detail"]["clusterId"]
        )

        describeClusterResponse = describe_cluster(event)

        shouldTerminateCluster = is_cluster_eligible_for_termination(
```

```
            event, describeClusterResponse
        )
        if shouldTerminateCluster:
            terminate_cluster(event)

            clusterId = create_cluster(event)
            print("Created a new cluster, clusterId: " + clusterId)
        else:
            print(
                "Cluster is not eligible for termination, clusterId: "
                + event["detail"]["clusterId"]
            )

    else:
        print("Received event is not spot provisioning timeout event, skipping")
```

Anzeigen von Cluster-Anwendungsmetriken mit Ganglia

Ganglia ist mit EMR Amazon-Versionen zwischen 4.2 und 6.15 erhältlich. Ganglia ist ein Open-Source-Projekt. Es handelt sich um ein skalierbares, verteiltes System zur Überwachung von Clustern und Grids, das zugleich die Auswirkungen auf die Leistung minimiert. Wenn Sie Ganglia in Ihrem Cluster aktivieren, können Sie Berichte erstellen und die Leistung des Clusters als Ganzes betrachten. Ebenso können Sie die Leistung einzelner Knoten-Instances überprüfen. Ganglia ist außerdem zur Aufnahme und Visualisierung von Hadoop- und Spark-Metriken konfiguriert. Weitere Informationen finden Sie unter Ganglia im Amazon EMR Release Guide.

EMRAPIAmazon-Anrufe protokollieren AWS CloudTrail

Amazon EMR ist in einen Service integriert AWS CloudTrail, der eine Aufzeichnung der Aktionen bereitstellt, die von einem Benutzer, einer Rolle oder einem AWS Service bei Amazon ausgeführt wurdenEMR. CloudTrail erfasst alle API Anrufe für Amazon EMR als Ereignisse. Zu den erfassten Anrufen gehören Anrufe von der EMR Amazon-Konsole und Code-Aufrufe an den EMR API Amazon-Betrieb. Wenn Sie einen Trail erstellen, können Sie die kontinuierliche Übermittlung von CloudTrail Ereignissen an einen Amazon S3 S3-Bucket aktivieren, einschließlich Ereignissen für AmazonEMR. Wenn Sie keinen Trail konfigurieren, können Sie die neuesten Ereignisse trotzdem in der CloudTrail Konsole im Ereignisverlauf anzeigen. Anhand der von gesammelten Informationen können Sie die Anfrage CloudTrail, die an Amazon gestellt wurdeEMR, die IP-Adresse, von der aus die Anfrage gestellt wurde, wer die Anfrage gestellt hat, wann sie gestellt wurde, und weitere Details ermitteln.

Weitere Informationen CloudTrail dazu finden Sie im AWS CloudTrail Benutzerhandbuch.

EMRAmazon-Informationen in CloudTrail

CloudTrail ist in Ihrem AWS Konto aktiviert, wenn Sie das Konto erstellen. Wenn in Amazon Aktivitäten auftretenEMR, wird diese Aktivität zusammen mit anderen AWS Serviceereignissen in der CloudTrail Ereignishistorie in einem Ereignis aufgezeichnet. Sie können aktuelle Ereignisse in Ihrem AWS Konto ansehen, suchen und herunterladen. Weitere Informationen finden Sie unter Ereignisse mit CloudTrail Ereignisverlauf anzeigen.

Für eine fortlaufende Aufzeichnung der Ereignisse in Ihrem AWS Konto, einschließlich Veranstaltungen für AmazonEMR, erstellen Sie einen Trail. Ein Trail ermöglicht CloudTrail die Übermittlung von Protokolldateien an einen Amazon S3 S3-Bucket. Wenn Sie einen Trail in der Konsole erstellen, gilt der Trail standardmäßig für alle AWS Regionen. Der Trail protokolliert Ereignisse aus allen Regionen der AWS Partition und übermittelt die Protokolldateien an den von Ihnen angegebenen Amazon S3 S3-Bucket. Darüber hinaus können Sie andere AWS Dienste konfigurieren, um die in den CloudTrail Protokollen gesammelten Ereignisdaten weiter zu analysieren und darauf zu reagieren. Weitere Informationen finden Sie hier:

- Übersicht zum Erstellen eines Trails
- CloudTrail unterstützte Dienste und Integrationen
- Konfiguration von SNS Amazon-Benachrichtigungen für CloudTrail
- Empfangen von CloudTrail Protokolldateien aus mehreren Regionen und Empfangen von CloudTrail Protokolldateien von mehreren Konten

Alle EMR Amazon-Aktionen werden von Amazon protokolliert CloudTrail und sind in der EMRAPIAmazon-Referenz dokumentiert. Beispielsweise generieren Aufrufe von `ListCluster` und `DescribeCluster` Aktionen Einträge in den CloudTrail Protokolldateien. `RunJobFlow`

Jeder Ereignis- oder Protokolleintrag enthält Informationen zu dem Benutzer, der die Anforderung generiert hat. Die Identitätsinformationen unterstützen Sie bei der Ermittlung der folgenden Punkte:

- Ob die Anfrage mit Root- oder AWS Identity and Access Management (IAM) Benutzeranmeldedaten gestellt wurde.
- Gibt an, ob die Anforderung mit temporären Sicherheitsanmeldeinformationen für eine Rolle oder einen Verbundbenutzer gesendet wurde.
- Ob die Anfrage von einem anderen AWS Dienst gestellt wurde.

Falls ein Prozess und nicht ein Benutzer einen Cluster erstellt, können Sie anhand der `principalId` ID den Benutzer ermitteln, der mit der Clustererstellung verknüpft ist. Weitere Informationen finden Sie im CloudTrail userIdentityElement.

Beispiel: EMR Amazon-Protokolldateieinträge

Ein Trail ist eine Konfiguration, die die Übertragung von Ereignissen als Protokolldateien an einen von Ihnen angegebenen Amazon S3 S3-Bucket ermöglicht. CloudTrail Protokolldateien enthalten einen oder mehrere Protokolleinträge. Ein Ereignis stellt eine einzelne Anforderung aus einer beliebigen Quelle dar und enthält Informationen über die angeforderte Aktion, Datum und Uhrzeit der Aktion, Anforderungsparameter usw. CloudTrail Protokolldateien sind kein geordneter Stack-Trace der öffentlichen API Aufrufe, sodass sie nicht in einer bestimmten Reihenfolge angezeigt werden.

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die RunJobFlowAktion demonstriert.


```
{
 "Records": [
 {
         "eventVersion":"1.01",
         "userIdentity":{
            "type":"IAMUser",
            "principalId":"EX_PRINCIPAL_ID",
            "arn":"arn:aws:iam::123456789012:user/temporary-user-xx-7M",
            "accountId":"123456789012",
            "userName":"temporary-user-xx-7M"
         },
         "eventTime":"2018-03-31T17:59:21Z",
         "eventSource":"elasticmapreduce.amazonaws.com",
         "eventName":"RunJobFlow",
         "awsRegion":"us-west-2",
         "sourceIPAddress":"192.0.2.1",
         "userAgent":"aws-sdk-java/unknown-version Linux/xx Java_HotSpot(TM)_64-
Bit_Server_VM/xx",
         "requestParameters":{
            "tags":[
               {
                  "value":"prod",
                  "key":"domain"
               },
               {
```

```
                  "value":"us-west-2",
                  "key":"realm"
               },
               {
                  "value":"VERIFICATION",
                  "key":"executionType"
               }
            ],
            "instances":{
               "slaveInstanceType":"m5.xlarge",
               "ec2KeyName":"emr-integtest",
               "instanceCount":1,
               "masterInstanceType":"m5.xlarge",
               "keepJobFlowAliveWhenNoSteps":true,
               "terminationProtected":false
            },
            "visibleToAllUsers":false,
            "name":"MyCluster",
            "ReleaseLabel":"emr-5.16.0"
         },
         "responseElements":{
            "jobFlowId":"j-2WDJCGEG4E6AJ"
         },
         "requestID":"2f482daf-b8fe-11e3-89e7-75a3d0e071c5",
         "eventID":"b348a38d-f744-4097-8b2a-e68c9b424698"
      },
  ...additional entries
   ]
}
```

Clusterskalierung verwenden

Sie können die Anzahl der für einen EMR Amazon-Cluster verfügbaren EC2 Amazon-Instances automatisch oder manuell anpassen, um auf Workloads mit unterschiedlichen Anforderungen zu reagieren. Um die automatische Skalierung zu verwenden, haben Sie zwei Optionen. Sie können Amazon EMR Managed Scaling aktivieren oder eine benutzerdefinierte automatische Skalierungsrichtlinie erstellen. Die folgende Tabelle beschreibt die Unterschiede zwischen den Optionen.

	Von Amazon EMR verwaltete Skalierung	Benutzerdefinierte automatische Skalierung
Skalieren von Richtlinien und Regeln	Keine Richtlinie erforderlich. Amazon EMR verwaltet die automatische Skalierung, indem es kontinuierlich Cluster-Metriken auswertet und optimierte Skalierungsentscheidungen trifft.	Sie müssen die Richtlinien und Regeln für das Auto Scaling definieren und verwalten, z. B. die spezifischen Bedingungen, die Skalierungsaktivitäten, Evaluierungszeiträume, Ruhephasen usw. auslösen.
Unterstützte EMR Amazon-Versionen	EMRAmazon-Version 5.30.0 und höher (außer EMR Amazon-Version 6.0.0)	Amazon EMR Version 4.0.0 und höher
Unterstützte Clusterzusammenstellung	Instance-Gruppen oder Instance-Flotten	Nur Instance-Gruppen
Konfiguration von Skalierungsgrenzen	Skalierungsgrenzwerte werden für den gesamten Cluster konfiguriert.	Skalierungslimits können nur für jede Instance-Gruppe konfiguriert werden.
Häufigkeit der Auswertung von Metriken	Alle 5 bis 10 Sekunden Eine häufigere Auswertung von Metriken ermöglicht es AmazonEMR, genauere Skalierungsentscheidungen zu treffen.	Sie können die Auswertungszeiträume nur in Fünf-Minuten-Schritten definieren.
Unterstützte Anwendungen	Es werden nur YARN Anwendungen wie Spark, Hadoop, Hive, Flink unterstützt. Amazon EMR Managed Scaling unterstützt keine Anwendungen, die nicht darauf basierenYARN, wie Presto oderHBase.	Sie können auswählen, welche Anwendungen unterstützt werden, wenn Sie die Regeln für eine automatische Skalierung definieren.

Überlegungen

- Ein EMR Amazon-Cluster besteht immer aus einem oder drei Primärknoten. Sobald Sie den Cluster zum ersten Mal konfiguriert haben, können Sie nur Core- und Aufgabenknoten skalieren. Sie können die Anzahl der Primärknoten für den Cluster nicht skalieren.
- Bei Instance-Gruppen werden Rekonfigurations- und Größenänderungsvorgänge nacheinander und nicht gleichzeitig ausgeführt. Wenn Sie eine Neukonfiguration initiieren, während die Größe einer Instance-Gruppe geändert wird, beginnt die Neukonfiguration, sobald die Instance-Gruppe die laufende Größenänderung abgeschlossen hat. Umgekehrt, wenn Sie eine Größenänderung einleiten, während eine Instance-Gruppe ihre Neukonfiguration durchführt.

Verwenden von verwalteter Skalierung in Amazon EMR

> **Important**
>
> Wir empfehlen dringend, die neueste EMR Amazon-Version (Amazon EMR 7.2.0) für verwaltete Skalierung zu verwenden. In einigen frühen Versionen kann es zu zeitweiligen Anwendungsausfällen oder Verzögerungen bei der Skalierung kommen. Amazon hat dieses Problem mit den 5.x-Versionen 5.30.2, 5.31.1, 5.32.1, 5.33.1 und höher sowie mit den 6.x-Versionen 6.1.1, 6.2.1, 6.3.1 und höher EMR behoben. Weitere Informationen zur Region und Release-Verfügbarkeit finden Sie unter Verwaltete Skalierungsverfügbarkeit

Übersicht

Mit EMR Amazon-Versionen 5.30.0 und höher (außer Amazon EMR 6.0.0) können Sie Amazon EMR Managed Scaling aktivieren. Managed Scaling hilft Ihnen, die Anzahl der Instances oder Einheiten in Ihrem Cluster basierend auf der Workload automatisch zu erhöhen oder zu verringern. Amazon bewertet EMR kontinuierlich Cluster-Metriken, um Skalierungsentscheidungen zu treffen, die Ihre Cluster im Hinblick auf Kosten und Geschwindigkeit optimieren. Verwaltete Skalierung ist für Cluster verfügbar, die entweder aus Instance-Gruppen oder Instance-Flotten bestehen.

Verwaltete Skalierungsverfügbarkeit

- Im Folgenden AWS-Regionen ist Amazon EMR Managed Scaling mit Amazon EMR 6.14.0 und höher verfügbar:
  - Asien-Pazifik (Hyderabad) (ap-south-2)

  - Asien-Pazifik (Jakarta) (ap-southeast-3)
  - Europa (Spanien) (eu-south-2)
- Im Folgenden AWS-Regionen ist Amazon EMR Managed Scaling mit Amazon EMR 5.30.0 und 6.1.0 und höher verfügbar:
  - USA Ost (Nord-Virginia): (us-east-1)
  - USA Ost (Ohio): (us-east-2)
  - USA West (Oregon): (us-west-2)
  - USA West (Nordkalifornien) (us-west-1)
  - Afrika (Kapstadt) (af-south-1)
  - Asien-Pazifik (Hongkong) (ap-east-1)
  - Asien-Pazifik (Mumbai): (ap-south-1)
  - Asien-Pazifik (Seoul): (ap-northeast-2)
  - Asien-Pazifik (Singapur): (ap-southeast-1)
  - Asien-Pazifik (Sydney): (ap-southeast-2)
  - Asien-Pazifik (Tokyo) (ap-northeast-1)
  - Kanada (Zentral): (ca-central-1)
  - Südamerika (São Paulo) (sa-east-1)
  - Europa (Frankfurt) (eu-central-1)
  - Europa (Irland) (eu-west-1)
  - Europa (London) (eu-west-2)
  - Europa (Mailand) (eu-south-1)
  - Europa (Paris) (eu-west-3)
  - Europa (Stockholm) (eu-north-1)
  - China (Peking) (cn-north-1)
  - China (Ningxia) (cn-northwest-1)
  - AWS GovCloud (US-Ost) (-1) us-gov-east
  - AWS GovCloud (US-West) (us-gov-west-1)
- Amazon EMR Managed Scaling funktioniert nur mit YARN Anwendungen wie Spark, Hadoop, Hive und Flink. Es unterstützt keine Anwendungen, die nicht darauf basierenYARN, wie Presto und. HBase

Verwaltete Skalierungsparameter

Sie müssen die folgenden Parameter für die verwaltete Skalierung konfigurieren. Das Limit gilt nur für die Kern- und Aufgabenknoten. Der Primärknoten kann nach der Erstkonfiguration nicht skaliert werden.

- Minimum (`MinimumCapacityUnits`) — Die untere Grenze der zulässigen EC2 Kapazität in einem Cluster. Sie wird anhand von Kernen oder Instanzen der virtuellen Zentraleinheit (VCPU) für Instanzgruppen gemessen. Sie wird in Einheiten für Instance-Flotten gemessen.
- Maximum (`MaximumCapacityUnits`) — Die Obergrenze der zulässigen EC2 Kapazität in einem Cluster. Sie wird anhand von Kernen oder Instanzen der virtuellen Zentraleinheit (VCPU) für Instanzgruppen gemessen. Sie wird in Einheiten für Instance-Flotten gemessen.
- On-Demand-Limit (`MaximumOnDemandCapacityUnits`) (optional) — Die Obergrenze der zulässigen EC2 Kapazität für den On-Demand-Markttyp in einem Cluster. Wenn dieser Parameter nicht angegeben wird, wird der Standardwert `MaximumCapacityUnits` verwendet.
  - Dieser Parameter wird verwendet, um die Kapazitätszuweisung zwischen On-Demand- und Spot Instances aufzuteilen. Wenn Sie beispielsweise den Minimalparameter auf 2 Instances, den Maximalparameter auf 100 Instances und das On-Demand-Limit auf 10 Instances festlegen, skaliert Amazon EMR Managed Scaling auf bis zu 10 On-Demand-Instances und weist die verbleibende Kapazität Spot-Instances zu. Weitere Informationen finden Sie unter Knotenzuweisungsszenarien.
- Maximale Anzahl an Kernknoten (`MaximumCoreCapacityUnits`) (optional) — Die Obergrenze der zulässigen EC2 Kapazität für den Core-Knotentyp in einem Cluster. Wenn dieser Parameter nicht angegeben wird, wird der Standardwert `MaximumCapacityUnits` verwendet.
  - Dieser Parameter wird verwendet, um die Kapazitätszuweisung zwischen Core- und Aufgabenknoten aufzuteilen. Wenn Sie beispielsweise den Minimalparameter auf 2 Instances, das Maximum auf 100 Instances und den maximalen Core-Knoten auf 17 Instances festlegen, skaliert Amazon EMR Managed Scaling auf bis zu 17 Kernknoten und weist die verbleibenden 83 Instances Task-Knoten zu. Weitere Informationen finden Sie unter Knotenzuweisungsszenarien.

Weitere Informationen zu verwalteten Skalierungsparametern finden Sie unter `ComputeLimits`.

Überlegungen zur von Amazon EMR verwalteten Skalierung

- Managed Scaling wird in limitierten Versionen AWS-Regionen und EMR Amazon-Versionen unterstützt. Weitere Informationen finden Sie unter Verwaltete Skalierungsverfügbarkeit.

- Sie müssen die erforderlichen Parameter für Amazon EMR Managed Scaling konfigurieren. Weitere Informationen finden Sie unter Verwaltete Skalierungsparameter.
- Um Managed Scaling verwenden zu können, muss der Metrics-Collector-Prozess in der Lage sein, eine Verbindung zum öffentlichen API Endpunkt für die verwaltete Skalierung in Gateway herzustellen. API Wenn Sie einen privaten DNS Namen mit verwenden Amazon Virtual Private Cloud, funktioniert die verwaltete Skalierung nicht ordnungsgemäß. Um sicherzustellen, dass die verwaltete Skalierung funktioniert, empfehlen wir, dass Sie eine der folgenden Aktionen ausführen:
  - Entfernen Sie den API VPC Gateway-Schnittstellenendpunkt von Ihrem AmazonVPC.
  - Folgen Sie den Anweisungen unter Warum erhalte ich die Fehlermeldung HTTP 403 Forbidden, wenn ich von einem APIs aus eine Verbindung zu meinem API Gateway herstelleVPC? um die Einstellung für private DNS Namen zu deaktivieren.
  - Starten Sie Ihren Cluster stattdessen in einem privaten Subnetz. Weitere Informationen finden Sie im Thema Private Subnetze.
- Wenn Ihre YARN Jobs während des Herunterskalierens zeitweise langsam sind und die YARN Resource Manager-Protokolle zeigen, dass die meisten Ihrer Knoten während dieser Zeit auf der Negativliste standen, können Sie den Schwellenwert für die Außerbetriebnahme anpassen.

  Reduzieren Sie den `spark.blacklist.decommissioning.timeout` von einer Stunde auf eine Minute, um den Knoten für andere ausstehende Container verfügbar zu machen, um die Aufgabenverarbeitung fortzusetzen.

  Sie sollten auch einen höheren Wert festlegenYARN`.resourcemanager.nodemanager-graceful-decommission-timeout-secs`, um sicherzustellen, dass Amazon EMR das Beenden des Knotens nicht erzwingt, solange die längste „Spark-Task“ noch auf dem Knoten läuft. Die aktuelle Standardeinstellung ist 60 Minuten, was bedeutet, dass der Container nach 60 Minuten YARN zwangsweise beendet wird, sobald der Knoten in den Stilllegungszustand übergeht.

  Das folgende Beispiel für eine YARN Resource Manager-Protokollzeile zeigt Knoten, die dem Status „Außerbetriebnahme“ hinzugefügt wurden:

```
2021-10-20 15:55:26,994 INFO
 org.apache.hadoop.YARN.server.resourcemanager.DefaultAMSProcessor
 (IPC Server handler 37 on default port 8030): blacklist are updated in
 Scheduler.blacklistAdditions: [ip-10-10-27-207.us-west-2.compute.internal,
 ip-10-10-29-216.us-west-2.compute.internal, ip-10-10-31-13.us-
west-2.compute.internal, ... , ip-10-10-30-77.us-west-2.compute.internal],
 blacklistRemovals: []
```

Erfahren Sie mehr darüber, wie Amazon bei der Außerbetriebnahme von Knoten in die YARN Deny-Listing-Funktion EMR integriert, in Fällen, in denen Knoten in Amazon auf die Ablehnungsliste gesetzt werden EMR können, und zur Konfiguration des Verhaltens bei der Außerbetriebnahme von Spark-Knoten.

- Eine übermäßige Auslastung von EBS Volumes kann zu Problemen mit Managed Scaling führen. Wir empfehlen, die Auslastung des EBS Volumens unter 90% zu halten. Weitere Informationen finden Sie unter Instance-Speicher.
- CloudWatch Amazon-Metriken sind entscheidend für den Betrieb von Amazon EMR Managed Scaling. Wir empfehlen Ihnen, die CloudWatch Amazon-Metriken genau zu beobachten, um sicherzustellen, dass keine Daten fehlen. Weitere Informationen darüber, wie Sie CloudWatch Alarme konfigurieren können, um fehlende Messwerte zu erkennen, finden Sie unter CloudWatch Amazon-Alarme verwenden.
- Verwaltete Skalierungsvorgänge auf Clustern der Versionen 5.30.0 und 5.30.1, ohne dass Presto installiert ist, können zu Anwendungsausfällen führen oder dazu führen, dass eine einheitliche Instance-Gruppe oder Instance-Flotte unverändert im Status `ARRESTED` bleibt, insbesondere wenn auf einen Herunterskalierungsvorgang schnell ein Skalierungsvorgang folgt.

  Um dieses Problem zu umgehen, wählen Sie Presto als zu installierende Anwendung, wenn Sie einen Cluster mit den EMR Amazon-Versionen 5.30.0 und 5.30.1 erstellen, auch wenn Ihr Job Presto nicht benötigt.
- Wenn Sie den maximalen Core-Knoten und das On-Demand-Limit für Amazon EMR Managed Scaling festlegen, sollten Sie die Unterschiede zwischen Instance-Gruppen und Instance-Flotten berücksichtigen. Jede Instance-Gruppenkonfiguration besteht jeder Knotentyp aus demselben Instance-Typ und derselben Kaufoption für Instances: On-Demand oder Spot. Für jede Instance-Flotte geben Sie bis zu fünf Instance-Typen an, die als On-Demand- und Spot Instances bereitgestellt werden können. Weitere Informationen finden Sie unter Erstellen eines Clusters mit Instance-Flotten oder einheitlichen Instance-Gruppen, Instance Flotten Optionen und Knotenzuweisungsszenarien.
- Wenn Sie bei Amazon EMR 5.30.0 und höher die Standardregel Allow All Outbound auf 0.0.0.0/ für die Master-Sicherheitsgruppe entfernen, müssen Sie eine Regel hinzufügen, die ausgehende TCP Konnektivität zu Ihrer Sicherheitsgruppe für den Servicezugriff auf Port 9443 zulässt. Ihre Sicherheitsgruppe für den Servicezugriff muss auch eingehenden TCP Datenverkehr über Port 9443 von der Master-Sicherheitsgruppe zulassen. Weitere Informationen zur Konfiguration von Sicherheitsgruppen finden Sie unter Von Amazon EMR verwaltete Sicherheitsgruppe für die primäre Instance (private Subnetze).

- Sie können es verwenden AWS CloudFormation , um Amazon EMR Managed Scaling zu konfigurieren. Weitere Informationen finden Sie unter AWS:EMR: :Cluster im AWS CloudFormation Benutzerhandbuch.
- Wenn Sie Spot-Knoten verwenden, sollten Sie die Verwendung EMR von Knotenbezeichnungen in Betracht ziehen, um zu verhindern, dass Amazon Anwendungsprozesse EMR entfernt, wenn Amazon Spot-Knoten entfernt. Weitere Informationen zu Knotenbezeichnungen finden Sie unter Task-Knoten.
- Die Knotenkennzeichnung wird in EMR Amazon-Versionen 6.15 oder niedriger standardmäßig nicht unterstützt. Weitere Informationen finden Sie unter Grundlegendes zu Knotentypen: Primär-, Kern- und Aufgabenknoten.
- Wenn Sie EMR Amazon-Versionen 6.15 oder niedriger verwenden, können Sie Knotenbezeichnungen nur nach Knotentyp zuweisen, z. B. Kern- und Aufgabenknoten. Wenn Sie jedoch Amazon EMR Version 7.0 oder höher verwenden, können Sie Node-Labels nach Knotentyp und Markttyp konfigurieren, z. B. On-Demand und Spot.
- Wenn die Nachfrage nach Anwendungsprozessen steigt und die Nachfrage nach Ausführern sinkt, wenn Sie den Anwendungsprozess auf Kernknoten beschränkt haben, können Sie bei derselben Größenänderung wieder Kernknoten hinzufügen und Aufgabenknoten entfernen. Weitere Informationen finden Sie unter Grundlegendes zu Strategien und Szenarien für die Knotenzuweisung.
- Amazon kennzeichnet Aufgabenknoten EMR nicht, sodass Sie die YARN Eigenschaften nicht festlegen können, um Anwendungsprozesse nur für Aufgabenknoten einzuschränken. Wenn Sie jedoch Markttypen als Knotenbezeichnungen verwenden möchten, können Sie die `SPOT` Bezeichnungen `ON_DEMAND` oder für die Platzierung von Antragsprozessen verwenden. Wir empfehlen nicht, Spot-Nodes für primäre Anwendungsprozesse zu verwenden.
- Wenn Sie Node Labels verwenden, kann die Gesamtzahl der laufenden Einheiten im Cluster vorübergehend die in Ihrer verwalteten Skalierungsrichtlinie festgelegte maximale Rechenleistung überschreiten, während Amazon EMR einige Ihrer Instances außer Betrieb nimmt. Die Gesamtzahl der angeforderten Einheiten bleibt immer auf oder unter der in Ihrer Richtlinie festgelegten maximalen Rechenleistung.
- Managed Scaling unterstützt nur die Knotenbezeichnungen `ON_DEMAND` und `SPOT` oder `CORE` und`TASK`. Benutzerdefinierte Knotenbezeichnungen werden nicht unterstützt.
- Amazon EMR erstellt bei der Erstellung des Clusters und der Bereitstellung von Ressourcen Knotenbezeichnungen. Amazon unterstützt das Hinzufügen von Knotenbezeichnungen bei der Neukonfiguration des Clusters EMR nicht. Sie können die Knotenbezeichnungen auch nicht ändern, wenn Sie die verwaltete Skalierung nach dem Start des Clusters konfigurieren.

- Bei der verwalteten Skalierung werden Kern- und Taskknoten unabhängig voneinander auf der Grundlage des Anwendungsprozesses und der Nachfrage der Executoren skaliert. Um HDFS Datenverlust beim Herunterfahren der Kerne zu vermeiden, sollten Sie sich an die Standardpraxis für Kernknoten halten. Weitere Informationen zu bewährten Methoden für Kernknoten und HDFS Replikation finden Sie unter Überlegungen und bewährte Methoden.
- Sie können nicht sowohl den Anwendungsprozess als auch die Executoren nur auf dem Knoten `core` oder dem `ON_DEMAND` Knoten platzieren. Wenn Sie sowohl den Anwendungsprozess als auch die Executoren auf einem der Knoten hinzufügen möchten, verwenden Sie die Konfiguration nicht. `yarn.node-labels.am.default-node-label-expression`

  Um beispielsweise sowohl den Anwendungsprozess als auch die Executoren in `ON_DEMAND` Knoten zu platzieren, setzen Sie max compute auf den Wert Maximum im Knoten. `ON_DEMAND` Entfernen Sie außerdem die Konfiguration`yarn.node-labels.am.default-node-label-expression`.

  Um sowohl den Anwendungsprozess als auch Executoren auf den `core` Knoten hinzuzufügen, entfernen Sie die `yarn.node-labels.am.default-node-label-expression` Konfiguration.
- Wenn Sie verwaltete Skalierung mit Knotenbezeichnungen verwenden, legen Sie die Eigenschaft fest, `yarn.scheduler.capacity.maximum-am-resource-percent: 1` wenn Sie mehrere Anwendungen parallel ausführen möchten. Dadurch wird sichergestellt, dass Ihre Anwendungsprozesse die verfügbaren `CORE` `ON_DEMAND` OR-Knoten vollständig nutzen.
- Wenn Sie verwaltete Skalierung mit Knotenbezeichnungen verwenden, legen Sie für `yarn.resourcemanager.decommissioning.timeout` die Eigenschaft einen Wert fest, der länger ist als die am längsten laufende Anwendung in Ihrem Cluster. Dadurch wird die Wahrscheinlichkeit verringert, dass Amazon EMR Managed Scaling Ihre Anwendungen für die Wiederinbetriebnahme oder Knoten neu planen muss. `CORE` `ON_DEMAND`

Feature-Verlauf

In dieser Tabelle sind Aktualisierungen der von Amazon EMR verwalteten Skalierungsfunktion aufgeführt.

Datum der Veröffentlichung	Funktion	EMRAmazon-Versionen
20. August 2024	Node-Labels sind jetzt in Managed Scaling verfügbar , sodass Sie Ihre Instances	7.2.0 und höher

Datum der Veröffentlichung	Funktion	EMRAmazon-Versionen
	nach Markt- oder Knotentyp kennzeichnen können, um die automatische Skalierung zu verbessern.	
31. März 2024	Managed Scaling ist in der Region `ap-south-2` Asien-Pazifik (Hyderabad) verfügbar.	6.14.0 und höher
13. Februar 2024	Managed Scaling ist in der Region `eu-south-2` Europa (Spanien) verfügbar.	6.14.0 und höher
10. Oktober 2023	Managed Scaling ist in der `ap-southeast-3` -Region Asien-Pazifik (Jakarta) verfügbar.	6.14.0 und höher
28. Juli 2023	Verbesserte verwaltete Skalierung, um beim Scale-up zu einer anderen Task-Instance-Gruppe zu wechseln, wenn Amazon EMR bei der Skalierung mit der aktuellen Instance-Gruppe eine Verzögerung feststellt.	5.34.0 und höher, 6.4.0 und höher

Datum der Veröffentlichung	Funktion	EMRAmazon-Versionen
16. Juni 2023	Verbesserte verwaltete Skalierung, sodass erkannt wird, auf welchen Knoten der Application Master ausgeführt wird, sodass diese Knoten nicht herunterskaliert werden. Weitere Informationen finden Sie unter Grundlegendes zu Strategien und Szenarien für die Knotenzuweisung.	5.34.0 und höher, 6.4.0 und höher

Datum der Veröffentlichung	Funktion	EMRAmazon-Versionen
21. März 2022	Spark Shuffle Data Awareness wurde hinzugefügt, das beim Herunterskalieren von Clustern verwendet wird. Bei EMR Amazon-Clustern mit Apache Spark und aktiviert er Managed Scaling-Funktion überwacht Amazon EMR kontinuierlich Spark-Executors und Zwischenspeicherorte für Shuffle-Daten. Anhand dieser Informationen EMR skaliert Amazon nur ungenutzte Instances herunter, die keine aktiv genutzten Shuffle-Daten enthalten. Dadurch wird eine Neuberechnung verloren gegangener Shuffle-Daten verhindert, was zur Senkung der Kosten und zur Verbesserung der Arbeitsleistung beiträgt. Weitere Informationen finden Sie unter im Spark-Programmierhandbuch.	5.34.0 und höher, 6.4.0 und höher

Konfiguration der verwalteten Skalierung für Amazon EMR

In den folgenden Abschnitten wird erklärt, wie Sie einen EMR Cluster starten, der verwaltete Skalierung mit dem AWS Management Console AWS SDK for Java, dem oder dem verwendet AWS Command Line Interface.

Themen

- Verwenden Sie den AWS Management Console , um die verwaltete Skalierung zu konfigurieren
- Verwenden Sie den AWS CLI , um die verwaltete Skalierung zu konfigurieren

- Wird verwendet AWS SDK for Java , um verwaltete Skalierung zu konfigurieren

Verwenden Sie den AWS Management Console , um die verwaltete Skalierung zu konfigurieren

Sie können die EMR Amazon-Konsole verwenden, um verwaltete Skalierung zu konfigurieren, wenn Sie einen Cluster erstellen, oder um eine verwaltete Skalierungsrichtlinie für einen laufenden Cluster zu ändern.

Console

Um verwaltete Skalierung zu konfigurieren, wenn Sie einen Cluster mit der Konsole erstellen

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und anschließend Create cluster aus.
3. Wählen Sie eine EMR Amazon-Version emr-5.30.0 oder höher, außer Version emr-6.0.0.
4. Wählen Sie unter Option Clusterskalierung und -bereitstellung die Option Use -managed scaling aus. EMR Geben Sie das Minimum – und Maximum von Instances, die maximale Anzahl an Core-Knoten-Instances und die maximale Anzahl von On-Demand-Instances an.
5. Wählen Sie alle anderen Optionen aus, die für Ihren Cluster gelten.
6. Um Ihren Cluster jetzt zu starten, wählen Sie Cluster erstellen aus.

Um verwaltete Skalierung auf einem vorhandenen Cluster mit der Konsole zu konfigurieren

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und dann den Cluster aus, den Sie aktualisieren möchten.
3. Suchen Sie auf der Registerkarte Instances der Cluster-Detailseite den Abschnitt Instance-Gruppen-Einstellungen. Geben Sie im Abschnitt Clusterskalierung bearbeiten neue Werte für die Minimum- und Maximum-Anzahl von Instances und das On-Demand-Limit an.

Verwenden Sie den AWS CLI , um die verwaltete Skalierung zu konfigurieren

Sie können AWS CLI Befehle für Amazon verwendenEMR, um die verwaltete Skalierung zu konfigurieren, wenn Sie einen Cluster erstellen. Sie können eine Kurzsyntax verwenden und die JSON Konfiguration direkt in den entsprechenden Befehlen angeben, oder Sie können auf eine Datei verweisen, die die Konfiguration enthält. JSON Sie können auch eine Richtlinie für verwaltete Skalierung auf einen vorhandenen Cluster anwenden und eine zuvor angewendete Richtlinie für verwaltete Skalierung entfernen. Darüber hinaus können Sie Details einer Skalierungsrichtlinien-Konfiguration aus einem aktuell ausgeführten Cluster abrufen.

Aktivieren der verwalteten Skalierung während des Clusterstarts

Sie können die verwaltete Skalierung während des Clusterstarts aktivieren, wie im folgenden Beispiel veranschaulicht wird.

```
aws emr create-cluster \
 --service-role EMR_DefaultRole \
 --release-label emr-7.2.0 \
 --name EMR_Managed_Scaling_Enabled_Cluster \
 --applications Name=Spark Name=Hbase \
 --ec2-attributes KeyName=keyName,InstanceProfile=EMR_EC2_DefaultRole \
 --instance-groups InstanceType=m4.xlarge,InstanceGroupType=MASTER,InstanceCount=1
 InstanceType=m4.xlarge,InstanceGroupType=CORE,InstanceCount=2 \
 --region us-east-1 \
 --managed-scaling-policy
 ComputeLimits='{MinimumCapacityUnits=2,MaximumCapacityUnits=4,UnitType=Instances}'
```

Sie können bei Verwendung der managed-scaling-policy Option -- auch eine verwaltete Richtlinienkonfiguration angeben. `create-cluster`

Anwenden einer Richtlinie für verwaltete Skalierung auf einen vorhandenen Cluster

Sie können eine Richtlinie für verwaltete Skalierung auf einen vorhandenen Cluster anwenden, wie im folgenden Beispiel veranschaulicht wird.

```
aws emr put-managed-scaling-policy
--cluster-id j-123456
--managed-scaling-policy ComputeLimits='{MinimumCapacityUnits=1,
MaximumCapacityUnits=10,  MaximumOnDemandCapacityUnits=10, UnitType=Instances}'
```

Sie können eine Richtlinie für verwaltete Skalierung auch auf einen vorhandenen Cluster anwenden, indem Sie den Befehl `aws emr put-managed-scaling-policy` verwenden. Im folgenden Beispiel wird ein Verweis auf eine JSON Datei verwendet`managedscaleconfig.json`, die die Konfiguration der verwalteten Skalierungsrichtlinie spezifiziert.

```
aws emr put-managed-scaling-policy --cluster-id j-123456 --managed-scaling-policy
 file://./managedscaleconfig.json
```

Das folgende Beispiel zeigt den Inhalt der Datei `managedscaleconfig.json`, in der die Richtlinie für verwaltete Skalierung definiert wird.

```
{
    "ComputeLimits": {
        "UnitType": "Instances",
        "MinimumCapacityUnits": 1,
        "MaximumCapacityUnits": 10,
        "MaximumOnDemandCapacityUnits": 10
    }
}
```

Abrufen einer Richtlinienkonfiguration für verwaltete Skalierung

Der Befehl `GetManagedScalingPolicy` ruft die Richtlinienkonfiguration ab. Mit dem folgenden Befehl wird beispielsweise die Konfiguration für den Cluster mit der Cluster-ID `j-123456` abgerufen.

```
aws emr get-managed-scaling-policy --cluster-id j-123456
```

Der Befehl generiert die folgende Beispielausgabe:

```
{
   "ManagedScalingPolicy": {
      "ComputeLimits": {
         "MinimumCapacityUnits": 1,
         "MaximumOnDemandCapacityUnits": 10,
         "MaximumCapacityUnits": 10,
         "UnitType": "Instances"
      }
   }
}
```

Weitere Informationen zur Verwendung von EMR Amazon-Befehlen in der AWS CLI finden Sie unterhttps://docs.aws.amazon.com/cli/latest/reference/emr.

Entfernen der Richtlinie für verwaltete Skalierung

Mit dem Befehl `RemoveManagedScalingPolicy` wird die Richtlinienkonfiguration entfernt. Mit dem folgenden Befehl wird beispielsweise die Konfiguration für den Cluster mit der Cluster-ID `j-123456` entfernt.

```
aws emr remove-managed-scaling-policy --cluster-id j-123456
```

Wird verwendet AWS SDK for Java , um verwaltete Skalierung zu konfigurieren

Der folgende Programmausschnitt zeigt, wie die verwaltete Skalierung mit dem AWS SDK for Java konfiguriert wird:

```
package com.amazonaws.emr.sample;

import java.util.ArrayList;
import java.util.List;

import com.amazonaws.AmazonClientException;
import com.amazonaws.auth.AWSCredentials;
import com.amazonaws.auth.AWSStaticCredentialsProvider;
import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.regions.Regions;
import com.amazonaws.services.elasticmapreduce.AmazonElasticMapReduce;
import com.amazonaws.services.elasticmapreduce.AmazonElasticMapReduceClientBuilder;
import com.amazonaws.services.elasticmapreduce.model.Application;
import com.amazonaws.services.elasticmapreduce.model.ComputeLimits;
import com.amazonaws.services.elasticmapreduce.model.ComputeLimitsUnitType;
import com.amazonaws.services.elasticmapreduce.model.InstanceGroupConfig;
import com.amazonaws.services.elasticmapreduce.model.JobFlowInstancesConfig;
import com.amazonaws.services.elasticmapreduce.model.ManagedScalingPolicy;
import com.amazonaws.services.elasticmapreduce.model.RunJobFlowRequest;
import com.amazonaws.services.elasticmapreduce.model.RunJobFlowResult;

public class CreateClusterWithManagedScalingWithIG {

 public static void main(String[] args) {
  AWSCredentials credentialsFromProfile = getCreadentials("AWS-Profile-Name-Here");

  /**
```

```
   * Create an Amazon EMR client with the credentials and region specified in order to
create the cluster
   */
 AmazonElasticMapReduce emr = AmazonElasticMapReduceClientBuilder.standard()
   .withCredentials(new AWSStaticCredentialsProvider(credentialsFromProfile))
   .withRegion(Regions.US_EAST_1)
   .build();

 /**
   * Create Instance Groups - Primary, Core, Task
   */
 InstanceGroupConfig instanceGroupConfigMaster = new InstanceGroupConfig()
    .withInstanceCount(1)
    .withInstanceRole("MASTER")
    .withInstanceType("m4.large")
    .withMarket("ON_DEMAND");

 InstanceGroupConfig instanceGroupConfigCore = new InstanceGroupConfig()
   .withInstanceCount(4)
   .withInstanceRole("CORE")
   .withInstanceType("m4.large")
   .withMarket("ON_DEMAND");

 InstanceGroupConfig instanceGroupConfigTask = new InstanceGroupConfig()
   .withInstanceCount(5)
   .withInstanceRole("TASK")
   .withInstanceType("m4.large")
   .withMarket("ON_DEMAND");

 List<InstanceGroupConfig> igConfigs = new ArrayList<>();
 igConfigs.add(instanceGroupConfigMaster);
 igConfigs.add(instanceGroupConfigCore);
 igConfigs.add(instanceGroupConfigTask);

       /**
        *  specify applications to be installed and configured when Amazon EMR creates
the cluster
        */
 Application hive = new Application().withName("Hive");
 Application spark = new Application().withName("Spark");
 Application ganglia = new Application().withName("Ganglia");
 Application zeppelin = new Application().withName("Zeppelin");

 /**
```

```
   * Managed Scaling Configuration -
         * Using UnitType=Instances for clusters composed of instance groups
   *
         * Other options are:
         * UnitType = VCPU ( for clusters composed of instance groups)
         * UnitType = InstanceFleetUnits ( for clusters composed of instance fleets)
         **/
 ComputeLimits computeLimits = new ComputeLimits()
   .withMinimumCapacityUnits(1)
   .withMaximumCapacityUnits(20)
   .withUnitType(ComputeLimitsUnitType.Instances);

 ManagedScalingPolicy managedScalingPolicy = new ManagedScalingPolicy();
 managedScalingPolicy.setComputeLimits(computeLimits);

 // create the cluster with a managed scaling policy
 RunJobFlowRequest request = new RunJobFlowRequest()
         .withName("EMR_Managed_Scaling_TestCluster")
         .withReleaseLabel("emr-7.2.0")          // Specifies the version label for
the Amazon EMR release; we recommend the latest release
         .withApplications(hive,spark,ganglia,zeppelin)
         .withLogUri("s3://path/to/my/emr/logs")  // A URI in S3 for log files is
required when debugging is enabled.
         .withServiceRole("EMR_DefaultRole")      // If you use a custom IAM service
role, replace the default role with the custom role.
         .withJobFlowRole("EMR_EC2_DefaultRole")  // If you use a custom Amazon EMR
role for EC2 instance profile, replace the default role with the custom Amazon EMR
role.
         .withInstances(new JobFlowInstancesConfig().withInstanceGroups(igConfigs)
             .withEc2SubnetId("subnet-123456789012345")
             .withEc2KeyName("my-ec2-key-name")
             .withKeepJobFlowAliveWhenNoSteps(true))
         .withManagedScalingPolicy(managedScalingPolicy);
   RunJobFlowResult result = emr.runJobFlow(request);

   System.out.println("The cluster ID is " + result.toString());
}

public static AWSCredentials getCredentials(String profileName) {
 // specifies any named profile in .aws/credentials as the credentials provider
 try {
  return new ProfileCredentialsProvider("AWS-Profile-Name-Here")
    .getCredentials();
       } catch (Exception e) {
```

```
            throw new AmazonClientException(
                    "Cannot load credentials from .aws/credentials file. " +
                    "Make sure that the credentials file exists and that the profile
 name is defined within it.",
                    e);
        }
 }

 public CreateClusterWithManagedScalingWithIG() { }
}
```

Grundlegendes zu Strategien und Szenarien für die Knotenzuweisung

Dieser Abschnitt gibt einen Überblick über die Strategie zur Knotenzuweisung und allgemeine Skalierungsszenarien, die Sie mit Amazon EMR Managed Scaling verwenden können.

Knotenzuweisungsstrategie

Amazon EMR Managed Scaling weist Kern- und Aufgabenknoten auf der Grundlage der folgenden Scale-Up- und Scale-Down-Strategien zu:

Strategie zum hochskalieren

- Bei EMR Amazon-Versionen 7.2 und höher fügt die verwaltete Skalierung zunächst Knoten hinzu, die auf Knotenbezeichnungen und der YARN Eigenschaft „Anwendungsprozesseinschränkung“ basieren.
- Für EMR Amazon-Versionen 7.2 und höher gilt: Wenn Sie Node Labels aktiviert und Anwendungsprozesse auf `CORE` Knoten beschränkt haben, skaliert Amazon EMR Managed Scaling die Kernknoten und Aufgabenknoten hoch, wenn die Nachfrage nach Anwendungsprozessen steigt und die Nachfrage nach Ausführern steigt. Wenn Sie Node Labels aktiviert und Anwendungsprozesse auf Knoten beschränkt haben, skaliert Managed Scaling entsprechend On-Demand-Knoten hoch, wenn die Nachfrage nach Anwendungsprozessen steigt, und skaliert Spot-Knoten, wenn die Nachfrage nach Ausführern steigt. `ON_DEMAND`
- Wenn Node Labels nicht aktiviert sind, ist die Platzierung von Anwendungsprozessen nicht auf einen Knoten oder Markttyp beschränkt.
- Durch die Verwendung von Node Labels kann Managed Scaling verschiedene Instanzgruppen und Instanzflotten bei derselben Größenänderung hoch- und herunterskalieren. Zum Beispiel in einem Szenario, in dem `instance_group1` ein `ON_DEMAND` Knoten und ein `SPOT` Knoten `instance_group2` vorhanden sind und die Knotenbezeichnungen aktiviert sind und Anwendungsprozesse auf Knoten mit der `ON_DEMAND` Bezeichnung beschränkt

sind. Die verwaltete Skalierung wird herunterskaliert `instance_group1` und hochskaliert, `instance_group2` wenn die Nachfrage nach Anwendungsprozessen sinkt und die Nachfrage nach Ausführern steigt.

- Wenn Amazon eine EMR Verzögerung beim Skalieren mit der aktuellen Instance-Gruppe feststellt, wechseln Cluster, die Managed Scaling verwenden, automatisch zu einer anderen Task-Instance-Gruppe.
- Wenn der `MaximumCoreCapacityUnits` Parameter gesetzt ist, EMR skaliert Amazon die Kernknoten, bis die Kerneinheiten das maximal zulässige Limit erreichen. Die gesamte verbleibende Kapazität wird den Aufgabenknoten hinzugefügt.
- Wenn der `MaximumOnDemandCapacityUnits` Parameter festgelegt ist, EMR skaliert Amazon den Cluster mithilfe der On-Demand-Instances, bis die On-Demand-Einheiten den maximal zulässigen Grenzwert erreichen. Die gesamte verbleibende Kapazität wird mithilfe von Spot Instances hinzugefügt.
- Wenn `MaximumCoreCapacityUnits` sowohl der als auch der `MaximumOnDemandCapacityUnits` Parameter festgelegt sind, EMR berücksichtigt Amazon bei der Skalierung beide Grenzwerte.

  Wenn der beispielsweise kleiner als `MaximumCoreCapacityUnits` ist`MaximumOnDemandCapacityUnits`, skaliert Amazon EMR zunächst die Kernknoten, bis die Kernkapazitätsgrenze erreicht ist. Für die verbleibende Kapazität verwendet Amazon EMR zunächst On-Demand-Instances, um Aufgabenknoten zu skalieren, bis das On-Demand-Limit erreicht ist, und verwendet dann Spot-Instances für Aufgabenknoten.

Strategie zum herunterskalieren

- Ähnlich wie bei der Scale-up-Strategie EMR entfernt Amazon Knoten, die auf Knotenbezeichnungen basieren. Weitere Informationen zu Knotenbezeichnungen finden Sie unter Grundlegendes zu Knotentypen: Primär-, Kern- und Aufgabenknoten.
- Wenn Sie Node Labels nicht aktiviert haben, entfernt Managed Scaling Task-Knoten und anschließend Core-Knoten, bis die gewünschte Scale-Down-Zielkapazität erreicht ist. Bei der verwalteten Skalierung wird der Cluster niemals unter die in der Richtlinie für verwaltete Skalierung angegebenen Mindestbeschränkungen herunterskaliert.
- Die EMR Amazon-Versionen 5.34.0 und höher sowie die EMR Amazon-Versionen 6.4.0 und höher unterstützen verwaltete Skalierung, die Spark-Shuffle-Daten berücksichtigt (Daten, die Spark partitionsübergreifend verteilt, um bestimmte Operationen auszuführen). Weitere Informationen zu Shuffle-Vorgängen finden Sie im Spark-Programmierhandbuch. Bei der verwalteten Skalierung

werden nur Instances herunterskaliert, die nicht ausreichend ausgelastet sind und keine aktiv genutzten Shuffle-Daten enthalten. Diese intelligente Skalierung verhindert den unbeabsichtigten Verlust von Shuffle-Daten, sodass keine erneuten Versuche und die Neuberechnung von Zwischendaten erforderlich sind.

- Bei der verwalteten Skalierung werden zuerst Aufgabenknoten und dann Kernknoten entfernt, bis die gewünschte Zielkapazität für das Herunterskalieren erreicht ist. Der Cluster wird niemals unter die in der Richtlinie für verwaltete Skalierung angegebenen Mindestbeschränkungen skaliert.
- Bei Clustern, die mit Amazon EMR 5.x-Versionen 5.34.0 und höher und 6.x-Versionen 6.4.0 und höher gestartet werden, skaliert die von Amazon EMR verwaltete Skalierung keine Knoten, auf denen Apache Spark ausgeführt wird. `ApplicationMaster` Dadurch werden Fehlschläge und Wiederholungen von Aufträgen minimiert, was zur Verbesserung der Auftragsleistung und zur Senkung der Kosten beiträgt. Um zu überprüfen, welche Knoten in Ihrem Cluster `ApplicationMaster` ausführen, besuchen Sie den Spark History Server und filtern Sie auf der Registerkarte Executors Ihrer Spark-Anwendungs-ID nach dem Treiber.

Wenn der Cluster nicht ausgelastet ist, EMR storniert Amazon das Hinzufügen neuer Instances aus einer früheren Evaluierung und führt Scale-Down-Operationen durch. Wenn der Cluster stark ausgelastet ist, EMR storniert Amazon das Entfernen von Instances und führt Scale-up-Operationen durch.

Überlegungen zur Knotenzuweisung

Wir empfehlen Ihnen, die On-Demand-Kaufoption für Core-Nodes zu verwenden, um HDFS Datenverlust im Falle einer Spot-Rückforderung zu vermeiden. Sie können die Spot-Kaufoption für Aufgabenknoten verwenden, um die Kosten zu senken und die Auftragsausführung zu beschleunigen, wenn mehr Spot Instances zu Aufgabenknoten hinzugefügt werden.

Knotenzuweisungsszenarien

Sie können je nach Bedarf verschiedene Skalierungsszenarien erstellen, indem Sie die Core-Knotenparameter Maximum, Minimum, On-Demand-Limit und Maximum in unterschiedlichen Kombinationen einrichten.

Szenario 1: Nur Core-Knoten skalieren

Um nur Core-Knoten zu skalieren, müssen die verwalteten Skalierungsparameter die folgenden Anforderungen erfüllen:

- Das On-Demand-Limit entspricht der maximalen Grenze.

- Der maximale Core-Knoten entspricht der maximalen Grenze.

Wenn das On-Demand-Limit und die maximale Anzahl an Core-Knoten nicht angegeben sind, verwenden beide Parameter standardmäßig die maximale Grenze.

Dieses Szenario ist nicht anwendbar, wenn Sie Managed Scaling mit Node-Labels verwenden und Ihre Anwendungsprozesse darauf beschränken, nur auf `CORE` Knoten ausgeführt zu werden, da Managed Scaling Task-Knoten skaliert, um der Nachfrage der Executoren gerecht zu werden.

Das folgende Beispiele zeigt das Szenario der ausschließlichen Skalierung von Core-Knoten.

Ausgangszustand des Clusters	Skalierungsparameter	Skalierungs-Verhalten
Instance-Gruppen Core: 1 On-Demand Aufgabe: 1 On-Demand- und 1 Spot	`UnitType`: Instances `MinimumCapacityUnits` : 1 `MaximumCapacityUnits` : 20 `MaximumOnDemandCapacityUnits` : 20 `MaximumCoreCapacityUnits` : 20	Skalieren Sie mithilfe des On-Demand-Typs zwischen 1 und 20 Instances oder Instance-Flotteneinheiten auf Core-Knoten. Keine Skalierung auf Aufgabenknoten. Wenn Sie Managed Scaling mit Node Labels verwenden und Ihre Anwendungsprozesse auf `ON_DEMAND` Knoten beschränken, skaliert
Instance-Flotten Core: 1 On-Demand Aufgabe: 1 On-Demand- und 1 Spot	UnitType: InstanceFleetUnits `MinimumCapacityUnits` : 1 `MaximumCapacityUnits` : 20 `MaximumOnDemandCapacityUnits` : 20 `MaximumCoreCapacityUnits` : 20	

Ausgangszustand des Clusters	Skalierungsparameter	Skalierungs-Verhalten
		der Cluster je nach Art der Nachfrage 1 bis 20 Instanzen oder Instanzfl otteneinh eiten auf CORE Knoten, die den Spot Typ On-Demand oder verwenden.

Szenario 2: Nur Aufgabenknoten skalieren

Um nur Aufgabenknoten zu skalieren, müssen die verwalteten Skalierungsparameter die folgenden Anforderungen erfüllen:

- Der maximale Core-Knoten muss der Mindestgrenze entsprechen.

Das folgende Beispiele zeigt das Szenario der ausschließlichen Skalierung von Aufgabenknoten.

Ausgangszustand des Clusters	Skalierungsparameter	Skalierungs-Verhalten
Instance-Gruppen Core: 2 On-Demand Aufgabe: 1 Spot	UnitType: Instances MinimumCapacityUnits : 2 MaximumCapacityUnits : 20 MaximumCoreCapacityUnits : 2	Halten Sie die Anzahl der Core-Knoten konstant bei 2 und skalieren Sie nur Aufgabenknoten zwischen 0 und 18 Instances oder Instance-
Instance-Flotten	UnitType: InstanceFleetUnits	

Ausgangszustand des Clusters	Skalierungsparameter	Skalierungs-Verhalten
Core: 2 On-Demand Aufgabe: 1 Spot	`MinimumCapacityUnits` : 2 `MaximumCapacityUnits` : 20 `MaximumCoreCapacityUnits` : 2	Flotteneinheiten. Die Kapazität zwischen Mindest- und Höchstgrenzen wird nur den Aufgabenknoten hinzugefügt. Wenn Sie Managed Scaling mit Node-Labels verwenden und Ihre Anwendung sprozesse auf DEMAND ON_-Knoten beschränken, hält der Cluster die Anzahl der Kernknoten konstant bei 2 und skaliert je nach `Spot` Art der Nachfrage nur Task-Knot en zwischen 0 und 18 Instances `0n-demand` oder Instance-Flotteneinheiten, die den Typ oder verwenden.

Szenario 4: Nur On-Demand-Instance im Cluster

Um nur über On-Demand-Instances zu verfügen, müssen Ihr Cluster und die verwalteten Skalierungsparameter die folgende Anforderung erfüllen:

- Das On-Demand-Limit entspricht der maximalen Grenze.

  Wenn das On-Demand-Limit nicht angegeben ist, entspricht der Parameterwert standardmäßig der Höchstgrenze. Der Standardwert gibt an, dass Amazon nur On-Demand-Instances EMR skaliert.

Wenn die maximale Anzahl an Core-Knoten kleiner als die maximale Grenze ist, kann der Parameter „Maximaler Core-Knoten“ verwendet werden, um die Kapazitätszuweisung zwischen Core- und Aufgabenknoten aufzuteilen.

Um dieses Szenario in einem Cluster zu aktivieren, der aus Instance-Gruppen besteht, müssen alle Knotengruppen im Cluster bei der Erstkonfiguration den Markttyp On-Demand verwenden.

Dieses Szenario ist nicht anwendbar, wenn Sie Managed Scaling mit Node-Labels verwenden und Ihre Anwendungsprozesse darauf beschränken, nur auf `ON_DEMAND` Knoten ausgeführt zu werden, da Managed Scaling `Spot` Knoten skaliert, um der Nachfrage der Executoren gerecht zu werden.

Die folgenden Beispiele veranschaulichen das Szenario, in dem On-Demand-Instances im gesamten Cluster vorhanden sind.

Ausgangszustand des Clusters	Skalierungsparameter	Skalierungs-Verhalten
Instance-Gruppen Core: 1 On-Demand Aufgabe: 1 On-Demand	`UnitType`: Instances `MinimumCapacityUnits` : 1 `MaximumCapacityUnits` : 20 `MaximumOnDemandCapacityUnits` : 20 `MaximumCoreCapacityUnits` : 12	Skalieren Sie mithilfe des On-Demand-Typs zwischen 1 und 12 Instances oder Instance-Flotteneinheiten auf Core-Knoten. Skalieren Sie die verbleibende Kapazität mithilfe
Instance-Flotten	`UnitType`: InstanceFleetUnits	

Ausgangszustand des Clusters	Skalierungsparameter	Skalierungs-Verhalten
Core: 1 On-Demand Aufgabe: 1 On-Demand	`MinimumCapacityUnits` : 1 `MaximumCapacityUnits` : 20 `MaximumOnDemandCapacityUnits` : 20 `MaximumCoreCapacityUnits` : 12	von On-Demand -Funktion auf Aufgabenk noten. Keine Skalierung mit Spot Instances. Wenn Sie Managed Scaling mit Node Labels verwenden und Ihre Anwendung sprozesse auf `CORE` Knoten beschränk en, skaliert der Cluster je nach Art der Nachfrage zwischen 1 und 20 Instanzen oder Instanzfl otteneinh eiten auf `CORE` `task` Knoten oder Knoten, die diesen `ON_DEMAND` Typ verwenden . Die Skalierun g auf Kernknote n wird 12 Instances

Ausgangszustand des Clusters	Skalierungsparameter	Skalierungs-Verhalten
		oder Instance-Flotteneinheiten nicht überschre iten.

Szenario 4: Nur Spot Instances im Cluster

Um nur Spot Instances zu verwenden, müssen die verwalteten Skalierungsparameter die folgenden Anforderungen erfüllen:

- Das On-Demand-Limit ist auf 0 gesetzt.

Wenn die maximale Anzahl an Core-Knoten kleiner als die maximale Grenze ist, kann der Parameter „Maximaler Core-Knoten“ verwendet werden, um die Kapazitätszuweisung zwischen Core- und Aufgabenknoten aufzuteilen.

Um dieses Szenario in einem Cluster zu aktivieren, der aus Instance-Gruppen besteht, muss die Kern-Instance-Gruppe bei der Erstkonfiguration die Spot-Kaufoption verwenden. Wenn die Task-Instance-Gruppe keine Spot-Instance enthält, erstellt Amazon EMR Managed Scaling bei Bedarf eine Auftragsgruppe mithilfe von Spot-Instances.

Dieses Szenario ist nicht anwendbar, wenn Sie Managed Scaling mit Node-Labels verwenden und Ihre Anwendungsprozesse darauf beschränken, nur auf `ON_DEMAND` Knoten ausgeführt zu werden, da Managed Scaling `ON_DEMAND` Knoten skaliert, um den Anforderungen der Anwendungsprozesse gerecht zu werden.

Die folgenden Beispiele veranschaulichen das Szenario, in dem Spot Instances im gesamten Cluster vorhanden sind.

Ausgangszustand des Clusters	Skalierungsparameter	Skalierungs-Verhalten
Instance-Gruppen	`UnitType`: Instances	Skalieren Sie mithilfe von Spot zwischen 1 und

Ausgangszustand des Clusters	Skalierungsparameter	Skalierungs-Verhalten
Core: 1 Spot Aufgabe: 1 Spot	`MinimumCapacityUnits` : 1 `MaximumCapacityUnits` : 20 `MaximumOnDemandCapacityUnits` : 0	20 Instances oder Instance-Flotteneinheiten auf Core-Knoten. Keine Skalierung beim On-Demand-Typ. Wenn Sie Managed Scaling mit Node Labels verwenden und Ihre Anwendungsprozesse auf `CORE` Knoten beschränken, skaliert der Cluster je nach Art des Bedarfs zwischen 1 und 20 Instances `CORE` oder Instance-Flotteneinheiten auf oder `TASK` Knoten, die Spot verwenden. Amazon skaliert EMR nicht mit dem `ON_DEMAND` Typ.
Instance-Flotten Core: 1 Spot Aufgabe: 1 Spot	`UnitType`: InstanceFleetUnits `MinimumCapacityUnits` : 1 `MaximumCapacityUnits` : 20 `MaximumOnDemandCapacityUnits` : 0	

Szenario 5: On-Demand-Instances auf Core-Knoten und Spot Instances auf Aufgabenknoten skalieren

Um On-Demand-Instances auf Core-Knoten und Spot Instances auf Aufgabenknoten zu skalieren, müssen die verwalteten Skalierungsparameter die folgenden Anforderungen erfüllen:

- Das On-Demand-Limit muss dem maximalen Core-Knoten entsprechen.
- Sowohl das On-Demand-Limit als auch die maximale Anzahl an Core-Knoten müssen unter der maximalen Grenze liegen.

Um dieses Szenario in einem Cluster zu aktivieren, der aus Instance-Gruppen besteht, muss die Core-Knotengruppe die On-Demand-Kaufoption verwenden.

Dieses Szenario ist nicht anwendbar, wenn Sie verwaltete Skalierung mit Knotenbezeichnungen verwenden und Ihre Anwendungsprozesse darauf beschränken, nur auf `ON_DEMAND` Knoten oder `CORE` Knoten ausgeführt zu werden.

Die folgenden Beispiele veranschaulichen das Szenario der Skalierung von On-Demand-Instances auf Core-Knoten und Spot Instances auf Aufgabenknoten.

Ausgangszustand des Clusters	Skalierungsparameter	Skalierungs-Verhalten
Instance-Gruppen Core: 1 On-Demand Aufgabe: 1 On-Demand- und 1 Spot	`UnitType`: Instances `MinimumCapacityUnits` : 1 `MaximumCapacityUnits` : 20 `MaximumOnDemandCapacityUnits` : 7 `MaximumCoreCapacityUnits` : 7	Hochskalieren Sie auf bis zu 6 On-Demand-Einheiten auf dem Core-Knoten, da sich bereits 1 On-Demand-Einheit auf dem Aufgabenknoten befindet und die maximale Anzahl für On-Demand-Einheiten
Instance-Flotten Core: 1 On-Demand	`UnitType`: InstanceFleetUnits `MinimumCapacityUnits` : 1	

Ausgangszustand des Clusters	Skalierungsparameter	Skalierungs-Verhalten
Aufgabe: 1 On-Demand- und 1 Spot	`MaximumCapacityUnits` : 20 `MaximumOnDemandCapacityUnits` : 7 `MaximumCoreCapacityUnits` : 7	7 beträgt. Hochskalieren Sie anschließend auf bis zu 13 Spot-Einheiten auf Aufgabenknoten.

Szenario 6: Skalieren Sie **`CORE`** Instances für den Bedarf von Anwendungsprozessen und **`TASK`** Instanzen für den Bedarf von Executoren.

Dieses Szenario ist nur anwendbar, wenn Sie verwaltete Skalierung mit Knotenbezeichnungen verwenden und Anwendungsprozesse so einschränken, dass sie nur auf `CORE` Knoten ausgeführt werden.

Um `CORE` Knoten auf der Grundlage der Anforderungen des Anwendungsprozesses und `TASK` Knoten auf der Grundlage der Anforderungen der Executoren zu skalieren, müssen Sie beim Clusterstart die folgenden Konfigurationen festlegen:

- `yarn.node-labels.enabled:true`
- `yarn.node-labels.am.default-node-label-expression: 'CORE'`

Wenn Sie den `ON_DEMAND` Grenzwert und den maximalen `CORE` Knotenparameter nicht angeben, verwenden beide Parameter standardmäßig die maximale Grenze.

Wenn die maximale Anzahl an `ON_DEMAND` Knoten kleiner als die maximale Grenze ist, verwendet die verwaltete Skalierung den maximalen `ON_DEMAND` Knotenparameter, um die Kapazitätszuweisung zwischen `SPOT` Knoten `ON_DEMAND` und Knoten aufzuteilen. Wenn Sie den Parameter für den maximalen `CORE` Knoten auf weniger als oder gleich dem Parameter für die minimale Kapazität festlegen, bleiben die `CORE` Knoten bei der maximalen Kernkapazität statisch.

Die folgenden Beispiele veranschaulichen das Szenario der Skalierung von CORE Instances auf der Grundlage der Anforderungen des Anwendungsprozesses und von TASK Instances auf der Grundlage der Nachfrage von Executoren.

Ausgangszustand des Clusters	Skalierungsparameter	Skalierungs-Verhalten
Instance-Gruppen Core: 1 On-Demand Aufgabe: 1 On-Demand	`UnitType`: Instances `MinimumCapacityUnits` : 1 `MaximumCapacityUnits` : 20 `MaximumOnDemandCapacityUnits` : 10 `MaximumCoreCapacityUnits` : 20	Skaliert `CORE` Knoten zwischen 1 und 20 Knoten basierend auf der Nachfrage nach dem Anwendung sprozess des Clusters unter Verwendung des Markttyps On-Demand oder Spot-Mark t. Skaliert `TASK` Knoten auf der Grundlage der Nachfrage der Executoren und der verbleibe nden verfügbar en Kapazität , nachdem Amazon Knoten zugewiesen hatEMR. `CORE` Die Summe der angeforderten `TASK` Knoten `CORE` und der Knoten wird den Wert von 20 nicht überschre
Instance-Flotten Core: 1 On-Demand Aufgabe: 1 On-Demand	`UnitType`: InstanceFleetUnits `MinimumCapacityUnits` : 1 `MaximumCapacityUnits` : 20 `MaximumOnDemandCapacityUnits` : 10 `MaximumCoreCapacityUnits` : 20	

Ausgangszustand des Clusters	Skalierungsparameter	Skalierungs-Verhalten
		iten. `maximumCapacity` Die Summe der angeforderten On-Demand-Core-Knoten und der On-Demand-Aufgabenknoten wird den Wert `maximumOnDemandCapacity` von 10 nicht überschreiten. Zusätzliche Kern- oder Taskknoten verwenden den Spotmarkttyp.

Szenario 7: Skalieren Sie **`ON_DEMAND`** Instances für den Bedarf an Anwendungsprozessen und **`SPOT`** Instances für den Bedarf an Executoren.

Dieses Szenario ist nur anwendbar, wenn Sie verwaltete Skalierung mit Knotenbezeichnungen verwenden und Anwendungsprozesse so einschränken, dass sie nur auf `ON_DEMAND` Knoten ausgeführt werden.

Um `ON_DEMAND` Knoten auf der Grundlage der Anforderungen des Anwendungsprozesses und `SPOT` Knoten auf der Grundlage der Anforderungen der Executoren zu skalieren, müssen Sie beim Clusterstart die folgenden Konfigurationen festlegen:

- `yarn.node-labels.enabled:true`
- `yarn.node-labels.am.default-node-label-expression: 'ON_DEMAND'`

Wenn Sie den ON_DEMAND Grenzwert und den maximalen CORE Knotenparameter nicht angeben, verwenden beide Parameter standardmäßig die maximale Grenze.

Wenn die maximale Anzahl an CORE Knoten kleiner als die maximale Grenze ist, verwendet die verwaltete Skalierung den maximalen CORE Knotenparameter, um die Kapazitätszuweisung zwischen TASK Knoten CORE und Knoten aufzuteilen. Wenn Sie den Parameter für den maximalen CORE Knoten auf weniger als oder gleich dem Parameter für die minimale Kapazität festlegen, bleiben die CORE Knoten bei der maximalen Kernkapazität statisch.

Die folgenden Beispiele veranschaulichen das Szenario der Skalierung von On-Demand-Instances auf der Grundlage der Anforderungen des Anwendungsprozesses und Spot-Instances auf der Grundlage der Nachfrage von Executoren.

Ausgangszustand des Clusters	Skalierungsparameter	Skalierungs-Verhalten
Instance-Gruppen Core: 1 On-Demand Aufgabe: 1 On-Demand	UnitType: Instances MinimumCapacityUnits : 1 MaximumCapacityUnits : 20 MaximumOnDemandCapacityUnits : 20 MaximumCoreCapacityUnits : 10	Skaliert ON_DEMAND Knoten zwischen 1 und 20 Knoten auf der Grundlage der Anforderungen des Clusters für den Anwendungsprozess unter Verwendung des CORE TASK Knotentyps oder. Skaliert SPOT Knoten auf der Grundlage der Nachfrage der Executoren und der verbleibenden verfügbaren Kapazität
Instance-Flotten Core: 1 On-Demand Aufgabe: 1 On-Demand	UnitType: InstanceFleetUnits MinimumCapacityUnits : 1 MaximumCapacityUnits : 20 MaximumOnDemandCapacityUnits : 20 MaximumCoreCapacityUnits : 10	

Ausgangszustand des Clusters	Skalierungsparameter	Skalierungs-Verhalten
		, nachdem Amazon Knoten zugewiese n hatEMR. `ON_DEMAND` Die Summe der angeforderten `SPOT` Knoten `ON_DEMAND` und der Knoten wird den Wert von 20 nicht überschreiten. `maximumCa pacity` Die Summe der angeforderten On-Demand -Core-Kno ten und Spot-Core-Knoten wird den Wert `maximumCo reCapacit y` von 10 nicht überschreiten. Zusätzliche On-Demand-Nodes oder Spot-Node s verwenden den `TASK` Knotentyp.

Grundlegendes zu Metriken für verwaltete Skalierung

Amazon EMR veröffentlicht hochauflösende Metriken mit Daten mit einer Granularität von einer Minute, wenn die verwaltete Skalierung für einen Cluster aktiviert ist. Sie können Ereignisse bei jeder Initiierung und Beendigung der Größenänderung anzeigen, die durch verwaltete Skalierung mit der EMR Amazon-Konsole oder der CloudWatch Amazon-Konsole gesteuert werden. CloudWatch Metriken sind entscheidend für den Betrieb von Amazon EMR Managed Scaling. Wir empfehlen Ihnen, die CloudWatch Metriken genau zu überwachen, um sicherzustellen, dass keine Daten fehlen. Weitere Informationen darüber, wie Sie CloudWatch Alarme konfigurieren können, um fehlende Messwerte zu erkennen, finden Sie unter CloudWatch Amazon-Alarme verwenden. Weitere Informationen zur Verwendung von CloudWatch Ereignissen mit Amazon EMR finden Sie unter CloudWatchEreignisse überwachen.

Die folgenden Metriken geben die aktuelle oder Zielkapazitäten eines Clusters an. Diese Metriken sind nur verfügbar, wenn die verwaltete Skalierung aktiviert ist. Bei Clustern, die aus Instance-Flotten bestehen, werden die Cluster-Kapazitätsmetriken in `Units` gemessen. Bei Clustern, die aus Instance-Gruppen bestehen, werden die Clusterkapazitätsmetriken in `Nodes` oder `vCPU` basierend auf dem Einheitentyp gemessen, der in der Richtlinie für verwaltete Skalierung verwendet wird.

Metrik	Beschreibung
• `TotalUnitsRequested` • `TotalNodesRequested` • `TotalVCPURequested`	Die angestrebte Gesamtzahl der Einheiten/Knoten/ vCPUs in einem Cluster, wie sie durch verwaltete Skalierung bestimmt wird. Einheiten: Anzahl
• `TotalUnitsRunning` • `TotalNodesRunning` • `TotalVCPURunning`	Die aktuelle Gesamtzahl der vCPUs Einheiten/Knoten/, die in einem laufenden Cluster verfügbar sind. Wenn eine Clustergrößenänderung angefordert wird, wird diese Metrik aktualisiert, nachdem die neuen Instances hinzugefügt oder aus dem Cluster entfernt wurden. Einheiten: Anzahl
•	

Metrik	Beschreibung
• `CoreUnitsRequested` • `CoreNodesRequested` • `CoreVCPURequested`	Die Zielanzahl von CORE Einheiten/Knoten/ vCPUs in einem Cluster, wie sie durch verwaltete Skalierung bestimmt wird. Einheiten: Anzahl
• `CoreUnitsRunning` • `CoreNodesRunning` • `CoreVCPURunning`	Die aktuelle Anzahl von CORE vCPUs Einheiten/Knoten/, die in einem Cluster ausgeführt werden. Einheiten: Anzahl
• `TaskUnitsRequested` • `TaskNodesRequested` • `TaskVCPURequested`	Die Zielanzahl von TASK Einheiten/Knoten/ vCPUs in einem Cluster, wie sie durch verwaltete Skalierung bestimmt wird. Einheiten: Anzahl
• `TaskUnitsRunning` • `TaskNodesRunning` • `TaskVCPURunning`	Die aktuelle Anzahl von TASK vCPUs Einheiten/Knoten/, die in einem Cluster ausgeführt werden. Einheiten: Anzahl

Die folgenden Metriken geben den Verwendungsstatus von Clustern und Anwendungen an. Diese Metriken sind für alle EMR Amazon-Funktionen verfügbar, werden jedoch in einer höheren Auflösung mit Daten mit einer Granularität von einer Minute veröffentlicht, wenn die verwaltete Skalierung für einen Cluster aktiviert ist. Sie können die folgenden Metriken mit den Clusterkapazitätsmetriken in der vorherigen Tabelle korrelieren, um die Entscheidungen bezüglich der verwalteten Skalierung zu verständlich zu machen.

Metrik	Beschreibung
AppsCompleted	Die Anzahl der eingereichten Anträge, YARN die abgeschlossen wurden. Anwendungsfall: Cluster-Fortschritt überwachen Einheiten: Anzahl
AppsPending	Die Anzahl der bei YARN diesem Unternehmen eingereichten Anträge ist noch nicht abgeschlossen. Anwendungsfall: Cluster-Fortschritt überwachen Einheiten: Anzahl
AppsRunning	Die Anzahl der Bewerbungen, die bei YARN diesem Dienst eingereicht wurden, laufen. Anwendungsfall: Cluster-Fortschritt überwachen Einheiten: Anzahl
ContainerAllocated	Die Anzahl der Ressourcencontainer, die von der zugewiesen wurdenResourceManager. Anwendungsfall: Cluster-Fortschritt überwachen Einheiten: Anzahl
ContainerPending	Anzahl der Container in der Warteschlange, die noch nicht zugeordnet worden sind. Anwendungsfall: Cluster-Fortschritt überwachen Einheiten: Anzahl
ContainerPendingRatio	

Metrik	Beschreibung
	Das Verhältnis von ausstehenden Containern zu zugewiesenen Containern (ContainerPendingRatio = ContainerPending / ContainerAllocated). Wenn ContainerAllocated = 0, dann ContainerPendingRatio =ContainerPending. Der Wert von ContainerPendingRatio steht für eine Zahl, nicht für einen Prozentsatz. Dieser Wert ist zum Skalieren von Cluster-Ressourcen anhand des Zuordnungsverhaltens des Containers hilfreich. Einheiten: Anzahl
`HDFSUtilization`	Der Prozentsatz des aktuell genutzten HDFS Speichers. Anwendungsfall: Cluster-Leistung analysieren Einheiten: Prozent
`IsIdle`	Gibt an, dass ein Cluster keine Arbeiten mehr ausführt, aber unverändert aktiv ist und Kosten verursacht. Der Wert beträgt 1, wenn weder Tasks noch Aufträge ausgeführt werden, andernfalls beträgt der Wert 0. Dieser Wert wird in 5-Minuten-Intervallen geprüft. Wenn der Wert 1 beträgt, bedeutet dies, dass der Cluster zum Zeitpunkt der Prüfung ungenutzt war, aber nicht die gesamten fünf Minuten. Um Fehlalarme zu vermeiden, sollten Sie einen Alarm auslösen, wenn dieser Wert mehrere aufeinander folgende fünfminütige Prüfungen lang 1 beträgt. Sie können zum Beispiel einen Alarm auslösen, wenn dieser Wert 30 Minuten oder länger 1 beträgt. Anwendungsfall: Cluster-Leistung überwachen Einheiten: boolescher Wert

Metrik	Beschreibung
`MemoryAvailableMB`	Verfügbarer zuzuordnender Speicher. Anwendungsfall: Cluster-Fortschritt überwachen Einheiten: Anzahl
`MRActiveNodes`	Die Anzahl der Knoten, auf denen derzeit MapReduce Aufgaben oder Jobs ausgeführt werden. Entspricht einer YARN Metrik`mapred.resourcemanager.NoOf ActiveNodes` . Anwendungsfall: Cluster-Fortschritt überwachen Einheiten: Anzahl
`YARNMemoryAvailabl ePercentage`	Der Prozentsatz des verbleibenden Speichers, der für verfügbar ist YARN (YARNMemoryAvailablePercentage= MemoryAvailable MB/MemoryTotalMB). Dieser Wert ist nützlich für die Skalierung von Clusterressourcen auf der Grundlage der YARN Speichernutzung. Einheiten: Prozent

Grafieren der Metriken für verwaltete Skalierung

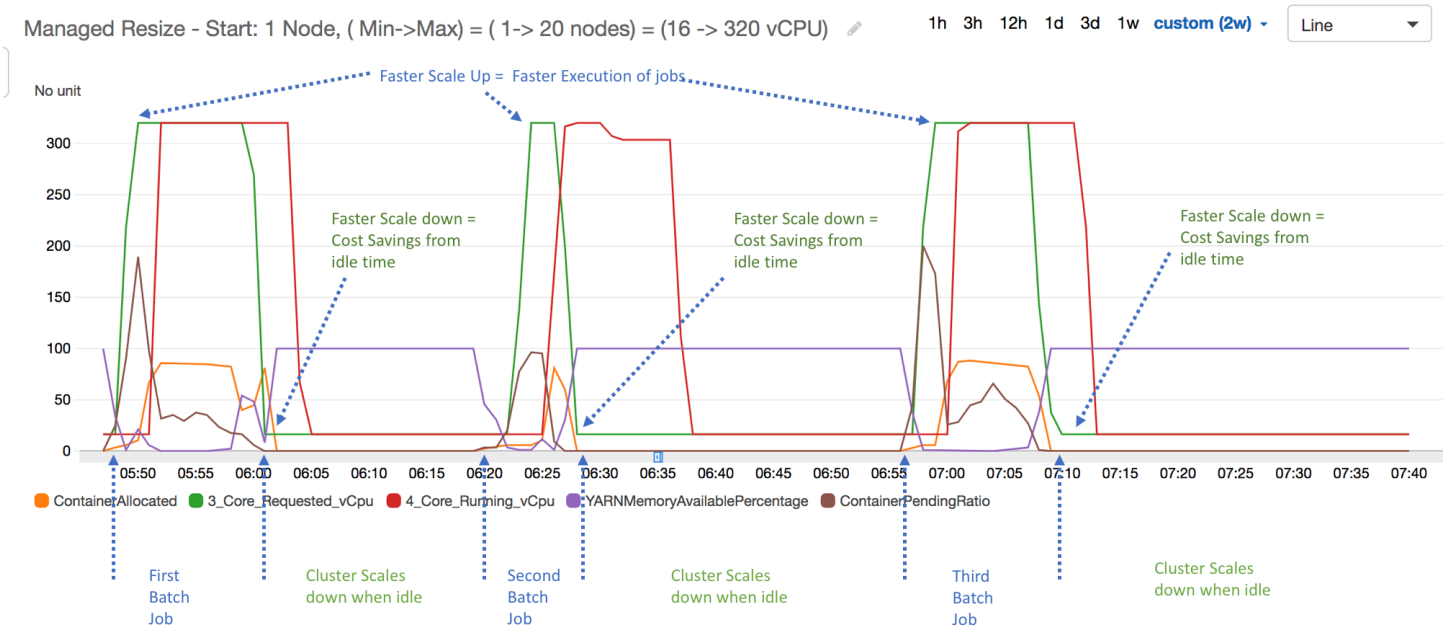
Sie können Metriken grafisch darstellen, um die Workload-Muster Ihres Clusters und die entsprechenden Skalierungsentscheidungen, die von Amazon EMR Managed Scaling getroffen wurden, zu visualisieren, wie die folgenden Schritte zeigen.

Um die Metriken für die verwaltete Skalierung in der CloudWatch Konsole grafisch darzustellen

1. Öffnen Sie die CloudWatchKonsole.
2. Wählen Sie im Navigationsbereich Amazon ausEMR. Sie können die Cluster-Kennung auch nach dem zu überwachenden Cluster durchsuchen.

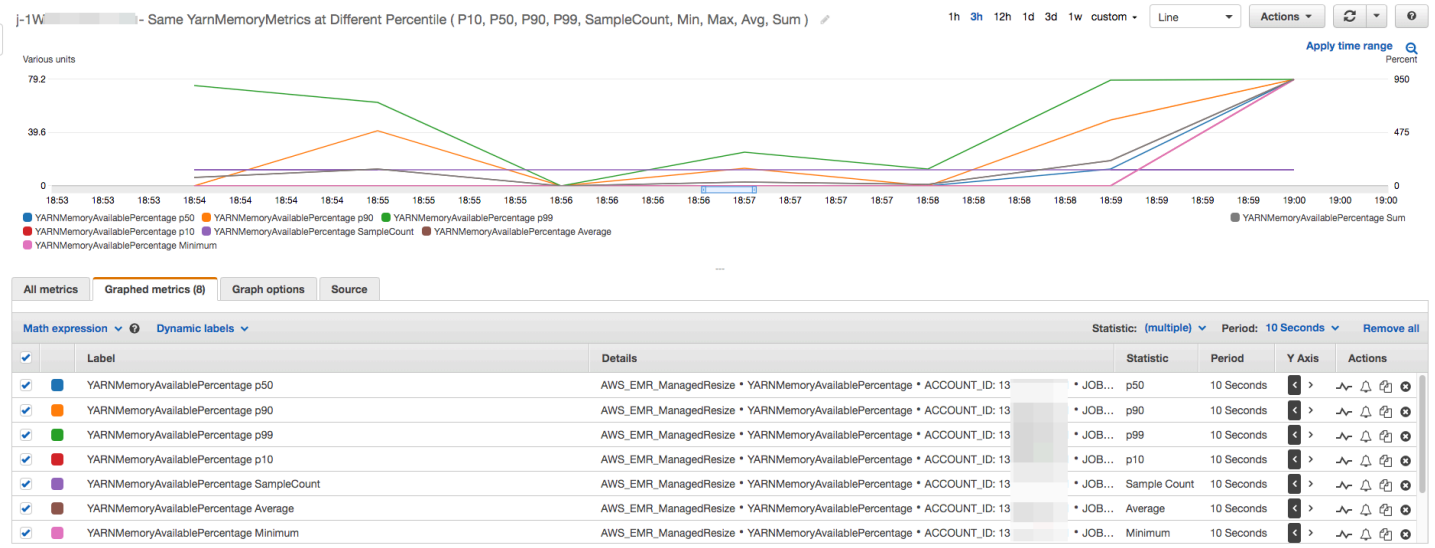
3. Scrollen Sie zur Metrik, die grafisch dargestellt werden soll. Öffnen Sie eine Metrik, um das Diagramm anzuzeigen.
4. Um eine oder mehrere Metriken grafisch darzustellen, aktivieren Sie das Kontrollkästchen neben jeder Metrik.

Das folgende Beispiel veranschaulicht die von Amazon EMR verwaltete Skalierungsaktivität eines Clusters. Das Diagramm zeigt drei automatische Scale-Down-Perioden, die Kosten sparen, wenn eine weniger aktive Workload vorliegt.



Alle Cluster-Kapazitäts- und Nutzungsmetriken werden in Intervallen von einer Minute veröffentlicht. Zusätzliche statistische Informationen sind auch jeweils mit allen einminütigen Daten verknüpft, sodass Sie verschiedene Funktionen wie `Percentiles`, `Min`, `Max`, `Sum`, `Average`, `SampleCount` darstellen können.

Im folgenden Diagramm wird beispielsweise dieselbe `YARNMemoryAvailablePercentage`-Metrik an verschiedenen Perzentilen (P10, P50, P90, P99) zusammen mit `Sum`, `Average`, `Min`, `SampleCount` dargestellt.



Verwenden der automatischen Skalierung mit einer benutzerdefinierten Richtlinie für Instance-Gruppen

Die automatische Skalierung mit einer benutzerdefinierten Richtlinie in EMR Amazon-Versionen 4.0 und höher ermöglicht Ihnen die programmatische Skalierung und Skalierung in Kernknoten und Aufgabenknoten auf der Grundlage einer CloudWatch Metrik und anderer Parameter, die Sie in einer Skalierungsrichtlinie angeben. Automatische Skalierung mit einer benutzerdefinierten Richtlinie ist bei der Instance-Gruppenkonfiguration verfügbar, aber nicht bei der Verwendung von Instance-Flotten. Weitere Informationen zu Instance-Gruppen und Instance-Flotten finden Sie unter Einen Cluster mit Instance-Flotten oder einheitlichen Instance-Gruppen erstellen.

> **Note**
>
> Um die automatische Skalierung mit einer benutzerdefinierten Richtlinienfunktion in Amazon zu verwendenEMR, müssen Sie `true` den `VisibleToAllUsers` Parameter festlegen, wenn Sie einen Cluster erstellen. Weitere Informationen finden Sie unter SetVisibleToAllUsers.

Die Skalierungsrichtlinie ist Teil einer Instance-Gruppen-Konfiguration. Sie können eine Richtlinie während der anfänglichen Konfiguration einer Instance-Gruppe oder durch Ändern einer Instance-Gruppe in einer vorhandenen Cluster-Gruppe festlegen (auch wenn die Instance aktiv ist). Jede Instance-Gruppe in einem Cluster,mit Ausnahme der Primär-Instance-Gruppe, kann ihre eigene Skalierungsrichtlinie haben. Diese besteht aus Regeln zur Hoch- und Herunter-Skalierung. Scale-

Out- und Scale-In-Regeln können unabhängig konfiguriert werden. Jede Regel kann andere Parameter haben.

Sie können Skalierungsrichtlinien mit dem AWS Management Console AWS CLI, dem oder dem Amazon konfigurieren EMRAPI. Wenn Sie Amazon AWS CLI oder Amazon verwenden EMRAPI, geben Sie die Skalierungsrichtlinie im JSON Format an. Wenn Sie mit dem AWS CLI oder Amazon arbeiten EMRAPI, können Sie außerdem benutzerdefinierte CloudWatch Metriken angeben. Benutzerdefinierte Metriken können nicht über die AWS Management Console ausgewählt werden. Wenn Sie eine Skalierungsrichtlinie zum ersten Mal über die Konsole erstellen, wird eine für viele Anwendungen geeignete Standardrichtlinie erstellt. Diese können Sie als Basis für Ihre eigene Richtlinie nutzen. Sie können die Standardregeln löschen oder ändern.

Auch wenn die automatische Skalierung es Ihnen ermöglicht, die EMR Cluster-Kapazität anzupassen on-the-fly, sollten Sie dennoch die grundlegenden Workload-Anforderungen berücksichtigen und Ihre Knoten- und Instance-Gruppenkonfigurationen planen. Weitere Informationen finden Sie unter Richtlinien zur Cluster-Konfiguration.

> **Note**
>
> Bei den meisten Workloads ist die Einrichtung von Scale-In- und Scale-Out-Regeln zur Optimierung der Ressourcenauslastung erstrebenswert. Wenn Sie eine Regel ohne Gegenstück erstellen, müssen Sie die Größe der Instanz nach einer Skalierung möglicherweise manuell anpassen. In diesem Fall richten Sie sozusagen ein "unidirektionales" Auto Scaling in eine Richtung (Scale-Out oder Scale-In) mit einem manuellen Reset ein.

Die IAM Rolle für die automatische Skalierung erstellen

Für die automatische Skalierung in Amazon EMR ist eine IAM Rolle mit Berechtigungen zum Hinzufügen und Beenden von Instances erforderlich, wenn Skalierungsaktivitäten ausgelöst werden. Eine Standardrolle, `EMR_AutoScaling_DefaultRole`, mit der entsprechenden Rollen- und Vertrauensrichtlinie, ist für diesen Zweck verfügbar. Wenn Sie mit dem zum ersten Mal einen Cluster mit einer Skalierungsrichtlinie erstellen AWS Management Console, EMR erstellt Amazon die Standardrolle und fügt die verwaltete Standardrichtlinie für Berechtigungen hinzu`AmazonElasticMapReduceforAutoScalingRole`.

Wenn Sie einen Cluster mit einer automatischen Skalierungsrichtlinie mit dem erstellen AWS CLI, müssen Sie zunächst sicherstellen, dass entweder die IAM Standardrolle vorhanden ist

oder dass Sie über eine benutzerdefinierte IAM Rolle mit einer angehängten Richtlinie verfügen, die die entsprechenden Berechtigungen bereitstellt. Um die Standardrolle zu erstellen, können Sie den Befehl `create-default-roles` ausführen, bevor Sie einen Cluster erstellen. Sie können dann die Option `--auto-scaling-role EMR_AutoScaling_DefaultRole` angeben, wenn Sie einen Cluster erstellen. Alternativ können Sie eine benutzerdefinierte Rolle mit Auto Scaling erstellen und diese angeben, wenn Sie einen Cluster erstellen, zum Beispiel `--auto-scaling-role` *`MyEMRAutoScalingRole`*. Wenn Sie eine benutzerdefinierte automatische Skalierungsrolle für Amazon erstellenEMR, empfehlen wir, dass Sie die Berechtigungsrichtlinien für Ihre benutzerdefinierte Rolle auf der Grundlage der verwalteten Richtlinie festlegen. Weitere Informationen finden Sie unter IAMServicerollen für EMR Amazon-Berechtigungen für AWS Dienste und Ressourcen konfigurieren.

Grundlegendes zu Auto-Scaling-Regeln

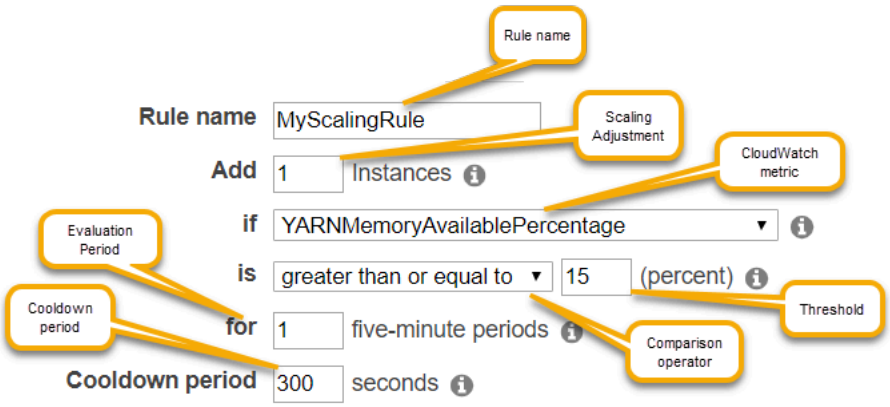
Wenn eine Scale-Out-Regel eine Skalierungsaktivität für eine Instance-Gruppe auslöst, werden EC2 Amazon-Instances gemäß Ihren Regeln zur Instance-Gruppe hinzugefügt. Neue Knoten können von Anwendungen wie Apache Spark, Apache Hive und Presto verwendet werden, sobald die EC2 Amazon-Instance in den `InService` Status wechselt. Sie können außerdem eine Scale-In-Regel erstellen, die Instances beendet und Knoten entfernt. Weitere Informationen zum Lebenszyklus von EC2 Amazon-Instances, die automatisch skalieren, finden Sie unter Auto Scaling Scaling-Lebenszyklus im Amazon EC2 Auto Scaling Scaling-Benutzerhandbuch.

Sie können konfigurieren, wie ein Cluster EC2 Amazon-Instances beendet. Sie können wählen, ob Sie entweder innerhalb der EC2 Amazon-Instance-Stundengrenze für die Abrechnung oder nach Abschluss der Aufgabe kündigen möchten. Diese Einstellung gilt sowohl für die Auto Scaling- als auch für manuelle Größenanpassungen. Weitere Informationen zu dieser Konfiguration finden Sie unter Cluster-Herunterskalierung.

Die folgenden Parameter für eine Regel in einer Richtlinie bestimmen das Auto Scaling-Verhalten.

> **Note**
>
> Die hier aufgeführten Parameter basieren auf dem AWS Management Console für AmazonEMR. Wenn Sie Amazon AWS CLI oder Amazon verwenden EMRAPI, sind zusätzliche erweiterte Konfigurationsoptionen verfügbar. Weitere Informationen zu erweiterten Optionen finden Sie SimpleScalingPolicyConfigurationin der EMRAPIAmazon-Referenz.

- Maximale und minimale Instances-Anzahl. Die Beschränkung Maximum Instances gibt die maximale Anzahl von EC2 Amazon-Instances an, die sich in der Instance-Gruppe befinden können, und gilt für alle Scale-out-Regeln. In ähnlicher Weise gibt die Beschränkung „Minimum Instances“ die Mindestanzahl von EC2 Amazon-Instances an und gilt für alle Scale-In-Regeln.
- Der Rule name (Regelname) muss innerhalb der Richtlinie eindeutig sein.
- Die Skalierungsanpassung, die die Anzahl der EC2 Instances bestimmt, die während der durch die Regel ausgelösten Skalierungsaktivität hinzugefügt (für Scale-Out-Regeln) oder beendet (für Scale-In-Regeln) werden sollen.
- Die CloudWatch Metrik, die im Hinblick auf einen Alarm überwacht wird.
- Ein Vergleichsoperator, der verwendet wird, um die CloudWatch Metrik mit dem Schwellenwert zu vergleichen und eine Auslösebedingung zu bestimmen.
- Ein Bewertungszeitraum in Schritten von fünf Minuten, für den sich die CloudWatch Metrik in einem Triggerzustand befinden muss, bevor die Skalierungsaktivität ausgelöst wird.
- Eine Ruhephase in Sekunden legt fest, wie viel Zeit zwischen einer durch eine Regel ausgelösten Skalierung und dem Start der nächsten Skalierung vergehen muss (unabhängig von der auslösenden Regel). Wenn eine Instanzgruppe eine Skalierungsaktivität abgeschlossen hat und ihren Status nach der Skalierung erreicht hat, bietet die Abklingzeit die Möglichkeit, dass sich die CloudWatch Metriken, die nachfolgende Skalierungsaktivitäten auslösen könnten, stabilisieren. Weitere Informationen finden Sie unter Auto Scaling Scaling-Abklingzeiten im Amazon EC2 Auto Scaling Scaling-Benutzerhandbuch.



Überlegungen und Einschränkungen

- CloudWatch Amazon-Metriken sind entscheidend für den Betrieb der EMR automatischen Skalierung von Amazon. Wir empfehlen Ihnen, die CloudWatch Amazon-Metriken genau zu beobachten, um sicherzustellen, dass keine Daten fehlen. Weitere Informationen darüber, wie Sie

CloudWatch Amazon-Alarme konfigurieren können, um fehlende Messwerte zu erkennen, finden Sie unter CloudWatch Amazon-Alarme verwenden.

- Eine übermäßige Auslastung von EBS Volumes kann zu Problemen mit Managed Scaling führen. Wir empfehlen, die EBS Volume-Nutzung genau zu überwachen, um sicherzustellen, dass EBS das Volumen unter 90% liegt. Informationen zur Angabe zusätzlicher EBS Volumes finden Sie unter Instance-Speicher.
- Bei der automatischen Skalierung mit einer benutzerdefinierten Richtlinie in den EMR Amazon-Versionen 5.18 bis 5.28 kann es zu Skalierungsfehlern kommen, die durch zeitweise fehlende Daten in den Amazon-Metriken verursacht werden. CloudWatch Wir empfehlen Ihnen, die neuesten EMR Amazon-Versionen zu verwenden, um die automatische Skalierung zu verbessern. Sie können sich auch an den AWS Support wenden, um einen Patch zu erhalten, wenn Sie eine EMR Amazon-Version zwischen 5.18 und 5.28 verwenden müssen.

Verwenden Sie die, AWS Management Console um die automatische Skalierung zu konfigurieren

Wenn Sie einen Cluster erstellen, konfigurieren Sie mithilfe der erweiterten Optionen für die Cluster-Konfiguration eine Skalierungsrichtlinie für Instance-Gruppen. Sie können außerdem eine Skalierungsrichtlinie für eine laufende Instance-Gruppe erstellen, indem Sie die Hardware-Einstellungen eines vorhandenen Clusters bearbeiten.

1. Navigieren Sie zur neuen EMR Amazon-Konsole und wählen Sie in der Seitennavigation die Option Zur alten Konsole wechseln aus. Weitere Informationen darüber, was Sie erwartet, wenn Sie zur alten Konsole wechseln, finden Sie unter Verwenden der alten Konsole.
2. Wenn Sie einen Cluster erstellen, wählen Sie in der EMR Amazon-Konsole Cluster erstellen, wählen Sie Gehe zu erweiterten Optionen, wählen Sie Optionen für Schritt 1: Software und Schritte aus und fahren Sie dann mit Schritt 2: Hardwarekonfiguration fort.

   – oder –

   Wenn Sie eine Instance-Gruppe in einem ausgeführten Cluster ändern, wählen Sie den Cluster aus der Cluster-Liste aus und erweitern Sie dann den Hardware-Abschnitt.
3. Wählen Sie im Abschnitt Clusterskalierung und Bereitstellungoption die Option Clusterskalierung aktivieren aus. Wählen Sie dann Benutzerdefinierte Richtlinie für automatische Skalierung erstellen aus.

Klicken Sie in der Tabelle Benutzerdefinierte Richtlinien für automatische Skalierung auf das Stiftsymbol in der Zeile der Instance-Gruppe, die Sie konfigurieren möchten. Die Seite Auto-Scaling-Regeln wird geöffnet.

4. Geben Sie die Maximum instances (maximale Instances)-Anzahl ein, die die Instance-Gruppe nach dem Scale-Out enthalten soll. Geben Sie die Minimum instances (minimale Instances)-Anzahl ein, die die Instance-Gruppe nach dem Scale-In enthalten soll.
5. Klicken Sie auf den Stift, um Regelparameter zu bearbeiten. Klicken Sie auf das X, um eine Regel aus der Richtlinie zu entfernen, und klicken Sie auf Add rule (Regel hinzufügen), um weitere Regeln hinzuzufügen.
6. Wählen Sie die weiter oben in diesem Thema beschriebenen Regelparameter aus. Eine Beschreibung der verfügbaren CloudWatch Metriken für Amazon EMR finden Sie unter EMRAmazon-Metriken und -Dimensionen im CloudWatch Amazon-Benutzerhandbuch.

Verwenden von AWS CLI , um die automatische Skalierung zu konfigurieren

Sie können AWS CLI Befehle für Amazon verwendenEMR, um die automatische Skalierung zu konfigurieren, wenn Sie einen Cluster und eine Instanzgruppe erstellen. Sie können eine Kurzsyntax verwenden und die JSON Konfiguration direkt in den entsprechenden Befehlen angeben, oder Sie können auf eine Datei verweisen, die die Konfiguration enthält. JSON Sie können außerdem eine Auto Scaling-Richtlinie auf eine vorhandene Instance-Gruppe anwenden und eine angewendete Auto Scaling-Richtlinie entfernen. Darüber hinaus können Sie Details einer Skalierungsrichtlinien-Konfiguration aus einem aktuell ausgeführten Cluster abrufen.

> **Important**
>
> Wenn Sie einen Cluster mit einer automatischen Skalierungsrichtlinie erstellen, müssen Sie den `--auto-scaling-role` *`MyAutoScalingRole`* Befehl verwenden, um die IAM Rolle für die automatische Skalierung anzugeben. Die Standard-Rolle ist *`EMR_AutoScaling_DefaultRole`* und kann mit dem Befehl `create-default-roles` erstellt werden. Die Rolle kann nur hinzugefügt werden, wenn der Cluster erstellt wird. Sie kann nicht zu einem vorhandenen Cluster hinzugefügt werden.

Eine ausführliche Beschreibung der Parameter, die bei der Konfiguration einer automatischen Skalierungsrichtlinie verfügbar sind, finden Sie PutAutoScalingPolicyin Amazon EMR API Reference.

Erstellen eines Clusters mit einer angewendeten Auto-Scaling-Richtlinie in einer Instance-Gruppe

Sie können eine Auto Scaling-Konfiguration innerhalb der Option `--instance-groups` des Befehls `aws emr create-cluster` festlegen. Das folgende Beispiel demonstriert einen create-Cluster-Befehl, in dem eine Auto Scaling-Richtlinie für die Core-Instance-Gruppe enthalten ist. Der Befehl erstellt eine Skalierungskonfiguration, die der Standard-Scale-Out-Richtlinie entspricht, die angezeigt wird, wenn Sie eine automatische Skalierungsrichtlinie mit dem AWS Management Console für Amazon erstellen. EMR Aus Gründen der Übersichtlichkeit verzichten wir auf die Abbildung einer Scale-In-Richtlinie. Das Erstellen einer Scale-Out-Regel ohne Verringern der Scale-In-Regel wird nicht empfohlen.

```
aws emr create-cluster --release-label emr-5.2.0 --service-role
 EMR_DefaultRole --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole
 --auto-scaling-role EMR_AutoScaling_DefaultRole  --instance-groups
 Name=MyMasterIG,InstanceGroupType=MASTER,InstanceType=m5.xlarge,InstanceCount=1
 'Name=MyCoreIG,InstanceGroupType=CORE,InstanceType=m5.xlarge,InstanceCount=2,AutoScalingPolicy
scale-out,Description=Replicates the default scale-out rule in the
 console.,Action={SimpleScalingPolicyConfiguration={AdjustmentType=CHANGE_IN_CAPACITY,ScalingAd
ElasticMapReduce,Period=300,Statistic=AVERAGE,Threshold=15,Unit=PERCENT,Dimensions=[{Key=JobFlo
```

Der folgende Befehl veranschaulicht die Verwendung der Befehlszeile zur Angabe einer Auto-Scaling-Richtliniendefinition im Rahmen einer Instance-Gruppe-Konfigurationsdatei mit dem Namen *instancegroupconfig.json*.

```
aws emr create-cluster --release-label emr-5.2.0 --service-role EMR_DefaultRole --ec2-
attributes InstanceProfile=EMR_EC2_DefaultRole --instance-groups file://your/path/to/
instancegroupconfig.json --auto-scaling-role EMR_AutoScaling_DefaultRole
```

Der Inhalt der Konfigurationsdatei sieht wie folgt aus:

```
[
{
  "InstanceCount": 1,
  "Name": "MyMasterIG",
  "InstanceGroupType": "MASTER",
  "InstanceType": "m5.xlarge"
},
{
  "InstanceCount": 2,
```

```
  "Name": "MyCoreIG",
  "InstanceGroupType": "CORE",
  "InstanceType": "m5.xlarge",
  "AutoScalingPolicy":
    {
     "Constraints":
      {
       "MinCapacity": 2,
       "MaxCapacity": 10
      },
     "Rules":
     [
      {
       "Name": "Default-scale-out",
       "Description": "Replicates the default scale-out rule in the console for YARN
memory.",
       "Action":{
        "SimpleScalingPolicyConfiguration":{
          "AdjustmentType": "CHANGE_IN_CAPACITY",
          "ScalingAdjustment": 1,
          "CoolDown": 300
        }
       },
       "Trigger":{
        "CloudWatchAlarmDefinition":{
          "ComparisonOperator": "LESS_THAN",
          "EvaluationPeriods": 1,
          "MetricName": "YARNMemoryAvailablePercentage",
          "Namespace": "AWS/ElasticMapReduce",
          "Period": 300,
          "Threshold": 15,
          "Statistic": "AVERAGE",
          "Unit": "PERCENT",
          "Dimensions":[
             {
               "Key" : "JobFlowId",
               "Value" : "${emr.clusterId}"
             }
          ]
        }
       }
      }
     ]
   }
```

```
}
]
```

Hinzufügen einer Instance-Gruppe mit einer Auto-Scaling-Richtlinie zu einem Cluster

Sie können genauso wie bei `--instance-groups` mithilfe der `add-instance-groups`-Option des `create-cluster`-Befehls eine Skalierungsrichtlinien-Konfiguration festlegen. Im folgenden Beispiel wird ein Verweis auf eine JSON Datei mit der Instanzgruppenkonfiguration verwendet. *`instancegroupconfig.json`*

```
aws emr add-instance-groups --cluster-id j-1EKZ3TYEVF1S2 --instance-groups file://your/
path/to/instancegroupconfig.json
```

Anwenden einer Auto-Scaling-Richtlinie auf eine vorhandene Instance-Gruppe oder Ändern einer angewandten Richtlinie

Verwenden Sie den `aws emr put-auto-scaling-policy`-Befehl, um eine Auto Scaling-Richtlinie auf eine vorhandene Instance-Gruppe anzuwenden. Die Instanzgruppe muss Teil eines Clusters sein, der die automatische IAM Skalierungsrolle verwendet. Im folgenden Beispiel wird ein Verweis auf eine JSON Datei verwendet*`autoscaleconfig.json`*, die die Konfiguration der automatischen Skalierungsrichtlinie spezifiziert.

```
aws emr put-auto-scaling-policy --cluster-id j-1EKZ3TYEVF1S2 --instance-group-
id ig-3PLUZBA6WLS07 --auto-scaling-policy file://your/path/to/autoscaleconfig.json
```

Der Inhalt der `autoscaleconfig.json`-Datei, die die gleiche Scale-Out-Regel wie im vorherigen Beispiel definiert, ist unten dargestellt.

```
{
          "Constraints": {
                  "MaxCapacity": 10,
                  "MinCapacity": 2
          },
          "Rules": [{
                  "Action": {
                          "SimpleScalingPolicyConfiguration": {
                                  "AdjustmentType": "CHANGE_IN_CAPACITY",
                                  "CoolDown": 300,
                                  "ScalingAdjustment": 1
                          }
                  },
```

```
                "Description": "Replicates the default scale-out rule in the console
 for YARN memory",
                "Name": "Default-scale-out",
                "Trigger": {
                        "CloudWatchAlarmDefinition": {
                                "ComparisonOperator": "LESS_THAN",
                                "Dimensions": [{
                                        "Key": "JobFlowId",
                                        "Value": "${emr.clusterID}"
                                }],
                                "EvaluationPeriods": 1,
                                "MetricName": "YARNMemoryAvailablePercentage",
                                "Namespace": "AWS/ElasticMapReduce",
                                "Period": 300,
                                "Statistic": "AVERAGE",
                                "Threshold": 15,
                                "Unit": "PERCENT"
                        }
                }
        }]
 }
```

Entfernen einer Auto-Scaling-Richtlinie aus einer Instance-Gruppe

```
aws emr remove-auto-scaling-policy --cluster-id j-1EKZ3TYEVF1S2 --instance-group-
id ig-3PLUZBA6WLS07
```

Abrufen einer Auto-Scaling-Richtlinienkonfiguration

Der `describe-cluster` Befehl ruft die Richtlinienkonfiguration im InstanceGroup Block ab. Mit dem folgenden Befehl wird beispielsweise die Konfiguration für den Cluster mit der Cluster-ID `j-1CW0HP4PI30VJ` abgerufen.

```
aws emr describe-cluster --cluster-id j-1CW0HP4PI30VJ
```

Der Befehl generiert die folgende Beispielausgabe:

```
{
    "Cluster": {
```

```
        "Configurations": [],
        "Id": "j-1CWOHP4PI30VJ",
        "NormalizedInstanceHours": 48,
        "Name": "Auto Scaling Cluster",
        "ReleaseLabel": "emr-5.2.0",
        "ServiceRole": "EMR_DefaultRole",
        "AutoTerminate": false,
        "TerminationProtected": true,
        "MasterPublicDnsName": "ec2-54-167-31-38.compute-1.amazonaws.com",
        "LogUri": "s3n://aws-logs-232939870606-us-east-1/elasticmapreduce/",
        "Ec2InstanceAttributes": {
            "Ec2KeyName": "performance",
            "AdditionalMasterSecurityGroups": [],
            "AdditionalSlaveSecurityGroups": [],
            "EmrManagedSlaveSecurityGroup": "sg-09fc9362",
            "Ec2AvailabilityZone": "us-east-1d",
            "EmrManagedMasterSecurityGroup": "sg-0bfc9360",
            "IamInstanceProfile": "EMR_EC2_DefaultRole"
        },
        "Applications": [
            {
                "Name": "Hadoop",
                "Version": "2.7.3"
            }
        ],
        "InstanceGroups": [
            {
                "AutoScalingPolicy": {
                    "Status": {
                        "State": "ATTACHED",
                        "StateChangeReason": {
                            "Message": ""
                        }
                    },
                    "Constraints": {
                        "MaxCapacity": 10,
                        "MinCapacity": 2
                    },
                    "Rules": [
                        {
                            "Name": "Default-scale-out",
                            "Trigger": {
                                "CloudWatchAlarmDefinition": {
                                    "MetricName": "YARNMemoryAvailablePercentage",
```

```
                    "Unit": "PERCENT",
                    "Namespace": "AWS/ElasticMapReduce",
                    "Threshold": 15,
                    "Dimensions": [
                        {
                            "Key": "JobFlowId",
                            "Value": "j-1CWOHP4PI30VJ"
                        }
                    ],
                    "EvaluationPeriods": 1,
                    "Period": 300,
                    "ComparisonOperator": "LESS_THAN",
                    "Statistic": "AVERAGE"
                }
            },
            "Description": "",
            "Action": {
                "SimpleScalingPolicyConfiguration": {
                    "CoolDown": 300,
                    "AdjustmentType": "CHANGE_IN_CAPACITY",
                    "ScalingAdjustment": 1
                }
            }
        },
        {
            "Name": "Default-scale-in",
            "Trigger": {
                "CloudWatchAlarmDefinition": {
                    "MetricName": "YARNMemoryAvailablePercentage",
                    "Unit": "PERCENT",
                    "Namespace": "AWS/ElasticMapReduce",
                    "Threshold": 75,
                    "Dimensions": [
                        {
                            "Key": "JobFlowId",
                            "Value": "j-1CWOHP4PI30VJ"
                        }
                    ],
                    "EvaluationPeriods": 1,
                    "Period": 300,
                    "ComparisonOperator": "GREATER_THAN",
                    "Statistic": "AVERAGE"
                }
            },
```

```
                        "Description": "",
                        "Action": {
                            "SimpleScalingPolicyConfiguration": {
                                "CoolDown": 300,
                                "AdjustmentType": "CHANGE_IN_CAPACITY",
                                "ScalingAdjustment": -1
                            }
                        }
                    }
                ]
            },
            "Configurations": [],
            "InstanceType": "m5.xlarge",
            "Market": "ON_DEMAND",
            "Name": "Core - 2",
            "ShrinkPolicy": {},
            "Status": {
                "Timeline": {
                    "CreationDateTime": 1479413437.342,
                    "ReadyDateTime": 1479413864.615
                },
                "State": "RUNNING",
                "StateChangeReason": {
                    "Message": ""
                }
            },
            "RunningInstanceCount": 2,
            "Id": "ig-3M16XBE8C3PH1",
            "InstanceGroupType": "CORE",
            "RequestedInstanceCount": 2,
            "EbsBlockDevices": []
        },
        {
            "Configurations": [],
            "Id": "ig-OP62I28NSE8M",
            "InstanceGroupType": "MASTER",
            "InstanceType": "m5.xlarge",
            "Market": "ON_DEMAND",
            "Name": "Master - 1",
            "ShrinkPolicy": {},
            "EbsBlockDevices": [],
            "RequestedInstanceCount": 1,
            "Status": {
                "Timeline": {
```

```
                    "CreationDateTime": 1479413437.342,
                    "ReadyDateTime": 1479413752.088
                },
                "State": "RUNNING",
                "StateChangeReason": {
                    "Message": ""
                }
            },
            "RunningInstanceCount": 1
        }
    ],
    "AutoScalingRole": "EMR_AutoScaling_DefaultRole",
    "Tags": [],
    "BootstrapActions": [],
    "Status": {
        "Timeline": {
            "CreationDateTime": 1479413437.339,
            "ReadyDateTime": 1479413863.666
        },
        "State": "WAITING",
        "StateChangeReason": {
            "Message": "Cluster ready after last step completed."
        }
    }
  }
}
```

Manuelle Größenanpassung eines aktiven Clusters

Sie können Instances zu Kern- und Task-Instance-Gruppen und Instance-Flotten in einem laufenden Cluster mit dem AWS Management Console AWS CLI, oder dem Amazon EMR API hinzufügen und daraus entfernen. Wenn ein Cluster Instance-Gruppen verwendet, müssen Sie die Anzahl der Instances explizit ändern. Wenn Ihr Cluster Instance-Flotten verwendet, können Sie die Zieleinheiten für On-Demand-Instances und Spot-Instances ändern. Die Instance-Flotte fügt anschließend Instances hinzu bzw. entfernt diese, um dem neuen Ziel zu entsprechen. Weitere Informationen finden Sie unter Instance-Flotten-Optionen. Anwendungen können neu bereitgestellte EC2 Amazon-Instances zum Hosten von Knoten verwenden, sobald die Instances verfügbar sind. Wenn Instances entfernt werden, EMR fährt Amazon Aufgaben so herunter, dass Jobs nicht unterbrochen werden und der Schutz vor Datenverlust gewährleistet ist. Weitere Informationen finden Sie unter Beendigung bei Aufgaben-Abschluss.

Die Größe eines Clusters mit der Konsole anpassen

Sie können die EMR Amazon-Konsole verwenden, um die Größe eines laufenden Clusters zu ändern.

Console

So ändern Sie die Anzahl der Instances für einen vorhandenen Cluster mithilfe der neuen Konsole

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und dann den Cluster aus, den Sie aktualisieren möchten. Der Cluster muss ausgeführt werden. Sie können die Größe eines bereitgestellten oder beendeten Clusters nicht ändern.
3. Sehen Sie sich auf der Cluster-Detailseite auf der Registerkarte Instances den Bereich Instance-Gruppen an.
4. Um die Größe einer vorhandenen Instance-Gruppe zu ändern, wählen Sie das Optionsfeld neben der Core- oder Aufgaben-Instance-Gruppe aus, deren Größe Sie ändern möchten, und wählen Sie dann Größe der Instance-Gruppe ändern. Geben Sie die neue Anzahl der Instances für die Instance-Gruppe an und wählen Sie anschließend Größe ändern aus.

   > **Note**
   >
   > Wenn Sie sich dafür entscheiden, die Größe einer laufenden Instance-Gruppe zu reduzieren, wählt Amazon intelligent die Instances aus, die aus der Gruppe entfernt EMR werden sollen, um den Datenverlust zu minimieren. Für eine genauere Steuerung Ihrer Größenänderungsaktion können Sie die ID für die Instance-Gruppe auswählen, die Instances auswählen, die Sie entfernen möchten, und dann die Option Terminate verwenden. Weitere Informationen zum intelligenten Herunterskalierungs-Verhalten finden Sie unter Cluster-Herunterskalierung.

5. Wenn Sie die Größenänderung abbrechen möchten, können Sie das Optionsfeld für eine Instance-Gruppe mit dem Status Größenänderung beenden auswählen und dann in der Liste der Aktionen die Option Größenänderung beenden auswählen.
6. Um Ihrem Cluster als Reaktion auf den steigenden Workload eine oder mehrere Aufgaben-Instance-Gruppen hinzuzufügen, wählen Sie Aufgaben-Instance-Gruppe hinzufügen aus der Liste der Aktionen aus. Wählen Sie den EC2 Amazon-Instance-Typ, geben Sie die Anzahl

der Instances für die Aufgabengruppe ein und wählen Sie dann Task-Instance-Gruppe hinzufügen, um zum Bereich Instance-Gruppen für Ihren Cluster zurückzukehren.

Wenn Sie die Anzahl der Knoten ändern, wird der Status der Instance-Gruppe aktualisiert. Wenn die gewünschte Änderung abgeschlossen ist, ändert sich der Status zu Running (Wird ausgeführt).

Ändern Sie die Größe eines Clusters mit AWS CLI

Sie können den verwenden AWS CLI , um die Größe eines laufenden Clusters zu ändern. Sie können die Anzahl der Aufgabenknoten erhöhen oder verringern. Sie können außerdem die Anzahl der Core-Knoten in einem ausgeführten Cluster erhöhen oder verringern. Es ist auch möglich, eine Instanz in der Core-Instanzgruppe mit dem AWS CLI oder dem API herunterzufahren. Dies sollte mit Vorsicht erfolgen. Beim Beenden einer Instance in der Core-Instance-Gruppe besteht das Risiko eines Datenverlustes. Die Instance wird zudem nicht automatisch ersetzt.

Zusätzlich zur Größenanpassung der Core- und Aufgaben-Gruppen können Sie mithilfe der AWS CLI auch eine oder mehrere Aufgaben-Instance-Gruppen zu einem ausgeführten Cluster hinzufügen.

Um die Größe eines Clusters zu ändern, indem Sie die Anzahl der Instanzen mit dem AWS CLI

Sie können der Kerngruppe oder Aufgabengruppe Instanzen hinzufügen und mit dem AWS CLI `modify-instance-groups` Unterbefehl mit dem Parameter Instanzen aus der Aufgabengruppe entfernen. `InstanceCount` Erhöhen Sie `InstanceCount`, um Instances zu Core- oder Task-Gruppen hinzuzufügen. Reduzieren Sie `InstanceCount`, um die Anzahl der Instances in der Gruppe zu verringern. Die Reduzierung der Anzahl der Instances einer Task-Gruppe auf null entfernt zwar alle Instances, nicht jedoch die Instance-Gruppe.

- Um die Anzahl der Instanzen in der Task-Instanzgruppe von 3 auf 4 zu erhöhen, geben Sie den folgenden Befehl ein und ersetzen *`ig-31JXXXXXXBT0`* mit der Instanzgruppen-ID.

  ```
  aws emr modify-instance-groups --instance-groups
   InstanceGroupId=ig-31JXXXXXXBT0,InstanceCount=4
  ```

  Verwenden Sie den Unterbefehl `InstanceGroupId`, um die `describe-cluster` abzurufen. Die Ausgabe ist ein JSON Objekt mit dem Namen`Cluster`, das die ID jeder Instanzgruppe enthält. Um diesen Befehl verwenden zu können, benötigen Sie die Cluster-ID (diese können Sie über den `aws emr list-clusters`-Befehl oder die Konsole abrufen). Um die Instanzgruppen-ID abzurufen, geben Sie den folgenden Befehl ein und ersetzen Sie *`j-2AXXXXXXGAPLF`* mit der Cluster-ID.

```
aws emr describe-cluster --cluster-id j-2AXXXXXXGAPLF
```

Mit dem AWS CLI können Sie auch eine Instanz in der Core-Instanzgruppe mit dem `--modify-instance-groups` Unterbefehl beenden.

> **Warning**
>
> Die Angabe von `EC2InstanceIdsToTerminate` muss mit Vorsicht erfolgen. Instances werden sofort beendet, unabhängig vom Status der Anwendungen, die auf ihnen ausgeführt werden, und die Instance wird nicht automatisch ersetzt. Dies gilt unabhängig von der Konfiguration vom Scale down behavior (Abwärtsskalierungsverhalten) für den Cluster. Wenn eine Instance auf diese Weise beendet wird, besteht das Risiko von Datenverlusten und unvorhersehbarem Clusterverhalten.

Um eine bestimmte Instanz zu beenden, benötigen Sie die Instanzgruppen-ID (vom `aws emr describe-cluster --cluster-id` Unterbefehl zurückgegeben) und die Instanz-ID (vom `aws emr list-instances --cluster-id` Unterbefehl zurückgegeben). Geben Sie den folgenden Befehl ein: replace *ig-6RXXXXXX07SA* mit der Instanzgruppen-ID und ersetze *i-f9XXXXf2* mit der Instanz-ID.

```
aws emr modify-instance-groups --instance-groups
 InstanceGroupId=ig-6RXXXXXX07SA,EC2InstanceIdsToTerminate=i-f9XXXXf2
```

Weitere Informationen zur Verwendung von EMR Amazon-Befehlen in der AWS CLI finden Sie unterhttps://docs.aws.amazon.com/cli/latest/reference/emr.

Um die Größe eines Clusters zu ändern, indem Sie Task-Instance-Gruppen hinzufügen, verwenden Sie AWS CLI

Mit dem AWS CLI können Sie einem Cluster 1—48 Task-Instance-Gruppen mit dem `--add-instance-groups` Unterbefehl hinzufügen. Aufgaben-Instances-Gruppen können nur zu einem Cluster mit einer Primär-Instance-Gruppe und einer Core-Instance-Gruppe hinzugefügt werden. Wenn Sie den verwenden AWS CLI, können Sie bei jeder Verwendung des Unterbefehls bis zu fünf Task-Instanzgruppen hinzufügen. `--add-instance-groups`

1. Um einem Cluster eine einzelne Task-Instanzgruppe hinzuzufügen, geben Sie den folgenden Befehl ein und ersetzen Sie *j-JXBXXXXXX37R* mit der Cluster-ID.

```
aws emr add-instance-groups --cluster-id j-JXBXXXXXX37R --instance-groups
 InstanceCount=6,InstanceGroupType=task,InstanceType=m5.xlarge
```

2. Um einem Cluster mehrere Task-Instance-Gruppen hinzuzufügen, geben Sie den folgenden Befehl ein und ersetzen Sie *j-JXBXXXXXX37R* mit der Cluster-ID. Sie können bis zu fünf Task-Instance-Gruppen pro Befehl hinzufügen.

```
aws emr add-instance-groups --cluster-id j-JXBXXXXXX37R --instance-
groups InstanceCount=6,InstanceGroupType=task,InstanceType=m5.xlarge
 InstanceCount=10,InstanceGroupType=task,InstanceType=m5.xlarge
```

   Weitere Informationen zur Verwendung von EMR Amazon-Befehlen in der AWS CLI finden Sie unterhttps://docs.aws.amazon.com/cli/latest/reference/emr.

Unterbrechen einer Größenänderung

Wenn Sie Amazon EMR Version 4.1.0 oder höher verwenden, können Sie eine Größenänderung während eines bestehenden Größenänderungsvorgangs vornehmen. Sie können außerdem eine zuvor gesendete Anfrage zur Größenanpassung stoppen oder eine neue Anfrage senden, um eine frühere Anfrage zu überschreiben, ohne auf deren Abschluss zu warten. Sie können eine bestehende Größenänderung auch von der Konsole aus oder mit dem `ModifyInstanceGroups` API Aufruf beenden, wobei die aktuelle Anzahl als Zielanzahl des Clusters verwendet wird.

Die folgende Abbildung zeigt eine Task-Instance-Gruppe, deren Größe gerade geändert wird und bei der die Größenänderung über Stop (Stopp) beendet werden kann.

Um eine Größenänderung zu unterbrechen mit dem AWS CLI

Sie können den verwenden AWS CLI , um eine Größenänderung mit dem Unterbefehl zu beenden. `modify-instance-groups` Nehmen wir an, Sie haben sechs Instances in der Instance-Gruppe und Sie möchten diese auf 10 erhöhen. Später entscheiden Sie, dass Sie diese Anforderung stornieren möchten:

- Die ursprüngliche Anforderung:

```
aws emr modify-instance-groups --instance-groups
 InstanceGroupId=ig-myInstanceGroupId,InstanceCount=10
```

Die zweite Anforderung zum Beenden der ersten Anforderung:

```
aws emr modify-instance-groups --instance-groups
 InstanceGroupId=ig-myInstanceGroupId,InstanceCount=6
```

> **Note**
>
> Da dieser Prozess asynchron ist, kann es vorkommen, dass sich die Anzahl der Instanzen im Vergleich zu früheren API Anfragen ändert, bevor nachfolgende Anfragen berücksichtigt werden. Bei einer Verkleinerung kann es sein, dass auf den Knoten noch Aufgaben ausgeführt werden. In diesem Fall wird die Instance-Gruppe nicht verkleinert, bis die Knoten ihre Arbeit abgeschlossen haben.

Suspendierter Zustand

Eine Instance-Gruppe geht in einen suspendierten Zustand über, wenn sie beim Versuch, die neuen Clusterknoten zu starten, auf zu viele Fehler stößt. Wenn beispielsweise neue Knoten bei der Ausführung von Bootstrap-Aktionen ausfallen, wechselt die Instanzgruppe in einen SUSPENDEDStatus, anstatt kontinuierlich neue Knoten bereitzustellen. Nachdem Sie das entsprechende Problem behoben haben, setzen Sie die Anzahl der gewünschten Knoten in der Instance-Gruppe des Clusters zurück. Anschließend fährt die Instance-Gruppe mit der Reservierung von Knoten fort. Durch das Ändern einer Instanzgruppe wird Amazon angewiesen, erneut EMR zu versuchen, Knoten bereitzustellen. Nicht ausgeführte Knoten werden neu gestartet oder beendet.

In der AWS CLI gibt der `list-instances` Unterbefehl alle Instances und deren Status zurück, ebenso wie der `describe-cluster` Unterbefehl. Wenn Amazon einen Fehler in einer Instance-Gruppe EMR feststellt, ändert es den Status der Gruppe auf`SUSPENDED`.

Um einen Cluster in einem SUSPENDED Zustand mit dem zurückzusetzen AWS CLI

Geben Sie den `describe-cluster`-Unterbefehl mit dem Parameter `--cluster-id` ein, um den Status der Instances in Ihrem Cluster anzuzeigen.

- Um Informationen zu allen Instances und Instanzgruppen in einem Cluster anzuzeigen, geben Sie den folgenden Befehl ein und ersetzen *j-3KVXXXXXXY7UG* mit der Cluster-ID.

```
aws emr describe-cluster --cluster-id j-3KVXXXXXXY7UG
```

Die Ausgabe zeigt Informationen über Ihre Instance-Gruppen und den Status der Instances an:

```
{
    "Cluster": {
        "Status": {
            "Timeline": {
                "ReadyDateTime": 1413187781.245,
                "CreationDateTime": 1413187405.356
            },
            "State": "WAITING",
            "StateChangeReason": {
                "Message": "Waiting after step completed"
            }
        },
        "Ec2InstanceAttributes": {
            "Ec2AvailabilityZone": "us-west-2b"
        },
        "Name": "Development Cluster",
        "Tags": [],
        "TerminationProtected": false,
        "RunningAmiVersion": "3.2.1",
        "NormalizedInstanceHours": 16,
        "InstanceGroups": [
            {
                "RequestedInstanceCount": 1,
                "Status": {
                    "Timeline": {
                        "ReadyDateTime": 1413187775.749,
                        "CreationDateTime": 1413187405.357
                    },
                    "State": "RUNNING",
                    "StateChangeReason": {
                        "Message": ""
                    }
                },
                "Name": "MASTER",
                "InstanceGroupType": "MASTER",
```

```
                "InstanceType": "m5.xlarge",
                "Id": "ig-3ETXXXXXXFYV8",
                "Market": "ON_DEMAND",
                "RunningInstanceCount": 1
            },
            {
                "RequestedInstanceCount": 1,
                "Status": {
                    "Timeline": {
                        "ReadyDateTime": 1413187781.301,
                        "CreationDateTime": 1413187405.357
                    },
                    "State": "RUNNING",
                    "StateChangeReason": {
                        "Message": ""
                    }
                },
                "Name": "CORE",
                "InstanceGroupType": "CORE",
                "InstanceType": "m5.xlarge",
                "Id": "ig-3SUXXXXXXQ9ZM",
                "Market": "ON_DEMAND",
                "RunningInstanceCount": 1
            }
...
}
```

Um Informationen zu einer bestimmten Instance-Gruppe anzuzeigen, geben Sie den Unterbefehl `list-instances` mit den Parametern `--cluster-id` und `--instance-group-types` ein. Sie können Informationen für Primär-, Core- oder Aufgaben-Gruppen anzeigen.

```
aws emr list-instances --cluster-id j-3KVXXXXXXY7UG --instance-group-types "CORE"
```

Verwenden Sie den Unterbefehl `modify-instance-groups` mit dem Parameter `--instance-groups`, um einen Cluster mit dem SUSPENDED-Status zurückzusetzen. Die Instance-Gruppen-ID wird vom Unterbefehl `describe-cluster` zurückgegeben.

```
aws emr modify-instance-groups --instance-groups
 InstanceGroupId=ig-3SUXXXXXXQ9ZM,InstanceCount=3
```

Überlegungen zur Reduzierung der Clustergröße

Wenn Sie sich dafür entscheiden, die Größe eines laufenden Clusters zu reduzieren, sollten Sie das folgende EMR Verhalten und die folgenden Best Practices von Amazon berücksichtigen:

- Um die Auswirkungen auf laufende Jobs zu reduzieren, wählt Amazon EMR intelligent die zu entfernenden Instances aus. Weitere Informationen zum Verhalten beim Herunterskalieren von Clustern finden Sie Beendigung bei Aufgaben-Abschluss im Amazon EMR Management Guide.
- Wenn Sie die Größe eines Clusters verkleinern, EMR kopiert Amazon die Daten aus den Instances, die es entfernt hat, in die verbleibenden Instances. Stellen Sie sicher, dass in den Instances, die in der Gruppe verbleiben, ausreichend Speicherkapazität für diese Daten vorhanden ist.
- Amazon EMR versucht, Instances in HDFS der Gruppe außer Betrieb zu nehmen. Bevor Sie die Größe eines Clusters reduzieren, empfehlen wir, die HDFS Schreib-E/A zu minimieren.
- Wenn Sie die Größe eines Clusters am genauesten steuern möchten, können Sie den Cluster in der Konsole anzeigen und zur Registerkarte Instances wechseln. Wählen Sie die ID für die Instance-Gruppe aus, deren Größe Sie ändern möchten. Verwenden Sie dann die Option Terminate für die spezifischen Instanceen, die Sie entfernen möchten.

Konfigurieren Sie Timeouts für die Bereitstellung von Kapazität

Wenn Sie Instanceflotten verwenden, können Sie Timeouts für die Bereitstellung konfigurieren. Ein Bereitstellungs-Timeout weist Amazon an, die Bereitstellung von Instance-Kapazität EMR zu beenden, wenn der Cluster während des Cluster-Starts oder der Cluster-Skalierung einen bestimmten Zeitschwellenwert überschreitet. In den folgenden Themen wird beschrieben, wie ein Bereitstellungs-Timeout für den Clusterstart und für Cluster-Hochskalierungsvorgänge konfiguriert wird.

Themen

- Konfigurieren Sie Bereitstellungs-Timeouts für den Cluster-Start in Amazon EMR
- Passen Sie einen Bereitstellungs-Timeout-Zeitraum für die Clustergrößenänderung in Amazon an EMR

Konfigurieren Sie Bereitstellungs-Timeouts für den Cluster-Start in Amazon EMR

Sie können einen Timeout-Zeitraum für die Bereitstellung von Spot Instances für jede Flotte in Ihrem Cluster definieren. Wenn Amazon keine Spot-Kapazität bereitstellen EMR kann, können Sie wählen, ob Sie den Cluster beenden oder stattdessen On-Demand-Kapazität bereitstellen möchten. Wenn der Timeout-Zeitraum während der Cluster-Größenänderung endet, EMR storniert Amazon nicht bereitgestellte Spot-Anfragen. Nicht bereitgestellte Spot Instances werden nicht in On-Demand-Kapazität übertragen.

Führen Sie die folgenden Schritte aus, um ein Bereitstellungs-Timeout für den Clusterstart mit der EMR Amazon-Konsole anzupassen.

Console

Um das Bereitstellungs-Timeout zu konfigurieren, wenn Sie einen Cluster mit der Konsole erstellen

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und anschließend Create cluster aus.
3. Navigieren Sie auf der Seite Cluster erstellen zur Cluster-Konfiguration und wählen Sie Instanceflotten.
4. Geben Sie unter Option Clusterskalierung und -Bereitstellung die Spotgröße für Ihre Core- und Taskflotten an.
5. Wählen Sie unter Spot-Timeout-Konfiguration entweder Cluster nach Spot-Timeout beenden oder Nach Spot-Timeout zu On-Demand wechseln. Geben Sie dann den Timeout-Zeitraum für die Bereitstellung von Spot Instances an. Der Standardwert lautet 1 Stunde.
6. Wählen Sie alle anderen Optionen aus, die für Ihren Cluster gelten.
7. Um Ihren Cluster mit dem konfigurierten Timeout zu starten, wählen Sie Cluster erstellen aus.

AWS CLI

Um ein Bereitstellungs-Timeout mit dem Befehl **`create-cluster`** anzugeben

```
aws emr create-cluster \
--release-label emr-5.35.0 \
--service-role EMR_DefaultRole \
```

```
--ec2-attributes '{"InstanceProfile":"EMR_EC2_DefaultRole","SubnetIds":["subnet-
XXXXX"]}' \
--instance-fleets
 '[{"InstanceFleetType":"MASTER","TargetOnDemandCapacity":1,"TargetSpotCapacity":0,"LaunchSp
{"OnDemandSpecification":{"AllocationStrategy":"lowest-
price"}},"InstanceTypeConfigs":[{"WeightedCapacity":1,"EbsConfiguration":
{"EbsBlockDeviceConfigs":[{"VolumeSpecification":
{"SizeInGB":32,"VolumeType":"gp2"},"VolumesPerInstance":2}]},"BidPriceAsPercentageOfOnDemand
 - 1"},
{"InstanceFleetType":"CORE","TargetOnDemandCapacity":1,"TargetSpotCapacity":1,"LaunchSpecifi
{"SpotSpecification":
{"TimeoutDurationMinutes":120,"TimeoutAction":"SWITCH_TO_ON_DEMAND"},"OnDemandSpecification"
{"AllocationStrategy":"lowest-price"}},"InstanceTypeConfigs":
[{"WeightedCapacity":1,"EbsConfiguration":{"EbsBlockDeviceConfigs":
[{"VolumeSpecification":
{"SizeInGB":32,"VolumeType":"gp2"},"VolumesPerInstance":2}]},"BidPriceAsPercentageOfOnDemand
 - 2"}]'
```

Passen Sie einen Bereitstellungs-Timeout-Zeitraum für die Clustergrößenänderung in Amazon an EMR

Definieren Sie einen Timeout-Zeitraum für die Bereitstellung von Spot Instances für jede Flotte in Ihrem Cluster. Wenn Amazon die Spot-Kapazität nicht bereitstellen EMR kann, storniert es die Anfrage zur Größenänderung und beendet seine Versuche, zusätzliche Spot-Kapazität bereitzustellen. Wenn Sie einen Cluster erstellen, können Sie das Timeout konfigurieren. Für einen laufenden Cluster können Sie ein Timeout hinzufügen oder aktualisieren.

Wenn der Timeout-Zeitraum abläuft, sendet Amazon Ereignisse EMR automatisch an einen Amazon CloudWatch Events-Stream. Mit können Sie Regeln erstellen CloudWatch, die Ereignisse nach einem bestimmten Muster zuordnen, und die Ereignisse dann an Ziele weiterleiten, um Maßnahmen zu ergreifen. Sie können beispielsweise eine Regel zum Senden einer E-Mail-Benachrichtigung konfigurieren. Weitere Informationen zum Erstellen von Regeln finden Sie unter Regeln für EMR Amazon-Events erstellen mit CloudWatch. Weitere Informationen zu verschiedenen Ereignisdetails finden Sie unter Ereignisse zur Änderung des Status der Instance-Flotte.

Beispiele für Bereitstellungs-Timeouts bei der Clustergrößenänderung

Geben Sie ein Bereitstellungs-Timeout für die Größenänderung mit dem AWS CLI an

Im folgenden Beispiel wird der `create-cluster`-Befehl verwendet, um ein Bereitstellungs-Timeout für die Größenänderung hinzuzufügen.

```
aws emr create-cluster \
--release-label emr-5.35.0 \
--service-role EMR_DefaultRole \
--ec2-attributes '{"InstanceProfile":"EMR_EC2_DefaultRole","SubnetIds":["subnet-
XXXXX"]}' \
--instance-fleets
 '[{"InstanceFleetType":"MASTER","TargetOnDemandCapacity":1,"TargetSpotCapacity":0,"InstanceTyp
[{"WeightedCapacity":1,"EbsConfiguration":{"EbsBlockDeviceConfigs":
[{"VolumeSpecification":
{"SizeInGB":32,"VolumeType":"gp2"},"VolumesPerInstance":2}]},"BidPriceAsPercentageOfOnDemandPri
 - 1"},
{"InstanceFleetType":"CORE","TargetOnDemandCapacity":1,"TargetSpotCapacity":1,"LaunchSpecificat
{"SpotSpecification":
{"TimeoutDurationMinutes":120,"TimeoutAction":"SWITCH_TO_ON_DEMAND"},"OnDemandSpecification":
{"AllocationStrategy":"lowest-price"}},"ResizeSpecifications":
{"SpotResizeSpecification":{"TimeoutDurationMinutes":20},"OnDemandResizeSpecification":
{"TimeoutDurationMinutes":25}},"InstanceTypeConfigs":
[{"WeightedCapacity":1,"EbsConfiguration":{"EbsBlockDeviceConfigs":
[{"VolumeSpecification":
{"SizeInGB":32,"VolumeType":"gp2"},"VolumesPerInstance":2}]},"BidPriceAsPercentageOfOnDemandPri
 - 2"}]'
```

Im folgenden Beispiel wird der `modify-instance-fleet`-Befehl verwendet, um ein Bereitstellungs-Timeout für die Größenänderung hinzuzufügen.

```
aws emr modify-instance-fleet \
--cluster-id j-XXXXXXXXXXXX \
--instance-fleet '{"InstanceFleetId":"if-XXXXXXXXXXX","ResizeSpecifications":
{"SpotResizeSpecification":{"TimeoutDurationMinutes":30},"OnDemandResizeSpecification":
{"TimeoutDurationMinutes":60}}}' \
--region us-east-1
```

Im folgenden Beispiel wird der `add-instance-fleet-command` verwendet, um ein Bereitstellungs-Timeout für die Größenänderung hinzuzufügen.

```
aws emr add-instance-fleet \
--cluster-id j-XXXXXXXXXXXX \
--instance-fleet
 '{"InstanceFleetType":"TASK","TargetOnDemandCapacity":1,"TargetSpotCapacity":0,"InstanceTypeCo
[{"WeightedCapacity":1,"EbsConfiguration":{"EbsBlockDeviceConfigs":
[{"VolumeSpecification":
{"SizeInGB":32,"VolumeType":"gp2"},"VolumesPerInstance":2}]},"BidPriceAsPercentageOfOnDemandPri
```

```
{"SpotResizeSpecification":{"TimeoutDurationMinutes":30},"OnDemandResizeSpecification":
{"TimeoutDurationMinutes":35}}}' \
--region us-east-1
```

Geben Sie ein Bereitstellungs-Timeout für die Größenänderung und den Start mit dem AWS CLI

Im folgenden Beispiel wird der `create-cluster`-Befehl verwendet, um ein Bereitstellungs-Timeout für die Größenänderung hinzuzufügen.

```
aws emr create-cluster \
--release-label emr-5.35.0 \
--service-role EMR_DefaultRole \
--ec2-attributes '{"InstanceProfile":"EMR_EC2_DefaultRole","SubnetIds":["subnet-
XXXXX"]}' \
--instance-fleets
 '[{"InstanceFleetType":"MASTER","TargetOnDemandCapacity":1,"TargetSpotCapacity":0,"LaunchSpeci
{"OnDemandSpecification":{"AllocationStrategy":"lowest-price"}},"InstanceTypeConfigs":
[{"WeightedCapacity":1,"EbsConfiguration":{"EbsBlockDeviceConfigs":
[{"VolumeSpecification":
{"SizeInGB":32,"VolumeType":"gp2"},"VolumesPerInstance":2}]},"BidPriceAsPercentageOfOnDemandPri
 - 1"},
{"InstanceFleetType":"CORE","TargetOnDemandCapacity":1,"TargetSpotCapacity":1,"LaunchSpecificat
{"SpotSpecification":
{"TimeoutDurationMinutes":120,"TimeoutAction":"SWITCH_TO_ON_DEMAND"},"OnDemandSpecification":
{"AllocationStrategy":"lowest-price"}},"ResizeSpecifications":
{"SpotResizeSpecification":{"TimeoutDurationMinutes":20},"OnDemandResizeSpecification":
{"TimeoutDurationMinutes":25}},"InstanceTypeConfigs":
[{"WeightedCapacity":1,"EbsConfiguration":{"EbsBlockDeviceConfigs":
[{"VolumeSpecification":
{"SizeInGB":32,"VolumeType":"gp2"},"VolumesPerInstance":2}]},"BidPriceAsPercentageOfOnDemandPri
 - 2"}]'
```

Überlegungen zur Größenänderung von Bereitstellungs-Timeouts

Wenn Sie Timeouts für die Cluster-Bereitstellung für Ihre Instanceflotten konfigurieren, sollten Sie die folgenden Verhaltensweisen berücksichtigen.

- Sie können Bereitstellungs-Timeouts sowohl für Spot Instances als auch für On-Demand-Instances konfigurieren. Das Mindestzeitlimit für die Bereitstellung beträgt 5 Minuten. Das maximale Bereitstellungszeitlimit beträgt 7 Tage.
- Sie können Bereitstellungs-Timeouts nur für einen EMR Cluster konfigurieren, der Instanzflotten verwendet. Sie müssen jeden Core und jede Aufgaben-Flotte separat konfigurieren.

- Wenn Sie einen Cluster erstellen, können Sie Bereitstellungs-Zeitlimits konfigurieren. Sie können ein Zeitlimit für einen laufenden Cluster hinzufügen oder ein vorhandenes Zeitlimit aktualisieren.
- Wenn Sie mehrere Größenänderungsvorgänge einreichen, EMR verfolgt Amazon die Bereitstellungs-Timeouts für jeden Größenänderungsvorgang. Legen Sie beispielsweise das Bereitstellungs-Timeout für einen Cluster auf fest *60* Minuten. Senden Sie dann einen Vorgang zur Größenänderung *R1* zu einem bestimmten Zeitpunkt *T1*. Reichen Sie einen zweiten Vorgang zur Größenänderung ein *R2* zu einem bestimmten Zeitpunkt *T2*. Das Bereitstellungs-Timeout für R1 läuft ab am *T1 + 60 minutes*. Das Bereitstellungs-Timeout für R2 läuft ab am *T2 + 60 minutes*.
- Wenn Sie vor Ablauf des Timeouts einen neuen Vorgang zur Skalierung der Größe einreichen, versucht Amazon EMR weiterhin, Kapazität für Ihren Cluster bereitzustellen. EMR

Cluster-Herunterskalierung

> **Note**
>
> Optionen für das Scale-down-Verhalten werden seit der EMR Amazon-Version 5.10.0 nicht mehr unterstützt. Aufgrund der Einführung der sekundengenauen Abrechnung in Amazon EC2 wird das Standard-Scale-Down-Verhalten für EMR Amazon-Cluster jetzt bei Abschluss der Aufgabe beendet.

In den EMR Amazon-Versionen 5.1.0 bis 5.9.1 gibt es zwei Optionen für das Scale-Down-Verhalten: Beenden innerhalb der Instanz-Stunden-Grenze für die EC2 Amazon-Abrechnung oder Beenden bei Abschluss der Aufgabe. Ab EMR Amazon-Version 5.10.0 ist die Einstellung für die Kündigung innerhalb der Instance-Stundengrenze veraltet, da Amazon die Abrechnung pro Sekunde eingeführt hat. EC2 Wir raten davon ab, die Beendigung zur Instance-Stundengrenze zu verwenden, wenn diese Option angeboten wird.

> **Warning**
>
> Wenn Sie die Option „ AWS CLI `modify-instance-groups`“ verwenden`EC2InstanceIdsToTerminate`, werden diese Instances sofort beendet, ohne Berücksichtigung dieser Einstellungen und unabhängig vom Status der Anwendungen, die auf ihnen ausgeführt werden. Wenn eine Instance auf diese Weise beendet wird, besteht das Risiko von Datenverlusten und unvorhersehbarem Clusterverhalten.

Wenn „Bei Abschluss der Aufgabe beenden“ angegeben ist, EMR lehnt Amazon die Aufgaben ab und entfernt sie von den Knoten, bevor die EC2 Amazon-Instances beendet werden. Wenn eines der beiden Verhaltensweisen angegeben ist, beendet Amazon EMR keine EC2 Amazon-Instances in Kerninstanzgruppen, wenn dies zu HDFS Beschädigungen führen könnte.

Beendigung bei Aufgaben-Abschluss

EMRMit Amazon können Sie Ihren Cluster herunterskalieren, ohne Ihre Arbeitslast zu beeinträchtigen. Amazon setzt während EMR einer Größenänderung ordnungsgemäß YARNHDFS, und andere Daemons auf Kern- und Taskknoten außer Betrieb, ohne Daten zu verlieren oder Jobs zu unterbrechen. Amazon reduziert die Größe der Instance-Gruppen EMR nur, wenn die den Gruppen zugewiesene Arbeit abgeschlossen ist und sie inaktiv sind. Bei YARN NodeManager Graceful Decommission können Sie die Zeit, die ein Knoten auf die Außerbetriebnahme wartet, manuell anpassen.

Die Dauer wird mit einer Eigenschaft in der `YARN-site`-Konfigurationsklassifizierung eingerichtet. Wenn Sie Amazon EMR Version 5.12.0 und höher verwenden, geben Sie die `YARN.resourcemanager.nodemanager-graceful-decommission-timeout-secs` Eigenschaft an. Geben Sie die `YARN.resourcemanager.decommissioning.timeout` Eigenschaft bei Verwendung früherer EMR Amazon-Versionen an.

Wenn nach Ablauf des Zeitlimits für die Außerbetriebnahme noch Container oder YARN Anwendungen ausgeführt werden, wird der Knoten zwangsweise außer Betrieb genommen und die betroffenen Container werden auf YARN anderen Knoten neu geplant. Die Standardwert ist 3600 Sekunden (eine Stunde). Sie können den Timeout auf einen extrem hohen Wert festlegen, um die ordnungsgemäße Verkleinerung zu verzögern. Weitere Informationen finden Sie unter Ordnungsgemäße Außerbetriebnahme von YARN Knoten in der Apache Hadoop-Dokumentation.

Aufgabenknoten-Gruppen

Amazon wählt EMR intelligent Instances aus, die keine Aufgaben haben, die für einen Schritt oder eine Anwendung ausgeführt werden, und entfernt diese Instances zunächst aus einem Cluster. Wenn alle Instances im Cluster verwendet werden, EMR wartet Amazon, bis die Aufgaben auf einer Instance abgeschlossen sind, bevor sie aus dem Cluster entfernt wird. Die standardmäßige Leerlaufzeit beträgt eine Stunde. Dieser Wert kann mit der Einstellung `YARN.resourcemanager.decommissioning.timeout` geändert werden. Amazon verwendet die neue Einstellung EMR dynamisch. Sie können dies auf eine beliebig große Zahl festlegen, um sicherzustellen, dass Amazon EMR keine Aufgaben beendet und gleichzeitig die Clustergröße reduziert.

Core-Knoten-Gruppen

Auf den Core-Knoten müssen YARN NodeManager sowohl HDFS DataNode Daemons als auch Daemons außer Betrieb genommen werden, damit die Instanzgruppe verkleinert werden kann. Durch die schrittweise Reduzierung wird nämlich sichergestelltYARN, dass ein für die Außerbetriebnahme markierter Knoten nur dann in den `DECOMMISSIONED` Status versetzt wird, wenn keine ausstehenden oder unvollständigen Container oder Anwendungen vorhanden sind. Die Außerbetriebnahme wird direkt beendet, falls es zu Beginn der Außerbetriebnahme keine laufenden Container auf dem Knoten gibt.

Denn HDFS durch eine schrittweise Reduzierung wird sichergestellt, dass die Zielkapazität von groß genug HDFS ist, um alle vorhandenen Blöcke aufzunehmen. Wenn die Zielkapazität nicht groß genug ist, wird nur ein Teil der Core-Instances außer Betrieb genommen, sodass die verbleibenden Knoten die aktuellen Daten verarbeiten können, in denen sie sich befinden. HDFS Sie sollten zusätzliche HDFS Kapazität sicherstellen, um eine weitere Außerbetriebnahme zu ermöglichen. Sie sollten auch versuchen, die Schreib-I/O zu minimieren, bevor Sie versuchen, Instance-Gruppen zu reduzieren. Übermäßiger Schreib-E/A-Vorgang kann den Abschluss des Größenänderungsvorgangs verzögern.

Ein weiterer Faktor ist der Standard-Replikationsfaktor (`dfs.replication`) in `/etc/hadoop/conf/hdfs-site`. Bei der Erstellung eines Clusters EMR konfiguriert Amazon den Wert auf der Grundlage der Anzahl der Instances im Cluster: 1 mit 1—3 Instances, 2 für Cluster mit 4—9 Instances und 3 für Cluster mit mehr als 10 Instances.

⚠ Warning

1. Die Einstellung `dfs.replication` auf 1 für Cluster mit weniger als vier Knoten kann zu HDFS Datenverlust führen, wenn ein einzelner Knoten ausfällt. Wir empfehlen, für Produktionsworkloads einen Cluster mit mindestens vier Core-Knoten zu verwenden.
2. Amazon EMR erlaubt Clustern nicht, Kernknoten nach unten zu skalierendfs.replication. Bei `dfs.replication = 2` z. B. beträgt die Mindestanzahl von Core-Knoten 2.
3. Wenn Sie verwaltete Skalierung oder Auto-Scaling verwenden oder die Größe Ihres Clusters manuell ändern möchten, empfehlen wir Ihnen, `dfs.replication` auf 2 oder höher einzustellen.

Durch die schrittweise Reduzierung können Sie die Anzahl der Kernknoten nicht unter den HDFS Replikationsfaktor reduzieren. Auf diese Weise können HDFS Dateien aufgrund unzureichender

Replikate geschlossen werden. Um dieses Limit zu umgehen, verringern Sie den Replikationsfaktor und starten Sie den NameNode Daemon neu.

Das EMR Scale-Down-Verhalten von Amazon konfigurieren

> **Note**
>
> Die Scale-Down-Verhaltensoption „Zur Instance-Stunde beenden“ wird für EMR Amazon-Version 5.10.0 und höher nicht mehr unterstützt. Die folgenden Optionen für das Scale-Down-Verhalten werden nur in der EMR Amazon-Konsole für die Versionen 5.1.0 bis 5.9.1 angezeigt.

Sie können Amazon verwenden AWS Management Console, um das AWS CLI Scale-Down-Verhalten EMR API zu konfigurieren, wenn Sie einen Cluster erstellen.

Console

Um das Scale-Down-Verhalten mit der Konsole zu konfigurieren

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie EC2 im linken Navigationsbereich unter EMRon die Option Clusters und anschließend Create cluster aus.
3. Wählen Sie im Abschnitt Optionen für Clusterskalierung und -bereitstellung die Option Benutzerdefinierte automatische Skalierung verwenden aus. Wählen Sie unter Benutzerdefinierte automatische Skalierungsrichtlinien die Plus-Aktionsschaltfläche aus, um Skalierungsrichtlinien hinzuzufügen. Wir empfehlen, dass Sie sowohl Richtlinien für die Skalierung als auch für die horizontale Skalierung hinzufügen. Wenn Sie nur einen Satz von Richtlinien hinzufügen, EMR führt Amazon nur eine unidirektionale Skalierung durch und Sie müssen die anderen Aktionen manuell ausführen.
4. Wählen Sie alle anderen Optionen aus, die für Ihren Cluster gelten.
5. Um Ihren Cluster jetzt zu starten, wählen Sie Cluster erstellen aus.

AWS CLI

Um das Scale-Down-Verhalten zu konfigurieren, verwenden Sie AWS CLI

- Verwenden Sie für die `--scale-down-behavior`-Option entweder `TERMINATE_AT_INSTANCE_HOUR` oder `TERMINATE_AT_TASK_COMPLETION`.

Einen Cluster beenden

In diesem Abschnitt werden die Methoden zum Beenden eines Clusters beschrieben. Informationen zum Aktivieren des Beendigungsschutzes und zum automatischen Beenden von Clustern finden Sie unter Steuern der Cluster-Beendigung. Sie können Cluster mit dem Status `STARTING`, `RUNNING`oder `WAITING` beenden. Ein Cluster mit dem Status `WAITING` muss beendet werden. Andernfalls wird er unbegrenzt ausgeführt, und verursacht Gebühren für Ihr Konto. Sie können einen Cluster beenden, der den Status `STARTING` nicht verlässt oder der einen bestimmten Schritt nicht durchführen kann.

Wenn Sie einen Cluster beenden, bei dem der Beendigungsschutz aktiviert ist, müssen Sie den Beendigungsschutz deaktivieren, bevor Sie den Cluster beenden können. Cluster können mithilfe der Konsole, der oder programmgesteuert mit dem AWS CLI beendet werden. `TerminateJobFlows` API

Je nach Konfiguration des Clusters kann es zwischen 5 und 20 Minuten dauern, bis der Cluster vollständig beendet und zugewiesene Ressourcen, z. B. EC2 Instanzen, freigegeben hat.

> **Note**
>
> Sie können einen beendeten Cluster nicht neu starten, aber Sie können einen beendeten Cluster klonen, um seine Konfiguration für einen neuen Cluster wiederzuverwenden. Weitere Informationen finden Sie unter Klonen eines Clusters mithilfe der Konsole.

> **Important**
>
> Amazon EMR verwendet die EMRAmazon-Servicerolle und die `AWSServiceRoleForEMRCleanup` Rolle, um Cluster-Ressourcen in Ihrem Konto zu bereinigen, die Sie nicht mehr verwenden, z. B. EC2 Amazon-Instances. Sie müssen Aktionen für die Rollenrichtlinien angeben, um die Ressourcen zu löschen oder zu beenden.

Andernfalls EMR kann Amazon diese Bereinigungsaktionen nicht durchführen, und es können Kosten für ungenutzte Ressourcen anfallen, die im Cluster verbleiben.

Einen Cluster mit der Konsole zu beenden

Sie können einen oder mehrere Cluster mit der EMR Amazon-Konsole beenden. Die Schritte zum Beenden eines Clusters über die Konsole variieren je nachdem, ob der Beendigungsschutz aktiviert oder deaktiviert ist. Um einen geschützten Cluster zu beenden, müssen Sie zuerst den Beendigungsschutz deaktivieren.

Console

Um einen Cluster mit der Konsole zu beenden

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen Sie Clusters und dann den Cluster aus, den Sie beenden möchten.
3. Wählen Sie im Dropdownmenü Aktionen die Option Cluster beenden aus, um die Aufforderung Cluster beenden zu öffnen.
4. Wählen Sie an der Eingabeaufforderung die Option Beenden. Je nach Clusterkonfiguration kann die Kündigung 5 bis 10 Minuten dauern. Weitere Informationen zur Verwendung von EMR Amazon-Clustern finden Sie unterEinen Cluster beenden.

Beenden eines Clusters mithilfe der AWS CLI

Um einen ungeschützten Cluster mit dem zu beenden AWS CLI

Um einen ungeschützten Cluster mit dem zu beenden AWS CLI, verwenden Sie den `terminate-clusters` Unterbefehl mit dem Parameter --cluster-ids.

- Geben Sie den folgenden Befehl ein, um einen einzelnen Cluster zu beenden und zu ersetzen *j-3KVXXXXXXX7UG* mit Ihrer Cluster-ID.

  ```
  aws emr terminate-clusters --cluster-ids j-3KVXXXXXXX7UG
  ```

  Um mehrere Cluster zu beenden, geben Sie den folgenden Befehl ein und ersetzen Sie *j-3KVXXXXXXX7UG* and *j-WJ2XXXXXX8EU* mit Ihrem ClusterIDs.

```
aws emr terminate-clusters --cluster-ids j-3KVXXXXXXX7UG j-WJ2XXXXXX8EU
```

Weitere Informationen zur Verwendung von EMR Amazon-Befehlen in der AWS CLI finden Sie unterhttps://docs.aws.amazon.com/cli/latest/reference/emr.

Um einen geschützten Cluster mit dem zu beenden AWS CLI

Um einen geschützten Cluster mit dem zu beenden AWS CLI, deaktivieren Sie zunächst den Kündigungsschutz mithilfe des `modify-cluster-attributes` Unterbefehls mit dem `--no-termination-protected` Parameter. Verwenden Sie dann den Unterbefehl `terminate-clusters` mit dem Parameter `--cluster-ids`, um den Cluster zu beenden.

1. Geben Sie den folgenden Befehl ein, um den Kündigungsschutz zu deaktivieren und zu ersetzen *j-3KVTXXXXXX7UG* mit Ihrer Cluster-ID.

   ```
   aws emr modify-cluster-attributes --cluster-id j-3KVTXXXXXX7UG --no-termination-protected
   ```

2. Um den Cluster zu beenden, geben Sie den folgenden Befehl ein und ersetzen Sie *j-3KVXXXXXXX7UG* mit Ihrer Cluster-ID.

   ```
   aws emr terminate-clusters --cluster-ids j-3KVXXXXXXX7UG
   ```

   Um mehrere Cluster zu beenden, geben Sie den folgenden Befehl ein und ersetzen Sie *j-3KVXXXXXXX7UG* and *j-WJ2XXXXXX8EU* mit Ihrem ClusterIDs.

   ```
   aws emr terminate-clusters --cluster-ids j-3KVXXXXXXX7UG j-WJ2XXXXXX8EU
   ```

   Weitere Informationen zur Verwendung von EMR Amazon-Befehlen in der AWS CLI finden Sie unterhttps://docs.aws.amazon.com/cli/latest/reference/emr.

Beenden eines Clusters mithilfe der API

Der `TerminateJobFlows` Vorgang beendet die Schrittverarbeitung, lädt alle Protokolldaten von Amazon EC2 auf Amazon S3 hoch (falls konfiguriert) und beendet den Hadoop-Cluster. Ein Cluster

wird außerdem automatisch beendet, wenn Sie in einer `KeepJobAliveWhenNoSteps`-Anforderung `False` auf `RunJobFlows` festlegen.

Sie können diese Aktion verwenden, um entweder einen einzelnen Cluster oder eine Liste von Clustern nach ihrem Cluster zu beenden. IDs

Weitere Hinweise zu den Eingabeparametern, die nur für gelten`TerminateJobFlows`, finden Sie unter TerminateJobFlows. Weitere Informationen zu den grundlegenden Parametern in der Anfrage finden Sie unter Allgemeine Anforderungsparameter.

Klonen eines Clusters mithilfe der Konsole

Sie können die EMR Amazon-Konsole verwenden, um einen Cluster zu klonen. Dabei wird eine Kopie der Konfiguration des ursprünglichen Clusters erstellt, die als Grundlage für einen neuen Cluster verwendet wird.

Console

Um einen Cluster mit der Konsole zu klonen

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die EMR Amazon-Konsole unter https://console.aws.amazon.com/emr.
2. Wählen EMRSie EC2 im linken Navigationsbereich unter on die Option Clusters aus.
3. Um einen Cluster aus der Cluster-LIS zu klonen

    a. Verwenden Sie die Such- und Filteroptionen, um den Cluster, den Sie klonen möchten, in der Listenansicht zu finden.

    b. Markieren Sie das Kontrollkästchen links neben der Zeile für den Cluster, den Sie klonen möchten.

    c. Die Option Klonen ist jetzt oben in der Listenansicht verfügbar. Wählen Sie Klonen aus, um den Klonvorgang zu starten. Wenn für den Cluster Schritte konfiguriert sind, wählen Sie Schritte einschließen und Weiter aus, wenn Sie die Schritte zusammen mit den anderen Clusterkonfigurationen klonen möchten.

    d. Überprüfen Sie die Einstellungen für den neuen Cluster, die aus dem geklonten Cluster kopiert wurden. Passen Sie die Einstellungen bei Bedarf an. Wenn Sie mit der Konfiguration des neuen Clusters zufrieden sind, wählen Sie Cluster erstellen aus, um den neuen Cluster zu starten.
4. Wie Sie einen Cluster von einer Cluster-Detailseite aus klonen

a. Um zur Detailseite des Clusters zu gelangen, den Sie klonen möchten, wählen Sie dessen Cluster-ID aus der Cluster-Listenansicht aus.

b. Wählen Sie oben auf der Cluster-Detailseite im Menü Aktionen die Option Cluster klonen aus, um den Klonvorgang zu starten. Wenn für den Cluster Schritte konfiguriert sind, wählen Sie Schritte einschließen und Weiter aus, wenn Sie die Schritte zusammen mit den anderen Clusterkonfigurationen klonen möchten.

c. Überprüfen Sie die Einstellungen für den neuen Cluster, die aus dem geklonten Cluster kopiert wurden. Passen Sie die Einstellungen bei Bedarf an. Wenn Sie mit der Konfiguration des neuen Clusters zufrieden sind, wählen Sie Cluster erstellen aus, um den neuen Cluster zu starten.

Automatisieren wiederkehrender Cluster mit AWS Data Pipeline

AWS Data Pipeline ist ein Dienst, der die Übertragung und Transformation von Daten automatisiert. Sie können ihn verwenden, um Eingabedaten in Amazon S3 zu verlagern und das Starten von Clustern zu planen, die diese Daten verarbeiten. Betrachten wir zum Beispiel den Fall, bei dem ein Webserver Datenverkehrsprotokolle aufzeichnet. Wenn Sie einen wöchentlichen Cluster zur Analyse der Verkehrsdaten ausführen möchten, können Sie ihn AWS Data Pipeline zur Planung dieser Cluster verwenden. AWS Data Pipeline ist ein datengesteuerter Workflow, sodass eine Aufgabe (Starten des Clusters) von einer anderen Aufgabe (Verschieben der Eingabedaten nach Amazon S3) abhängig sein kann. Der Workflow verfügt außerdem über eine robuste Wiederholungsfunktionalität.

Weitere Informationen zu AWS Data Pipeline finden Sie im AWS Data Pipeline Entwicklerhandbuch, insbesondere in den Tutorials zu AmazonEMR:

- Tutorial: Einen EMR Amazon-Jobflow starten
- Erste Schritte: Webprotokolle mit AWS Data Pipeline Amazon EMR und Hive verarbeiten
- Tutorial: Amazon DynamoDB importieren und exportieren mit AWS Data Pipeline

Fehlersuche bei Clustern

Ein EMR Cluster wird in einem komplexen Ökosystem ausgeführt, das Open-Source-Software, benutzerdefinierten Anwendungscode und AWS -Services umfasst. Wenn bei einem dieser Teile ein Problem auftritt, schlägt der Cluster möglicherweise fehl oder es dauert länger als erwartet, bis er abgeschlossen ist. Die folgenden Themen können Ihnen bei der Identifizierung von Cluster-Problemen und deren Behebung helfen.

Themen

Wenn Sie eine neue Hadoop-Anwendung entwickeln, empfehlen wir Ihnen, das Debugging zu aktivieren und eine kleine, aber repräsentative Teilmenge Ihrer Daten zu verarbeiten, um die Anwendung zu testen. Möglicherweise möchten Sie die Anwendung auch ausführen, step-by-step um jeden Schritt separat zu testen. Weitere Informationen erhalten Sie unter Konfigurieren der Cluster-Protokollierung und des Debuggings und Schritt 5: Den Cluster Schritt für Schritt testen.

Welche Tools sind zur Fehlerbehebung verfügbar?

Um Clusterfehler zu identifizieren und zu beheben, können Sie die auf dieser Seite beschriebenen Tools verwenden. Möglicherweise müssen Sie einige der Tools initialisieren, wenn Sie den Cluster starten. Andere Tools sind standardmäßig für jeden Cluster verfügbar.

Themen

- Überwachen Sie die EMR Cluster-Leistung

EMRCluster-Details anzeigen

Sie können das AWS Management Console, oder verwenden AWS CLI, EMR API um detaillierte Informationen über einen EMR Cluster und die Auftragsausführung abzurufen. Weitere Hinweise zur Verwendung von AWS Management Console und AWS CLI finden Sie unterCluster-Status und -Details anzeigen.

Detailbereich EMR der Amazon-Konsole

In der Clusterliste auf der EMR Amazon-Konsole finden Sie allgemeine Informationen zum Status der einzelnen Cluster in Ihrem Konto und AWS-Region. Die Liste zeigt alle aktiven und beendeten Cluster an, die Sie in den vergangenen zwei Monaten gestartet haben. Sie können in der Liste Clusters (Cluster) den Name (Namen) eines Clusters auswählen, um Details zu diesem anzuzeigen. Diese Informationen sind in verschiedene Kategorien unterteilt, um das Navigieren zu vereinfachen.

Die auf der Cluster-Detailseite verfügbaren Anwendungsbenutzeroberflächen können bei der Fehlerbehebung bei Clustern hilfreich sein. Sie enthält Informationen zum Status von YARN Anwendungen. Bei einigen Anwendungen, wie z. B. Spark-Anwendungen, können Sie verschiedene Kennzahlen und Facetten wie Jobs, Phasen und Executors genauer untersuchen. Weitere Informationen finden Sie unter Anwendungsverlauf anzeigen. Diese Funktion ist nur für EMR Amazon-Versionen 5.8.0 und höher verfügbar.

EMRAmazon-Befehlszeilenschnittstelle

Sie können Details zu einem Cluster AWS CLI anhand des `--describe` Arguments finden.

Amazon EMR API

Einzelheiten zu einem Cluster finden Sie API unter Verwendung der `DescribeJobFlows` Aktion.

EMRCluster-Fehlerdetails anzeigen

Wenn ein EMR Cluster mit einem Fehler beendet wird, werden ein Fehlercode und eine Fehlermeldung `ListClusters` APIs zurückgegeben. `DescribeCluster` Bei ausgewählten Clusterfehlern kann Ihnen das `ErrorDetail`-Datenarray bei der Behebung des Fehlers helfen.

Eine Liste der Fehlercodes, die `ErrorDetail` Daten enthalten, finden Sie unter Fehlercodes mit ErrorDetail Informationen.

> **Note**
>
> Wir verfeinern unsere Fehlermeldungen kontinuierlich, damit Sie die aktuellsten und relevantesten Informationen erhalten. Es wird nicht empfohlen, den Text von `ErrorMessage` zu analysieren, da sich dieser Text ändern kann.

Führen Sie Skripts aus und konfigurieren Sie EMR Amazon-Prozesse

Im Rahmen Ihrer Problembehandlung kann es hilfreich sein, benutzerdefinierte Skripts auf Ihrem Cluster auszuführen oder Clusterprozesse anzuzeigen und zu konfigurieren.

Anwendungsprozesse anzeigen und neu starten

Es kann hilfreich sein, sich die laufenden Prozesse auf Ihrem Cluster anzusehen, um potenzielle Probleme zu diagnostizieren. Sie können Clusterprozesse beenden und neu starten, indem Sie eine Verbindung zum Hauptknoten Ihres Clusters herstellen. Weitere Informationen finden Sie unter Amazon- EMR und Anwendungsprozesse (Daemons) anzeigen und neu starten.

Führen Sie Befehle und Skripts ohne SSH Verbindung aus

Um als Schritt einen Befehl oder ein Skript auf Ihrem Cluster auszuführen, können Sie die `script-runner.jar` Tools `command-runner.jar` oder verwenden, ohne eine SSH Verbindung zum Master-Knoten herzustellen. Weitere Informationen finden Sie unter Befehle und Skripts auf einem EMR Amazon-Cluster ausführen.

Anzeige von -Protokolldateien

Amazon EMR und Hadoop generieren beide Protokolldateien, während der Cluster ausgeführt wird. Sie können auf diese Protokolldateien mit mehreren Tools zugreifen, abhängig von der Konfiguration, die Sie beim Starten des Clusters angegeben haben. Weitere Informationen finden Sie unter Konfigurieren der Cluster-Protokollierung und des Debuggings.

Protokolldateien auf dem Hauptknoten

Jeder Cluster veröffentlicht Protokolldateien im Verzeichnis /mnt/var/log/ auf dem Master-Knoten. Diese Protokolldateien sind nur verfügbar, während der Cluster ausgeführt wird.

So archivieren Sie Protokolldateien in Amazon S3

Wenn Sie den Cluster starten und einen Amazon S3 -Pfad angeben, kopiert der Cluster die in / mnt/var/log/ gespeicherten Protokolldateien auf dem Hauptknoten nach Amazon S3 in 5-Minuten-Intervallen. So wird sichergestellt, dass Sie Zugriff auf die Protokolldateien auch nach Beendigung des Clusters haben. Da die Dateien in 5-Minuten-Intervallen archiviert werden, stehen die letzten Minuten eines unvermittelt beendeten Clusters ggf. nicht zur Verfügung.

Überwachen Sie die EMR Cluster-Leistung

Amazon EMR bietet verschiedene Tools zur Überwachung der Leistung Ihres Clusters.

Hadoop-Webschnittstellen

Jeder Cluster veröffentlicht eine Reihe von Webschnittstellen auf dem Master-Knoten, die Informationen über den Cluster enthalten. Sie können auf diese Webseiten zugreifen, indem Sie sie über einen SSH Tunnel auf dem Master-Knoten verbinden. Weitere Informationen finden Sie unter Auf EMR Amazon-Clustern gehostete Weboberflächen anzeigen.

CloudWatch Metriken

Jeder Cluster meldet Metriken an CloudWatch. CloudWatch ist ein Webservice, der Metriken verfolgt und mit dem Sie Alarme für diese Metriken einrichten können. Weitere Informationen finden Sie unter Überwachung von EMR Amazon-Metriken mit CloudWatch.

Amazon- EMR und Anwendungsprozesse (Daemons) anzeigen und neu starten

Wenn Sie in einem Cluster Fehler beheben, möchten Sie möglicherweise laufende Prozesse auflisten. Möglicherweise möchten Sie Prozesse auch beenden oder neu starten. Sie können beispielsweise einen Prozess neu starten, nachdem Sie eine Konfiguration geändert haben, oder ein Problem mit einem bestimmten Prozess feststellen, nachdem Sie Protokolldateien und Fehlermeldungen analysiert haben.

Es gibt zwei Arten von Prozessen, die auf einem Cluster ausgeführt werden: EMR Amazon-Prozesse (z. B. Instance-Controller und Log Pusher) und Prozesse, die den auf dem Cluster installierten Anwendungen zugeordnet sind (z. B. hadoop-hdfs-namenode, und). hadoop-yarn-resourcemanager

Um mit Prozessen direkt auf einem Cluster zu arbeiten, stellen Sie eine Verbindung mit dem Hauptknoten her. Weitere Informationen finden Sie unter Verbinden mit einem Cluster.

Anzeigen von ausgeführten Prozessen

Die Methode, die Sie verwenden, um laufende Prozesse in einem Cluster anzuzeigen, unterscheidet sich je nach der von Ihnen verwendeten EMR Amazon-Version.

EMR 5.30.0 and 6.0.0 and later

Example : Listet alle laufenden Prozesse auf

Im folgenden Beispiel wird `systemctl` verwendet und `--type` angegeben, um alle Prozesse anzuzeigen.

```
systemctl --type=service
```

Example : Listet bestimmte Prozesse auf

Im folgenden Beispiel werden alle Prozesse aufgeführt, deren Namen `hadoop` enthalten.

```
systemctl --type=service | grep -i hadoop
```

Beispielausgabe:

```
 hadoop-hdfs-namenode.service           loaded active running Hadoop namenode
 hadoop-httpfs.service                  loaded active running Hadoop httpfs
 hadoop-kms.service                     loaded active running Hadoop kms
 hadoop-mapreduce-historyserver.service loaded active running Hadoop historyserver
 hadoop-state-pusher.service            loaded active running Daemon process that
 processes and serves EMR metrics data.
 hadoop-yarn-proxyserver.service        loaded active running Hadoop proxyserver
 hadoop-yarn-resourcemanager.service    loaded active running Hadoop resourcemanager
 hadoop-yarn-timelineserver.service     loaded active running Hadoop timelineserver
```

Example : Sehen Sie sich einen detaillierten Statusbericht für einen bestimmten Prozess an

Im folgenden Beispiel wird ein detaillierter Statusbericht für den `hadoop-hdfs-namenode`-Service angezeigt.

```
sudo systemctl status hadoop-hdfs-namenode
```

Beispielausgabe:

```
hadoop-hdfs-namenode.service - Hadoop namenode
   Loaded: loaded (/etc/systemd/system/hadoop-hdfs-namenode.service; enabled; vendor
 preset: disabled)
   Active: active (running) since Wed 2021-08-18 21:01:46 UTC; 26min ago
 Main PID: 9733 (java)
    Tasks: 0
   Memory: 1.1M
   CGroup: /system.slice/hadoop-hdfs-namenode.service
           # 9733 /etc/alternatives/jre/bin/java -Dproc_namenode -Xmx1843m -server -
XX:OnOutOfMemoryError=kill -9 %p ...

Aug 18 21:01:37 ip-172-31-20-123 systemd[1]: Starting Hadoop namenode...
Aug 18 21:01:37 ip-172-31-20-123 su[9715]: (to hdfs) root on none
Aug 18 21:01:37 ip-172-31-20-123 hadoop-hdfs-namenode[9683]: starting namenode,
 logging to /var/log/hadoop-hdfs/ha...out
Aug 18 21:01:46 ip-172-31-20-123 hadoop-hdfs-namenode[9683]: Started Hadoop
 namenode:[  OK  ]
Aug 18 21:01:46 ip-172-31-20-123 systemd[1]: Started Hadoop namenode.
Hint: Some lines were ellipsized, use -l to show in full.
```

EMR 4.x - 5.29.0

Example : Listet alle laufenden Prozesse auf

Das folgende Beispiel listet alle laufenden Prozesse auf.

```
initctl list
```

EMR 2.x - 3.x

Example : Listet alle laufenden Prozesse auf

Das folgende Beispiel listet alle laufenden Prozesse auf.

```
ls /etc/init.d/
```

Beenden und Neustarten von Prozessen

Nachdem Sie bestimmen, welche Prozesse ausgeführt werden, können Sie diese beenden und dann neu starten.

EMR 5.30.0 and 6.0.0 and later

Example : Stoppt einen Prozess

Das folgende Beispiel stoppt den `hadoop-hdfs-namenode`-Prozess.

```
sudo systemctl stop hadoop-hdfs-namenode
```

Sie können `status` abfragen, um zu überprüfen, ob der Prozess gestoppt wurde.

```
sudo systemctl status hadoop-hdfs-namenode
```

Beispielausgabe:

```
hadoop-hdfs-namenode.service - Hadoop namenode
  Loaded: loaded (/etc/systemd/system/hadoop-hdfs-namenode.service; enabled; vendor
 preset: disabled)
  Active: failed (Result: exit-code) since Wed 2021-08-18 21:37:50 UTC; 8s ago
Main PID: 9733 (code=exited, status=143)
```

Example : Startet einen Prozess

Das folgende Beispiel startet den `hadoop-hdfs-namenode`-Prozess.

```
sudo systemctl start hadoop-hdfs-namenode
```

Sie können den Status überprüfen, um sicherzustellen, dass der Prozess ausgeführt wird.

```
sudo systemctl status hadoop-hdfs-namenode
```

Beispielausgabe:

```
hadoop-hdfs-namenode.service - Hadoop namenode
   Loaded: loaded (/etc/systemd/system/hadoop-hdfs-namenode.service; enabled; vendor
 preset: disabled)
   Active: active (running) since Wed 2021-08-18 21:38:24 UTC; 2s ago
  Process: 13748 ExecStart=/etc/init.d/hadoop-hdfs-namenode start (code=exited,
 status=0/SUCCESS)
 Main PID: 13800 (java)
    Tasks: 0
```

```
   Memory: 1.1M
   CGroup: /system.slice/hadoop-hdfs-namenode.service
           # 13800 /etc/alternatives/jre/bin/java -Dproc_namenode -Xmx1843m -server
 -XX:OnOutOfMemoryError=kill -9 %p...
```

EMR 4.x - 5.29.0

Example : Stoppt einen laufenden Prozess

Im folgenden Beispiel wird der `hadoop-hdfs-namenode`-Service gestoppt.

```
sudo stop hadoop-hdfs-namenode
```

Example : Startet einen gestoppten Prozess neu

Im folgenden Beispiel wird der `hadoop-hdfs-namenode`-Service neu gestartet. Sie müssen den `start`-Befehl verwenden und nicht `restart`.

```
sudo start hadoop-hdfs-namenode
```

Example : Überprüfen des Prozessstatus

Im Folgenden wird der Status für `hadoop-hdfs-namenode` abgerufen. Sie können den `status` Befehl verwenden, um zu überprüfen, ob der Prozess gestoppt oder gestartet wurde.

```
sudo status hadoop-hdfs-namenode
```

EMR 2.x - 3.x

Example : Beenden eines Anwendungsprozesses

Im folgenden Beispiel wird der `hadoop-hdfs-namenode` Service beendet, der mit der auf dem Cluster EMR installierten Version von Amazon verknüpft ist.

```
sudo /etc/init.d/hadoop-hdfs-namenode stop
```

Example : Startet einen Anwendungsprozess neu

Geben Sie den folgenden Befehl ein, um den Prozess `hadoop-hdfs-namenode` neu zu starten:

```
sudo /etc/init.d/hadoop-hdfs-namenode start
```

Example : Stoppen Sie einen EMR Amazon-Prozess

Das folgende Beispiel stoppt einen Prozess, wie z. B. instance-controller, der nicht mit der Version von Amazon EMR auf dem Cluster verknüpft ist.

```
sudo /sbin/stop instance-controller
```

Example : Starten Sie einen EMR Amazon-Prozess neu

Im folgenden Beispiel wird ein Prozess neu gestartet, z. B. instance-controller, der nicht mit der Version von Amazon EMR auf dem Cluster verknüpft ist.

```
sudo /sbin/start instance-controller
```

> **Note**
>
> Die Befehle `/sbin/start, stop` und `restart` sind symbolische Links zu `/sbin/initctl`. Weitere Informationen zu `initctl`finden Sie auf der initctl man-Seite. Geben Sie `man initctl` in die Befehlszeile ein.

Häufige Fehler bei Amazon EMR

Manchmal schlagen Cluster fehl oder verarbeiten Daten nur langsam. In den folgenden Abschnitten werden einige häufig auftretende Clusterprobleme mit Vorschlägen zur Behebung dieser Probleme aufgeführt.

Themen

- Streaming-Cluster-Fehler
- Benutzerdefinierte Cluster-Fehler JAR
- AWS GovCloud Fehler (US-West)
- Finden Sie einen fehlenden Cluster

Fehlercodes mit ErrorDetail Informationen

Wenn ein EMR Cluster mit einem Fehler beendet wird, werden ein Fehlercode und eine Fehlermeldung `ListClusters` APIs zurückgegeben. `DescribeCluster` Bei einigen Clusterfehlern kann Ihnen das `ErrorDetail`-Datenarray bei der Behebung des Fehlers helfen.

Fehler, die ein `ErrorDetail`-Array beinhalten, enthalten die folgenden Informationen:

`ErrorCode`

Ein eindeutiger Fehlercode, den Sie für den programmatischen Zugriff verwenden können.

`ErrorData`

Eine Liste von Bezeichnern in Schlüssel-Wert-Paaren, die Sie für die Programmierung oder die manuelle Suche verwenden können. Eine Beschreibung der `ErrorData` Werte, die ein Fehlercode enthält, finden Sie auf der Seite zur Problembehandlung für den Fehlercode.

`ErrorMessage`

Beschreibung des Fehlers mit einem Link zu weiteren Informationen in der EMR Amazon-Dokumentation.

> **Note**
>
> Es wird nicht empfohlen, den Text von `ErrorMessage` zu analysieren, da sich dieser Text ändern kann.

Fehlercodes nach Kategorie

- Fehlercodes für Bootstrap-Fehler
- Interne Fehlercodes
- Fehlercodes für Fehler bei der Validierung

Fehlercodes für Bootstrap-Fehler

Die folgenden Abschnitte enthalten Informationen zur Fehlerbehebung bei Bootstrap-Fehlercodes.

Themen

BOOTSTRAP_FAILURE_PRIMARY_WITH_NON_ZERO_CODE

Übersicht

Wenn ein Cluster mit einem `BOOTSTRAP_FAILURE_PRIMARY_WITH_NON_ZERO_CODE`-Fehler beendet wird, ist eine Bootstrap-Aktion in der primären Instance fehlgeschlagen. Weitere Informationen zu Bootstrap-Aktionen finden Sie unter Erstellen von Bootstrap-Aktionen zur Installation zusätzlicher Software.

Auflösung

Um diesen Fehler zu beheben, überprüfen Sie die im API Fehler zurückgegebenen Details, ändern Sie Ihr Bootstrap-Aktionsskript und erstellen Sie einen neuen Cluster mit der aktualisierten Bootstrap-Aktion.

Informationen zur Behebung des ausgefallenen EMR Clusters finden Sie in den `ErrorDetail` Informationen, die `DescribeCluster` von und zurückgegeben wurden. `ListClusters` APIs Weitere Informationen finden Sie unter Fehlercodes mit ErrorDetail Informationen. Das in `ErrorDetail` enthaltene `ErrorData`-Array gibt die folgenden Informationen für diesen Fehlercode zurück:

`primary-instance-id`

Die ID der primären Instance, bei der die Bootstrap-Aktion fehlgeschlagen ist.

`bootstrap-action`

Die Ordinalzahl für die fehlgeschlagene Bootstrap-Aktion. Ein Skript mit dem `bootstrap-action`-Wert von `1` ist die erste Bootstrap-Aktion, die auf der Instance ausgeführt wird.

`return-code`

Der Rückgabecode für die fehlgeschlagene Bootstrap-Aktion.

`amazon-s3-path`

Der Amazon-S3-Speicherort der Bootstrap-Aktion, die fehlgeschlagen ist.

`public-doc`

Die Öffentlichkeit URL der Dokumentation für den Fehlercode.

Schritte zum Absolvieren

Gehen Sie wie folgt vor, um die Ursache des Bootstrap-Aktionsfehlers zu ermitteln und zu beheben. Starten Sie dann einen neuen Cluster.

1. Überprüfen Sie die Bootstrap-Aktionsprotokolldateien in Amazon S3, um die Hauptursache für den Fehler zu ermitteln. Weitere Informationen zum Anzeigen von EMR Amazon-Protokollen finden Sie unterAnzeige von -Protokolldateien.
2. Wenn Sie bei der Erstellung der Instance Cluster-Protokolle aktiviert haben, finden Sie weitere Informationen im `stdout`-Protokoll. Sie finden das `stdout`-Protokoll für die Bootstrap-Aktion an diesem Amazon-S3-Speicherort:

   ```
   s3://EXAMPLE-BUCKET/logs/Your_Cluster_Id/node/Primary_Instance_Id/bootstrap-
   actions/Failed_Bootstrap_Action_Number/stdout.gz
   ```

   Weitere Informationen zu Clusterprotokollen finden Sie im Abschnitt Konfigurieren der Cluster-Protokollierung und des Debuggings.
3. Um festzustellen, ob die Bootstrap-Aktion fehlgeschlagen ist, überprüfen Sie die Ausnahmen in den `stdout`-Protokollen und den `return-code`-Wert in `ErrorData`.
4. Verwenden Sie Ihre Ergebnisse aus dem vorherigen Schritt, um Ihre Bootstrap-Aktion so zu überarbeiten, dass Ausnahmen vermieden werden oder Ausnahmen ordnungsgemäß behandelt werden können, wenn sie auftreten.
5. Starten Sie einen neuen Cluster mit Ihrer aktualisierten Bootstrap-Aktion.

BOOTSTRAP_ FAILURE _BA_ _ _ DOWNLOAD FAILED PRIMARY

Übersicht

Ein Cluster wird mit dem `BOOTSTRAP_FAILURE_BA_DOWNLOAD_FAILED_PRIMARY`-Fehler beendet, wenn die primäre Instance kein Bootstrap-Aktionsskript von dem von Ihnen angegebenen Amazon-S3-Speicherort herunterladen kann. Zu den potentiellen Ursachen zählen auch die Folgenden:

- Die Bootstrap-Aktionsskriptdatei befindet sich nicht am angegebenen Amazon-S3-Speicherort.
- Die Servicerolle für EC2 Amazon-Instances auf dem Cluster (auch EC2Instance-Profil für Amazon genanntEMR) hat keine Berechtigungen für den Zugriff auf den Amazon S3-Bucket, in dem sich das Bootstrap-Aktionsskript befindet. Weitere Informationen zu Servicerollen finden Sie unter Servicerolle für EC2 Cluster-Instances (EC2Instance-Profil).

Weitere Informationen zu Bootstrap-Aktionen finden Sie unter Erstellen von Bootstrap-Aktionen zur Installation zusätzlicher Software.

Auflösung

Um diesen Fehler zu beheben, stellen Sie sicher, dass Ihre primäre Instance über angemessenen Zugriff auf das Bootstrap-Aktionsskript verfügt.

Informationen zur Behebung des ausgefallenen EMR Clusters finden Sie in den `ErrorDetail` Informationen, die `DescribeCluster` von und zurückgegeben wurden. `ListClusters` APIs Weitere Informationen finden Sie unter Fehlercodes mit ErrorDetail Informationen. Das in `ErrorDetail` enthaltene `ErrorData`-Array gibt die folgenden Informationen für diesen Fehlercode zurück:

`primary-instance-id`

Die ID der primären Instance, bei der die Bootstrap-Aktion fehlgeschlagen ist.

`bootstrap-action`

Die Ordinalzahl für die fehlgeschlagene Bootstrap-Aktion. Ein Skript mit dem `bootstrap-action`-Wert von `1` ist die erste Bootstrap-Aktion, die auf der Instance ausgeführt wird.

`amazon-s3-path`

Der Amazon-S3-Speicherort der Bootstrap-Aktion, die fehlgeschlagen ist.

`public-doc`

Die Öffentlichkeit URL der Dokumentation für den Fehlercode.

Schritte zum Absolvieren

Gehen Sie wie folgt vor, um die Ursache des Bootstrap-Aktionsfehlers zu ermitteln und zu beheben. Starten Sie dann einen neuen Cluster.

Fehlerbehebungsschritte

1. Verwenden Sie den `amazon-s3-path`-Wert aus dem `ErrorData`-Array, um das entsprechende Bootstrap-Aktionsskript in Amazon S3 zu finden.
2. Wenn Sie bei der Erstellung der Instance Cluster-Protokolle aktiviert haben, finden Sie weitere Informationen im `stdout`-Protokoll. Sie finden das `stdout`-Protokoll für die Bootstrap-Aktion an diesem Amazon-S3-Speicherort:

   ```
   s3://EXAMPLE-BUCKET/logs/Your_Cluster_Id/node/Primary_Instance_Id/bootstrap-actions/Failed_Bootstrap_Action_Number/stdout.gz
   ```

   Weitere Informationen zu Clusterprotokollen finden Sie im Abschnitt Konfigurieren der Cluster-Protokollierung und des Debuggings.
3. Um festzustellen, ob die Bootstrap-Aktion fehlgeschlagen ist, überprüfen Sie die Ausnahmen in den `stdout`-Protokollen und den `return-code`-Wert in `ErrorData`.
4. Verwenden Sie Ihre Ergebnisse aus dem vorherigen Schritt, um Ihre Bootstrap-Aktion so zu überarbeiten, dass Ausnahmen vermieden werden oder Ausnahmen ordnungsgemäß behandelt werden können, wenn sie auftreten.
5. Starten Sie einen neuen Cluster mit Ihrer aktualisierten Bootstrap-Aktion.

BOOTSTRAP_FAILURE_FILE_NOT_FOUND_PRIMARY

Übersicht

Der `BOOTSTRAP_FAILURE_FILE_NOT_FOUND_PRIMARY`-Fehler weist darauf hin, dass die primäre Instance das Bootstrap-Aktionsskript nicht finden kann, das die Instance gerade aus dem angegebenen Amazon-S3-Bucket heruntergeladen hat.

Auflösung

Um diesen Fehler zu beheben, stellen Sie sicher, dass Ihre primäre Instance über angemessenen Zugriff auf das Bootstrap-Aktionsskript verfügt.

Informationen zur Behebung des ausgefallenen EMR Clusters finden Sie in den `ErrorDetail` Informationen, die von `DescribeCluster` und zurückgegeben wurden `ListClusters`APIs. Weitere Informationen finden Sie unter Fehlercodes mit ErrorDetail Informationen. Das in `ErrorDetail` enthaltene `ErrorData`-Array gibt die folgenden Informationen für diesen Fehlercode zurück:

`primary-instance-id`

Die ID der primären Instance, bei der die Bootstrap-Aktion fehlgeschlagen ist.

`bootstrap-action`

Die Ordinalzahl für die fehlgeschlagene Bootstrap-Aktion. Ein Skript mit dem `bootstrap-action`-Wert von 1 ist die erste Bootstrap-Aktion, die auf der Instance ausgeführt wird.

`amazon-s3-path`

Der Amazon-S3-Speicherort der Bootstrap-Aktion, die fehlgeschlagen ist.

`public-doc`

Die Öffentlichkeit URL der Dokumentation für den Fehlercode.

Schritte zum Absolvieren

Gehen Sie wie folgt vor, um die Ursache des Bootstrap-Aktionsfehlers zu ermitteln und zu beheben. Starten Sie dann einen neuen Cluster.

1. Verwenden Sie den `amazon-s3-path`-Wert aus dem `ErrorData`-Array, um das entsprechende Bootstrap-Aktionsskript in Amazon S3 zu finden.
2. Überprüfen Sie die Bootstrap-Aktionsprotokolldateien in Amazon S3, um die Hauptursache für den Fehler zu ermitteln. Weitere Informationen zum Anzeigen von EMR Amazon-Protokollen finden Sie unterAnzeige von -Protokolldateien.

   > **Note**
   >
   > Wenn Sie die Protokolle für Ihren Cluster nicht aktiviert haben, müssen Sie einen neuen Cluster mit denselben Konfigurationen und Bootstrap-Aktionen erstellen. Informationen dazu, wie Sie sicherstellen können, dass die Clusterprotokolle aktiviert sind, finden Sie unter Konfigurieren der Cluster-Protokollierung und des Debuggings.

3. Überprüfen Sie das `stdout`-Protokoll auf Ihre Bootstrap-Aktionen und stellen Sie sicher, dass es keine benutzerdefinierten Prozesse gibt, die Dateien im `/emr/instance-controller/lib/bootstrap-actions`-Ordner auf Ihren primären Instances löschen. Sie finden das `stdout`-Protokoll für die Bootstrap-Aktion an diesem Amazon-S3-Speicherort:

```
s3://EXAMPLE-BUCKET/logs/Your_Cluster_Id/node/Primary_Instance_Id/bootstrap-
actions/Failed_Bootstrap_Action_Number/stdout.gz
```

4. Starten Sie einen neuen Cluster mit Ihrer aktualisierten Bootstrap-Aktion.

Interne Fehlercodes

Die folgenden Abschnitte enthalten Informationen zur Fehlerbehebung bei internen Fehlercodes.

Themen

- INTERNAL_ ERROR _ _ EC2 INSUFFICIENT _ CAPACITY _AZ
- INTERNAL_ERROR_SPOT_PRICE_INCREASE_PRIMARY
- INTERNAL_ ERROR _ SPOT CAPACITY _NEIN_ PRIMARY

INTERNAL_ ERROR _ _ EC2 INSUFFICIENT _ CAPACITY _AZ

Übersicht

Ein Cluster wird mit einem `INTERNAL_ERROR_EC2_INSUFFICIENT_CAPACITY_AZ` Fehler beendet, wenn die ausgewählte Availability Zone nicht über genügend Kapazität verfügt, um Ihre Anfrage vom EC2 Amazon-Instance-Typ zu erfüllen. Die Availability Zone ist von dem von Ihnen für einen Cluster ausgewählten Subnetz abhängig. Weitere Informationen zu Subnetzen für Amazon finden Sie EMR unterNetzwerk konfigurieren.

Auflösung

Um diesen Fehler zu beheben, ändern Sie Ihre Instance-Typ-Konfigurationen und erstellen Sie einen neuen Cluster mit Ihrer aktualisierten Anfrage.

Informationen zur Behebung des ausgefallenen EMR Clusters finden Sie in den `ErrorDetail` Informationen, die von `DescribeCluster` und `ListClusters` APIs zurückgegeben wurden. Weitere Informationen finden Sie unter Fehlercodes mit ErrorDetail Informationen. Das in `ErrorDetail` enthaltene `ErrorData`-Array gibt die folgenden Informationen für diesen Fehlercode zurück:

`instance-type`

Der Instance-Typ, dessen Kapazität aufgebraucht ist.

`availability-zone`

Die Availability Zone, in die Ihr Subnetz aufgelöst wird.

`public-doc`

Die Öffentlichkeit URL der Dokumentation für den Fehlercode.

Schritte zum Absolvieren

Gehen Sie wie folgt vor, um die Ursache des Cluster Konfigurationsfehlers zu ermitteln und zu beheben:

- Für andere Clusterkonfiguration lesen Sie die bewährten Methoden. Weitere Informationen finden Sie Bewährte Methoden für die Konfiguration des Clusters im Amazon EMR Management Guide.
- Beheben Sie die Startprobleme und überprüfen Sie Ihre Konfiguration. Weitere Informationen finden Sie unter Problembehandlung beim Starten von Instances im EC2 Amazon-Benutzerhandbuch.
- Starten Sie einen neuen Cluster mit Ihrer aktualisierten Cluster-Konfiguration.

INTERNAL_ERROR_SPOT_PRICE_INCREASE_PRIMARY

Übersicht

Ein Cluster wird mit einem `INTERNAL_ERROR_SPOT_PRICE_INCREASE_PRIMARY` Fehler beendet, wenn Amazon Ihre Spot-Instance-Anfrage für den primären Knoten nicht erfüllen EMR kann, weil Instances nicht zu oder unter Ihrem maximalen Spot-Preis verfügbar sind. Weitere Informationen finden Sie unter Spot-Instances im EC2Amazon-Benutzerhandbuch.

Auflösung

Um diesen Fehler zu beheben, geben Sie Instance-Typen für Ihren Cluster an, die innerhalb Ihres Preisziels liegen, oder erhöhen Sie Ihr Preislimit für denselben Instance-Typ.

Informationen zur Behebung des ausgefallenen EMR Clusters finden Sie in den `ErrorDetail` Informationen, die von `DescribeCluster` und zurückgegeben wurden `ListClusters`APIs. Weitere Informationen finden Sie unter Fehlercodes mit ErrorDetail Informationen. Das in `ErrorDetail` enthaltene `ErrorData`-Array gibt die folgenden Informationen für diesen Fehlercode zurück:

`primary-instance-id`

Die ID für die primäre Instance des Clusters, die fehlgeschlagen ist.

`instance-type`

Der Instance-Typ, dessen Kapazität aufgebraucht ist.

`availability-zone`

Die Availability Zone, in der sich Ihr Subnetz befindet.

`public-doc`

Die Öffentlichkeit URL der Dokumentation für den Fehlercode.

Schritte zum Absolvieren

Führen Sie die folgenden Schritte aus, um Probleme mit Ihrer Cluster-Konfigurationsstrategie zu beheben, und starten Sie dann einen neuen Cluster:

1. Lesen Sie die Best Practices für Amazon EC2 Spot-Instances und überprüfen Sie Ihre Cluster-Konfigurationsstrategie. Weitere Informationen finden Sie unter Bewährte Methoden für EC2 Spot im EC2Amazon-Benutzerhandbuch undBewährte Methoden für die Konfiguration des Clusters.
2. Um diesen Fehler zu beheben, ändern Sie Ihre Instance-Typ-Konfigurationen oder Availability Zone und erstellen Sie einen neuen Cluster mit Ihrer aktualisierten Anfrage.
3. Wenn das Problem weiterhin besteht, verwenden Sie On-Demand-Kapazität für Ihre primäre Instance.

INTERNAL_ ERROR _ SPOT CAPACITY _NEIN_ PRIMARY

Übersicht

Ein Cluster wird mit einem `INTERNAL_ERROR_SPOT_NO_CAPACITY_PRIMARY`-Fehler beendet, wenn nicht genügend Kapazität vorhanden ist, um eine Spot-Instance-Anfrage für Ihren Primärknoten zu erfüllen. Weitere Informationen finden Sie unter Spot-Instances im EC2Amazon-Benutzerhandbuch.

Auflösung

Um diesen Fehler zu beheben, geben Sie Instance-Typen für Ihren Cluster an, die innerhalb Ihres Preisziels liegen, oder erhöhen Sie Ihr Preislimit für denselben Instance-Typ.

Informationen zur Behebung des ausgefallenen EMR Clusters finden Sie in den `ErrorDetail` Informationen, die von `DescribeCluster` und zurückgegeben wurden `ListClusters`APIs. Weitere Informationen finden Sie unter Fehlercodes mit ErrorDetail Informationen. Das in `ErrorDetail` enthaltene `ErrorData`-Array gibt die folgenden Informationen für diesen Fehlercode zurück:

`primary-instance-id`

Die ID für die primäre Instance des Clusters, die fehlgeschlagen ist.

`instance-type`

Der Instance-Typ, dessen Kapazität aufgebraucht ist.

`availability-zone`

Die Availability Zone, in die Ihr Subnetz aufgelöst wird.

`public-doc`

Die Öffentlichkeit URL der Dokumentation für den Fehlercode.

Schritte zum Absolvieren

Führen Sie die folgenden Schritte aus, um Probleme mit Ihrer Cluster-Konfigurationsstrategie zu beheben, und starten Sie dann einen neuen Cluster:

1. Lesen Sie die Best Practices für Amazon EC2 Spot-Instances und überprüfen Sie Ihre Cluster-Konfigurationsstrategie. Weitere Informationen finden Sie unter Bewährte Methoden für EC2 Spot im EC2Amazon-Benutzerhandbuch undBewährte Methoden für die Konfiguration des Clusters.
2. Ändern Sie Ihre Instance-Typ-Konfigurationen und erstellen Sie einen neuen Cluster mit Ihrer aktualisierten Anfrage.
3. Wenn das Problem weiterhin besteht, verwenden Sie On-Demand-Kapazität für Ihre primäre Instance.

Fehlercodes für Fehler bei der Validierung

Die folgenden Abschnitte enthalten Informationen zur Fehlerbehebung bei Validierung-Fehlercodes.

Themen

VALIDATION_ERROR_SUBNET_NOT_FROM_ONE_VPC

Übersicht

Wenn Ihr Cluster und die Subnetze, auf die Sie für Ihren Cluster verweisen, zu verschiedenen virtuellen privaten Clouds (VPCs) gehören, wird der Cluster mit einem Fehler beendet. `VALIDATION_ERROR_SUBNET_NOT_FROM_ONE_VPC` Sie können Cluster mit Amazon EMR mit der Konfiguration der Instance-Flotten über Subnetze in a starten. VPC Weitere Informationen zu Instance-Flotten finden Sie Instance-Flotten konfigurieren im Amazon EMR Management Guide.

Auflösung

Um diesen Fehler zu beheben, verwenden Sie Subnetze, die zu demselben Cluster VPC gehören.

Informationen zur Behebung des ausgefallenen EMR Clusters finden Sie in den `ErrorDetail` Informationen, die von `DescribeCluster` und `ListClusters` APIs zurückgegeben wurden. Weitere Informationen finden Sie unter Fehlercodes mit ErrorDetail Informationen. Das in `ErrorDetail` enthaltene `ErrorData`-Array gibt die folgenden Informationen für diesen Fehlercode zurück:

`vpc`

Für jedes VPC Subnetz-Paar die ID des Subnetzes, zu dem VPC das Subnetz gehört.

`subnet`

Für jedes VPC Subnetz-Paar die ID für das Subnetz.

`public-doc`

Die Öffentlichkeit URL der Dokumentation für den Fehlercode.

Schritte zum Absolvieren

Führen Sie die folgenden Schritte aus, um den Fehler zu identifizieren und zu beheben:

1. Überprüfen Sie die SubnetzeIDs, die im `ErrorData` Array aufgeführt sind, und stellen Sie sicher, dass sie zu dem Subnetz gehörenVPC, in dem Sie den EMR Cluster starten möchten.
2. Ändern Sie Ihre Subnetzkonfigurationen. Sie können eine der folgenden Methoden verwenden, um alle verfügbaren öffentlichen und privaten Subnetze in a zu finden. VPC

   - Navigieren Sie zur VPC Amazon-Konsole. Wählen Sie Subnetze und listen Sie alle Subnetze auf, die sich in Ihrem Cluster befinden. AWS-Region Um nur öffentliche oder private Subnetze zu finden, wenden Sie den Filter Öffentliche Adresse automatisch zuweisen an. IPv4 Verwenden Sie die Option Filtern nach, um Subnetze in den Subnetzen zu finden und auszuwählen, VPC die Ihr Cluster verwendet. VPC Weitere Informationen zum Erstellen von Subnetzen finden Sie unter Erstellen eines Subnetzes im Benutzerhandbuch von Amazon Virtual Private Cloud.
   - Verwenden Sie die AWS CLI , um alle verfügbaren öffentlichen und privaten Subnetze in dem zu findenVPC, das Ihr Cluster verwendet. Weitere Informationen finden Sie unter APIDescribe-Subnetze. Informationen zum Erstellen neuer Subnetze in einem VPC finden Sie unter Create-Subnet. API
3. Starten Sie einen neuen Cluster mit Subnetzen aus demselben Cluster wie der Cluster. VPC

VALIDATION_ERROR_SECURITY_GROUP_NOT_FROM_ONE_VPC

Übersicht

Wenn Ihr Cluster und die Sicherheitsgruppen, die Sie Ihrem Cluster zuweisen, zu verschiedenen virtuellen privaten Clouds (VPCs) gehören, wird der Cluster mit einem `VALIDATION_ERROR_SECURITY_GROUP_NOT_FROM_ONE_VPC` Fehler beendet. Weitere Informationen zu Sicherheitsgruppen finden Sie unter Angabe von von EMR Amazon verwalteten und zusätzlichen Sicherheitsgruppen und Steuerung des Netzwerkverkehrs mit Sicherheitsgruppen.

Auflösung

Um diesen Fehler zu beheben, verwenden Sie Sicherheitsgruppen, die zu demselben VPC Cluster gehören.

Informationen zur Behebung des ausgefallenen EMR Clusters finden Sie in den `ErrorDetail` Informationen, die von `DescribeCluster` und zurückgegeben wurden `ListClusters`APIs.

Weitere Informationen finden Sie unter Fehlercodes mit ErrorDetail Informationen. Das in `ErrorDetail` enthaltene `ErrorData`-Array gibt die folgenden Informationen für diesen Fehlercode zurück:

`vpc`

Für jedes VPC Sicherheitsgruppenpaar: die ID der Sicherheitsgruppe, zu der VPC die Sicherheitsgruppe gehört.

`security-group`

Für jede Sicherheitsgruppe: VPC Paar, die ID für die Sicherheitsgruppe.

`public-doc`

Die Öffentlichkeit URL der Dokumentation für den Fehlercode.

Schritte zum Absolvieren

Führen Sie die folgenden Schritte aus, um den Fehler zu identifizieren und zu beheben:

1. Überprüfen Sie die SicherheitsgruppeIDs, die im `ErrorData` Array aufgeführt ist, und stellen Sie sicher, dass sie zu der Gruppe gehörenVPC, in der Sie den EMR Cluster starten möchten.
2. Navigieren Sie zur VPC Amazon-Konsole. Wählen Sie Sicherheitsgruppen aus, um alle Sicherheitsgruppen in der ausgewählten Region aufzulisten. Suchen Sie die Sicherheitsgruppen aus demselben Cluster VPC wie Ihr Cluster und ändern Sie dann Ihre Sicherheitsgruppenkonfiguration.
3. Starten Sie einen neuen Cluster mit Sicherheitsgruppen aus demselben Cluster VPC wie der Cluster.

VALIDATION_ERROR_INVALID_SSH_KEY_NAME

Übersicht

Ein Cluster wird mit einem `VALIDATION_ERROR_INVALID_SSH_KEY_NAME` Fehler beendet, wenn Sie ein EC2 Amazon-Schlüsselpaar verwenden, das für SSH die primäre Instance nicht gültig ist. Der Name des Schlüsselpaars ist möglicherweise falsch, oder das key pair ist in der angeforderten Datei nicht vorhanden AWS-Region. Weitere Informationen zu Schlüsselpaaren finden Sie unter EC2Amazon-Schlüsselpaare und Linux-Instances im EC2Amazon-Benutzerhandbuch.

Auflösung

Um diesen Fehler zu beheben, erstellen Sie einen neuen Cluster mit einem gültigen SSH Schlüsselpaarnamen.

Informationen zur Behebung des ausgefallenen EMR Clusters finden Sie in den `ErrorDetail` Informationen, die von `DescribeCluster` und zurückgegeben wurden `ListClusters`APIs. Weitere Informationen finden Sie unter Fehlercodes mit ErrorDetail Informationen. Das in `ErrorDetail` enthaltene `ErrorData`-Array gibt die folgenden Informationen für diesen Fehlercode zurück:

`ssh-key`

Der Name des SSH key pair, den Sie bei der Erstellung des Clusters angegeben haben.

`public-doc`

Die Öffentlichkeit URL der Dokumentation für den Fehlercode.

Schritte zum Absolvieren

Führen Sie die folgenden Schritte aus, um den Fehler zu identifizieren und zu beheben:

1. Überprüfe deine *`keypair`*.pem-Datei und vergewissern Sie sich, dass sie mit dem Namen des SSH Schlüssels übereinstimmt, den Sie in der EMR Amazon-Konsole sehen.
2. Navigieren Sie zur EC2 Amazon-Konsole. Stellen Sie sicher, dass der von Ihnen verwendete SSH Schlüsselname in dem AWS-Region , den Ihr Cluster verwendet, verfügbar ist. Sie finden Ihre ID AWS-Region neben Ihrer Konto-ID oben im AWS Management Console.
3. Starten Sie einen neuen Cluster mit einem gültigen SSH Schlüsselnamen.

VALIDATION_ERROR_INSTANCE_TYPE_NOT_SUPPORTED

Übersicht

Ein Cluster wird mit einem `VALIDATION_ERROR_INSTANCE_TYPE_NOT_SUPPORTED`-Fehler beendet, wenn die AWS-Region und Availability Zones für Ihren Cluster den angegebenen Instance-Typ für eine oder mehrere Instance-Gruppen nicht unterstützen. Amazon unterstützt EMR möglicherweise einen Instance-Typ in einer Availability Zone innerhalb einer Region, aber nicht in einer anderen. Die Availability Zone innerhalb der Region ist von dem von Ihnen für einen Cluster

ausgewählten Subnetz abhängig. Eine Liste der Instance-Typen und Regionen, die Amazon EMR unterstützt, finden Sie unterUnterstützte Instance-Typen.

Auflösung

Um diesen Fehler zu beheben, geben Sie Instance-Typen für Ihren Cluster an, die Amazon in der Region und Availability Zone EMR unterstützt, in der Sie den Cluster anfordern.

Informationen zur Behebung des ausgefallenen EMR Clusters finden Sie in den `ErrorDetail` Informationen, die von `DescribeCluster` und zurückgegeben wurden `ListClusters`APIs. Weitere Informationen finden Sie unter Fehlercodes mit ErrorDetail Informationen. Das in `ErrorDetail` enthaltene `ErrorData`-Array gibt die folgenden Informationen für diesen Fehlercode zurück:

`instance-types`

Die Liste der nicht unterstützten Instance-Typen.

`availability-zones`

Die Availability Zone Liste, in die Ihr Subnetz aufgelöst wird.

`public-doc`

Die Öffentlichkeit URL der Dokumentation für den Fehlercode.

Schritte zum Absolvieren

Führen Sie die folgenden Schritte aus, um den Fehler zu identifizieren und zu beheben:

1. Verwenden Sie die AWS CLI , um die verfügbaren Instance-Typen in einer Availability Zone abzurufen. Zu diesem Zweck können Sie den `ec2 describe-instance-type-offerings` Befehl verwenden, um verfügbare Instance-Typen nach Standort (AWS-Region oder Availability Zone) zu filtern. Beispielsweise können Sie den folgenden Befehl verwenden, um die Instance-Typen anzuzeigen, die in der angegebenen AZ angeboten werden. *`us-east-2a`*

   ```
   aws ec2 describe-instance-type-offerings --location-type "availability-
   zone" --filters Name=location,Values=us-east-2a --region us-east-2 --query
    "InstanceTypeOfferings[*].[InstanceType]" --output text | sort
   ```

   Weitere Informationen darüber, wie Sie verfügbare Instance-Typen ermitteln können, finden Sie unter Finden Sie einen EC2 Amazon-Instance-Typ.

2. Nachdem Sie die Instance-Typen ermittelt haben, die in derselben Region und Availability Zone wie der Cluster verfügbar sind, wählen Sie eine der folgenden Lösungen, um fortzufahren:

   a. Erstellen Sie einen neuen Cluster und wählen Sie ein Subnetz für den Cluster aus, der sich in einer Availability Zone befindet, in der der von Ihnen ausgewählte Instance-Typ verfügbar ist und von Amazon EMR unterstützt wird.

   b. Erstellen Sie einen neuen Cluster in derselben Region und demselben EC2 Amazon-Subnetz wie der ausgefallene Cluster, jedoch mit einem Instance-Typ, der an diesem Standort von Amazon EMR unterstützt wird.

Eine Liste der Instance-Typen und Regionen, die Amazon EMR unterstützt, finden Sie unterUnterstützte Instance-Typen. Einen Vergleich der Funktionen der Instance-Typen finden Sie unter EC2Amazon-Instance-Typen.

Ressourcenfehler

Die folgenden Fehler werden häufig durch eingeschränkte Ressourcen im Cluster verursacht.

Themen

- Der Cluster endet mit SLAVE NO_ _ und den Kernknoten _BY_ LEFT FAILED MASTER
- Replizieren von Block nicht möglich, nur Replizieren auf null Knoten möglich.
- EC2 QUOTA EXCEEDED
- Zu viele Abruffehler
- Datei konnte nur auf 0 Knoten anstatt auf 1 repliziert werden
- Knoten, die auf der Liste stehen
- Drosselungsfehler
- Instance-Typ nicht unterstützt
- EC2hat keine Kapazität
- HDFSFehler beim Replikationsfaktor
- HDFSFehler bei unzureichendem Speicherplatz

Der Cluster endet mit SLAVE NO_ _ und den Kernknoten _BY_ LEFT FAILED MASTER

Dies passiert in der Regel, da der Beendigungsschutz deaktiviert ist, und alle Core-Knoten überschreiten die Datenträger-Speicherkapazität, die durch einen Schwellenwert für die maximale Auslastung in der `yarn-site`-Konfigurationsklassifizierung angegeben ist, die der `yarn-site.xml`-Datei entspricht. Dieser Wert liegt standardmäßig bei 90 %. Wenn die Festplattenauslastung für einen Kernknoten den Auslastungsschwellenwert überschreitet, meldet der YARN NodeManager Health Service den Knoten als. `UNHEALTHY` In diesem Zustand listet Amazon EMR Deny den Knoten auf und weist ihm keine YARN Container zu. Wenn der Knoten 45 Minuten lang fehlerhaft bleibt, EMR markiert Amazon die zugehörige EC2 Amazon-Instance zur Kündigung als`FAILED_BY_MASTER`. Wenn alle EC2 Amazon-Instances, die mit Kernknoten verknüpft sind, für die Kündigung markiert sind, wird der Cluster mit dem Status beendet, `NO_SLAVE_LEFT` da keine Ressourcen zur Ausführung von Jobs vorhanden sind.

Das Überschreiten der Datenträgernutzung auf einem Core-Knoten könnte eine Kettenreaktion auslösen. Wenn ein einzelner Knoten den Schwellenwert für die Festplattenauslastung aus diesem Grund überschreitetHDFS, befinden sich wahrscheinlich auch andere Knoten in der Nähe des Schwellenwerts. Der erste Knoten überschreitet den Schwellenwert für die Festplattenauslastung, weshalb Amazon EMR Deny ihn auflistet. Dies erhöht die Belastung der verbleibenden Knoten durch die Festplattenauslastung, da sie damit beginnen, HDFS Daten, die sie auf dem Knoten auf der Sperrliste verloren haben, untereinander zu replizieren. Jeder Knoten wird anschließend auf die gleiche Weise in den Zustand `UNHEALTHY` versetzt und der Cluster wird schließlich beendet.

Bewährte Methoden und Empfehlungen

Konfigurieren von Cluster-Hardware mit ausreichend Speicher

Wenn Sie einen Cluster erstellen, stellen Sie sicher, dass genügend Kernknoten vorhanden sind und dass jeder über einen geeigneten Instance-Speicher und EBS Speichervolumes für verfügt. HDFS Weitere Informationen finden Sie unter Berechnung der erforderlichen HDFS Kapazität eines Clusters. Sie können auch Core-Instances manuell oder mithilfe der automatischen Skalierung zu vorhandenen Instance-Gruppen hinzuzufügen. Die neuen Instances haben dieselbe Speicherkonfiguration wie andere Instances in der Instance-Gruppe. Weitere Informationen finden Sie unter Clusterskalierung verwenden.

Aktivieren des Beendigungsschutzes

Beendigungsschutz aktivieren. Auf diese Weise können Sie, wenn ein Core-Node auf der Deny-Liste steht, eine Verbindung zu der zugehörigen EC2 Amazon-Instance herstellenSSH, die zur Fehlerbehebung und Wiederherstellung von Daten verwendet wird. Wenn Sie den Kündigungsschutz aktivieren, beachten Sie, dass Amazon die EC2 Amazon-Instance EMR nicht durch eine neue Instance ersetzt. Weitere Informationen finden Sie unter Verwenden des Beendigungsschutzes.

Erstellen Sie einen Alarm für die MRUnhealthyNodes CloudWatch Metrik

Diese Metrik meldet die Anzahl der Knoten mit dem Status `UNHEALTHY`. Es entspricht der YARN Metrik`mapred.resourcemanager.NoOfUnhealthyNodes`. Sie können eine Benachrichtigung für diesen Alarm einrichten, um über fehlerhafte Knoten informiert zu werden, bevor der 45-Minuten-Timeout erreicht ist. Weitere Informationen finden Sie unter Überwachung von EMR Amazon-Metriken mit CloudWatch.

Anpassen von Einstellungen mit yarn-site

Die folgenden Einstellungen können an Ihre Anwendungsanforderungen angepasst werden. Beispiel: Sie möchten den Schwellenwert für die Datenträgernutzung erhöhen, bei dem ein Knoten `UNHEALTHY` melden, indem Sie den Wert von `yarn.nodemanager.disk-health-checker.max-disk-utilization-per-disk-percentage` erhöhen.

Sie können diese Werte festlegen, wenn Sie einen Cluster mithilfe der `yarn-site`-Konfigurationsklassifizierung erstellen. Weitere Informationen finden Sie unter Konfiguration von Anwendungen im EMRAmazon-Versionshandbuch. Sie können auch mithilfe eines Texteditors eine Verbindung zu den EC2 Amazon-Instances herstellenSSH, die mit den Kernknoten verknüpft sind, und dann die Werte hinzufügen. `/etc/hadoop/conf.empty/yarn-site.xml` Nachdem Sie die Änderung vorgenommen haben, müssen Sie hadoop-yarn-nodemanager wie unten gezeigt neu starten.

> **Important**
>
> Wenn Sie den NodeManager Dienst neu starten, werden aktive YARN Container beendet, es `yarn.nodemanager.recovery.enabled` sei denn, Sie haben bei der Erstellung des Clusters die `true` Verwendung der `yarn-site` Konfigurationsklassifizierung festgelegt. Darüber hinaus müssen Sie über die Eigenschaft `yarn.nodemanager.recovery.dir` das Verzeichnis angeben, in dem der Containerstatus gespeichert werden soll.

```
sudo /sbin/stop hadoop-yarn-nodemanager
sudo /sbin/start hadoop-yarn-nodemanager
```

Weitere Informationen zu aktuellen `yarn-site` Eigenschaften und Standardwerten finden Sie unter YARNStandardeinstellungen in der Apache Hadoop-Dokumentation.

Eigenschaft	Standardwert	Beschreibung
yarn.nodemanager. disk-heal th-checker.interval-ms	120000	Die Häufigkeit (in Sekunden), mit der die Datenträger-Zustan dsprüfung ausgeführt wird.
garn.nodemanager. disk-heal th-checker. min-healthy-disks	0,25	Der Mindestanteil der Anzahl der Festplatten, die fehlerfre i sein müssen, NodeManag er damit neue Container gestartet werden können. Dies entspricht sowohl yarn.node manager.local-dirs (standard mäßig in Amazon) als auch yarn.nodemanager.log-dirs (standardmäßig, mit dem `/ mnt/yarn` in Amazon ein Symlink verknüpft istEMR). `/ var/log/hadoop-yarn/ containers mnt/var/l og/hadoop-yarn/con tainers` EMR
`yarn.nodemanager.d isk-health-checker .max-disk-utilizat ion-per-disk-perce ntage`	90.0	Der maximale Prozentsatz der zulässigen Speicherplatzausla stung, ab der ein Datenträg er als fehlerhaft markiert wird. Die Werte können zwischen 0,0 und 100,0 liegen. Wenn der Wert größer oder gleich 100 ist, wird

Eigenschaft	Standardwert	Beschreibung
		geprüft, ob eine volle Festplatte vorhanden ist. NodeManager Dies gilt für `yarn-nodemanager.local-dirs` und `yarn.nodemanager.log-dirs` .
`yarn.nodemanager.disk-health-checker.min-free-space-per-disk-mb`	0	Der mindestens erforderliche verfügbare Speicherplatz, damit ein Datenträger verwendet werden kann. Dies gilt für `yarn-nodemanager.local-dirs` und `yarn.nodemanager.log-dirs` .

Replizieren von Block nicht möglich, nur Replizieren auf null Knoten möglich.

Der Fehler „Replizieren von Block nicht möglich, nur Replizieren auf null Knoten möglich“ tritt in der Regel auf, wenn ein Cluster nicht über genügend HDFS Speicherplatz verfügt. Dieser Fehler tritt auf, wenn Sie in Ihrem Cluster mehr Daten generieren, als darin gespeichert werden könnenHDFS. Dieser Fehler wird nur angezeigt, wenn der Cluster ausgeführt wird, denn wenn der Job endet, gibt er den HDFS Speicherplatz frei, den er belegt hat.

Der für einen Cluster verfügbare HDFS Speicherplatz hängt von der Anzahl und dem Typ der EC2 Amazon-Instances ab, die als Kernknoten verwendet werden. Task-Knoten werden nicht für die HDFS Speicherung verwendet. Der gesamte Speicherplatz auf jeder EC2 Amazon-Instance, einschließlich der angehängten EBS Speichervolumes, steht für zur VerfügungHDFS. Weitere Informationen zur Größe des lokalen Speichers für jeden EC2 Instance-Typ finden Sie unter Instance-Typen und -Familien im EC2Amazon-Benutzerhandbuch.

Der andere Faktor, der sich auf den verfügbaren HDFS Speicherplatz auswirken kann, ist der Replikationsfaktor. Dabei handelt es sich um die Anzahl der Kopien jedes Datenblocks, die aus HDFS Redundanzgründen gespeichert werden. Der Replikationsfaktor steigt mit der Anzahl der Knoten im Cluster: Es gibt 3 Kopien jedes Datenblocks für einen Cluster mit 10 oder mehr Knoten, 2 Kopien jedes Blocks für einen Cluster mit 4 bis 9 Knoten und 1 Kopie (keine Redundanz) für

Cluster mit 3 oder weniger Knoten. Der insgesamt verfügbare HDFS Speicherplatz wird durch den Replikationsfaktor geteilt. In einigen Fällen, z. B. wenn die Anzahl der Knoten von 9 auf 10 erhöht wird, kann die Erhöhung des Replikationsfaktors sogar dazu führen, dass der verfügbare HDFS Speicherplatz sinkt.

In einem Cluster mit zehn Kernknoten des Typs m1.large stehen beispielsweise 2833 GB Speicherplatz zur Verfügung HDFS ((10 Knoten X 850 GB pro Knoten) /Replikationsfaktor 3).

Wenn Ihr Cluster den verfügbaren Speicherplatz überschreitetHDFS, können Sie Ihrem Cluster zusätzliche Kernknoten hinzufügen oder Datenkomprimierung verwenden, um mehr Speicherplatz zu schaffen. HDFS Wenn es sich bei Ihrem Cluster um einen Cluster handelt, der gestoppt und neu gestartet werden kann, sollten Sie die Verwendung von Kernknoten eines größeren EC2 Amazon-Instance-Typs in Betracht ziehen. Sie können auch den Replikationsfaktor anpassen. Beachten Sie jedoch, dass eine Verringerung des Replikationsfaktors die Redundanz der HDFS Daten und die Fähigkeit Ihres Clusters, sich nach verlorenen oder beschädigten HDFS Blöcken wiederherzustellen, verringert.

EC2 QUOTA EXCEEDED

Wenn Sie die Meldung `EC2 QUOTA EXCEEDED` erhalten, gibt es möglicherweise mehrere Ursachen. Je nach Konfigurationsunterschieden kann es zwischen 5 und 20 Minuten dauern, bis vorherige Cluster beendet und die entsprechenden Ressourcen wieder freigegeben werden. Wenn Sie beim Versuch, einen Custer zu starten, die Fehlermeldung `EC2 QUOTA EXCEEDED` erhalten, kann es daran liegen, dass Ressourcen eines kürzlich beendeten Clusters noch nicht zur Verfügung stehen. Diese Meldung kann auch durch die Größenanpassung einer Instance-Gruppe oder Instance-Flotte an eine Zielgröße, die das aktuelle Instance-Kontingent für das Konto überschreitet, verursacht werden. Dies kann manuell oder automatisch durch Auto Scaling geschehen.

Sie können das Problem u. U. mit den folgenden Optionen beheben:

- Folgen Sie den Anweisungen unter AWS -Service-Quotas in Allgemeine Amazon Web Services-Referenz, um eine Erhöhung des Servicelimits zu beantragen. Für manche APIs ist die Einrichtung einer CloudWatch Veranstaltung möglicherweise eine bessere Option als die Erhöhung der Grenzwerte. Weitere Details finden Sie unter Wann sollten EMR-Ereignisse eingerichtet werden in CloudWatch.
- Wenn einer oder mehrere der aktiven Cluster nicht ausgelastet sind, skalieren Sie Instance-Gruppen oder reduzieren Sie Zielkapazitäten von Instance-Flotten für aktive Cluster.
- Erstellen Sie Cluster mit weniger EC2 Instanzen oder reduzierter Zielkapazität.

Zu viele Abruffehler

Die Fehlermeldung „Too many fetch-failures (Zu viele Abruffehler)“ oder „Error reading task output (Fehler beim Lesen der Aufgabenausgabe)“ in Schritt- oder Aufgabenversuchsprotokollen gibt an, dass die auszuführende Aufgabe von der Ausgabe einer anderen Aufgabe abhängt. Dies geschieht häufig, wenn eine Reduce-Aufgabe zur Ausführung in die Warteschlange gestellt wird und die Ausgabe einer oder mehrerer Map-Aufgaben erfordert, die jedoch noch nicht verfügbar ist.

Es gibt mehrere Gründe, warum die Ausgabe noch nicht verfügbar ist:

- Die erforderliche Aufgabe befindet sich noch in Bearbeitung. Dies ist oft eine Map-Aufgabe.
- Die Daten sind möglicherweise aufgrund einer schlechten Netzwerkverbindung nicht verfügbar, wenn sie sich auf einer anderen Instance befinden.
- Wenn zum Abrufen der Ausgabe verwendet HDFS wird, liegt möglicherweise ein Problem mit vorHDFS.

Der häufigste Grund ist, dass sich die vorherige Aufgabe noch in Bearbeitung befindet. Dies ist besonders wahrscheinlich, wenn die Fehler beim ersten Ausführen der Reduce-Aufgaben auftreten. Sie können prüfen, ob dies der Fall ist, indem Sie sich das Syslog-Protokoll für den Cluster-Schritt ansehen, der den Fehler zurückgibt. Wenn das Syslog den Fortschritt beider Map- und Reduce-Aufgaben belegt, weist dies darauf hin, dass die Reduce-Phase gestartet wurde und einige Map-Aufgaben noch nicht abgeschlossen sind.

Sehen Sie sich in den Protokollen den Prozentsatz für den Map-Fortschritt an, der auf 100 % ansteigt und dann wieder auf einen niedrigeren Wert zurückfällt. Wenn der Map-Prozentsatz 100 % beträgt, bedeutet das nicht, dass alle Map-Aufgaben abgeschlossen sind. Es bedeutet lediglich, dass Hadoop alle Map-Aufgaben ausführt. Wenn dieser Wert unter 100 % fällt, bedeutet dies, dass eine Map-Aufgabe fehlgeschlagen ist und Hadoop je nach Konfiguration versucht, die Aufgabe neu zu planen. Bleibt der Kartenprozentsatz in den Protokollen bei 100%, schauen Sie sich insbesondere die CloudWatch Metriken an`RunningMapTasks`, um zu überprüfen, ob die Kartenaufgabe noch bearbeitet wird. Sie finden diese Informationen auch mithilfe der Hadoop-Weboberfläche auf dem Master-Knoten.

Wenn dieses Problem auftritt, können Sie verschiedene Schritte versuchen:

- Weisen Sie die Reduce-Phase an, länger zu warten, bis sie startet. Ändern Sie dazu die Konfigurationseinstellung mapred.reduce.slowstart.completed.maps in Hadoop und legen Sie sie

auf einen längeren Zeitraum fest. Weitere Informationen finden Sie unter Erstellen von Bootstrap-Aktionen zur Installation zusätzlicher Software.

- Passen Sie die Reducer-Anzahl der gesamten Reducer-Kapazität des Clusters an. Ändern Sie dazu die Konfigurationseinstellung mapred.reduce.tasks für den Auftrag in Hadoop.
- Verwenden Sie einen Kombinationsklassencode zum Minimieren der Anzahl der Ausgaben, die abgerufen werden müssen.
- Vergewissern Sie sich, dass es keine Probleme mit dem EC2 Amazon-Service gibt, die sich auf die Netzwerkleistung des Clusters auswirken. Verwenden Sie dazu das Dashboard zum Servicestatus.
- Überprüfen Sie die Ressourcen CPU und die Speicherressourcen der Instances in Ihrem Cluster, um sicherzustellen, dass Ihre Datenverarbeitung die Ressourcen Ihrer Knoten nicht überlastet. Weitere Informationen finden Sie unter Cluster-Hardware und Netzwerken konfigurieren.
- Überprüfen Sie die Version des Amazon Machine Image (AMI), das in Ihrem EMR Amazon-Cluster verwendet wird. Wenn die Version 2.3.0 bis einschließlich 2.4.4 ist, aktualisieren Sie auf eine neuere Version. AMIVersionen im angegebenen Bereich verwenden eine Version von Jetty, die möglicherweise keine Ausgabe aus der Map-Phase liefert. Der Abruf-Fehler tritt auf, wenn die Reducer keine Ausgabe aus der Map-Phase abrufen können.

  Jetty ist ein HTTP Open-Source-Server, der für die Kommunikation von Maschine zu Maschine innerhalb eines Hadoop-Clusters verwendet wird.

Datei konnte nur auf 0 Knoten anstatt auf 1 repliziert werden

Wenn in eine Datei geschrieben wirdHDFS, wird sie auf mehrere Kernknoten repliziert. Wenn Sie diesen Fehler sehen, bedeutet das, dass der NameNode Daemon keine verfügbaren DataNode Instanzen hat, in die er Daten schreiben kann. HDFS Mit anderen Worten, es findet keine Block-Replikation statt. Dieser Fehler kann durch eine Reihe von Problemen verursacht werden:

- Dem HDFS Dateisystem ist möglicherweise der Speicherplatz ausgegangen. Dies ist die wahrscheinlichste Ursache.
- DataNode Instanzen waren möglicherweise nicht verfügbar, als der Job ausgeführt wurde.
- DataNode Instanzen wurden möglicherweise für die Kommunikation mit dem Master-Knoten gesperrt.
- Instances in der Core-Instance-Gruppe sind möglicherweise nicht verfügbar.
- Berechtigungen können fehlen. Beispielsweise ist der JobTracker Daemon möglicherweise nicht berechtigt, Job-Tracker-Informationen zu erstellen.

- Die Einstellung für den reservierten Speicherplatz für eine DataNode Instanz ist möglicherweise unzureichend. Stellen Sie fest, ob dies der Fall ist, indem Sie die Konfigurationseinstellung dfs.datanode.du.reserved prüfen.

Um zu überprüfen, ob dieses Problem durch HDFS zu wenig Festplattenspeicher verursacht wird, schauen Sie sich die `HDFSUtilization` Metrik unter an CloudWatch. Wenn dieser Wert zu hoch ist, können Sie zusätzliche Core-Knoten zum Cluster hinzufügen. Wenn Sie einen Cluster haben, von dem Sie glauben, dass der HDFS Speicherplatz knapp wird, können Sie einen Alarm einrichten, der Sie benachrichtigt, wenn der Wert `HDFSUtilization` von einen bestimmten Wert überschreitet. CloudWatch Weitere Informationen erhalten Sie unter Manuelle Größenanpassung eines aktiven Clusters und Überwachung von EMR Amazon-Metriken mit CloudWatch.

Wenn HDFS der Platzmangel nicht das Problem war, überprüfen Sie die Protokolle, die DataNode NameNode Protokolle und die Netzwerkkonnektivität auf andere Probleme, die die Datenreplikation hätten HDFS verhindern können. Weitere Informationen finden Sie unter Anzeige von -Protokolldateien.

Knoten, die auf der Liste stehen

Der NodeManager Daemon ist für den Start und die Verwaltung von Containern auf Kern- und Taskknoten verantwortlich. Die Container werden dem NodeManager Daemon von dem Daemon zugewiesen, der ResourceManager auf dem Master-Knoten läuft. Der ResourceManager überwacht den NodeManager Knoten über einen Heartbeat.

Es gibt eine Reihe von Situationen, in denen der ResourceManager Daemon eine Liste auflistet und sie aus dem Pool der Knoten entfernt NodeManager, die für die Bearbeitung von Aufgaben zur Verfügung stehen:

- Wenn der in den NodeManager letzten 10 Minuten (600.000 Millisekunden) keinen Heartbeat an den ResourceManager Daemon gesendet hat. Dieser Zeitraum kann über die Konfigurationseinstellung `yarn.nm.liveness-monitor.expiry-interval-ms` festgelegt werden. Weitere Informationen zum Ändern der Yarn-Konfigurationseinstellungen finden Sie unter Konfiguration von Anwendungen im Amazon EMR Release Guide.
- NodeManager überprüft den Zustand der Festplatten, der durch `yarn.nodemanager.local-dirs` und bestimmt wird`yarn.nodemanager.log-dirs`. Die Prüfungen umfassen Berechtigungen und freien Speicherplatz (< 90 %). Wenn eine Festplatte die Prüfung nicht besteht, verwendet sie diese bestimmte Festplatte nicht NodeManager mehr, meldet den Knotenstatus aber trotzdem als fehlerfrei. Wenn mehrere Festplatten die Prüfung nicht bestehen, wird der

Knoten als fehlerhaft gemeldet ResourceManager und dem Knoten werden keine neuen Container zugewiesen.

Der Anwendungsmaster kann einen NodeManager Knoten auch ablehnen, wenn er mehr als drei fehlgeschlagene Aufgaben hat. Sie können hierfür mithilfe des Konfigurationsparameters `mapreduce.job.maxtaskfailures.per.tracker` einen höheren Wert einstellen. Andere Konfigurationseinstellungen, die Sie ändern können, steuern, wie oft versucht wird, eine Aufgabe auszuführen, bevor ein Fehler gemeldet wird: `mapreduce.map.max.attempts` für Map-Aufgaben und `mapreduce.reduce.maxattempts` für Reduce-Aufgaben. Weitere Informationen zum Ändern der Konfigurationseinstellungen finden Sie unter Configuring Applications im Amazon EMR Release Guide.

Drosselungsfehler

Die Fehler „Throttled from *Amazon EC2* beim Starten des Clusters“ und „Instanzen konnten aufgrund der Drosselung von nicht bereitgestellt werden *Amazon EC2*„treten auf, wenn Amazon eine Anfrage EMR nicht bearbeiten kann, weil ein anderer Service die Aktivität gedrosselt hat. Amazon EC2 ist die häufigste Quelle für Drosselungsfehler, aber auch andere Dienste können die Ursache für Drosselungsfehler sein. AWS Servicebeschränkungen gelten für jede Region, um die Leistung zu verbessern. Ein Drosselungsfehler weist darauf hin, dass Sie das Servicelimit für Ihr Konto in dieser Region überschritten haben.

Mögliche Ursachen

Die häufigste Ursache für EC2 Drosselungsfehler bei Amazon ist eine große Anzahl von Cluster-Instances, die gestartet werden, sodass Ihr Service-Limit für EC2 Instances überschritten wird. Cluster-Instances können aus den folgenden Gründen gestartet werden:

- Es werden neue Cluster erstellt.
- Die Clustergröße wird manuell angepasst. Weitere Informationen finden Sie unter Manuelle Größenanpassung eines aktiven Clusters.
- Instance-Gruppen in einem Cluster fügen Instances als Ergebnis einer Auto Scaling-Regel hinzu ("Scale-Out" oder horizontales Skalieren). Weitere Informationen finden Sie unter Grundlegendes zu Auto-Scaling-Regeln.
- Instance-Flotten in einem Cluster fügen Instances hinzu, um eine erhöhte Zielkapazität zu erreichen. Weitere Informationen finden Sie unter Instance-Flotten konfigurieren.

Es ist auch möglich, dass die Häufigkeit oder Art der API Anfrage an Amazon zu Drosselungsfehlern EC2 führt. Weitere Informationen darüber, wie Amazon API Anfragen EC2 drosselt, finden Sie unter Query API request rate in der EC2APIAmazon-Referenz.

Lösungen

Erwägen Sie die folgenden Lösungen:

- Folgen Sie den Anweisungen unter AWS -Service-Quotas in Allgemeine Amazon Web Services-Referenz, um eine Erhöhung des Servicelimits zu beantragen. Für manche APIs ist die Einrichtung einer CloudWatch Veranstaltung möglicherweise die bessere Option, als die Limits zu erhöhen. Weitere Details finden Sie unter Wann sollten EMR-Ereignisse eingerichtet werden in CloudWatch.
- Wenn Sie Cluster haben, die nach demselben Zeitplan gestartet werden, z. B. zu Beginn der Stunde, sollten Sie gestaffelte Startzeiten in Betracht ziehen.
- Wenn die Nachfragespitzen für Ihre Cluster zu groß angelegt sind und Sie Ihre Instance-Kapazitäten in regelmäßigen Abständen angeben, sollten Sie Ihre Instance mit Auto Scaling nach Bedarf hinzufügen und entfernen. Auf diese Weise werden Instances effizienter genutzt und können je nach Bedarfsprofil zu jedem beliebigen Zeitpunkt für ein Konto weniger Instances angefordert werden. Weitere Informationen finden Sie unter Verwenden der automatischen Skalierung mit einer benutzerdefinierten Richtlinie für Instance-Gruppen.

Instance-Typ nicht unterstützt

Wenn Sie einen Cluster erstellen und dieser mit der Fehlermeldung „Der angeforderte Instanztyp“ fehlschlägt *`InstanceType`* wird in der angeforderten Availability Zone nicht unterstützt. Dies bedeutet, dass Sie den Cluster erstellt und einen Instance-Typ für eine oder mehrere Instance-Gruppen angegeben haben, der von Amazon EMR in der Region und Availability Zone, in der der Cluster erstellt wurde, nicht unterstützt wird. Amazon unterstützt EMR möglicherweise einen Instance-Typ in einer Availability Zone innerhalb einer Region und nicht in einer anderen. Die Availability Zone innerhalb der Region ist von dem von Ihnen für einen Cluster ausgewählten Subnetz abhängig.

Lösung

Ermitteln Sie die verfügbaren Instance-Typen in einer Availability Zone mithilfe der AWS CLI

- Verwenden Sie den Befehl `ec2 run-instances` mit der Option `--dry-run`. Ersetzen Sie im folgenden Beispiel *`m5.xlarge`* durch den Instanztyp, den Sie verwenden möchten

ami-035be7bafff33b6b6 mit dem zu diesem Instanztyp AMI gehörenden Typ und *subnet-12ab3c45* mit einem Subnetz in der Availability Zone, das Sie abfragen möchten.

```
aws ec2 run-instances --instance-type m5.xlarge --dry-run --image-
id ami-035be7bafff33b6b6 --subnet-id subnet-12ab3c45
```

Anweisungen zum Suchen einer AMI ID finden Sie unter Finden Sie ein Linux AMI. Um eine Subnetz-ID zu finden, können Sie den Befehl describe-subnets verwenden.

Weitere Informationen darüber, wie Sie verfügbare Instance-Typen ermitteln können, finden Sie unter Finden Sie einen EC2 Amazon-Instance-Typ.

Nachdem Sie die verfügbaren Instance-Typen bestimmt haben, können Sie beliebige der folgenden Aktionen ausführen:

- Erstellen Sie den Cluster in derselben Region und demselben EC2 Subnetz und wählen Sie einen anderen Instance-Typ mit ähnlichen Funktionen wie Sie ursprünglich ausgewählt haben. Eine Liste mit unterstützten Instance-Typen finden Sie unter Unterstützte Instance-Typen. Informationen zum Vergleich der Funktionen von EC2 Instance-Typen finden Sie unter EC2Amazon-Instance-Typen.
- Wählen Sie ein Subnetz für den Cluster in einer Availability Zone aus, in der der Instance-Typ verfügbar ist und von Amazon EMR unterstützt wird.

Vermeiden Sie Fehler beim Start von Instance-Flotten und Clustern, die auf nicht unterstützte primäre Instance-Typen in Amazon zurückzuführen sind EMR

Primäre Knoten sind in EMR Amazon-Clustern unverzichtbar. Ein EMR Clusterstart kann mit einem `instance type not supported` Fehler fehlschlagen, wenn Amazon EMR versucht, den Cluster in einer Availability Zone zu starten, in der der primäre Instance-Typ nicht unterstützt wird. Die erweiterte Availability Zone-Auswahl für Instance-Flottencluster in Amazon filtert EMR automatisch nicht unterstützte AZs Instance-Typen heraus, die Sie in der Cluster-Konfiguration angegeben haben. Das bedeutet, dass Amazon EMR keine Availability Zone auswählt, in der die konfigurierten primären Instance-Typen nicht unterstützt werden, wodurch Cluster-Startfehler aufgrund nicht unterstützter Instance-Typen verhindert werden.

Um diese Verbesserung zu aktivieren, fügen Sie der Servicerolle oder -richtlinie für Ihren Cluster die erforderliche Berechtigung hinzu. Die neueste Version von `AmazonEMRServicePolicy_v2` beinhaltet diese Berechtigung. Wenn Sie also diese Richtlinie verwenden, ist die Verbesserung

bereits verfügbar. Wenn Sie eine benutzerdefinierte Servicerolle oder Richtlinie verwenden, fügen Sie die Berechtigung hinzu, `ec2:DescribeInstanceTypeOfferings` wenn Sie Ihren Cluster starten.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": "ec2:DescribeInstanceTypeOfferings",
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

EC2hat keine Kapazität

Ein "EC2hat keine Kapazität für *`InstanceType`*"tritt ein Fehler auf, wenn Sie versuchen, in einer Availability Zone, in der es keine weiteren Instanzen des angegebenen EC2 Instanztyps gibt, einen Cluster zu erstellen oder Instances zu einem Cluster hinzuzufügen. Die Availability Zone ist von dem von Ihnen für einen Cluster ausgewählten Subnetz abhängig.

Um einen Cluster zu erstellen, führen Sie einen der folgenden Schritte aus:

- Geben Sie einen anderen Instance-Typ mit ähnlichen Funktionen an
- Erstellen des Clusters in einer anderen Region
- Wählen Sie ein Subnetz in einer Availability Zone aus, in dem der gewünschte Instance-Typ möglicherweise verfügbar ist.

Führen Sie einen der folgenden Schritte aus, um Instances zu einem laufenden Cluster hinzuzufügen:

- Ändern Sie Instance-Gruppenkonfigurationen oder Instance-Flottenkonfigurationen so bearbeiten, dass verfügbare Instance-Typen mit ähnlichen Funktionen hinzugefügt werden. Eine Liste mit unterstützten Instance-Typen finden Sie unter Unterstützte Instance-Typen. Informationen zum Vergleich der Funktionen von EC2 Instance-Typen finden Sie unter EC2Amazon-Instance-Typen.
- Beenden Sie den Cluster und erstellen Sie ihn in einer Region und Verfügbarkeitszone neu, in der der Instancetyp verfügbar ist.

HDFSFehler beim Replikationsfaktor

Wenn Sie einen Core-Knoten aus einer Core-Instance-Gruppe oder Instance-Flotte entfernen, tritt bei Amazon EMR möglicherweise ein HDFS Replikationsfehler auf. Dieser Fehler tritt auf, wenn Sie Kernknoten entfernen und die Anzahl der Kernknoten unter den konfigurierten dfs.replication-Faktor für das Hadoop Distributed File System () fällt. HDFS Daher EMR kann Amazon den Vorgang nicht sicher durchführen. Um den Standardwert der `dfs.replication` Konfiguration zu ermitteln, HDFSKonfiguration.

Mögliche Ursachen

Im Folgenden finden Sie die möglichen Ursachen für Fehler beim HDFS Replikationsfaktor:

- Wenn Sie die Größe einer Core-Instance-Gruppe oder Instance-Flotte manuell unter den konfigurierten `dfs.replication` Faktor ändern.
- Ihre Richtlinien für verwaltete Skalierung oder Autoscaling ermöglichen möglicherweise eine Skalierung, um die Anzahl der Kernknoten unter den Schwellenwert von zu reduzieren. `dfs.replication`
- Dieser Fehler kann auch auftreten, wenn Amazon EMR versucht, einen fehlerhaften Kernknoten zu ersetzen, obwohl ein Cluster die minimale Anzahl von Kernknoten hat, die von `dfs.replication`definiert ist.

Lösungen und bewährte Verfahren

Im Folgenden finden Sie Lösungen und bewährte Verfahren:

- Wenn Sie die Größe eines EMR Amazon-Clusters manuell ändern, sollten Sie nicht unter den Wert herunterskalieren, `dfs.replication` da Amazon die Größenänderung nicht sicher abschließen EMR kann.
- Wenn Sie verwaltete Skalierung oder Autoscaling verwenden, stellen Sie sicher, dass die Mindestkapazität Ihres Clusters nicht unter dem Faktor liegt. `dfs.replication`
- Die Anzahl der Core-Instances sollte mindestens `dfs.replication` plus eins sein. Dadurch wird sichergestellt, dass Amazon einen fehlerhaften Core-Node erfolgreich ersetzen EMR kann, wenn Sie den Austausch fehlerhafter Kerne aktiviert haben.

> **Important**
>
> Der Ausfall eines einzelnen Core-Knotens kann zu HDFS Datenverlust führen, wenn Sie `dfs.replication` auf 1 setzen. Wenn Ihr Cluster über HDFS Speicher verfügt, empfehlen wir, den Cluster mit mindestens vier Kernknoten für Produktionsworkloads zu konfigurieren, um Datenverlust zu vermeiden, und außerdem den `dfs.replication` Faktor auf mindestens 2 festzulegen.

HDFSFehler bei unzureichendem Speicherplatz

Ein Hadoop Distributed File System (HDFS) -Fehler mit unzureichendem Speicherplatz kann auftreten, wenn Sie versuchen, einen Core-Knoten zu entfernen, Amazon den Vorgang jedoch nicht sicher abschließen EMR kann, da nicht genügend Speicherplatz in der vorhanden ist. HDFS Bevor Amazon einen Core-Node EMR entfernt, müssen alle HDFS Daten auf dem Node auf andere Core-Nodes übertragen werden, um Datenredundanz zu gewährleisten. Wenn auf den anderen Kernknoten jedoch nicht genügend Speicherplatz für die Replikation vorhanden ist, EMR kann Amazon den Knoten nicht ordnungsgemäß außer Betrieb nehmen.

Mögliche Ursachen

Im Folgenden finden Sie eine Liste der möglichen Ursachen für Fehler bei HDFS unzureichendem Speicherplatz:

- Wenn Sie eine Core-Instance-Gruppe oder Instance-Flotte manuell herunterskalieren, obwohl auf den verbleibenden Knoten vor dem Herunterskalieren nicht genügend HDFS Speicherplatz für die Datenreplikation vorhanden ist.
- Durch verwaltetes Skalieren oder Autoscaling wird eine Core-Instanzgruppe oder Instanzflotte herunterskaliert, wenn nicht genügend HDFS Speicherplatz für die Datenreplikation vorhanden ist.
- Amazon EMR versucht, einen fehlerhaften Kernknoten zu ersetzen, kann den Knoten jedoch aufgrund des unzureichenden HDFS Speicherplatzes nicht sicher ersetzen.

Lösungen und bewährte Methoden

Im Folgenden finden Sie Lösungen und bewährte Verfahren:

- Erhöhen Sie die Anzahl der Kernknoten in Ihrem EMR Amazon-Cluster. Wenn Sie Managed Scaling oder Autoscaling verwenden, erhöhen Sie die Mindestkapazität Ihrer Kernknoten.

- Verwenden Sie größere EBS Volumes für Ihre Kernknoten, wenn Sie Ihren EMR Cluster erstellen.
- Löschen Sie nicht benötigte HDFS Daten in Ihrem EMR Cluster. Wir empfehlen Ihnen, CloudWatch Alarme einzurichten, um die `HDFSUtilization` Metrik in Ihrem Cluster zu überwachen und festzustellen, ob in Ihrem EMR Cluster wenig Speicherplatz zur Verfügung steht.

Fehler bei der Ein- und Ausgabe

Die folgenden Fehler treten in Cluster-Ein- und Ausgabeoperationen häufig auf.

Themen

Hat Ihr Pfad zu Amazon Simple Storage Service (Amazon S3) mindestens drei Schrägstriche?

Wenn Sie einen Amazon S3 S3-Bucket angeben, müssen Sie einen abschließenden Schrägstrich am Ende von einfügen. URL Anstatt einen Bucket beispielsweise als „s3n: //amzn-s3-demo-bucket1" zu referenzieren, sollten Sie „s3n: //amzn-s3-demo-bucket1/“ verwenden, da Hadoop sonst in den meisten Fällen einen Ausfall Ihres Clusters verursacht.

Versuchen Sie Eingabeverzeichnisse rekursiv zu durchlaufen?

Hadoop durchsucht Eingabeverzeichnisse nicht rekursiv nach Dateien. Wenn Sie über eine Verzeichnisstruktur wie beispielsweise /corpus/01/01.txt, /corpus/01/02.txt, /corpus/02/01.txt usw. verfügen und /corpus/ als Eingabeparameter für Ihren Cluster angeben, findet Hadoop keine Eingabedateien, da das Verzeichnis /corpus/ leer ist und Hadoop den Inhalt der Unterverzeichnisse nicht überprüft. Entsprechend überprüft Hadoop die Unterverzeichnisse von Amazon-S3-Buckets nicht rekursiv.

Die Eingabedateien müssen sich direkt in dem Eingabeverzeichnis oder dem Amazon-S3-Bucket, das bzw. den Sie angeben, befinden und nicht in Unterverzeichnissen.

Ist Ihr Ausgabeverzeichnis bereits vorhanden?

Wenn Sie einen Ausgabepfad angeben, der bereits vorhanden ist, schlägt der Hadoop-Cluster in den meisten Fällen fehl. Das bedeutet, dass Sie, wenn Sie einen Cluster ausführen und dann diesen Vorgang mit denselben Parametern wiederholen, der erste Lauf und kein weiterer funktioniert. Nach dem ersten Lauf ist der Ausgabepfad vorhanden, was dazu führt, dass alle nachfolgenden Läufe fehlschlagen.

Versuchen Sie, eine Ressource mit einem zu spezifizieren? HTTP URL

Hadoop akzeptiert keine Ressourcenspeicherorte, die mit dem Präfix http:// angegeben werden. Sie können nicht mit einem auf eine Ressource verweisen HTTPURL. Wenn Sie beispielsweise http://mysite/myjar.jar als JAR Parameter angeben, schlägt der Cluster fehl.

Verweisen Sie mit einem ungültigen Namensformat auf einen Amazon-S3-Bucket?

Wenn Sie versuchen, einen Bucket-Namen wie „amzn-s3-demo-bucket1.1“ mit Amazon zu verwenden, schlägt Ihr Cluster fehl, weil Amazon EMR verlangtEMR, dass Bucket-Namen gültige RFC 2396 Hostnamen sind; der Name darf nicht mit einer Zahl enden. Aufgrund der Anforderungen von Hadoop EMR dürfen Amazon S3 S3-Bucket-Namen, die mit Amazon verwendet werden, außerdem nur Kleinbuchstaben, Zahlen, Punkte (.) und Bindestriche (-) enthalten. Weitere Informationen zum Formatieren von Amazon-S3-Bucket-Namen finden Sie unter Bucket-Einschränkungen und -Limits im Benutzerhandbuch für Amazon Simple Storage Service.

Haben Sie Probleme beim Laden von Daten in oder aus Amazon S3?

Amazon S3 ist die beliebteste Eingabe- und Ausgabequelle für AmazonEMR. Ein häufiger Fehler besteht darin, Amazon S3 so zu behandeln wie ein typisches Dateisystem. Es gibt Unterschiede zwischen Amazon S3 und einem Dateisystem, die Sie berücksichtigen müssen, wenn Sie Ihren Cluster ausführen.

- Wenn ein interner Fehler in Amazon S3 auftritt, muss Ihre Anwendung diesen problemlos behandeln und die Operation wiederholen.
- Wenn Aufrufe in Amazon S3 zu lange dauern, muss Ihre Anwendung die Häufigkeit der Amazon-S3-Aufrufe ggf. reduzieren.
- Das Auflisten aller Objekte in einem Amazon-S3-Bucket ist ein teurer Aufruf. Ihre Anwendung sollte die Anzahl solcher Aufrufe minimieren.

Es gibt mehrere Möglichkeiten, wie Sie die Interaktion Ihres Cluster mit Amazon S3 verbessern können.

- Starten Sie Ihren Cluster mit der neuesten Release-Version von AmazonEMR.
- Verwenden Sie S3DistCp , um Objekte in und aus Amazon S3 zu verschieben. S3 DistCp implementiert Fehlerbehandlung, Wiederholungsversuche und Back-offs, um die Anforderungen von Amazon S3 zu erfüllen. Weitere Informationen finden Sie unter Verteilte Kopie mit S3. DistCp
- Entwickeln Sie Ihre Anwendung mit letztendlicher Datenkonsistenz im Blick. Wird HDFS für die Zwischenspeicherung von Daten verwendet, während der Cluster läuft, und Amazon S3 nur zur Eingabe der Anfangsdaten und Ausgabe der Endergebnisse.
- Wenn Ihre Cluster einen Commit für mindestens 200 Transaktionen pro Sekunde in Amazon S3 contact supportdurchführen, wenden Sie sich an den Support, um Ihren Bucket auf größere Transaktionen pro Sekunde vorzubereiten. Ziehen Sie dazu die unter Tipps und Tricks zur Leistung von Amazon S3 beschriebenen Strategien in Erwägung.
- Legen Sie die Hadoop-Konfigurationseinstellung "io.file.buffer.size" auf "65536" fest. Diese bewirkt, dass Hadoop weniger Zeit damit verbringt, Amazon-S3-Objekte zu durchsuchen.
- Überlegen Sie sich, ob Sie dasie Speculative-Execution-Feature in Hadoop deaktivieren, wenn Ihr Cluster Probleme mit der gleichzeitigen Ausführung von Amazon S3 hat. Diese Vorgehensweise ist auch bei der Problembehandlung eines langsamen Clusters nützlich. Sie tun dies, indem Sie die `mapreduce.map.speculative`- und die `mapreduce.reduce.speculative`-Eigenschaften auf `false` festlegen. Wenn Sie einen Cluster starten, können Sie diese Werte mithilfe der `mapred-env`-Konfigurationsklassifizierung festlegen. Weitere Informationen finden Sie unter Configuring Applications im EMRAmazon-Versionshandbuch.
- Wenn Sie einen Hive-Cluster ausführen, finden Sie weitere Informationen unter Haben Sie Probleme beim Laden von Daten in oder aus Amazon S3 in Hive?.

Weitere Informationen finden Sie unter Bewährte Methoden für Amazon-S3-Fehler im Benutzerhandbuch von Amazon Simple Storage Service.

Berechtigungsfehler

Die folgenden Fehler treten häufig im Zusammenhang mit Berechtigungen oder Anmeldeinformationen auf.

Themen

- Übergeben Sie die richtigen Anmeldeinformationen anSSH?

- Wenn Sie verwendenIAM, haben Sie die richtigen EC2 Amazon-Richtlinien festgelegt?

Übergeben Sie die richtigen Anmeldeinformationen anSSH?

Wenn Sie keine Verbindung zum Master-Knoten herstellen könnenSSH, liegt höchstwahrscheinlich ein Problem mit Ihren Sicherheitsanmeldedaten vor.

Überprüfen Sie zunächst, ob die PEM-Datei, die Ihren SSH Schlüssel enthält, über die richtigen Berechtigungen verfügt. Verwenden Sie chmod, um die Berechtigungen für Ihre .PEM-Datei zu ändern. Im folgenden Beispiel würden Sie mykey.pem durch den Namen Ihre eigenen PEM-Datei ersetzen.

```
chmod og-rwx mykey.pem
```

Die zweite Fehlerquelle besteht darin, dass Sie nicht das Schlüsselpaar verwenden, das Sie beim Erstellen des Clusters angegeben haben. Dies passiert schnell, falls Sie mehrere Schlüsselpaare erstellt haben. Suchen Sie in den Cluster-Details in der EMR Amazon-Konsole (oder verwenden Sie die `--describe` Option inCLI) für den Namen des Schlüsselpaars, das bei der Erstellung des Clusters angegeben wurde.

Nachdem Sie sich vergewissert haben, dass Sie das richtige key pair verwenden und dass die Berechtigungen für die .pem-Datei korrekt gesetzt sind, können Sie den folgenden Befehl verwenden, um eine Verbindung SSH zum Master-Knoten herzustellen. Dabei ersetzen Sie mykey.pem durch den Namen Ihrer .pem-Datei und `hadoop@ec2-01-001-001-1.compute-1.amazonaws.com` durch den öffentlichen DNS Namen des Master-Knotens (verfügbar über die `--describe` Option in der oder über die Amazon-Konsole). CLI EMR

> **Important**
>
> Sie müssen den Anmeldenamen verwenden, `hadoop` wenn Sie eine Verbindung zu einem EMR Amazon-Cluster-Knoten herstellen. Andernfalls kann ein Fehler auftreten, der einem `Server refused our key` Fehler ähnelt.

```
ssh -i mykey.pem hadoop@ec2-01-001-001-1.compute-1.amazonaws.com
```

Weitere Informationen finden Sie unter Connect zum Primärknoten her mit SSH.

Wenn Sie verwendenIAM, haben Sie die richtigen EC2 Amazon-Richtlinien festgelegt?

Da Amazon EC2 Instances als Knoten EMR verwendet, müssen für Amazon-Benutzer EMR auch bestimmte EC2 Amazon-Richtlinien festgelegt seinEMR, damit Amazon diese Instances im Namen eines Benutzers verwalten kann. Wenn Sie nicht über die erforderlichen Berechtigungen verfügen, EMR gibt Amazon die folgende Fehlermeldung zurück: „Account is not authorized to call“EC2.

Weitere Informationen zu den EC2 Amazon-Richtlinien, die Ihr IAM Konto für die Ausführung von Amazon einrichten mussEMR, finden Sie unterSo EMR arbeitet Amazon mit IAM.

Hive-Cluster-Fehler

Den Grund für einen Hive-Fehler finden Sie in der Regel in der Datei `syslog`, auf die Sie im Bereich Steps (Schritte) zugreifen können. Wenn Sie das Problem nicht ermitteln können, sehen Sie sich die Fehlermeldung für die versuchte Hadoop-Aufgabe an. Erstellen Sie einen Link dahin im Abschnitt Task Attempts (Aufgaben-Versuche).

Die folgenden Fehler treten häufig bei Hive-Clustern auf.

Themen

- Verwenden Sie die neueste Version von Hive?
- Ist im Hive-Skript ein Syntaxfehler aufgetreten?
- Ist ein interaktiv ausgeführter Auftrag fehlgeschlagen?
- Haben Sie Probleme beim Laden von Daten in oder aus Amazon S3 in Hive?

Verwenden Sie die neueste Version von Hive?

Die neueste Version von Hive verfügt über alle aktuellen Patches und Fehlerbehebungen und kann Ihr Problem lösen.

Ist im Hive-Skript ein Syntaxfehler aufgetreten?

Wenn ein Schritt fehlschlägt, sehen Sie sich die Datei `stdout` der Protokolle für den Schritt an, die das Hive-Skript ausgeführt hat. Wenn der Fehler nicht vorhanden ist, sehen Sie sich die Datei

`syslog` der Aufgabenprotokolle für die versuchte Aufgabe an, die fehlgeschlagen ist. Weitere Informationen finden Sie unter Anzeige von -Protokolldateien.

Ist ein interaktiv ausgeführter Auftrag fehlgeschlagen?

Wenn Sie Hive interaktiv auf dem Master-Knoten ausführen und der Cluster fehlschlägt, sehen Sie sich die Einträge `syslog` im Aufgabenprotokoll für die fehlgeschlagene Aufgabe an. Weitere Informationen finden Sie unter Anzeige von -Protokolldateien.

Haben Sie Probleme beim Laden von Daten in oder aus Amazon S3 in Hive?

Falls Sie Probleme mit dem Zugriff auf Daten in Amazon S3 haben, überprüfen Sie zuerst die möglichen Ursachen, die in Haben Sie Probleme beim Laden von Daten in oder aus Amazon S3? aufgeführt sind. Wenn keines dieser Probleme die Ursache ist, ziehen Sie die folgenden, für Hive spezifischen Optionen in Betracht.

- Stellen Sie sicher, dass Sie die neueste Version von Hive verwenden, die über alle aktuellen Patches und Fehlerbehebungen verfügt, die Ihr Problem lösen können. Weitere Informationen finden Sie unter Apache Hive.
- Wenn Sie `INSERT OVERWRITE` verwenden, müssen Sie die Inhalte des Amazon-S3-Buckets oder -Ordners auflisten. Dies ist eine teure Operation. Wenn möglich, optimieren Sie den Pfad manuell die vorhandenen Objekte von Hive auflisten und löschen zu lassen.
- Wenn Sie EMR Amazon-Release-Versionen vor 5.0 verwenden, können Sie den folgenden Befehl in HiveQL verwenden, um die Ergebnisse einer Amazon S3-Listenoperation lokal auf dem Cluster vorab zwischenzuspeichern:

```
set hive.optimize.s3.query=true;
```

- Verwenden Sie statische Partitionen, wenn möglich.
- In einigen Versionen von Hive und Amazon ist es möglichEMR, dass die Verwendung ALTER TABLES fehlschlägt, weil die Tabelle an einem anderen Ort als von Hive erwartet gespeichert ist. Die Lösung ist, Folgendes in `/home/hadoop/conf/core-site.xml` hinzuzufügen oder zu aktualisieren:

```
<property>
    <name>fs.s3n.endpoint</name>
    <value>s3.amazonaws.com</value>
</property>
```

VPCFehler

Die folgenden Fehler treten häufig bei der VPC Konfiguration in Amazon aufEMR.

Themen

Ungültige Subnetzkonfiguration

Auf der Seite Cluster Details (Cluster-Details) im Feld Status sehen Sie eine Fehlermeldung wie folgende:

```
The subnet configuration was invalid: Cannot find route to InternetGateway
in main RouteTable rtb-id for vpc vpc-id.
```

Um dieses Problem zu lösen, müssen Sie ein Internet Gateway erstellen und es an Ihr anschließenVPC. Weitere Informationen finden Sie unter Hinzufügen eines Internet-Gateways zu Ihrem VPC.

Stellen Sie alternativ sicher, dass Sie Ihre VPC Konfiguration so konfiguriert haben, dass die Optionen DNS Auflösung aktivieren und DNS Hostnamenunterstützung aktivieren aktiviert sind. Weitere Informationen finden Sie unter Verwenden DNS mit Ihrem VPC.

Fehlender DHCP Optionssatz

Sie sehen einen Schrittfehler im Cluster-Systemprotokoll (syslog) mit einer Fehlermeldung ähnlich der folgenden:

```
ERROR org.apache.hadoop.security.UserGroupInformation
(main): PriviledgedActionException as:hadoop (auth:SIMPLE)
cause:java.io.IOException:
org.apache.hadoop.yarn.exceptions.ApplicationNotFoundException: Application
with id 'application_id' doesn't exist in RM.
```

or

```
ERROR org.apache.hadoop.streaming.StreamJob (main): Error Launching job :
org.apache.hadoop.yarn.exceptions.ApplicationNotFoundException: Application
with id 'application_id' doesn't exist in RM.
```

Um dieses Problem zu lösen, müssen Sie einen konfigurierenVPC, der einen DHCP Optionssatz enthält, dessen Parameter auf die folgenden Werte festgelegt sind:

> **Note**
>
> Wenn Sie die Region AWS GovCloud (US-West) verwenden, setzen Sie domain-name auf **us-gov-west-1.compute.internal** anstelle des im folgenden Beispiel verwendeten Werts.

- domain-name = **ec2.internal**

  Verwenden Sie **ec2.internal**, wenn Ihre Region USA Ost (Nord-Virginia) ist. Verwenden Sie für andere Regionen *region-name*.**compute.internal**. Verwenden Sie beispielsweise in us-west-2 domain-name =. **us-west-2.compute.internal**
- domain-name-servers = **AmazonProvidedDNS**

Weitere Informationen finden Sie unter Optionssätze. DHCP

Berechtigungsfehler

Ein Fehler im `stderr`-Protokoll für einen Schritt gibt an, dass eine Amazon-S3-Ressource nicht über die entsprechenden Berechtigungen verfügt. Dies ist ein Fehler 403, der wie folgt aussieht:

```
Exception in thread "main" com.amazonaws.services.s3.model.AmazonS3Exception: Access
 Denied (Service: Amazon S3; Status Code: 403; Error Code: AccessDenied; Request
 ID: REQUEST_ID
```

Wenn der auf gesetzt ActionOnFailure ist`TERMINATE_JOB_FLOW`, würde dies dazu führen, dass der Cluster mit dem Status, `SHUTDOWN_COMPLETED_WITH_ERRORS` beendet wird.

Möglichkeiten, um dieses Problem zu beheben, sind beispielsweise:

- Wenn Sie eine Amazon S3 S3-Bucket-Richtlinie innerhalb eines verwendenVPC, stellen Sie sicher, dass Sie Zugriff auf alle Buckets gewähren, indem Sie einen VPC Endpunkt erstellen und bei der Erstellung des Endpunkts unter der Option Richtlinie die Option Alle zulassen auswählen.
- Stellen Sie sicher, dass alle mit S3-Ressourcen verknüpften Richtlinien auch die Richtlinien enthalten, VPC in denen Sie den Cluster starten.
- Führen Sie den folgenden Befehl über Ihren Cluster aus, um zu überprüfen, ob Sie auf den Bucket zugreifen können.

```
hadoop fs -copyToLocal s3://path-to-bucket /tmp/
```

- Sie können spezifischere Debugging-Informationen abrufen, indem Sie den Parameter `log4j.logger.org.apache.http.wire` in der Datei DEBUG-Datei im Cluster auf `/home/hadoop/conf/log4j.properties` festlegen. Sie können die `stderr`-Protokolldatei prüfen, nachdem Sie versucht haben, über den Cluster auf den Bucket zuzugreifen. Die Protokolldatei enthält detaillierte Informationen:

```
Access denied for getting the prefix for bucket - us-west-2.elasticmapreduce with
 path samples/wordcount/input/
15/03/25 23:46:20 DEBUG http.wire: >> "GET /?prefix=samples%2Fwordcount%2Finput
%2F&delimiter=%2F&max-keys=1 HTTP/1.1[\r][\n]"
15/03/25 23:46:20 DEBUG http.wire: >> "Host: us-
west-2.elasticmapreduce.s3.amazonaws.com[\r][\n]"
```

Fehler, die zu **`START_FAILED`** führen

Vor AMI Version 3.7.0, für die VPCs ein Hostname angegeben wurde, EMR ordnet Amazon die internen Hostnamen des Subnetzes den benutzerdefinierten Domainadressen wie folgt zu:. `ip-X.X.X.X.customdomain.com.tld` Wenn der Hostname beispielsweise customdomain.com wäre `ip-10.0.0.10` und die Option Domainname auf customdomain.com gesetzt VPC ist, wäre der resultierende Hostname von Amazon zugeordnet. EMR `ip-10.0.1.0.customdomain.com` Ein Eintrag wird in `/etc/hosts` hinzugefügt, um den Hostnamen in 10.0.0.10 aufzulösen. Dieses Verhalten wurde mit AMI 3.7.0 geändert und jetzt berücksichtigt EMR Amazon die DHCP Konfiguration von vollständig. VPC Bislang konnten Kunden eine Zuweisung des Hostnamens auch mit einer Bootstrap-Aktion angeben.

Wenn Sie dieses Verhalten beibehalten möchten, müssen Sie das für die benutzerdefinierte Domain erforderliche Setup DNS und die Weiterleitungslösung angeben.

Cluster **`Terminated with errors`** und NameNode kann nicht gestartet werden

Wenn Sie einen EMR Cluster in einem startenVPC, der einen benutzerdefinierten DNS Domänennamen verwendet, schlägt Ihr Cluster möglicherweise fehl und es wird die folgende Fehlermeldung in der Konsole angezeigt:

```
Terminated with errors  On the master instance(instance-id), bootstrap action 1
 returned a  non-zero return code
```

Der Fehler ist darauf zurückzuführen, dass der Start NameNode nicht möglich ist. Dies führt zu dem folgenden Fehler in den NameNode Protokollen, deren Amazon S3 URI die Form hats3://*mybucket*/*logs*/*cluster-id*/daemons/*master instance-id*/hadoop-hadoop-namenode-*master node hostname*.log.gz:

```
2015-07-23 20:17:06,266 WARN
     org.apache.hadoop.hdfs.server.namenode.FSNamesystem (main): Encountered
 exception
     loading fsimage  java.io.IOException: NameNode is not formatted.
     at

 org.apache.hadoop.hdfs.server.namenode.FSImage.recoverTransitionRead(FSImage.java:212)
          at

 org.apache.hadoop.hdfs.server.namenode.FSNamesystem.loadFSImage(FSNamesystem.java:1020)
          at

 org.apache.hadoop.hdfs.server.namenode.FSNamesystem.loadFromDisk(FSNamesystem.java:739)
          at
     org.apache.hadoop.hdfs.server.namenode.NameNode.loadNamesystem(NameNode.java:537)
          at
     org.apache.hadoop.hdfs.server.namenode.NameNode.initialize(NameNode.java:596)

     at  org.apache.hadoop.hdfs.server.namenode.NameNode.<init>(NameNode.java:765)
          at
     org.apache.hadoop.hdfs.server.namenode.NameNode.<init>(NameNode.java:749)
     at

 org.apache.hadoop.hdfs.server.namenode.NameNode.createNameNode(NameNode.java:1441)
          at
     org.apache.hadoop.hdfs.server.namenode.NameNode.main(NameNode.java:1507)
```

Dies ist auf ein potenzielles Problem zurückzuführen, bei dem eine EC2 Instance mehrere Sätze vollqualifizierter Domainnamen haben kann, wenn EMR Cluster in einem gestartet werdenVPC, der sowohl einen von AWS-bereitgestellten DNS Server als auch einen benutzerdefinierten, vom Benutzer bereitgestellten DNS Server verwendet. Wenn der vom Benutzer bereitgestellte DNS Server keine Zeigerdatensätze (PTR) für A-Datensätze bereitstellt, die zur Bezeichnung von Knoten in einem EMR Cluster verwendet werden, schlägt der Start von Clustern fehl, wenn sie auf diese Weise konfiguriert sind. Die Lösung besteht darin, für jeden PTR A-Eintrag, der erstellt wird, wenn eine EC2 Instanz in einem der Subnetze in einem der Subnetze in der gestartet wird, einen Datensatz hinzuzufügen. VPC

Streaming-Cluster-Fehler

Sie können in der Regel die Ursache für einen Streaming-Fehler in einer `syslog`-Datei finden. Erstellen Sie einen Link dahin im Abschnitt Steps (Schritte).

Die folgenden Fehler treten häufig bei Streaming-Clustern auf.

Themen

- Werden Daten an den Mapper im falschen Format gesendet?
- Gibt es eine Zeitüberschreitung bei der Skriptausführung?
- Werden ungültige Streaming-Argumente übergeben?
- Wurde Ihr Skript mit einem Fehler beendet?

Werden Daten an den Mapper im falschen Format gesendet?

Suchen Sie in der `syslog`-Datei nach einer Fehlermeldung über einen fehlgeschlagenen Aufgabenversuch in den Protokolldateien der Aufgabenversuche, um dies zu überprüfen. Weitere Informationen finden Sie unter Anzeige von -Protokolldateien.

Gibt es eine Zeitüberschreitung bei der Skriptausführung?

Die standardmäßige Zeitbeschränkung für ein Mapper- oder Reducer-Skript beträgt 600 Sekunden. Wenn Ihr Skript mehr Zeit benötigt, schlägt der Aufgabenversuch fehl. Suchen Sie in der `syslog`-Datei nach einem fehlgeschlagenen Aufgabenversuch in den Protokolldateien der Aufgabenversuche, um dies zu überprüfen. Weitere Informationen finden Sie unter Anzeige von -Protokolldateien.

Sie können die Zeitbeschränkung ändern, indem Sie einen neuen Wert für die Konfigurationseinstellung `mapred.task.timeout` festlegen. Diese Einstellung gibt die Anzahl der Millisekunden an, nach denen Amazon EMR eine Aufgabe beendet, die keine Eingabe gelesen, keine Ausgabe geschrieben oder ihre Statuszeichenfolge nicht aktualisiert hat. Sie können diesen Wert aktualisieren, indem Sie ein zusätzliches Streaming-Argument `-jobconf mapred.task.timeout=800000` übergeben.

Werden ungültige Streaming-Argumente übergeben?

Hadoop-Streaming unterstützt nur die folgenden Argumente. Wenn Sie andere als die unten aufgeführten Argumente übergeben, schlägt der Cluster fehl.

```
-blockAutoGenerateCacheFiles
-cacheArchive
-cacheFile
-cmdenv
-combiner
-debug
-input
-inputformat
-inputreader
-jobconf
-mapper
-numReduceTasks
-output
-outputformat
-partitioner
-reducer
-verbose
```

Darüber hinaus erkennt Hadoop-Streaming nur in Java-Syntax übergebene Argumente, also mit einem vorangestellten einzelnen Bindestrich. Wenn Argumente mit vorangestelltem doppelten Bindestrich übergeben werden, schlägt der Cluster fehl.

Wurde Ihr Skript mit einem Fehler beendet?

Wenn Ihr Mapper- oder Reducer-Skript mit einem Fehler beendet wird, können Sie den Fehler in der `stderr`-Datei des fehlgeschlagenen Aufgabenversuchs in den Protokolldateien der Aufgabenversuche ermitteln. Weitere Informationen finden Sie unter Anzeige von -Protokolldateien.

Benutzerdefinierte Cluster-Fehler JAR

Die folgenden Fehler treten häufig bei benutzerdefinierten JAR Clustern auf.

Themen

Löst du JAR eine Ausnahme aus, bevor du einen Job erstellst?

Wenn das von Ihnen benutzerdefinierte Hauptprogramm beim Erstellen des Hadoop-Jobs eine Ausnahme JAR auslöst, schauen Sie am besten in der `syslog` Datei mit den Schrittprotokollen nach. Weitere Informationen finden Sie unter Anzeige von -Protokolldateien.

Wirft Ihr JAR innerhalb einer Map-Aufgabe einen Fehler aus?

Wenn Ihr Benutzerprogramm JAR und Ihr Mapper bei der Verarbeitung von Eingabedaten eine Ausnahme auslösen, suchen Sie am besten in der `syslog` Datei mit den Protokollen der Aufgabenversuche. Weitere Informationen finden Sie unter Anzeige von -Protokolldateien.

AWS GovCloud Fehler (US-West)

Die Region AWS GovCloud (US-West) unterscheidet sich von anderen Regionen in Bezug auf Sicherheit, Konfiguration und Standardeinstellungen. Verwenden Sie daher die folgende Checkliste, um EMR Amazon-Fehler zu beheben, die für die Region AWS GovCloud (USA West) spezifisch sind, bevor Sie allgemeinere Empfehlungen zur Fehlerbehebung verwenden.

- Stellen Sie sicher, dass Ihre IAM Rollen korrekt konfiguriert sind. Weitere Informationen finden Sie unter IAMServicerollen für EMR Amazon-Berechtigungen für AWS Dienste und Ressourcen konfigurieren.
- Stellen Sie sicher, dass Ihre VPC Konfiguration die DNS Auflösungs-/Hostnamenunterstützung, das Internet Gateway und DHCP die Optionssatz-Parameter korrekt konfiguriert hat. Weitere Informationen finden Sie unter VPCFehler.

Wenn diese Schritte das Problem nicht lösen, fahren Sie mit den Schritten zur Behebung häufiger EMR Amazon-Fehler fort. Weitere Informationen finden Sie unter Häufige Fehler bei Amazon EMR.

Finden Sie einen fehlenden Cluster

Wenn Ihr Cluster in der Konsolenliste fehlt oder `ListClusters`API, überprüfen Sie Folgendes:

- Vergewissern Sie sich, dass das Alter des Clusters ab dem Zeitpunkt der Fertigstellung weniger als zwei Monate beträgt. Amazon EMR bewahrt Metadateninformationen für abgeschlossene Cluster zwei Monate lang kostenlos auf. Sie können abgeschlossene Cluster nicht aus der Konsole löschen. Stattdessen löscht Amazon EMR abgeschlossene Cluster automatisch nach zwei Monaten.
- Bestätigen Sie, dass Sie über Rollenberechtigungen zum Anzeigen des Clusters verfügen.
- Vergewissern Sie sich, dass Sie dort, AWS-Region wo sich der Cluster befindet, dieselbe Ansicht sehen.

Fehlerbehebung für einen ausgefallenen Cluster

In diesem Abschnitt werden Sie durch den Vorgang zur Fehlerbehebung eines Cluster geführt, der ausgefallen ist. Das bedeutet, dass der Cluster mit einem Fehlercode beendet wurde.

> **Note**
>
> Wenn ein EMR Cluster mit einem Fehler beendet wird, werden ein Fehlercode und eine Fehlermeldung `ListClusters` APIs zurückgegeben. `DescribeCluster` Bei einigen Clusterfehlern kann Ihnen das `ErrorDetail`-Datenarray bei der Behebung des Fehlers helfen. Weitere Informationen finden Sie unter Fehlercodes mit ErrorDetail Informationen.

Wenn Ihr Cluster ausgeführt wird, aber lange braucht, bis Ergebnisse zurückgegeben werden, finden Sie unter Fehlerbehebung für einen langsamen Cluster.

Themen

- Schritt 1: Daten über das Problem sammeln
- Schritt 2: Die Umgebung prüfen
- Schritt 3: Die letzte Statusänderung überprüfen
- Schritt 4: Die Protokolldateien überprüfen
- Schritt 5: Den Cluster Schritt für Schritt testen

Schritt 1: Daten über das Problem sammeln

Der erste Schritt bei der Fehlerbehebung bei einem Cluster besteht darin, Informationen darüber zu sammeln, was schief gelaufen ist, sowie über den aktuellen Status und die Konfiguration des Clusters. Diese Informationen werden in den folgenden Schritten verwendet, um mögliche Ursachen des Problems zu bestätigen oder auszuschließen.

Definieren des Problems

Eine klare Definition des Problems ist der erste Ausgangspunkt. Einige Fragen, die Sie sich stellen sollten:

- Was habe ich erwartet? Was ist stattdessen passiert?
- Wann ist dieses Problem zum ersten Mal aufgetreten? Wie oft ist es seitdem passiert?
- Hat sich etwas an der Konfiguration oder Ausführung meines Clusters geändert?

Cluster-Details

Die folgenden Clusterdetails sind hilfreich, um Probleme aufzuspüren. Weitere Informationen zum Sammeln dieser Informationen finden Sie unter Cluster-Status und -Details anzeigen.

- Die Cluster-ID. (Wird auch als Job-Flow-Identifier bezeichnet.)
- AWS-Region und Availability Zone, in der der Cluster gestartet wurde.
- Status des Clusters, einschließlich Details zur letzten Statusänderung.
- Typ und Anzahl der für die Master-, Core- und Task-Knoten angegebenen EC2 Instanzen.

Schritt 2: Die Umgebung prüfen

Amazon EMR ist Teil eines Ökosystems aus Webdiensten und Open-Source-Software. Dinge, die sich auf diese Abhängigkeiten auswirken, können sich auf die Leistung von Amazon auswirkenEMR.

Themen

- Prüfen auf Service-Ausfälle
- Prüfen auf Nutzungsgrenzen
- Überprüfen der Version

- Überprüfen Sie die VPC Amazon-Subnetzkonfiguration

Prüfen auf Service-Ausfälle

Amazon EMR verwendet intern mehrere Amazon Web Services. Es betreibt virtuelle Server auf AmazonEC2, speichert Daten und Skripts auf Amazon S3 und meldet Metriken an CloudWatch. Ereignisse, die diese Dienste stören, sind selten — aber wenn sie auftreten, können sie zu Problemen bei Amazon EMR führen.

Überprüfen Sie die Übersicht zum Servicestatus, bevor Sie fortfahren. Prüfen Sie in der Region, in der Sie Ihren Cluster gestartet haben, ob es bei einem dieser Services zu Störungen gekommen ist.

Prüfen auf Nutzungsgrenzen

Wenn Sie einen großen Cluster starten, viele Cluster gleichzeitig gestartet haben oder wenn Sie ein Benutzer sind, der einen Cluster AWS-Konto mit anderen Benutzern teilt, ist der Cluster möglicherweise ausgefallen, weil Sie ein AWS Service-Limit überschritten haben.

Amazon EC2 begrenzt die Anzahl der virtuellen Server-Instances, die in einer einzelnen AWS Region ausgeführt werden, auf 20 On-Demand-Instances oder Reserved Instances. Wenn Sie einen Cluster mit mehr als 20 Knoten starten oder einen Cluster starten, der dazu führt, dass die Gesamtzahl der AWS-Konto auf Ihrem Computer aktiven EC2 Instances 20 überschreitet, kann der Cluster nicht alle benötigten EC2 Instances starten und schlägt möglicherweise fehl. In diesem Fall EMR gibt Amazon einen `EC2 QUOTA EXCEEDED` Fehler zurück. Sie können beantragen, dass die Anzahl der EC2 Instances, die Sie auf Ihrem Konto ausführen können, AWS erhöht wird, indem Sie einen Antrag auf Erhöhung des EC2 Amazon-Instance-Limits einreichen.

Eine weitere Sache, die dazu führen kann, dass Sie Ihre Nutzungslimits überschreiten, ist die Verzögerung zwischen der Beendigung eines Clusters und der Freigabe aller seiner Ressourcen. Je nach Konfiguration kann es bis zu 5–20 Minuten dauern, bis ein Cluster vollständig beendet ist und zugewiesene Ressourcen freigibt. Wenn Sie beim Versuch, einen Custer zu starten, die Fehlermeldung `EC2 QUOTA EXCEEDED` erhalten, kann es daran liegen, dass Ressourcen eines kürzlich beendeten Clusters noch nicht zur Verfügung stehen. In diesem Fall können Sie entweder beantragen, dass Ihr EC2 Amazon-Kontingent erhöht wird, oder Sie können zwanzig Minuten warten und den Cluster neu starten.

Amazon S3 begrenzt die Anzahl der auf einem Konto erstellten Buckets auf 100. Wenn Ihr Cluster einen neuen Bucket erstellt, der dieses Limit überschreitet, schlägt die Bucket-Erstellung fehl und kann dazu führen, dass der Cluster fehlschlägt.

Überprüfen der Version

Vergleichen Sie das Release-Label, mit dem Sie den Cluster gestartet haben, mit der neuesten EMR Amazon-Version. Jede Version von Amazon EMR enthält Verbesserungen wie neue Anwendungen, Funktionen, Patches und Bugfixes. Das Problem, das Ihren Cluster betrifft, wurde in der aktuellen Version möglicherweise bereits behoben. Führen Sie Ihren Cluster wenn möglich mit der aktuellen Version erneut aus.

Überprüfen Sie die VPC Amazon-Subnetzkonfiguration

Wenn Ihr Cluster in einem VPC Amazon-Subnetz gestartet wurde, muss das Subnetz wie unter beschrieben konfiguriert werden. Netzwerk konfigurieren Überprüfen Sie außerdem, ob das Subnetz, in dem Sie den Cluster starten, über genügend freie elastische IP-Adressen verfügt, um jedem Knoten im Cluster eine zuzuweisen.

Schritt 3: Die letzte Statusänderung überprüfen

Die letzte Statusänderung gibt Aufschluss darüber, welches Ereignis bei der letzten Statusänderung des Clusters aufgetreten ist. Häufig lassen sich Informationen gewinnen, die Hinweise darauf geben, welcher Fehler auftrat, als sich der Cluster-Status in `FAILED` änderte. Wenn Sie beispielsweise einen Streaming-Cluster starten und einen Ausgabespeicherort angeben, der in Amazon S3 bereits vorhanden ist, tritt beim Cluster ein Fehler auf und die letzte Statusänderung weist darauf hin, dass das Streaming-Ausgabeverzeichnis bereits vorhanden ist.

Sie können den Wert für die letzte Statusänderung in der Konsole finden, indem Sie den Detailbereich für den Cluster aufrufen, die `describe-cluster` Argumente `list-steps` oder CLI verwenden oder die Aktionen `DescribeCluster` und API `ListSteps` verwenden. Weitere Informationen finden Sie unter Cluster-Status und -Details anzeigen.

Schritt 4: Die Protokolldateien überprüfen

Der nächste Schritt besteht darin, die Protokolldateien zu untersuchen, um einen Fehlercode oder einen anderen Hinweis auf das Problem zu finden, das in Ihrem Cluster aufgetreten ist. Informationen zu den verfügbaren Protokolldateien, wo sie zu finden sind und wie Sie sie anzeigen können, finden Sie unter Anzeige von -Protokolldateien.

Es kann einige Nachforschungen erfordern, um herauszufinden, was passiert ist. Hadoop führt die Arbeit der Aufträge in Aufgabenversuchen auf verschiedenen Knoten im Cluster aus. Amazon EMR kann spekulative Aufgabenversuche einleiten und die anderen Aufgabenversuche beenden, die

nicht zuerst abgeschlossen werden. Dadurch werden umfangreiche Aktivitäten generiert, die in den Controller-, Stderr- und Syslog-Protokolldateien protokolliert werden. Darüber hinaus werden mehrere Aufgaben gleichzeitig ausgeführt, aber eine Protokolldatei kann die Ergebnisse nur linear anzeigen.

Überprüfen Sie zunächst die Bootstrap-Aktionsprotokolle auf Fehler oder unerwartete Konfigurationsänderungen beim Start des Clusters. Suchen Sie anschließend in den Schrittprotokollen nach Hadoop-Aufträgen, die als Teil eines fehlerhaften Schritts gestartet wurden. Untersuchen Sie die Hadoop-Auftragsprotokolle, um die fehlgeschlagenen Aufgabenversuche zu identifizieren. Das Protokoll der Aufgabenversuche wird Details darüber enthalten, was zum Fehlschlagen eines Aufgabenversuchs geführt hat.

In den folgenden Abschnitten wird erläutert, wie die verschiedenen Protokolldateien verwendet werden, um Fehler in Ihrem Cluster zu identifizieren.

Die Bootstrap-Aktionsprotokolle überprüfen

Bootstrap-Aktionen führen Skripts auf dem Cluster aus, während dieser gestartet wird. Sie werden häufig verwendet, um zusätzliche Software auf dem Cluster zu installieren oder um Konfigurationseinstellungen gegenüber den Standardwerten zu ändern. Die Überprüfung dieser Protokolle kann Aufschluss über Fehler geben, die bei der Einrichtung des Clusters aufgetreten sind, sowie über Änderungen der Konfigurationseinstellungen, die sich auf die Leistung auswirken könnten.

Die Schrittprotokolle überprüfen

Es gibt vier Arten von Schrittprotokollen.

- Controller — Enthält von Amazon EMR (AmazonEMR) generierte Dateien, die auf Fehler zurückzuführen sind, die bei der Ausführung Ihres Schritts aufgetreten sind. Wenn Ihr Schritt beim Laden fehlschlägt, finden Sie den Stack-Trace in diesem Protokoll. Fehler beim Laden oder Zugreifen auf Ihre Anwendung werden hier häufig beschrieben, ebenso wie Fehler in der fehlenden Mapper-Datei.
- stderr – Enthält Fehlermeldungen, die bei der Verarbeitung des Schritts aufgetreten sind. Fehler beim Laden von Anwendungen werden hier häufig beschrieben. Dieses Protokoll enthält manchmal einen Stack-Trace.
- stdout – Enthält den Status, der von Ihren ausführbaren Mapper- und Reducer-Dateien generiert wurde. Fehler beim Laden von Anwendungen werden hier häufig beschrieben. Dieses Protokoll enthält manchmal Anwendungsfehlermeldungen.

- syslog – Enthält Protokolle von Software, die nicht von Amazon stammt, wie Apache und Hadoop. Streaming-Fehler werden hier häufig beschrieben.

Überprüfen Sie stderr auf offensichtliche Fehler. Wenn stderr eine kurze Liste von Fehlern anzeigt, wurde der Schritt schnell beendet und es wurde ein Fehler ausgelöst. Dies wird meistens durch einen Fehler in den Mapper- und Reducer-Anwendungen verursacht, die im Cluster ausgeführt werden.

Untersuchen Sie die letzten Zeilen von Controller und Syslog auf Hinweise auf Fehler oder Ausfälle. Folgen Sie allen Hinweisen zu fehlgeschlagenen Aufgaben, insbesondere wenn dort „Auftrag fehlgeschlagen“ steht.

Die Aufgabenversuchsprotokolle überprüfen

Wenn die vorherige Analyse der Schrittprotokolle eine oder mehrere fehlgeschlagene Aufgaben ergeben hat, suchen Sie in den Protokollen der entsprechenden Aufgabenversuche nach detaillierteren Fehlerinformationen.

Schritt 5: Den Cluster Schritt für Schritt testen

Eine nützliche Strategie zum Nachverfolgen der Ursache für einen Fehler besteht darin, den Cluster neu zu starten und die Schritte einzeln auszuführen. So können Sie die Ergebnisse für jeden Schritt überprüfen, bevor Sie die Verarbeitung des nächsten Schritts starten, und erhalten die Möglichkeit, einen fehlgeschlagenen Schritt zu korrigieren und erneut auszuführen. Dies hat den Vorteil, dass Sie Ihre Eingabedaten nur einmal laden müssen.

So testen Sie den Cluster Schritt für Schritt

1. Starten Sie einen neuen Cluster mit aktiviertem Keepalive und Beendigungsschutz. Keepalive sorgt dafür, dass der Cluster weiterhin ausgeführt wird, nachdem er alle ausstehenden Schritte verarbeitet hat. Der Beendigungsschutz verhindert, dass ein Cluster im Falle eines Fehlers heruntergefahren wird. Weitere Informationen erhalten Sie unter Konfigurieren eines Clusters zum Fortfahren oder Beenden nach der Schrittausführung und Verwenden des Beendigungsschutzes.
2. Senden Sie einen Schritt an den Cluster. Weitere Informationen finden Sie unter Übermitteln von Arbeit an einen Cluster.
3. Wenn die Verarbeitung des Schritts abgeschlossen ist, prüfen Sie die Schrittprotokolldateien auf Fehler. Weitere Informationen finden Sie unter Schritt 4: Die Protokolldateien überprüfen. Die schnellste Möglichkeit zum Auffinden dieser Protokolldateien besteht darin, eine

Verbindung mit dem Master-Knoten herzustellen und die Protokolldateien hier anzuzeigen. Die Schrittprotokolldateien werden erst angezeigt, wenn der Schritt einige Zeit ausgeführt wird, beendet wird oder ein Fehler auftritt.

4. Wenn der Schritt erfolgreich ohne Fehler abgeschlossen wurde, führen Sie den nächsten Schritt aus. Wenn Fehler vorliegen, ermitteln Sie den Fehler in den Protokolldateien. Wenn in Ihrem Code ein Fehler aufgetreten ist, korrigieren Sie ihn und führen Sie den Schritt erneut aus. Fahren Sie fort, bis alle Schritte ohne Fehler ausgeführt werden.
5. Wenn Sie das Debuggen des Clusters abgeschlossen haben, müssen Sie den Cluster ggf. manuell beenden. Dies ist erforderlich, da der Cluster mit aktiviertem Beendigungsschutz gestartet wurde. Weitere Informationen finden Sie unter Verwenden des Beendigungsschutzes.

Fehlerbehebung für einen langsamen Cluster

In diesem Abschnitt wird die Fehlerbehebung eines Clusters beschrieben, der noch ausgeführt wird, aber viel Zeit benötigt, um Ergebnisse zurückzugeben. Weitere Informationen zu Verfahren, die Sie anwenden können, wenn der Cluster mit einem Fehlercode beendet wurde, finden Sie unter Fehlerbehebung für einen ausgefallenen Cluster

EMRMit Amazon können Sie die Anzahl und Art der Instances im Cluster angeben. Diese Spezifikationen sind die beste Möglichkeit, die Geschwindigkeit, mit der Ihre Daten verarbeitet werden, zu beeinflussen. Eine Sache, die Sie in Betracht ziehen könnten, ist die erneute Ausführung des Clusters, wobei Sie diesmal EC2 Instances mit größeren Ressourcen oder eine größere Anzahl von Instances im Cluster angeben. Weitere Informationen finden Sie unter Cluster-Hardware und Netzwerken konfigurieren.

In den folgenden Themen wird erklärt, wie Sie alternative Ursachen für einen langsamen Cluster identifizieren.

Themen

- Schritt 7: Eingabedaten überprüfen

Schritt 1: Daten über das Problem sammeln

Der erste Schritt bei der Fehlerbehebung bei einem Cluster besteht darin, Informationen darüber zu sammeln, was schief gelaufen ist, sowie über den aktuellen Status und die Konfiguration des Clusters. Diese Informationen werden in den folgenden Schritten verwendet, um mögliche Ursachen des Problems zu bestätigen oder auszuschließen.

Definieren des Problems

Eine klare Definition des Problems ist der erste Ausgangspunkt. Einige Fragen, die Sie sich stellen sollten:

- Was habe ich erwartet? Was ist stattdessen passiert?
- Wann ist dieses Problem zum ersten Mal aufgetreten? Wie oft ist es seitdem passiert?
- Hat sich etwas an der Konfiguration oder Ausführung meines Clusters geändert?

Cluster-Details

Die folgenden Clusterdetails sind hilfreich, um Probleme aufzuspüren. Weitere Informationen zum Sammeln dieser Informationen finden Sie unter Cluster-Status und -Details anzeigen.

- Die Cluster-ID. (Wird auch als Job-Flow-Identifier bezeichnet.)
- AWS-Region und Availability Zone, in der der Cluster gestartet wurde.
- Status des Clusters, einschließlich Details zur letzten Statusänderung.
- Typ und Anzahl der für die Master-, Core- und Task-Knoten angegebenen EC2 Instanzen.

Schritt 2: Die Umgebung prüfen

Themen

- Prüfen auf Service-Ausfälle
- Prüfen auf Nutzungsgrenzen
- Überprüfen Sie die VPC Amazon-Subnetzkonfiguration
- Neustarten des Clusters

Prüfen auf Service-Ausfälle

Amazon EMR verwendet intern mehrere Amazon Web Services. Es betreibt virtuelle Server auf AmazonEC2, speichert Daten und Skripts auf Amazon S3 und meldet Metriken an CloudWatch. Ereignisse, die diese Dienste stören, sind selten — aber wenn sie auftreten, können sie zu Problemen bei Amazon EMR führen.

Überprüfen Sie die Übersicht zum Servicestatus, bevor Sie fortfahren. Prüfen Sie in der Region, in der Sie Ihren Cluster gestartet haben, ob es bei einem dieser Services zu Störungen gekommen ist.

Prüfen auf Nutzungsgrenzen

Wenn Sie einen großen Cluster starten, viele Cluster gleichzeitig gestartet haben oder wenn Sie ein Benutzer sind, der einen Cluster AWS-Konto mit anderen Benutzern teilt, ist der Cluster möglicherweise ausgefallen, weil Sie ein AWS Service-Limit überschritten haben.

Amazon EC2 begrenzt die Anzahl der virtuellen Server-Instances, die in einer einzelnen AWS Region ausgeführt werden, auf 20 On-Demand-Instances oder Reserved Instances. Wenn Sie einen Cluster mit mehr als 20 Knoten starten oder einen Cluster starten, der dazu führt, dass die Gesamtzahl der AWS-Konto auf Ihrem Computer aktiven EC2 Instances 20 überschreitet, kann der Cluster nicht alle benötigten EC2 Instances starten und schlägt möglicherweise fehl. In diesem Fall EMR gibt Amazon einen `EC2 QUOTA EXCEEDED` Fehler zurück. Sie können beantragen, dass die Anzahl der EC2 Instances, die Sie auf Ihrem Konto ausführen können, AWS erhöht wird, indem Sie einen Antrag auf Erhöhung des EC2 Amazon-Instance-Limits einreichen.

Eine weitere Sache, die dazu führen kann, dass Sie Ihre Nutzungslimits überschreiten, ist die Verzögerung zwischen der Beendigung eines Clusters und der Freigabe aller seiner Ressourcen. Je nach Konfiguration kann es bis zu 5–20 Minuten dauern, bis ein Cluster vollständig beendet ist und zugewiesene Ressourcen freigibt. Wenn Sie beim Versuch, einen Custer zu starten, die Fehlermeldung `EC2 QUOTA EXCEEDED` erhalten, kann es daran liegen, dass Ressourcen eines kürzlich beendeten Clusters noch nicht zur Verfügung stehen. In diesem Fall können Sie entweder beantragen, dass Ihr EC2 Amazon-Kontingent erhöht wird, oder Sie können zwanzig Minuten warten und den Cluster neu starten.

Amazon S3 begrenzt die Anzahl der auf einem Konto erstellten Buckets auf 100. Wenn Ihr Cluster einen neuen Bucket erstellt, der dieses Limit überschreitet, schlägt die Bucket-Erstellung fehl und kann dazu führen, dass der Cluster fehlschlägt.

Überprüfen Sie die VPC Amazon-Subnetzkonfiguration

Wenn Ihr Cluster in einem VPC Amazon-Subnetz gestartet wurde, muss das Subnetz wie unter beschrieben konfiguriert werden. Netzwerk konfigurieren Überprüfen Sie außerdem, ob das Subnetz, in dem Sie den Cluster starten, über genügend freie elastische IP-Adressen verfügt, um jedem Knoten im Cluster eine zuzuweisen.

Neustarten des Clusters

Die Verlangsamung der Verarbeitung kann von einer vorübergehenden Bedingung herrühren. Überlegen Sie sich, ob Sie den Cluster beenden und neu starten möchten, um zu prüfen, ob sich die Leistung verbessert.

Schritt 3: Die Protokolldateien prüfen

Der nächste Schritt besteht darin, die Protokolldateien zu untersuchen, um einen Fehlercode oder einen anderen Hinweis auf das Problem zu finden, das in Ihrem Cluster aufgetreten ist. Informationen zu den verfügbaren Protokolldateien, wo sie zu finden sind und wie Sie sie anzeigen können, finden Sie unter Anzeige von -Protokolldateien.

Es kann einige Nachforschungen erfordern, um herauszufinden, was passiert ist. Hadoop führt die Arbeit der Aufträge in Aufgabenversuchen auf verschiedenen Knoten im Cluster aus. Amazon EMR kann spekulative Aufgabenversuche einleiten und die anderen Aufgabenversuche beenden, die nicht zuerst abgeschlossen werden. Dadurch werden umfangreiche Aktivitäten generiert, die in den Controller-, Stderr- und Syslog-Protokolldateien protokolliert werden. Darüber hinaus werden mehrere Aufgaben gleichzeitig ausgeführt, aber eine Protokolldatei kann die Ergebnisse nur linear anzeigen.

Überprüfen Sie zunächst die Bootstrap-Aktionsprotokolle auf Fehler oder unerwartete Konfigurationsänderungen beim Start des Clusters. Suchen Sie anschließend in den Schrittprotokollen nach Hadoop-Aufträgen, die als Teil eines fehlerhaften Schritts gestartet wurden. Untersuchen Sie die Hadoop-Auftragsprotokolle, um die fehlgeschlagenen Aufgabenversuche zu identifizieren. Das Protokoll der Aufgabenversuche wird Details darüber enthalten, was zum Fehlschlagen eines Aufgabenversuchs geführt hat.

In den folgenden Abschnitten wird erläutert, wie die verschiedenen Protokolldateien verwendet werden, um Fehler in Ihrem Cluster zu identifizieren.

Die Bootstrap-Aktionsprotokolle überprüfen

Bootstrap-Aktionen führen Skripts auf dem Cluster aus, während dieser gestartet wird. Sie werden häufig verwendet, um zusätzliche Software auf dem Cluster zu installieren oder um Konfigurationseinstellungen gegenüber den Standardwerten zu ändern. Die Überprüfung dieser Protokolle kann Aufschluss über Fehler geben, die bei der Einrichtung des Clusters aufgetreten sind, sowie über Änderungen der Konfigurationseinstellungen, die sich auf die Leistung auswirken könnten.

Die Schrittprotokolle überprüfen

Es gibt vier Arten von Schrittprotokollen.

- Controller — Enthält von Amazon EMR (AmazonEMR) generierte Dateien, die auf Fehler zurückzuführen sind, die bei der Ausführung Ihres Schritts aufgetreten sind. Wenn Ihr Schritt beim Laden fehlschlägt, finden Sie den Stack-Trace in diesem Protokoll. Fehler beim Laden oder Zugreifen auf Ihre Anwendung werden hier häufig beschrieben, ebenso wie Fehler in der fehlenden Mapper-Datei.
- stderr – Enthält Fehlermeldungen, die bei der Verarbeitung des Schritts aufgetreten sind. Fehler beim Laden von Anwendungen werden hier häufig beschrieben. Dieses Protokoll enthält manchmal einen Stack-Trace.
- stdout – Enthält den Status, der von Ihren ausführbaren Mapper- und Reducer-Dateien generiert wurde. Fehler beim Laden von Anwendungen werden hier häufig beschrieben. Dieses Protokoll enthält manchmal Anwendungsfehlermeldungen.
- syslog – Enthält Protokolle von Software, die nicht von Amazon stammt, wie Apache und Hadoop. Streaming-Fehler werden hier häufig beschrieben.

Überprüfen Sie stderr auf offensichtliche Fehler. Wenn stderr eine kurze Liste von Fehlern anzeigt, wurde der Schritt schnell beendet und es wurde ein Fehler ausgelöst. Dies wird meistens durch einen Fehler in den Mapper- und Reducer-Anwendungen verursacht, die im Cluster ausgeführt werden.

Untersuchen Sie die letzten Zeilen von Controller und Syslog auf Hinweise auf Fehler oder Ausfälle. Folgen Sie allen Hinweisen zu fehlgeschlagenen Aufgaben, insbesondere wenn dort „Auftrag fehlgeschlagen“ steht.

Die Aufgabenversuchsprotokolle überprüfen

Wenn die vorherige Analyse der Schrittprotokolle eine oder mehrere fehlgeschlagene Aufgaben ergeben hat, suchen Sie in den Protokollen der entsprechenden Aufgabenversuche nach detaillierteren Fehlerinformationen.

Die Hadoop-Daemon-Protokolle überprüfen

In seltenen Fällen kann Hadoop selbst ausfallen. Um zu sehen, ob das der Fall ist, müssen Sie sich die Hadoop-Protokolle ansehen. Sie befinden sich auf `/var/log/hadoop/` auf jedem Knoten.

Sie können die JobTracker Protokolle verwenden, um einen fehlgeschlagenen Aufgabenversuch dem Knoten zuzuordnen, auf dem er ausgeführt wurde. Sobald Sie den Knoten kennen, der mit dem Aufgabenversuch verknüpft ist, können Sie den Zustand der EC2 Instanz überprüfen, die diesen Knoten hostet, um festzustellen, ob Probleme wie Speichermangel CPU aufgetreten sind.

Schritt 4: Den Zustand des Clusters und der Instance überprüfen

Ein EMR Amazon-Cluster besteht aus Knoten, die auf EC2 Amazon-Instances ausgeführt werden. Wenn diese Instances an Ressourcen gebunden sind (z. B. wenn der CPU Arbeitsspeicher knapp wird), Probleme mit der Netzwerkkonnektivität auftreten oder beendet werden, leidet die Geschwindigkeit der Cluster-Verarbeitung.

Es gibt bis zu drei Arten von Knoten in einem Cluster:

- Hauptknoten – verwaltet den Cluster. Wenn ein Leistungsproblem auftritt, ist der gesamte Cluster betroffen.
- Kernknoten — verarbeiten Aufgaben zur Reduzierung von Zuordnungen und verwalten das Hadoop Distributed Filesystem (). HDFS Wenn bei einem dieser Knoten ein Leistungsproblem auftritt, kann dies sowohl den HDFS Betrieb als auch die Map-Reduce-Verarbeitung verlangsamen. Sie können einem Cluster zusätzliche Core-Knoten hinzufügen, um die Leistung zu verbessern, aber keine Core-Knoten entfernen. Weitere Informationen finden Sie unter Manuelle Größenanpassung eines aktiven Clusters.
- Aufgabenknoten – verarbeiten Map- und Reduce-Aufgaben. Dies sind reine Rechenressourcen und speichern keine Daten. Sie können einem Cluster Aufgabenknoten hinzufügen, um die Leistung zu beschleunigen, oder nicht benötigte Aufgabenknoten entfernen. Weitere Informationen finden Sie unter Manuelle Größenanpassung eines aktiven Clusters.

Wenn Sie den Zustand eines Clusters prüfen, sollten Sie sich sowohl die Leistung des Clusters insgesamt als auch die Leistung der einzelnen Instances anschauen. Es gibt mehrere Tools, die Sie verwenden können:

Überprüfen Sie den Zustand des Clusters mit CloudWatch

Jeder EMR Amazon-Cluster meldet Metriken an CloudWatch. Diese Metriken bieten zusammenfassende Leistungsinformationen über den Cluster, z. B. Gesamtlast, HDFS Auslastung, laufende Aufgaben, verbleibende Aufgaben, beschädigte Blöcke und mehr. Wenn Sie sich die CloudWatch Metriken ansehen, erhalten Sie einen Überblick darüber, was in Ihrem Cluster vor sich geht, und Sie erhalten einen Einblick in die Ursachen für die Verlangsamung der Verarbeitung. Sie können nicht nur ein vorhandenes Leistungsproblem analysieren, sondern auch Alarme einrichten, die eine Warnung auslösen CloudWatch , wenn ein future Leistungsproblem auftritt. CloudWatch Weitere Informationen finden Sie unter Überwachung von EMR Amazon-Metriken mit CloudWatch.

Überprüfen Sie den Status und HDFS den Zustand des Jobs

Verwenden Sie die Registerkarte Anwendungsbenutzeroberflächen auf der Cluster-Detailseite, um YARN Anwendungsdetails anzuzeigen. Bei bestimmten Anwendungen können Sie weitere Details und Zugriffsprotokolle direkt anzeigen. Dies ist besonders nützlich für Spark-Anwendungen. Weitere Informationen finden Sie unter Anwendungsverlauf anzeigen.

Hadoop bietet eine Reihe von Webschnittstellen, mit denen Sie Informationen anzeigen lassen können. Weitere Informationen darüber, wie Sie auf diese Webschnittstellen zugreifen können, finden Sie unter Auf EMR Amazon-Clustern gehostete Weboberflächen anzeigen.

- JobTracker — enthält Informationen über den Status des Jobs, der vom Cluster verarbeitet wird. Mit dieser Schnittstelle können Sie ermitteln, wann ein Auftrag blockiert ist.
- HDFS NameNode — liefert Informationen über den Prozentsatz der HDFS Auslastung und den verfügbaren Speicherplatz auf jedem Knoten. Mithilfe dieser Schnittstelle können Sie feststellen, wann Ressourcen knapp werden und zusätzliche Kapazität benötigt HDFS wird.
- TaskTracker — liefert Informationen über die Aufgaben des Jobs, der vom Cluster verarbeitet wird. Mit dieser Schnittstelle können Sie ermitteln, wann eine Aufgabe blockiert ist.

Überprüfen Sie den Zustand Ihrer Instance mit Amazon EC2

Eine andere Möglichkeit, nach Informationen über den Status der Instances in Ihrem Cluster zu suchen, ist die Verwendung der EC2 Amazon-Konsole. Da jeder Knoten im Cluster auf einer EC2

Instance ausgeführt wird, können Sie die von Amazon bereitgestellten Tools verwendenEC2, um ihren Status zu überprüfen. Weitere Informationen finden Sie unter Cluster-Instances in Amazon anzeigen EC2.

Schritt 5: Nach gesperrten Gruppen suchen

Eine Instance-Truppe wird angehalten, wenn beim Versuch, einen Knoten zu starten, zu viele Fehler auftreten. Wenn z. B. neue Knoten während der Durchführung von Bootstrap-Aktionen wiederholt fehlschlagen, wechselt die Instance-Gruppe nach einiger Zeit in den Status `SUSPENDED`, anstatt fortlaufend zu versuchen, neue Knoten bereitzustellen.

In folgenden Fällen kann ein Knoten fehlschlagen:

- Hadoop oder der Cluster ist irgendwie beschädigt und akzeptiert keinen neuen Knoten im Cluster.
- Eine Bootstrap-Aktion schlägt auf dem neuen Knoten fehl.
- Der Knoten arbeitet nicht ordnungsgemäß und kann nicht mit Hadoop einchecken

Wenn sich eine Instance-Gruppe im Status `SUSPENDED` befindet und der Cluster den Status `WAITING` hat, können Sie einen Cluster-Schritt hinzufügen, um die gewünschte Anzahl von Core- und Aufgabenknoten zurückzusetzen. Durch Hinzufügen des Schritts wird die Verarbeitung des Clusters fortgesetzt und die Instance-Gruppe wieder in den Status `RUNNING` versetzt.

Weitere Informationen zum Zurücksetzen eines Clusters im angehaltenen Zustand finden Sie unter Suspendierter Zustand.

Schritt 6: Konfigurationseinstellungen überprüfen

Konfigurationseinstellungen legen die Ausführung eines Clusters im Detail fest, z. B. wie häufig eine Aufgabe wiederholt wird und wie viel Arbeitsspeicher zum Sortieren verfügbar ist. Wenn Sie einen Cluster mit Amazon startenEMR, gibt es zusätzlich zu den standardmäßigen Hadoop-Konfigurationseinstellungen EMR Amazon-spezifische Einstellungen. Die Konfigurationseinstellungen werden im Master-Knoten des Clusters gespeichert. Sie können die Konfigurationseinstellungen überprüfen, um sicherzustellen, dass Ihr Cluster über die benötigten Ressourcen für einen effizienten Betrieb verfügt.

Amazon EMR definiert standardmäßige Hadoop-Konfigurationseinstellungen, die zum Starten eines Clusters verwendet werden. Die Werte basieren auf dem AMI und dem Instance-Typ, den Sie für den Cluster angeben. Ändern können Sie die Standardwerte der Konfigurationseinstellungen mithilfe einer Bootstrap-Aktion oder indem Sie neue Wert in den Parametern für die Auftragsausführung festlegen.

Weitere Informationen finden Sie unter Erstellen von Bootstrap-Aktionen zur Installation zusätzlicher Software. Um zu bestimmen, ob eine Bootstrap-Aktion die Konfigurationseinstellungen geändert hat, prüfen Sie die Bootstrap-Aktionsprotokolle.

Amazon EMR protokolliert die Hadoop-Einstellungen, die zur Ausführung der einzelnen Jobs verwendet wurden. Die Protokolldaten werden in einer Datei gespeichert, die `job_`*`job-id`*`_conf.xml` unter dem `/mnt/var/log/hadoop/history/` Verzeichnis des Master-Knotens benannt ist, wobei *`job-id`* wird durch den Bezeichner des Jobs ersetzt. Wenn Sie die Protokollarchivierung aktiviert haben, werden diese Daten in den `logs/`*`date`*`/`*`jobflow-id`*`/jobs` Ordner nach Amazon S3 kopiert, wo *`date`* ist das Datum, an dem der Job ausgeführt wurde, und *`jobflow-id`* ist der Identifier des Clusters.

Die folgenden Konfigurationseinstellungen des Hadoop-Auftrags eignen sich besonders für die Untersuchung von Leistungsproblemen. Weitere Informationen zu den Hadoop-Konfigurationseinstellungen und deren Auswirkungen auf das Verhalten von Hadoop finden Sie unter http://hadoop.apache.org/docs/.

> **Warning**
>
> 1. Die Einstellung `dfs.replication` auf 1 für Cluster mit weniger als vier Knoten kann zu HDFS Datenverlust führen, wenn ein einzelner Knoten ausfällt. Wir empfehlen, für Produktionsworkloads einen Cluster mit mindestens vier Core-Knoten zu verwenden.
> 2. Amazon EMR erlaubt Clustern nicht, Kernknoten nach unten zu skalierendfs.replication. Bei `dfs.replication = 2` z. B. beträgt die Mindestanzahl von Core-Knoten 2.
> 3. Wenn Sie verwaltete Skalierung oder Auto-Scaling verwenden oder die Größe Ihres Clusters manuell ändern möchten, empfehlen wir Ihnen, `dfs.replication` auf 2 oder höher einzustellen.

Konfigurationseinstellung	Beschreibung
dfs.replication	Die Anzahl der HDFS Knoten, auf die ein einzelner Block (wie der Festplattenblock) kopiert wird, um eine RAID ähnliche Umgebung zu erzeugen. Bestimmt die Anzahl der HDFS Knoten, die eine Kopie des Blocks enthalten.

Konfigurationseinstellung	Beschreibung
io.sort.mb	Für die Sortierung verfügbarer Gesamtspeicher. Dieser Wert sollte das Zehnfache von "io.sort.factor" sein. Diese Einstellung kann auch für die Berechnung des vom Aufgabenknoten genutzten Gesamtspeichers durch Berechnen von "io.sort.mb" multipliziert mit "mapred.t asktracker.ap.tasks.maximum" verwendet werden.
io.sort.spill.percent	Wird während der Sortierung verwendet. An diesem Punkt beginnt die Verwendung des Datenträgers, da der für die Sortierung zugewiesene Speicherplatz knapp wird.
mapred.child.java.opts	Als veraltet gekennzeichnet. Verwenden Sie stattdess en "mapred.map.child.java.opts" und "mapred.reduce.chi ld.java.opts". Die Java-Optionen, die beim Starten einer JVM für eine Aufgabe zur Ausführung innerhalb von TaskTracker verwendet werden. "-Xmx" ist ein üblicher Parameter zum Festlegen der maximalen Arbeitssp eichergröße.
mapred.map.child.java.opts	Die Java-Optionen, die beim Starten einer JVM für eine Map zur Ausführung innerhalb von TaskTracker verwendet werden. "-Xmx" ist ein üblicher Parameter zum Festlegen der maximalen Heap-Arbeitsspeichergröße.
mapred.map.tasks.speculativ e.execution	Legt fest, ob Map-Aufgabenversuche derselben Aufgabe parallel gestartet werden können.
mapred.reduce.tasks.specula tive.execution	Legt fest, ob Reduce-Aufgabenversuche derselben Aufgabe parallel gestartet werden können.
mapred.map.max.attempts	Die maximale Anzahl an Map-Aufgabenversuchen. Wenn alle fehlschlagen, wird die Map-Aufgabe als fehlgesch lagen markiert.

Konfigurationseinstellung	Beschreibung
mapred.reduce.child.java.opts	Die Java-Optionen, die beim Starten einer JVM für eine Reduce-Aufgabe zur Ausführung innerhalb von TaskTracker verwendet werden. "-Xmx" ist ein üblicher Parameter zum Festlegen der maximalen Heap-Arbe itsspeichergröße.
mapred.reduce.max.attempts	Die maximale Anzahl an Reduce-Aufgabenversuchen. Wenn alle fehlschlagen, wird die Map-Aufgabe als fehlgeschlagen markiert.
mapred.reduce.slowstart.com pleted.maps	Die Anzahl an Map-Aufgaben, die abgeschlossen werden, bevor Reduce-Aufgabenversuche durchgeführt werden. Bei zu geringer Wartezeit kann der Fehler „Too many fetch" in Versuchen ausgelöst werden.
mapred.reuse.jvm.num.tasks	Eine Aufgabe wird innerhalb einer einzigen ausgeführ tJVM. Gibt an, wie viele Aufgaben dieselbe wiederver wenden könnenJVM.
mapred.tasktracker.map.task s.maximum	Die maximale Anzahl von Aufgaben, die während des Map-Vorgangs pro Aufgabenknoten parallel ausgeführt werden können.
mapred.tasktracker.reduce.t asks.maximum	Die maximale Anzahl von Aufgaben, die während des Reduce-Vorgangs pro Aufgabenknoten parallel ausgeführ t werden können.

Wenn Ihre Cluster-Aufgaben arbeitsspeicherintensiv sind, können Sie die Leistung verbessern, indem Sie weniger Aufgaben pro Core-Knoten verwenden und die Heap-Größe des JobTrackers reduzieren.

Schritt 7: Eingabedaten überprüfen

Schauen Sie sich Ihre Eingabedaten an. Sind diese gleichmäßig auf Ihre Schlüsselwerte verteilt? Bei einer starken Datenschiefe in Richtung eines oder weniger Schlüsselwerte wird die Verarbeitungslast möglicherweise einer kleinen Anzahl von Knoten zugeordnet, während sich andere Knoten

im Leerlauf befinden. Diese ungleichmäßige Verteilung der Arbeit kann zu einer langsameren Verarbeitung führen.

Um einen ungleichmäßigen Datensatz handelt es sich z. B., wenn ein Cluster ausgeführt wird, um Wörter alphabetisch anzuordnen, aber ein Datensatz zur Verfügung steht, dessen Wörter alle nur mit "a" beginnen. Beim Map-Vorgang wird dann der Knoten überfordert, der Werte verarbeitet, die mit "a" anfangen, während diejenigen Knoten nicht beschäftigt sind, die Wörter mit anderen Anfangsbuchstaben verarbeiten.

Problembehandlung bei einem Lake-Formation-Cluster

In diesem Abschnitt erfahren Sie, wie Sie häufig auftretende Probleme bei der Nutzung von Amazon EMR mit beheben AWS Lake Formation.

Der Zugriff auf den Data Lake ist nicht zulässig

Sie müssen sich ausdrücklich für die Datenfilterung auf EMR Amazon-Clustern entscheiden, bevor Sie Daten in Ihrem Data Lake analysieren und verarbeiten können. Wenn der Datenzugriff fehlschlägt, wird in der Ausgabe Ihrer Notebookeinträge eine allgemeine `Access is not allowed`-Meldung angezeigt.

Anweisungen dazu, wie Sie die Datenfilterung bei Amazon aktivieren und zulassen könnenEMR, finden Sie unter Datenfilterung bei Amazon zulassen EMR im AWS Lake Formation Entwicklerhandbuch.

Sitzungsablauf

Das Sitzungs-Timeout für EMR Notebooks und Zeppelin wird durch die Einstellung „IAMRolle für Lake Formation“ gesteuert. `Maximum CLI/API session duration` Der Standardwert für diese Einstellung ist eine Stunde. Wenn ein Sitzungs-Timeout eintritt, sehen Sie in der Ausgabe Ihrer Notizbucheinträge die folgende Meldung, wenn Sie versuchen, Spark-Befehle auszuführen. SQL

```
Error 401    HTTP ERROR: 401 Problem accessing /sessions/2/statements.
Reason:  JWT token included in request failed validation.
Powered by Jetty:// 9.3.24.v20180605
 org.springframework.web.client.HttpClientErrorException: 401 JWT token included in
 request failed validation…
```

Aktualisieren Sie die Seite, um Ihre Sitzung zu validieren. Sie werden aufgefordert, sich erneut über Ihren Identitätsanbieter zu authentifizieren und dann zu dem Notebook zurückgeleitet. Sie können nach der erneuten Authentifizierung weiter Abfragen ausführen.

Keine Berechtigungen für Benutzer in der angeforderten Tabelle

Wenn Sie versuchen, auf eine Tabelle zuzugreifen, auf die Sie keinen Zugriff haben, wird die folgende Ausnahme in der Ausgabe Ihrer Notizbucheinträge angezeigt, wenn Sie versuchen, SQL Spark-Befehle auszuführen.

```
org.apache.spark.sql.AnalysisException:
 org.apache.hadoop.hive.ql.metadata.HiveException: Unable to fetch table table.
Resource does not exist or requester is not authorized to access requested
 permissions.
(Service: AWSGlue; Status Code: 400; Error Code: AccessDeniedException; Request ID: …
```

Um auf die Tabelle zuzugreifen, müssen Sie dem Benutzer Zugriff gewähren, indem Sie die mit dieser Tabelle verknüpften Berechtigungen in Lake Formation aktualisieren.

Abfragen von kontenübergreifenden Daten, die mit Lake Formation geteilt wurden

Wenn Sie Amazon verwendenEMR, um von einem anderen Konto aus auf Daten zuzugreifen, die mit Ihnen geteilt wurden, versuchen einige Spark-Bibliotheken, `Glue:GetUserDefinedFunctions` API Operation aufzurufen. Da die Versionen 1 und 2 der AWS RAM verwalteten Berechtigungen diese Aktion nicht unterstützen, erhalten Sie die folgende Fehlermeldung:

`"ERROR: User: arn:aws:sts::012345678901:assumed-role/my-spark-role/i-06ab8c2b59299508a is not authorized to perform: glue:GetUserDefinedFunctions on resource: arn:exampleCatalogResource because no resource-based policy allows the glue:GetUserDefinedFunctions action"`

Um diesen Fehler zu beheben, muss der Data Lake-Administrator, der die Ressourcenfreigabe erstellt hat, die AWS RAM verwalteten Berechtigungen aktualisieren, die der Ressourcenfreigabe zugeordnet sind. Version 3 der von AWS RAM verwalteten Berechtigungen ermöglicht es Prinzipalen, die `glue:GetUserDefinedFunctions`-Aktion auszuführen.

Wenn Sie eine neue Ressourcenfreigabe erstellen, wendet Lake Formation standardmäßig die neueste Version der AWS RAM verwalteten Berechtigung an, sodass Sie nichts unternehmen

müssen. Um den kontenübergreifenden Datenzugriff für bestehende Ressourcenfreigaben zu ermöglichen, müssen Sie die AWS RAM verwalteten Berechtigungen auf Version 3 aktualisieren.

Die AWS RAM Berechtigungen, die Ressourcen zugewiesen wurden, die mit Ihnen geteilt wurden, finden Sie unter AWS RAM. Die folgenden Berechtigungen sind in Version 3 enthalten:

```
Databases
  AWSRAMPermissionGlueDatabaseReadWriteForCatalog
  AWSRAMPermissionGlueDatabaseReadWrite

Tables
  AWSRAMPermissionGlueTableReadWriteForCatalog
  AWSRAMPermissionGlueTableReadWriteForDatabase

AllTables
  AWSRAMPermissionGlueAllTablesReadWriteForCatalog
  AWSRAMPermissionGlueAllTablesReadWriteForDatabase
```

Um die Version mit AWS RAM verwalteten Berechtigungen vorhandener Ressourcenfreigaben zu aktualisieren

Sie (Data Lake-Administrator) können entweder AWS RAM verwaltete Berechtigungen auf eine neuere Version aktualisieren, indem Sie den Anweisungen im AWS RAM Benutzerhandbuch folgen, oder Sie können alle vorhandenen Berechtigungen für den Ressourcentyp widerrufen und sie erneut gewähren. Wenn Sie Berechtigungen widerrufen, wird die mit dem AWS RAM Ressourcentyp verknüpfte Ressourcenfreigabe AWS RAM gelöscht. Wenn Sie Berechtigungen erneut gewähren, AWS RAM erstellt es neue Ressourcenfreigaben, denen die neueste Version der verwalteten Berechtigungen angehängt wird. AWS RAM

Einfügen in, Erstellen und Ändern von Tabellen

Das Einfügen von Daten in Tabellen in Datenbanken und das Erstellen und Ändern von Datenbanken, die durch Lake Formation Richtlinien geschützt sind, wird nicht unterstützt. Wenn Sie diese Operationen ausführen, wird in der Ausgabe Ihrer Notizbucheinträge die folgende Ausnahme angezeigt, wenn Sie versuchen, SQL Spark-Befehle auszuführen:

```
java.io.IOException:
 com.amazon.ws.emr.hadoop.fs.shaded.com.amazonaws.services.s3.model.AmazonS3Exception:
```

```
            Access Denied (Service: Amazon S3; Status Code: 403; Error Code:
AccessDenied; Request ID: …
```

Weitere Informationen finden Sie unter Einschränkungen der EMR Amazon-Integration mit AWS Lake Formation.

Schreiben von Anwendungen, die Cluster starten und verwalten

Themen

Sie können auf die von der Amazon EMR-API bereitgestellten Funktionen zugreifen, indem Sie Wrapper-Funktionen in einem der AWS SDKs aufrufen. Die AWS SDKs bieten sprachspezifische Funktionen, die die API des Webservices umschließen und die Verbindung zum Webservice vereinfachen, da sie viele Verbindungsdetails für Sie übernehmen. Weitere Informationen zum Aufrufen von Amazon EMR mit einem der SDKs finden Sie unter So verwenden Sie SDKs zum Aufrufen von Amazon-EMR-APIs.

> **Important**
>
> Die maximale Anforderungsrate für Amazon EMR beträgt eine Anforderung alle zehn Sekunden.

E nd-to-end Beispiel für Amazon EMR Java-Quellcode

Entwickler können die Amazon-EMR-API über benutzerdefinierten Java-Code aufrufen, um die über die Amazon-EMR-Konsole und CLI verfügbaren Funktionen zu nutzen. Dieser Abschnitt enthält die end-to-end Schritte, die für die Installation AWS Toolkit for Eclipse und Ausführung eines voll funktionsfähigen Java-Quellcode-Beispiels erforderlich sind, das Schritte zu einem Amazon EMR-Cluster hinzufügt.

> **Note**
>
> Dieses Beispiel konzentriert sich auf Java. Amazon EMR unterstützt über verschiedene Amazon-EMR-SDKs jedoch auch andere Programmiersprachen. Weitere Informationen finden Sie unter So verwenden Sie SDKs zum Aufrufen von Amazon-EMR-APIs.

In diesem Java-Beispiel wird gezeigt, wie die folgenden Aufgaben mit der Amazon-EMR-API durchgeführt werden:

- AWS Anmeldeinformationen abrufen und an Amazon EMR senden, um API-Aufrufe zu tätigen
- Konfigurieren eines neuen, benutzerdefinierten Schritts und eines neuen, vordefinierten Schritts
- Hinzufügen neuer Schritte zu einem vorhandenen Amazon-EMR-Cluster
- Abrufen der Cluster-Schritt-IDs aus einem ausgeführten Cluster

> **Note**
>
> In diesem Beispiel wird gezeigt, wie Sie Schritte zu einem vorhandene, Cluster hinzufügen. Daher ist ein aktiver Cluster in Ihrem Konto erforderlich.

Bevor Sie beginnen, installieren Sie die Version von Eclipse IDE for Java EE Developers, die Ihrer Plattform entspricht. Weitere Informationen erhalten Sie unter Eclipse-Downloads.

Als Nächstes installieren Sie das Database Development Plug-in für Eclipse.

So installieren Sie das Database Development Plug-in für Eclipse

1. Öffnen Sie die Eclipse-IDE.
2. Wählen Sie Help (Hilfe) und dann Install New Software (Neue Software installieren) aus.
3. Geben Sie im Feld Work with: (Arbeiten mit:) **`http://download.eclipse.org/releases/kepler`** oder den Pfad ein, der der Versionsnummer Ihrer Eclipse IDE entspricht.
4. Wählen Sie in der Liste Database Development (Datenbankentwicklung) und Finish (Fertig stellen) aus.
5. Starten Sie Eclipse neu, wenn Sie dazu aufgefordert werden.

Als Nächstes installieren Sie das Toolkit für Eclipse, um hilfreiche, vorkonfigurierte Quellcode-Projektvorlagen nutzen zu können.

So installieren Sie das Toolkit für Eclipse

1. Öffnen Sie die Eclipse-IDE.
2. Wählen Sie Help (Hilfe) und dann Install New Software (Neue Software installieren) aus.
3. Geben Sie im Feld Work with: (Arbeiten mit:) **`https://aws.amazon.com/eclipse`** ein.
4. Wählen Sie in der Artikelliste die Option AWS Toolkit for Eclipse und Fertigstellen aus.
5. Starten Sie Eclipse neu, wenn Sie dazu aufgefordert werden.

Erstellen Sie als Nächstes ein neues AWS Java-Projekt und führen Sie den Java-Beispielquellcode aus.

Um ein neues AWS Java-Projekt zu erstellen

1. Öffnen Sie die Eclipse-IDE.
2. Wählen Sie File (Datei), New (Neu) und Other (Sonstiges) aus.
3. Wählen Sie im Dialogfeld Einen Assistenten auswählen AWS -Java-Projekt und Weiter aus.
4. Geben Sie im Dialogfeld Neues AWS Java-Projekt in das **`Project name:`** Feld beispielsweise den Namen Ihres neuen Projekts ein**`EMR-sample-code`**.
5. Wählen Sie AWS Konten konfigurieren..., geben Sie Ihre öffentlichen und privaten Zugangsschlüssel ein und wählen Sie Fertig stellen. Weitere Informationen zum Erstellen von Zugriffsschlüsseln finden Sie unter Wie erhalte ich Sicherheitsanmeldeinformationen? in Allgemeine Amazon-Web-Services-Referenz.

   > **Note**
   >
   > Sie sollten Zugriffsschlüssel nicht direkt in den Code einbetten. Das Amazon-EMR-SDK ermöglicht es Ihnen, Zugriffsschlüssel in bekannten Speicherorten abzulegen, sodass Sie sie nicht in den Code integrieren müssen.

6. Klicken Sie im neuen Java-Projekt mit der rechten Maustaste auf den src--Ordner und wählen Sie dann New (Neu) und Class (Klasse) aus.
7. Geben Sie im Dialogfeld Java Class (Java-Klasse) in das Feld Name einen Namen für Ihre neue Klasse ein (z. B. **`main`**).

8. Wählen Sie im Abschnitt Which method stubs would you like to create? (Welche Method-Stubs möchten Sie erstellen?) public static void main (String [] args) und Finish (Fertig stellen) aus.
9. Geben Sie den Java-Quellcode in Ihrer neuen Klasse ein und fügen Sie die entsprechenden import (Importieren)-Anweisungen für die Klassen und Methoden des Beispiels hinzu. Den vollständigen Quellcode finden Sie unten.

> **Note**
>
> Ersetzen Sie im folgenden Beispielcode die Beispiel-Cluster-ID (JobFlowId) *j-xxxxxxxxxxxx*, durch eine gültige Cluster-ID in Ihrem Konto, die Sie entweder in AWS Management Console oder finden, indem Sie den folgenden AWS CLI Befehl verwenden:
>
> ```
> aws emr list-clusters --active | grep "Id"
> ```
>
> Ersetzen Sie außerdem den Amazon-S3-Beispielpfad *s3://path/to/my/jarfolder* durch den gültigen Pfad der JAR-Datei. Ersetzen Sie den Beispiel-Klassennamen (*com.my.Main1*) durch den richtigen Namen der Klasse in der JAR-Datei (falls relevant).

```
import com.amazonaws.AmazonClientException;
import com.amazonaws.auth.AWSCredentials;
import com.amazonaws.auth.AWSStaticCredentialsProvider;
import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.services.elasticmapreduce.AmazonElasticMapReduce;
import com.amazonaws.services.elasticmapreduce.AmazonElasticMapReduceClientBuilder;
import com.amazonaws.services.elasticmapreduce.model.*;
import com.amazonaws.services.elasticmapreduce.util.StepFactory;

public class Main {

 public static void main(String[] args) {
  AWSCredentials credentials_profile = null;
  try {
   credentials_profile = new
 ProfileCredentialsProvider("default").getCredentials();
  } catch (Exception e) {
   throw new AmazonClientException(
```

```
      "Cannot load credentials from .aws/credentials file. " +
        "Make sure that the credentials file exists and the profile name is
 specified within it.",
      e);
  }

  AmazonElasticMapReduce emr = AmazonElasticMapReduceClientBuilder.standard()
    .withCredentials(new AWSStaticCredentialsProvider(credentials_profile))
    .withRegion(Regions.US_WEST_1)
    .build();

  // Run a bash script using a predefined step in the StepFactory helper class
  StepFactory stepFactory = new StepFactory();
  StepConfig runBashScript = new StepConfig()
    .withName("Run a bash script")
    .withHadoopJarStep(stepFactory.newScriptRunnerStep("s3://jeffgoll/emr-scripts/
create_users.sh"))
    .withActionOnFailure("CONTINUE");

  // Run a custom jar file as a step
  HadoopJarStepConfig hadoopConfig1 = new HadoopJarStepConfig()
    .withJar("s3://path/to/my/jarfolder") // replace with the location of the jar
 to run as a step
    .withMainClass("com.my.Main1") // optional main class, this can be omitted if
 jar above has a manifest
    .withArgs("--verbose"); // optional list of arguments to pass to the jar
  StepConfig myCustomJarStep = new StepConfig("RunHadoopJar", hadoopConfig1);

  AddJobFlowStepsResult result = emr.addJobFlowSteps(new AddJobFlowStepsRequest()
    .withJobFlowId("j-xxxxxxxxxxxx") // replace with cluster id to run the steps
    .withSteps(runBashScript, myCustomJarStep));

  System.out.println(result.getStepIds());

 }
}
```

10. Wählen Sie Run (Ausführen), Run As (Ausführen als) und Java Application (Java-Anwendung) aus.
11. Wenn das Beispiel korrekt ausgeführt wird, wird eine Liste der IDs für die neuen Schritte in der Eclipse-IDE-Konsole angezeigt. Die korrekte Ausgabe sieht folgendermaßen oder ähnlich aus:

```
[s-39BLQZRJB2E5E, s-1L6A4ZU2SAURC]
```

Grundlegende Konzepte für API-Aufrufe

Themen

- Endpunkte für Amazon EMR
- Angeben von Cluster-Parametern in Amazon EMR
- Availability Zones in Amazon EMR
- So verwenden Sie weitere Dateien und Bibliotheken in Amazon-EMR-Clustern

Wenn Sie eine Anwendung entwickeln, die Amazon-EMR-API-Aufrufe durchführt, gibt es mehrere Konzepte, die Sie beim Aufruf einer der Wrapper-Funktionen in einem SDK einsetzen können.

Endpunkte für Amazon EMR

Ein Endpunkt ist eine URL, die als Eintrittspunkt für einen Webservice fungiert. Jede Webserviceanforderung muss einen Endpunkt umfassen. Der Endpunkt gibt die AWS Region an, in der Cluster erstellt, beschrieben oder beendet werden. Er hat die Form `elasticmapreduce.`*`regionname`*`.amazonaws.com`. Wenn Sie den allgemeinen Endpunkt (`elasticmapreduce.amazonaws.com`) angeben, leitet Amazon EMR Ihre Anforderung an einen Endpunkt in der Standardregion weiter. Für Konten, die am oder nach dem 8. März 2013 erstellt wurden, lautet die Standardregion "us-west-2"; für ältere Konten ist die Standardregion "us-east-1".

Weitere Informationen über Regionen und Endpunkte für Amazon EMR finden Sie unter Regionen und Endpunkte im Allgemeine Amazon Web Services-Referenz.

Angeben von Cluster-Parametern in Amazon EMR

Die `Instances`-Parameter ermöglichen das Konfigurieren des Typs und der Anzahl der EC2-Instances zum Erstellen von Knoten für die Verarbeitung der Daten. Hadoop verteilt die Verarbeitung der Daten über mehrere Cluster-Knoten. Der Master-Knoten ist für die Integrität der Core- und Aufgabenknoten sowie für das Abfragen des Auftragsergebnisstatus der Knoten verantwortlich. Die Core- und Aufgabenknoten erledigen die tatsächliche Verarbeitung der Daten. Wenn Sie einen Cluster mit einem Knoten haben, agiert dieser als Master-Knoten und als Core-Knoten.

Der `KeepJobAlive`-Parameter in einer `RunJobFlow`-Anforderung bestimmt, ob der Cluster beendet wird, wenn der Cluster keine auszuführenden Schritte mehr hat. Legen Sie diesen Wert auf `False` fest, wenn Sie wissen, dass der Cluster wie erwartet ausgeführt wird. Bei der Fehlerbehebung des Auftragverlaufs und beim Hinzufügen von Schritten während der ausgesetzten Cluster-Ausführung legen Sie den Wert auf `True` fest. Das reduziert die Zeit und die Kosten für das Hochladen der Ergebnisse in Amazon Simple Storage Service (Amazon S3) der Neustart des Clusters nach dem Bearbeiten eines Schritts müsste wiederholt werden).

`KeepJobAlive`Ist dies der `true` Fall, müssen Sie, nachdem Sie den Cluster erfolgreich zum Abschluss gebracht haben, eine `TerminateJobFlows` Anfrage senden. Andernfalls wird der Cluster weiter ausgeführt und es AWS fallen Gebühren an.

Weitere Hinweise zu Parametern, die nur für geltenRunJobFlow, finden Sie unter RunJobFlow. Weitere Informationen zu den grundlegenden Parametern in der Anfrage finden Sie unter Allgemeine Anforderungsparameter.

Availability Zones in Amazon EMR

Amazon EMR arbeitet mit EC2-Instances als Knoten zur Cluster-Verarbeitung. Diese EC2-Instances arbeiten mit Standorten, die aus Regionen und Availability Zones bestehen. Regionen sind verteilt und befinden sich in unterschiedlichen geografischen Zonen. Availability Zones sind eigenständige Standorte innerhalb einer Region, die von Ausfällen anderen Availability Zones isoliert sind. Jede Availability Zone bietet eine kostengünstige Netzwerkkonnektivität mit geringer Latenz zu anderen Availability Zones in der gleichen Region. Eine Liste der Regionen und Endpunkte für Amazon EMR finden Sie unter Regionen und Endpunkte in der Allgemeine Amazon Web Services-Referenz.

Der `AvailabilityZone`-Parameter gibt den grundlegenden Speicherort des Clusters an. Dieser Parameter ist optional. Wir empfehlen seine Verwendung. Wenn `AvailabilityZone` nicht angegeben ist, wählt Amazon EMR automatisch den besten `AvailabilityZone`-Wert für den Cluster aus. Der Parameter kann z. B. dann nützlich sein, wenn Sie Ihre Instances mit anderen aktiven Instances gemeinsam platzieren möchten und Ihr Cluster Daten aus diesen Instances lesen oder schreiben muss. Weitere Informationen finden Sie im Amazon EC2 EC2-Benutzerhandbuch.

So verwenden Sie weitere Dateien und Bibliotheken in Amazon-EMR-Clustern

Es kann vorkommen, dass Sie weiteren Dateien oder benutzerdefinierte Bibliotheken für Ihre Mapper oder Reducer-Anwendungen verwenden möchten. Sie können beispielsweise eine Bibliothek nutzen, die eine PDF-Datei in eine Textdatei konvertiert.

So speichern Sie eine Datei für den Mapper oder Reducer bei der Verwendung von Hadoop-Streaming zwischen

- Fügen Sie im JAR-`args`-Feld das folgende Argument hinzu:

  ```
  -cacheFile s3://bucket/path_to_executable#local_path
  ```

  Die Datei (`local_path`) befindet sich im Arbeitsverzeichnis des Mappers. Dieser kann auf die Datei verweisen.

So verwenden Sie SDKs zum Aufrufen von Amazon-EMR-APIs

Themen

Die AWS SDKs bieten Funktionen, die die API umschließen und sich um viele Verbindungsdetails kümmern, z. B. um die Berechnung von Signaturen, die Bearbeitung von Wiederholungsversuchen von Anfragen und die Fehlerbehandlung. Die SDKs enthalten außerdem Beispielcode, Tutorials und andere Ressourcen, die Ihnen den Einstieg in das Schreiben von Anwendungen erleichtern, die aufrufen. AWS Das Aufrufen der Wrapper-Funktionen in einem SDK kann das Schreiben einer AWS Anwendung erheblich vereinfachen.

Weitere Informationen zum Herunterladen und Verwenden der AWS SDKs finden Sie unter SDKs unter Tools für Amazon Web Services.

Verwenden von AWS SDK for Java , um einen Amazon EMR-Cluster zu erstellen

Das AWS SDK for Java bietet drei Pakete mit Amazon EMR-Funktionalität:

- com.amazonaws.services.elasticmapreduce
- com.amazonaws.services.elasticmapreduce.model
- com.amazonaws.services.elasticmapreduce.util

Weitere Informationen zu diesen Paketen finden Sie in der AWS SDK for Java -API-Referenz.

Das folgende Beispiel veranschaulicht, wie die SDKs die Programmierung mit Amazon EMR vereinfachen. Das folgende Codebeispiel verwendet das `StepFactory`-Objekt (eine Hilfsklasse zum Erstellen von typischen Amazon-EMR-Schritttypen) zum Erstellen eines interaktiven Hive-Clusters mit aktiviertem Debugging.

```
import com.amazonaws.AmazonClientException;
import com.amazonaws.auth.AWSCredentials;
import com.amazonaws.auth.AWSStaticCredentialsProvider;
import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.services.elasticmapreduce.AmazonElasticMapReduce;
import com.amazonaws.services.elasticmapreduce.AmazonElasticMapReduceClientBuilder;
import com.amazonaws.services.elasticmapreduce.model.*;
import com.amazonaws.services.elasticmapreduce.util.StepFactory;

public class Main {

 public static void main(String[] args) {
  AWSCredentialsProvider profile = null;
  try {
   credentials_profile = new ProfileCredentialsProvider("default"); // specifies any
 named profile in
                                    // .aws/credentials as the credentials provider
  } catch (Exception e) {
   throw new AmazonClientException(
     "Cannot load credentials from .aws/credentials file. " +
       "Make sure that the credentials file exists and that the profile name is defined
 within it.",
     e);
  }

  // create an EMR client using the credentials and region specified in order to
  // create the cluster
  AmazonElasticMapReduce emr = AmazonElasticMapReduceClientBuilder.standard()
    .withCredentials(credentials_profile)
    .withRegion(Regions.US_WEST_1)
    .build();

  // create a step to enable debugging in the AWS Management Console
  StepFactory stepFactory = new StepFactory();
  StepConfig enabledebugging = new StepConfig()
    .withName("Enable debugging")
    .withActionOnFailure("TERMINATE_JOB_FLOW")
    .withHadoopJarStep(stepFactory.newEnableDebuggingStep());
```

```
 // specify applications to be installed and configured when EMR creates the
 // cluster
 Application hive = new Application().withName("Hive");
 Application spark = new Application().withName("Spark");
 Application ganglia = new Application().withName("Ganglia");
 Application zeppelin = new Application().withName("Zeppelin");

 // create the cluster
 RunJobFlowRequest request = new RunJobFlowRequest()
   .withName("MyClusterCreatedFromJava")
   .withReleaseLabel("emr-5.20.0") // specifies the EMR release version label, we
recommend the latest release
   .withSteps(enabledebugging)
   .withApplications(hive, spark, ganglia, zeppelin)
   .withLogUri("s3://path/to/my/emr/logs") // a URI in S3 for log files is required
when debugging is enabled
   .withServiceRole("EMR_DefaultRole") // replace the default with a custom IAM
service role if one is used
   .withJobFlowRole("EMR_EC2_DefaultRole") // replace the default with a custom EMR
role for the EC2 instance
                       // profile if one is used
   .withInstances(new JobFlowInstancesConfig()
     .withEc2SubnetId("subnet-12ab34c56")
     .withEc2KeyName("myEc2Key")
     .withInstanceCount(3)
     .withKeepJobFlowAliveWhenNoSteps(true)
     .withMasterInstanceType("m4.large")
     .withSlaveInstanceType("m4.large"));

 RunJobFlowResult result = emr.runJobFlow(request);
 System.out.println("The cluster ID is " + result.toString());

 }

}
```

Sie müssen mindestens eine Service- und eine Jobflow-Rolle übergeben, die EMR_ bzw. EMR_EC2_ DefaultRole entsprechen. DefaultRole Sie können dies tun, indem Sie diesen Befehl für dasselbe Konto aufrufen. AWS CLI Überprüfen Sie zuerst, ob die Rollen bereits vorhanden sind:

```
aws iam list-roles | grep EMR
```

Sowohl das Instanzprofil (EMR_EC2_DefaultRole) als auch die Servicerolle (EMR_DefaultRole) werden angezeigt, sofern sie existieren:

```
"RoleName": "EMR_DefaultRole",
            "Arn": "arn:aws:iam::AccountID:role/EMR_DefaultRole"
            "RoleName": "EMR_EC2_DefaultRole",
            "Arn": "arn:aws:iam::AccountID:role/EMR_EC2_DefaultRole"
```

Wenn die Standardrollen nicht vorhanden sind, können Sie sie über den folgenden Befehl erstellen:

```
aws emr create-default-roles
```

Amazon EMR Service Quotas verwalten

Themen

In den Themen in diesem Abschnitt werden EMR-Dienstkontingente (früher als Service Limits bezeichnet) beschrieben, wie sie in der verwaltet werden und wann es von Vorteil ist AWS Management Console, CloudWatch Ereignisse anstelle von Servicekontingenten zu verwenden, um Cluster zu überwachen und Aktionen auszulösen.

Was sind Amazon EMR Service Quotas?

Ihr AWS Konto verfügt über Standard-Servicekontingenten, auch Limits genannt, für jeden AWS Dienst. Für den EMR-Service gibt es zwei Arten von Grenzwerten:

- Ressourcenbeschränkungen – Sie können EMR verwenden, um EC2-Ressourcen zu erstellen. Diese EC2-Ressourcen unterliegen jedoch Service Quotas. Die Ressourcenbeschränkungen in dieser Kategorie sind:
  - Die maximale Anzahl der aktiven Cluster, die gleichzeitig ausgeführt werden können.
  - Die maximale Anzahl aktiver Instances pro Instance-Gruppe.
- Limits für APIs – Bei der Verwendung von EMR-APIs gibt es zwei Arten von Einschränkungen:

- Burst-Limit – Dies ist die maximale Anzahl von API-Aufrufen, die Sie gleichzeitig tätigen können. Beispielsweise ist die maximale Anzahl von AddInstanceFleet API-Anfragen, die Sie pro Sekunde stellen können, standardmäßig auf 5 Aufrufe/Sekunde festgelegt. Dies bedeutet, dass das Burst-Limit der AddInstanceFleet API bei 5 Aufrufen/Sekunde liegt oder dass Sie zu einem bestimmten Zeitpunkt maximal 5 API-Aufrufe tätigen können. AddInstanceFleet Nachdem Sie das Burst-Limit verwendet haben, sind Ihre nachfolgenden Aufrufe jedoch durch das Ratenlimit begrenzt.
- Ratenlimit – Dies ist die Wiederauffüllrate der Burst-Kapazität der API. Beispielsweise ist die Wiederauffüllrate von AddInstanceFleet Anrufen standardmäßig auf 0,5 Anrufe/Sekunde festgelegt. Das bedeutet, dass Sie, nachdem Sie das Burst-Limit erreicht haben, mindestens 2 Sekunden warten müssen (0,5 Aufrufe/Sekunde X 2 Sekunden = 1 Anruf), um den API-Aufruf zu tätigen. Wenn Sie vorher einen Aufruf tätigen, werden Sie vom EMR-Webservice gedrosselt. Zu jedem Zeitpunkt können Sie nur so viele Aufrufe tätigen, wie die Burst-Kapazität ausreicht, ohne dass dies gedrosselt wird. Mit jeder weiteren Sekunde, die Sie warten, erhöht sich Ihre Burst-Kapazität um 0,5 Aufrufe, bis sie das maximale Limit von 5, dem Burst-Limit, erreicht.

Amazon EMR Service Quotas verwalten

Service Quotas ist eine AWS Funktion, mit der Sie Ihre Amazon EMR-Servicekontingente oder -Limits über die API oder die AWS Management Console CLI von einem zentralen Ort aus einsehen und verwalten können. Weitere Informationen zum Anzeigen von Quotas und zum Beantragen einer Erhöhung finden Sie unter AWS -Service Quotas in der Allgemeine Amazon Web Services-Referenz.

Für einige APIs ist die Einrichtung einer CloudWatch Veranstaltung möglicherweise die bessere Option als die Erhöhung der Servicekontingenten. Sie können auch Zeit sparen, indem CloudWatch Sie proaktiv Alarme einrichten und Erhöhungsanforderungen auslösen, bevor Sie das Servicekontingent erreichen. Weitere Details finden Sie unter Wann sollten EMR-Ereignisse eingerichtet werden in CloudWatch.

Wann sollten EMR-Ereignisse eingerichtet werden in CloudWatch

Bei einigen Abfrage-APIs, wie DescribeCluster, und DescribeStep ListClusters, kann die Einrichtung eines CloudWatch Ereignisses die Reaktionszeit auf Änderungen reduzieren und Ihre Servicekontingenten freisetzen. Wenn Sie beispielsweise eine Lambda-Funktion so eingerichtet haben, dass sie ausgeführt wird, wenn sich der Status eines Clusters ändert, z. B. wenn ein Schritt abgeschlossen oder ein Cluster beendet wird, können Sie diesen Auslöser verwenden, um die nächste Aktion in Ihrem Workflow zu starten, anstatt auf die nächste Abfrage zu warten. Andernfalls,

wenn Sie über dedizierte Amazon-EC2-Instances oder Lambda-Funktionen verfügen, die ständig die EMR-API nach Änderungen abfragen, verschwenden Sie nicht nur Rechenressourcen, sondern könnten auch Ihr Service Quota erreichen.

Im Folgenden sind einige Fälle aufgeführt, in denen Sie von einer Umstellung auf eine ereignisgesteuerte Architektur profitieren könnten.

Fall 1: EMR-Abfrage mithilfe von DescribeCluster API-Aufrufen zur Schrittabwicklung

Example EMR mithilfe von DescribeCluster API-Aufrufen zur Schrittabwicklung abfragen

Ein gängiges Muster besteht darin, einen Schritt an einen laufenden Cluster zu senden und Amazon EMR nach dem Status des Schritts abzufragen, normalerweise mithilfe der DescribeStep APIs DescribeCluster oder. Diese Aufgabe kann auch mit minimaler Verzögerung erledigt werden, indem Sie sich in das Amazon-EMR-Schrittstatusänderungsereignis einklinken.

Dieses Ereignis enthält die folgenden Informationen in seiner Nutzlast.

```
{
  "version": "0",
  "id": "999cccaa-eaaa-0000-1111-123456789012",
  "detail-type": "EMR Step Status Change",
  "source": "aws.emr",
  "account": "123456789012",
  "time": "2016-12-16T20:53:09Z",
  "region": "us-east-1",
  "resources": [],
  "detail": {
    "severity": "ERROR",
    "actionOnFailure": "CONTINUE",
    "stepId": "s-ZYXWVUTSRQPON",
    "name": "CustomJAR",
    "clusterId": "j-123456789ABCD",
    "state": "FAILED",
    "message": "Step s-ZYXWVUTSRQPON (CustomJAR) in Amazon EMR cluster j-123456789ABCD
 (Development Cluster) failed at 2016-12-16 20:53 UTC."
  }
}
```

In der Detailmap könnte eine Lambda-Funktion nach „state“, „stepId“ oder „clusterId“ suchen, um relevante Informationen zu finden.

Fall 2: EMR nach verfügbaren Clustern abfragen, um Workflows auszuführen

Example EMR nach verfügbaren Clustern abfragen, um Workflows auszuführen

Ein Muster für Kunden, die mehrere Cluster ausführen, besteht darin, Workflows auf Clustern auszuführen, sobald sie verfügbar sind. Wenn viele Cluster laufen und ein Workflow auf einem wartenden Cluster ausgeführt werden muss, könnte ein Muster darin bestehen, EMR mithilfe von API-Aufrufen DescribeCluster oder ListClusters API-Aufrufen nach verfügbaren Clustern abzufragen. Eine weitere Möglichkeit, die Verzögerung zu verringern, wenn Sie wissen, wann ein Cluster für einen Schritt bereit ist, besteht darin, das Amazon-EMR-Cluster-Statusänderungsereignis in zu verarbeiten.

Dieses Ereignis enthält die folgenden Informationen in seiner Nutzlast.

```
{
  "version": "0",
  "id": "999cccaa-eaaa-0000-1111-123456789012",
  "detail-type": "EMR Cluster State Change",
  "source": "aws.emr",
  "account": "123456789012",
  "time": "2016-12-16T20:43:05Z",
  "region": "us-east-1",
  "resources": [],
  "detail": {
    "severity": "INFO",
    "stateChangeReason": "{\"code\":\"\"}",
    "name": "Development Cluster",
    "clusterId": "j-123456789ABCD",
    "state": "WAITING",
    "message": "Amazon EMR cluster j-123456789ABCD ..."
  }
}
```

Für dieses Ereignis könnte eine Lambda-Funktion eingerichtet werden, um einen wartenden Workflow sofort an einen Cluster zu senden, sobald sich sein Status in WAITING ändert.

Fall 3: EMR nach Cluster-Terminierung abfragen

Example EMR nach Cluster-Terminierung abfragen

Kunden, die viele EMR-Cluster betreiben, fragen häufig bei Amazon EMR nach beendeten Clustern ab, sodass keine Arbeit mehr an Amazon EMR gesendet wird. Sie können dieses Muster mit den

ListClusters API-Aufrufen DescribeCluster und oder mithilfe des Amazon EMR Cluster State Change-Ereignisses in implementieren.

Nach der Clusterbeendigung sieht das ausgegebene Ereignis wie im folgenden Beispiel aus.

```
{
  "version": "0",
  "id": "1234abb0-f87e-1234-b7b6-000000123456",
  "detail-type": "EMR Cluster State Change",
  "source": "aws.emr",
  "account": "123456789012",
  "time": "2016-12-16T21:00:23Z",
  "region": "us-east-1",
  "resources": [],
  "detail": {
    "severity": "INFO",
    "stateChangeReason": "{\"code\":\"USER_REQUEST\",\"message\":\"Terminated by user
 request\"}",
    "name": "Development Cluster",
    "clusterId": "j-123456789ABCD",
    "state": "TERMINATED",
    "message": "Amazon EMR Cluster jj-123456789ABCD (Development Cluster) has
 terminated at 2016-12-16 21:00 UTC with a reason of USER_REQUEST."
  }
}
```

Der Abschnitt „Detail“ der Nutzlast enthält die ClusterID und den Status, auf den reagiert werden kann.

AWS-Glossar

Die neueste AWS-Terminologie finden Sie im AWS-Glossar in der AWS-Glossar-Referenz.

Die vorliegende Übersetzung wurde maschinell erstellt. Im Falle eines Konflikts oder eines Widerspruchs zwischen dieser übersetzten Fassung und der englischen Fassung (einschließlich infolge von Verzögerungen bei der Übersetzung) ist die englische Fassung maßgeblich.