



Guía del usuario

Amazon ElastiCache (RedisOSS)



Versión de API 2015-02-02

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon ElastiCache (RedisOSS): Guía del usuario

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Las marcas comerciales y la imagen comercial de Amazon no se pueden utilizar en relación con ningún producto o servicio que no sea de Amazon de ninguna manera que pueda causar confusión entre los clientes y que menosprecie o desacredite a Amazon. Todas las demás marcas registradas que no son propiedad de Amazon son propiedad de sus respectivos propietarios, que pueden o no estar afiliados, conectados o patrocinados por Amazon.

Table of Contents

¿Qué es ElastiCache (Redis OSS)?	1
Almacenamiento en caché sin servidor	1
Clústeres de autodiseño	2
Servicios relacionados	2
Funcionamiento	3
Caché y motores de almacenamiento en caché	3
Elección de la implementación	8
Comparación de características	10
Recursos de ElastiCache	16
AWS Regiones y zonas de disponibilidad	18
Casos de uso	19
Almacenamiento de datos en memoria	20
Tablas de clasificación de juegos (conjuntos ordenados en RedisOSS)	21
Mensajería (Redis Pub/SubOSS)	23
Datos de recomendación (hashes de Redis) OSS	26
Otros usos de Redis OSS	27
ElastiCache Testimonios de clientes	27
Cómo empezar con ElastiCache (Redis OSS)	28
Configuración	28
Inscríbase en una Cuenta de AWS	28
Creación de un usuario con acceso administrativo	29
Concesión de acceso programático	30
Configuración de permisos	32
Configurar EC2	33
Otorgar acceso de red	33
Configuración de redis-cli	34
Creación de una caché	35
Lectura y escritura de datos	36
Limpieza	38
Sigüientes pasos	39
Introducción a ElastiCache y a los SDK de AWS	40
Python y ElastiCache	40
Tutorial: Configuración de una función de Lambda para acceder a Amazon ElastiCache en una Amazon VPC	58

Paso 1: Crear una memoria caché sin servidor	58
Paso 2: Crear una función de Lambda	61
Paso 3: comprobación de la función de Lambda	65
Paso 4: limpiar (opcional)	66
Diseñar su propio clúster ElastiCache	68
Componentes y características	68
Nodos	69
ElastiCache Fragmentos (Redis OSS)	70
ElastiCache Clústeres (Redis OSS)	70
ElastiCache replicación	72
AWS Regiones y zonas de disponibilidad	74
ElastiCache puntos finales	75
Grupos de parámetros	76
ElastiCache seguridad	76
Grupos de subredes	77
ElastiCache (Redis OSS) copias de seguridad	77
Eventos	77
ElastiCache Terminología (Redis OSS)	79
Diseño de su propio clúster	82
Configuración de un grupo de subredes	82
Paso 1: crear un grupo de subredes	82
Paso 2: crear un clúster	85
Paso 3: autorizar acceso al clúster	93
Paso 4: conectarse al nodo de un clúster	96
Paso 5: eliminar un clúster	104
Tutoriales y videos	106
¿Qué tengo que hacer ahora?	112
Administración de nodos	113
Visualización ElastiCache del estado de los nodos	114
Nodos y fragmentos de Redis OSS	120
Conexión a los nodos	122
Tipos de nodos compatibles	125
Reinicio de nodos (solo modo de clúster deshabilitado)	136
Sustitución de nodos	138
Nodos reservados	146
Migración de nodos de generación anterior	158

Administración de clústeres	161
Elección de un tipo de red	163
Organización de datos en niveles	167
Preparación de un clúster	174
Creación de un clúster	182
Visualización de los detalles de un clúster	193
Modificación de un clúster	206
Adición de nodos a un clúster	211
Eliminación de nodos de un clúster	219
Cancelación de operaciones pendientes de adición o eliminación de nodos	227
Eliminación de un clúster	228
Acceso al clúster o al grupo de reproducción	231
Búsqueda de puntos de conexión	237
Particiones	249
Comparación de las cachés autodiseñadas de Memcached y Redis OSS	254
Migración en línea a ElastiCache	260
Información general	260
Pasos para realizar la migración	261
Preparación de los nodos OSS de Redis de origen y destino para la migración	261
Prueba de la migración de datos	263
Inicio de la migración	264
Verificación del progreso de migración de datos	265
Finalización de la migración de datos	266
Realización de la migración de datos en línea con la consola	267
Elección de regiones y zonas de disponibilidad	268
Ubicación de los nodos	270
Regiones y puntos de enlace admitidos	270
Uso de Local zones	275
Uso de Outposts	277
Trabajando con ElastiCache	281
Instantánea y restauración	281
Restricciones	282
El impacto en el rendimiento de las copias de seguridad de los clústeres de autodiseño	283
Programación de copias de seguridad automáticas	285
Copias de seguridad manuales	286
Creación de una copia de seguridad final	292

Descripción de copias de seguridad	295
Copiar copias de seguridad	297
Exportación de una copia de seguridad	300
Restauración desde una copia de seguridad	308
Eliminación de una copia de seguridad	310
Etiquetado de copias de seguridad	312
Inicialización de datos en un clúster de autodiseño con una copia de seguridad	313
Versiones del motor y actualizaciones	322
Versiones del motor y actualizaciones	323
Versiones de Redis OSS compatibles	328
Calendario de fin de vida de las versiones de Redis OSS	342
Cómo actualizar las versiones del motor	326
Resolución de actualizaciones del motor bloqueadas	327
Diferencias principales de compatibilidad y comportamiento de versiones	345
Procedimientos recomendados y estrategias de almacenamiento en caché	349
Trabajando con Redis OSS	350
Mejores prácticas con los clientes de Redis OSS	390
Administración de la memoria reservada	418
Procedimientos recomendados al trabajar con clústeres de autodiseño	425
Mejores prácticas de Redis OSS	431
Estrategias de almacenamiento en caché	433
Administración de un clúster de autodiseño	438
Clústeres de Auto Scaling ElastiCache (Redis OSS)	439
Modificación del modo de clúster	488
Replicación entre AWS regiones mediante almacenes de datos globales	491
Alta disponibilidad a través de grupos de reproducción	519
Administración del mantenimiento	609
Configuración de los parámetros de motor mediante los grupos de parámetros	612
Escalado ElastiCache (Redis OSS)	715
Escalado sin servidor ElastiCache	715
Establecimiento de límites de escalado para administrar los costes	716
Escalado previo con Serverless ElastiCache	716
Establecer límites de escalado mediante la consola y AWS CLI	717
Escalar clústeres de ElastiCache diseño propio (RedisOSS)	719
Introducción a JSON en ElastiCache (Redis OSS)	792
Descripción general del tipo de datos JSON de Redis OSS	793

Comandos JSON	805
Etiquetado de los recursos de ElastiCache	847
Monitoreo de costos con etiquetas	859
Administrar las etiquetas mediante el AWS CLI	860
Administrar las etiquetas mediante la ElastiCache API	864
Enfoque Well-Architected de Amazon ElastiCache	867
Pilar de excelencia operativa	868
Pilar de seguridad	877
Pilar de fiabilidad	883
Pilar de eficiencia de rendimiento	889
Pilar de optimización de costos	901
Resolución de problemas	908
Problemas de conectividad	908
Errores del cliente OSS de Redis	909
Solución de problemas de alta latencia en Serverless ElastiCache	910
Solución de problemas de limitación en Serverless ElastiCache	912
Temas relacionados	913
Pasos adicionales de solución de problemas	913
Grupos de seguridad	913
ACL de red	914
Tablas de enrutamiento	916
Resolución de los DNS	916
Identificación de los problemas con los diagnósticos del lado del servidor	916
Validación de la conectividad de red	923
Límites relacionados con la red	925
Uso de la CPU	926
Conexiones que terminan desde el lado del servidor	930
Solución de problemas del lado del cliente para instancias de Amazon EC2	931
Análisis del tiempo que se tarda en completar una sola solicitud	932
Seguridad	936
Protección de datos	937
Seguridad de datos en Amazon ElastiCache	938
Privacidad del tráfico entre redes	1011
Seguridad de ElastiCache y Amazon VPC	1012
Puntos de conexión de VPC de interfaz y API de Amazon ElastiCache (AWS PrivateLink)	1037
Subredes y grupos de subredes	1041

Identity and Access Management	1049
Público	1049
Autenticación con identidades	1050
Administración de acceso mediante políticas	1054
Cómo ElastiCache trabaja Amazon con IAM	1056
Ejemplos de políticas basadas en identidades	1063
Resolución de problemas	1066
Control de acceso	1068
Información general sobre la administración del acceso	1070
Validación de conformidad	1113
Más información	1115
Resiliencia	1115
Mitigación de errores	1116
Seguridad de la infraestructura	1119
Actualizaciones de servicio	1120
Administrar las actualizaciones de los servicios	1120
Se abordaron las vulnerabilidades de seguridad	1126
Registro y monitorización	1128
Eventos y métricas sin servidor	1128
Métricas sin servidor	1128
Eventos sin servidor	1138
Eventos y métricas de clústeres de autodiseño	1152
Métricas para clústeres de autodiseño	1153
Eventos de clústeres de autodiseño	1153
Entrega de registro	1162
Monitoreo del uso	1175
Monitorización de SNS eventos de Amazon	1205
Registro de llamadas a la API de Amazon ElastiCache con AWS CloudTrail	1223
Información sobre Amazon ElastiCache en CloudTrail	1223
Descripción de las entradas de archivo de registros de Amazon ElastiCache	1224
Cuotas	1229
Referencia	1231
Uso de la API de ElastiCache	1231
Uso de la API de consultas	1231
Bibliotecas disponibles	1235
Solución de problemas de aplicaciones	1235

Configurar la AWS CLI de ElastiCache	1236
Requisitos previos	1237
Obtención de las herramientas de línea de comandos	1239
Configuración de las herramientas	1239
Suministro de credenciales para las herramientas	1240
Variables de entorno	1241
Mensajes de error	1242
Notificaciones	1244
ElastiCache Notificaciones generales	1244
ElastiCache (Redis OSS) notificaciones específicas	1244
ElastiCache Historial de documentación de (RedisOSS)	1245
AWS Glosario	1280
.....	mcclxxxix

¿Qué es Amazon ElastiCache (Redis OSS)?

Bienvenido a la Guía del usuario de Amazon ElastiCache (Redis OSS). Amazon ElastiCache es un servicio web que facilita la configuración, la administración y el escalado de un almacén de datos en memoria distribuido o un entorno de caché en la nube. Proporciona una capacidad de almacenamiento en caché de alto rendimiento, escalable y rentable. Al mismo tiempo, ayuda a eliminar la complejidad propia de la implementación y la administración de un entorno de caché distribuido.

Puedes operar Amazon ElastiCache en dos formatos. Puede empezar con una memoria caché sin servidor o diseñar su propio clúster de caché.

Note

Amazon ElastiCache funciona con los motores Redis OSS y Memcached. Utilice la guía del motor en la que esté interesado. Si no está seguro de qué motor desea utilizar, consulte [Comparación de las cachés autodiseñadas de Memcached y Redis OSS](#) en esta guía.

Almacenamiento en caché sin servidor

ElastiCache (Redis OSS) ofrece almacenamiento en caché sin servidor, lo que simplifica la adición y el funcionamiento de una caché basada en Redis para su aplicación. ElastiCache (Redis OSS) Sin servidor le permite crear una caché de alta disponibilidad en menos de un minuto y elimina la necesidad de aprovisionar instancias o configurar nodos o clústeres. Los desarrolladores pueden crear una caché sin servidor especificando el nombre de la caché mediante la ElastiCache consola, el SDK o la CLI.

ElastiCache (Redis OSS) Serverless también elimina la necesidad de planificar y administrar la capacidad de almacenamiento en caché. ElastiCache (Redis OSS) supervisa constantemente la memoria caché, el cómputo y el ancho de banda de red que utiliza la aplicación, y se amplía para adaptarse a las necesidades de la aplicación. ElastiCache (Redis OSS) ofrece una experiencia de punto final sencilla para los desarrolladores, al abstraer la infraestructura de caché subyacente y el diseño del clúster. ElastiCache (Redis OSS) gestiona el aprovisionamiento del hardware, la supervisión, la sustitución de nodos y la aplicación de parches de software de forma automática y transparente, para que pueda centrarse en el desarrollo de las aplicaciones, en lugar de en utilizar la memoria caché.

ElastiCache (Redis OSS) Serverless es compatible con Redis OSS 7.1 y versiones posteriores.

Diseñe su propio clúster ElastiCache (Redis OSS)

Si necesita un control detallado de su clúster ElastiCache (Redis OSS), puede optar por diseñar su propio clúster de Redis OSS con él. ElastiCache le permite diseñar su clúster eligiendo el tipo de nodo, la cantidad de nodos y la ubicación de los nodos en las zonas de disponibilidad del clúster. Como ElastiCache se trata de un servicio totalmente gestionado, gestiona automáticamente el aprovisionamiento del hardware, la supervisión, la sustitución de nodos y la aplicación de parches de software para el clúster.

Diseñar su propio clúster ElastiCache (Redis OSS) ofrece una mayor flexibilidad y control sobre los clústeres. Por ejemplo, puede operar un clúster con disponibilidad en una o muchas zonas de disponibilidad, en función de sus necesidades. También puede optar por ejecutar Redis OSS en modo de clúster, lo que permite el escalado horizontal, o sin modo de clúster para escalar solo verticalmente. Al diseñar sus propios clústeres, usted será el responsable de elegir el tipo y la cantidad de nodos correctamente para garantizar que la memoria caché tenga la capacidad suficiente según lo requiera la aplicación. También puede elegir cuándo aplicar los nuevos parches de software a su clúster de Redis OSS.

Al diseñar su propio clúster ElastiCache (Redis OSS), puede optar por ejecutar Redis OSS 3.0 y versiones posteriores.

Servicios relacionados

[Base de datos de memoria](#)

Al decidir si utilizar ElastiCache (Redis OSS) o MemoryDB, tenga en cuenta las siguientes comparaciones:

- ElastiCache (Redis OSS) es un servicio que se utiliza habitualmente para almacenar en caché datos de otras bases de datos y almacenes de datos que utilizan Redis OSS. Debería tener en cuenta ElastiCache (Redis OSS) para almacenar en caché las cargas de trabajo en las que desee acelerar el acceso a los datos con su base de datos principal o almacén de datos existente (rendimiento de lectura y escritura en microsegundos). También debería tener en cuenta ElastiCache (Redis OSS) en los casos de uso en los que desee utilizar las estructuras de datos y las API de Redis OSS para acceder a los datos almacenados en una base de datos o almacén de datos principal.

- MemoryDB es una base de datos en memoria duradera para cargas de trabajo que requieren una base de datos principal ultrarrápida. Debería considerar el uso de MemoryDB si la carga de trabajo requiere una base de datos duradera que ofrezca un rendimiento ultrarrápido (lectura en microsegundos y latencia de escritura de un solo dígito en milisegundos). MemoryDB también puede ser una buena opción para su caso de uso si desea crear una aplicación utilizando las estructuras de datos y las API de Redis OSS con una base de datos principal y duradera. Por último, debería considerar el uso de MemoryDB para simplificar la arquitectura de la aplicación y reducir los costos al sustituir el uso de una base de datos por una memoria caché para aumentar la durabilidad y el rendimiento.

[Amazon RDS](#)

ElastiCache (Redis OSS) puede ayudarle a ahorrar costes en la base de datos al almacenar en caché los datos a los que se accede con frecuencia. Si su aplicación tiene requisitos de alto rendimiento de lectura, puede lograr un rendimiento rápido y de alta escala y reducir los costes de almacenamiento de datos utilizando ElastiCache, en lugar de escalar, la base de datos subyacente.

Funcionamiento

Aquí puede encontrar una descripción general de los componentes principales de una implementación ElastiCache (Redis OSS).

Caché y motores de almacenamiento en caché

Una caché es un almacén de datos en memoria que puede utilizar para almacenar datos en caché. Por lo general, la aplicación almacenará en caché los datos a los que se accede con frecuencia para optimizar los tiempos de respuesta. ElastiCache (Redis OSS) ofrece dos opciones de implementación: clústeres sin servidor y clústeres de diseño propio. Consulte [Cómo elegir una opción de implementación](#)

Note

Amazon ElastiCache funciona con los motores Redis OSS y Memcached. Utilice la guía del motor en la que esté interesado. Si no está seguro de qué motor desea utilizar, consulte [Comparación de las cachés autodiseñadas de Memcached y Redis OSS](#) en esta guía.

Temas

- [¿Cómo funciona ElastiCache \(Redis OSS\)](#)
- [Dimensiones de precios](#)
- [ElastiCache Copias de seguridad \(Redis OSS\)](#)

¿Cómo funciona ElastiCache (Redis OSS)

ElastiCache (Redis OSS) Sin servidor

ElastiCache (Redis OSS) Sin servidor le permite crear una memoria caché sin preocuparse por la planificación de la capacidad, la administración del hardware o el diseño del clúster. Solo tiene que proporcionar un nombre para la memoria caché y recibirá un único punto de conexión que podrá configurar en su cliente OSS de Redis para empezar a acceder a la memoria caché.

Note

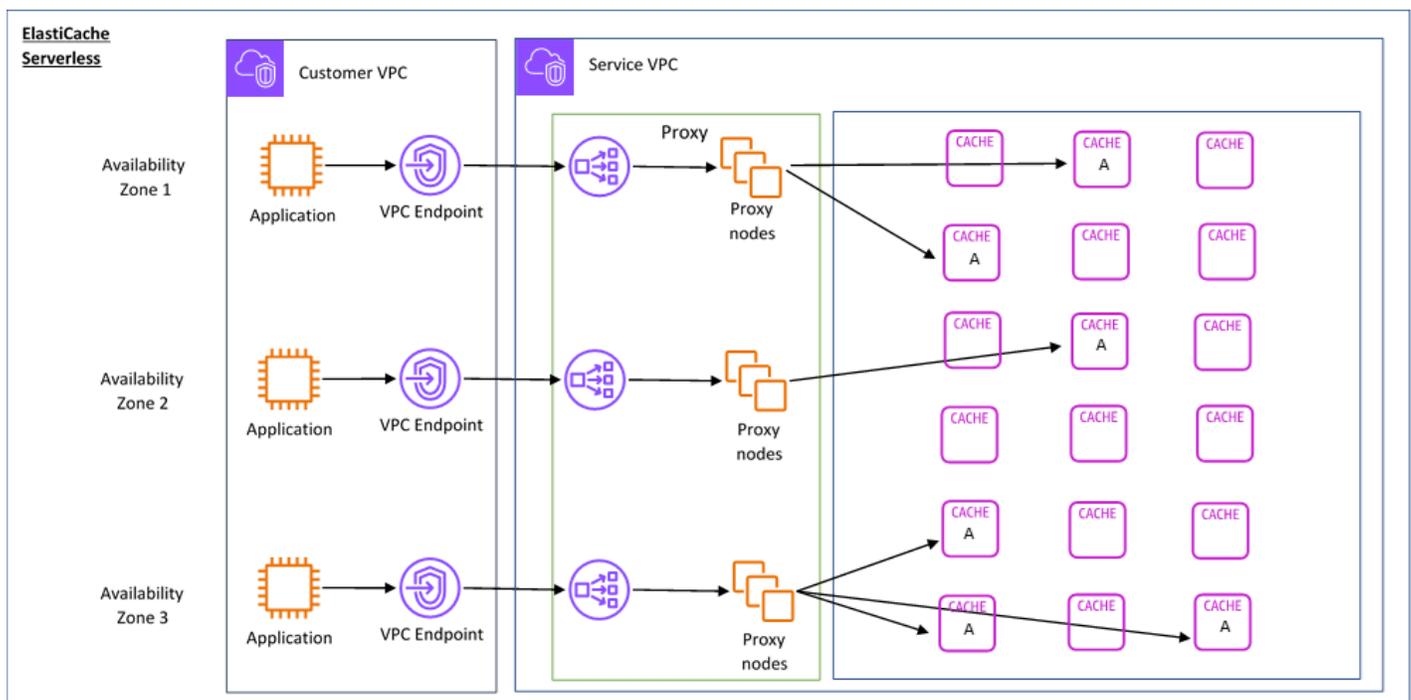
ElastiCache (Redis OSS) Serverless ejecuta Redis OSS en modo clúster y solo es compatible con los clientes de Redis OSS que admiten TLS y el protocolo de clúster Redis OSS.

Ventajas principales

- Sin planificación de la capacidad: ElastiCache Serverless elimina la necesidad de planificar la capacidad. ElastiCache Serverless monitorea continuamente el uso de la memoria, el cómputo y el ancho de banda de la red de la memoria caché y escala tanto vertical como horizontalmente. Permite que un nodo de caché aumente de tamaño y, en paralelo, inicia una operación de escalado horizontal para garantizar que la caché pueda escalarse a fin de cumplir con los requisitos de la aplicación en todo momento.
- Pay-per-use: Con ElastiCache Serverless, usted paga por los datos almacenados y el procesamiento utilizados por su carga de trabajo en la memoria caché. Consulte [Dimensiones de precios](#).
- Alta disponibilidad: ElastiCache Serverless replica automáticamente sus datos en varias zonas de disponibilidad (AZ) para lograr una alta disponibilidad. Supervisa automáticamente los nodos de caché subyacentes y los reemplaza en caso de errores. Ofrece un SLA de disponibilidad del 99,99 % para cada caché.

- **Actualizaciones de software automáticas:** ElastiCache Serverless actualiza automáticamente la memoria caché a la última versión secundaria y actualiza el software sin que ello afecte a la disponibilidad de la aplicación. Cuando haya disponible una nueva versión principal de Redis OSS, ElastiCache le enviará una notificación.
- **Seguridad:** la tecnología sin servidor cifra siempre los datos en tránsito y en reposo. Puede utilizar una clave administrada por el servicio o su propia clave administrada por el cliente para cifrar los datos en reposo.

El siguiente diagrama ilustra cómo funciona ElastiCache Serverless.



Al crear una nueva caché sin servidor, ElastiCache crea un punto final de Virtual Private Cloud (VPC) en las subredes que elija de la VPC. La aplicación se puede conectar a la memoria caché a través de estos puntos de conexión de VPC.

Con ElastiCache Serverless, recibirá un único punto de enlace DNS al que se conectará su aplicación. Cuando solicita una nueva conexión al punto final, ElastiCache Serverless gestiona todas las conexiones de caché a través de una capa de proxy. La capa de proxy contribuye a reducir la compleja configuración del cliente, ya que este no necesita volver a detectar de nuevo la topología del clúster en caso de que haya cambios en el clúster subyacente. La capa proxy es un conjunto de nodos proxy que gestionan las conexiones mediante un equilibrador de carga de red. Cuando la aplicación crea una nueva conexión de caché, el equilibrador de carga de red envía la solicitud a un

nodo proxy. Cuando la aplicación ejecuta comandos de caché, el nodo proxy que está conectado a la aplicación ejecuta las solicitudes en un nodo de caché de la memoria caché. La capa proxy abstrae la topología del clúster de caché y los nodos del cliente. Esto permite ElastiCache equilibrar la carga, escalar y añadir nuevos nodos de caché de forma inteligente, reemplazar los nodos de caché cuando fallan y actualizar el software de los nodos de caché, todo ello sin que la disponibilidad afecte a la aplicación ni tenga que restablecer las conexiones.

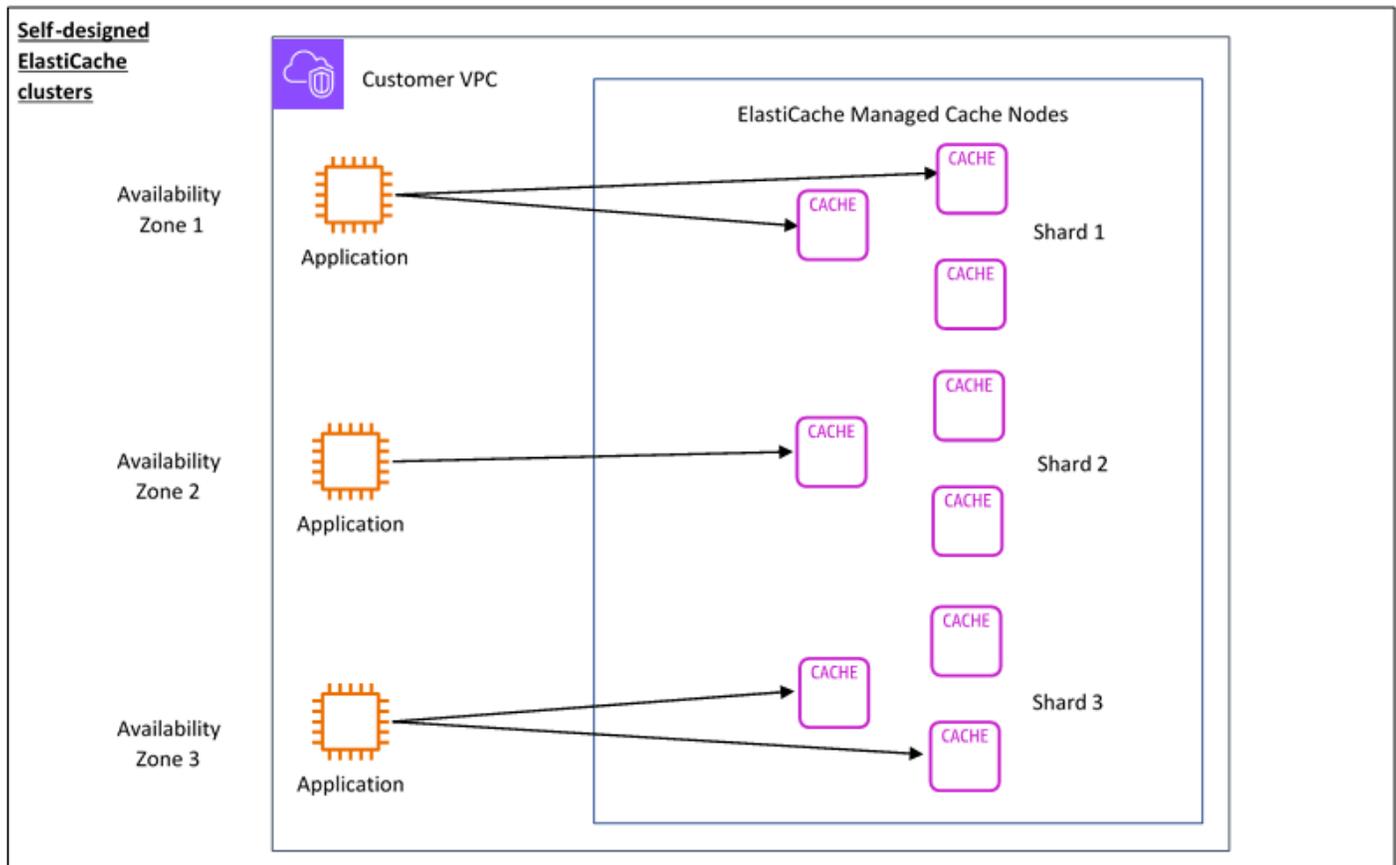
Clústeres de diseño propio ElastiCache

Puede optar por diseñar sus propios ElastiCache clústeres seleccionando una familia de nodos de caché, un tamaño y una cantidad de nodos para su clúster. Diseñar su propio clúster le proporciona un control más detallado y le permite elegir la cantidad de particiones de la memoria caché y el número de nodos (principales y réplicas) de cada partición. Puede optar por utilizar Redis OSS en modo clúster creando un clúster con varios fragmentos o en modo sin clúster con un solo fragmento.

Ventajas principales

- **Diseñe su propio clúster:** con él ElastiCache, puede diseñar su propio clúster y elegir dónde quiere colocar los nodos de la memoria caché. Por ejemplo, si tiene una aplicación en la que quiere sacrificar una alta disponibilidad por una baja latencia, puede optar por implementar los nodos de caché en una única zona de disponibilidad. Si lo prefiere, puede diseñar el clúster con nodos en varias zonas de disponibilidad para lograr una alta disponibilidad.
- **Control detallado:** al diseñar nuestro propio clúster, tenemos más control a la hora de ajustar la configuración de la memoria caché. Por ejemplo, puede utilizar [Parámetros específicos de Redis OSS](#) para configurar el motor de la caché.
- **Escalar vertical y horizontalmente:** puede escalar el clúster manualmente aumentando o disminuyendo el tamaño del nodo de caché cuando sea necesario. Además, puede escalar horizontalmente añadiendo particiones nuevas o añadiendo más réplicas a sus particiones. También puedes usar la función de escalado automático para configurar el escalado en función de un cronograma o el escalado en función de métricas como el uso de CPU y memoria en la memoria caché.

El siguiente diagrama ilustra cómo funcionan los clústeres de ElastiCache diseño propio.



Dimensiones de precios

Puede realizar la implementación ElastiCache en dos opciones de implementación. Al implementar ElastiCache Serverless, se paga por el uso de los datos almacenados en GB-hora y el procesamiento en unidades de ElastiCache procesamiento (ECPU). Si decide diseñar sus propios clústeres ElastiCache (Redis OSS), paga por hora de uso del nodo de caché. Puede encontrar más información sobre los precios [aquí](#).

Almacenamiento de datos

Usted paga por los datos almacenados en ElastiCache Serverless y facturados en gigabytes-hora (GB-hora). ElastiCache Serverless supervisa de forma continua los datos almacenados en la memoria caché, toma muestras varias veces por minuto y calcula un promedio por hora para determinar el uso de almacenamiento de datos de la memoria caché en GB por hora. Cada caché ElastiCache sin servidor se mide para un mínimo de 1 GB de datos almacenados.

ElastiCache Unidades de procesamiento (ECPUs)

Usted paga por las solicitudes de Redis OSS que su aplicación ejecuta en unidades de ElastiCache procesamiento (ECPUs) ElastiCache sin servidor, una unidad que incluye tanto el tiempo de vCPU como los datos transferidos.

- Las lecturas y escrituras sencillas requieren 1 ECPU por cada kilobyte (KB) de datos transferidos. Por ejemplo, un comando GET que transfiere hasta 1 KB de datos consume 1 ECPU. Una solicitud SET que transfiera 3,2 KB de datos consumirá 3,2 ECPU.
- Los comandos que requieren más tiempo de vCPU consumirán proporcionalmente más ECPU. Por ejemplo, si la aplicación utiliza el [comando HMGET](#) OSS de Redis y consume 3 veces el tiempo de vCPU que un simple comando SET/GET, consumirá 3 ECPUs.
- Los comandos que consumen más tiempo de vCPU y transfieren más datos consumen ECPU en función del valor más alto entre las dos dimensiones. Por ejemplo, si la aplicación usa el comando HMGET, consume el triple de tiempo de vCPU que un simple comando SET/GET y transfiere 3,2 KB de datos, consumirá 3,2 ECPU. Sin embargo, si transfiere solo 2 KB de datos, consumirá 3 ECPU.

ElastiCache Serverless emite una nueva métrica denominada `ElastiCacheProcessingUnits` que le ayuda a comprender las ECPUs que consume su carga de trabajo.

Horas de nodo

Puede optar por diseñar su propio clúster de caché de OSS de Redis eligiendo la familia de nodos EC2, el tamaño, la cantidad de nodos y la ubicación en las distintas zonas de disponibilidad. Al diseñar su propio clúster, pagará por hora por cada nodo de caché.

ElastiCache Copias de seguridad (Redis OSS)

Una copia de seguridad es una point-in-time copia de una memoria caché de Redis OSS. ElastiCache le permite realizar una copia de seguridad de sus datos en cualquier momento o configurar copias de seguridad automáticas. Dichas copias de seguridad pueden usarse para restaurar una caché existente o para propagar datos en una nueva caché. Las copias de seguridad se componen de todos los datos en una caché, además de algunos metadatos. Para obtener más información consulte [Instantánea y restauración](#).

Cómo elegir una opción de implementación

Amazon ElastiCache tiene dos opciones de implementación:

- Almacenamiento en caché sin servidor
- Clústeres de autodiseño

Para obtener una lista de los comandos compatibles con ambos, consulte [Comandos de Redis OSS compatibles y restringidos](#).

Almacenamiento en caché sin servidor

Amazon ElastiCache Serverless simplifica la creación de caché y se escala al instante para dar soporte a las aplicaciones más exigentes de los clientes. Con ElastiCache Serverless, puede crear una caché escalable y de alta disponibilidad en menos de un minuto, lo que elimina la necesidad de aprovisionar, planificar y administrar la capacidad del clúster de caché. ElastiCache Serverless almacena automáticamente los datos de forma redundante en tres zonas de disponibilidad y proporciona un acuerdo de nivel de servicio (SLA) de disponibilidad del 99,99%. Las copias de seguridad son compatibles entre sí y se pueden exportar y restaurar desde clústeres de diseño propio.

Clústeres de autodiseño

Si necesita un control detallado de su clúster ElastiCache (Redis OSS), puede optar por diseñar su propio clúster de Redis OSS. ElastiCache le permite operar un clúster basado en nodos al elegir el tipo de nodo, la cantidad de nodos y la ubicación de los nodos en las zonas de disponibilidad del clúster. AWS ElastiCache se trata de un servicio totalmente gestionado, ayuda a gestionar el aprovisionamiento del hardware, la supervisión, las sustituciones de nodos y los parches de software del clúster. Los clústeres de diseño propio se pueden diseñar para ofrecer un SLA de disponibilidad de hasta el 99,99%. Las copias de seguridad son compatibles entre sí y se pueden exportar y restaurar desde las cachés Serverless.

Cómo elegir una opción de implementación

Elija el almacenamiento en caché sin servidor en los siguientes casos:

- Está creando una memoria caché para cargas de trabajo nuevas o difíciles de predecir.
- El tráfico de la aplicación es impredecible.
- Desea optar por la forma más sencilla de comenzar a utilizar una caché.

Elija diseñar su propio ElastiCache clúster si:

- Ya utiliza ElastiCache Serverless y desea tener un control más preciso sobre el tipo de nodo que ejecuta Redis OSS, el número de nodos y la ubicación de dichos nodos.
- Espera que el tráfico de sus aplicaciones sea relativamente predecible y desea tener un control pormenorizado del rendimiento, la disponibilidad y el coste.
- Puede prever los requisitos de capacidad para controlar los costes.

Comparación del almacenamiento en caché sin servidor y los clústeres de diseño propio

Característica	Almacenamiento en caché sin servidor	Clústeres de autodiseño
Configuración de caché	Crea una caché con solo un nombre en menos de un minuto	Proporciona un control detallado sobre el diseño del clúster de caché. El usuario puede elegir el tipo de nodo, la cantidad de nodos y la ubicación en las distintas zonas de disponibilidad AWS
Versión compatible ElastiCache (Redis OSS)	ElastiCache (Redis OSS) versión 7.1 y superior	ElastiCache (Redis OSS) versión 4.0 y superior
Modo agrupado en clúster	Funciona únicamente con Redis OSS. <code>cluster mode enabled</code> Los clientes OSS de Redis deben ser compatibles con <code>cluster mode enabled</code> para conectarse a ElastiCache Serverless.	Se puede configurar para que funcione en modo clúster activado o modo clúster desactivado.
Escalado	Se escala automáticamente tanto vertical como horizontalmente sin administrar la capacidad.	Permite controlar el escalado y, al mismo tiempo, requiere supervisión para garantizar que la capacidad actual

Característica	Almacenamiento en caché sin servidor	Clústeres de autodiseño
		<p>satisfaga adecuadamente la demanda.</p> <p>Puede optar por escalar verticalmente, aumentando o disminuyendo el tamaño del nodo de caché cuando sea necesario. También puede escalar horizontalmente, añadiendo nuevos fragmentos o añadiendo más réplicas a los fragmentos.</p> <p>Con la función de escalado automático, también puede configurar el escalado en función de un cronograma o escalarlo en función de métricas como el uso de CPU y memoria en la memoria caché.</p>
Conexión de cliente	Los clientes se conectan a un único punto final. Esto permite que la topología del nodo de caché subyacente (escalado, reemplazos y actualizaciones) cambie sin desconectar el cliente.	Los clientes se conectan a cada nodo de caché individual. Si se reemplaza un nodo, el cliente vuelve a descubrir la topología del clúster y restablece las conexiones.

Característica	Almacenamiento en caché sin servidor	Clústeres de autodiseño
Configurabilidad	No hay una configuración detallada disponible. Los clientes pueden configurar los ajustes básicos, como las subredes que pueden acceder a la memoria caché, si se activan o desactivan las copias de seguridad automáticas y los límites máximos de uso de la memoria caché.	Los clústeres de diseño propio ofrecen opciones de configuración detalladas. Los clientes pueden usar grupos de parámetros para un control detallado. Para ver una tabla de estos valores de los parámetros por tipo de nodo, consulte Parámetros específicos del tipo de nodo de Redis OSS .
Multi-AZ	Los datos se replican de forma asíncrona en varias zonas de disponibilidad para aumentar la disponibilidad y mejorar la latencia de lectura.	Ofrece la opción de diseñar el clúster en una única zona de disponibilidad o en varias zonas de disponibilidad (AZ). En el caso de los clústeres Multi-AZ, los datos se replican de forma asíncrona en varias zonas de disponibilidad para aumentar la disponibilidad y mejorar la latencia de lectura.
Cifrado en reposo	Siempre habilitados. Los clientes pueden utilizar una clave Clave administrada de AWS o una gestionada por el cliente AWS KMS.	Opción para activar o desactivar el cifrado en reposo. Cuando está habilitada, los clientes pueden usar una clave Clave administrada de AWS o una entrada gestionada por el cliente AWS KMS.

Característica	Almacenamiento en caché sin servidor	Clústeres de autodiseño
Cifrado en tránsito (TLS)	Siempre activado. Los clientes deben admitir la conectividad TLS.	Opción para activar o desactivar.
Copias de seguridad	<p>Admite copias de seguridad automáticas y manuales de las cachés sin que ello afecte al rendimiento.</p> <p>Las copias de seguridad son compatibles entre sí y se pueden restaurar en una caché ElastiCache sin servidor o en un clúster de diseño propio.</p>	<p>Admite copias de seguridad automáticas y manuales. Es posible que el rendimiento de los clústeres se vea afectado en cierta medida en función de la memoria reservada disponible. Para obtener más información, consulte Administración de la memoria reservada.</p> <p>Las copias de seguridad son compatibles entre sí y se pueden restaurar en una caché ElastiCache sin servidor o en un clúster de diseño propio.</p>

Característica	Almacenamiento en caché sin servidor	Clústeres de autodiseño
Supervisión	<p>Support métricas de nivel de caché, incluida la tasa de aciertos de caché, la tasa de errores de caché, el tamaño de los datos y las ECPUs consumidas.</p> <p>ElastiCache Serverless envía los eventos EventBridge cuando ocurren eventos importantes en la memoria caché. Puedes elegir monitorizar, asimilar, transformar y actuar en función de ElastiCache los eventos con Amazon EventBridge. Para obtener más información, consulte Eventos de caché sin servidor.</p>	<p>ElastiCache Los clústeres de diseño propio emiten métricas en cada nivel de nodo, incluidas las métricas a nivel de host y las métricas de caché.</p> <p>Los clústeres de diseño propio emiten notificaciones de SNS en caso de eventos importantes. Consulte Métricas para Redis OSS.</p>
Disponibilidad	<p>Acuerdo de nivel de servicio (SLA) con una disponibilidad del 99,99%</p>	<p>Los clústeres de diseño propio se pueden diseñar para alcanzar un acuerdo de nivel de servicio (SLA) de disponibilidad de hasta el 99,99%, según la configuración.</p>

Característica	Almacenamiento en caché sin servidor	Clústeres de autodiseño
Actualizaciones y parches de software	<p>Actualiza automáticamente el software de caché a la última versión secundaria y a la versión de parche más reciente, sin que ello afecte a la aplicación. Los clientes reciben una notificación sobre las actualizaciones de las versiones principales y pueden actualizar a la versión principal más reciente cuando lo deseen.</p>	<p>Los clústeres de diseño propio ofrecen un autoservicio habilitado por el cliente para actualizaciones de versiones menores y con parches, así como para actualizaciones de versiones principales. Las actualizaciones gestionadas se aplican automáticamente durante los períodos de mantenimiento definidos por el cliente. Los clientes también pueden optar por aplicar una actualización de la versión secundaria o de un parche a pedido.</p>
Almacén de datos global	No compatible	<p>Es compatible con el almacén de datos global, que permite la replicación entre regiones con escrituras en una sola región y lecturas en varias regiones</p>

Característica	Almacenamiento en caché sin servidor	Clústeres de autodiseño
Organización de datos en niveles	No compatible	Los clústeres que se diseñan con nodos de la familia r6gd tienen sus datos agrupados en niveles entre la memoria y el almacenamiento SSD (unidades de estado sólido) local. La organización de datos en niveles ofrece una opción de relación precio-rendimiento para las cargas de trabajo de Redis OSS, ya que utiliza unidades de estado sólido (SSD) de menor costo en cada nodo del clúster, además de almacenar los datos en la memoria.
Modelo de precios	Pay-per-use, en función de los datos almacenados en GB-hora y las solicitudes en las unidades de procesamiento (ECPU). ElastiCache Puede encontrar más información sobre los precios aquí .	Pay-per-hour, en función del uso de los nodos de caché. Puede encontrar más información sobre los precios aquí .

Temas relacionados:

- [Diseñar y administrar su propio clúster ElastiCache \(Redis OSS\)](#)

Recursos de Amazon ElastiCache

Se recomienda empezar a leer las siguientes secciones y consultarlas conforme las necesite:

- Precios y aspectos resaltados del servicio: en la [página de detalles del producto](#) se proporciona información general sobre ElastiCache junto con los aspectos resaltados del servicio y los precios.
- Videos de ElastiCache: la sección [ElastiCache Vídeos](#) contiene videos que presentan a Amazon ElastiCache. En los videos se cubren casos de uso frecuentes de ElastiCache y se muestran cómo utilizar ElastiCache para reducir la latencia y mejorar el rendimiento de las aplicaciones.
- Introducción: la sección [Introducción a Amazon ElastiCache \(Redis OSS\)](#) incluye información sobre la creación de un clúster de caché. Incluye también cómo autorizar el acceso al clúster de caché, conectarse a un nodo de caché y eliminar el clúster de caché.
- Rendimiento a escala: en el documento técnico [Performance at scale with Amazon ElastiCache](#) se abordan estrategias de almacenamiento en caché que asisten a la aplicación para realizar un escalado correcto.

Si desea utilizar la AWS Command Line Interface (AWS CLI), puede utilizar estos documentos, que lo ayudarán a empezar:

- [Documentación de AWS Command Line Interface](#)

En esta sección, se proporciona información sobre cómo descargar la AWS CLI, poner a funcionar la AWS CLI en su sistema y proporcionar sus credenciales de AWS.

- [Documentación de la AWS CLI para ElastiCache](#)

Este documento independiente abarca todas las AWS CLI para los comandos de ElastiCache, incluida la sintaxis, además de ejemplos.

Puede escribir programas de aplicación para utilizar la API de ElastiCache con diversos lenguajes de programación conocidos. Estos son algunos de los recursos:

- [Herramientas para Amazon Web Services](#)

Amazon Web Services proporciona una serie de kits de desarrollo de software (SDK) compatibles con ElastiCache. Puede codificar para ElastiCache con Java, .NET, PHP, Ruby y otros lenguajes. Estos SDK pueden simplificar en gran medida el desarrollo de la aplicación al aplicar formato a sus solicitudes a ElastiCache, analizar las respuestas y proporcionar gestión de errores y lógica de reintentos.

- [Uso de la API de ElastiCache](#)

Si no desea utilizar los SDK de AWS, puede interactuar con ElastiCache directamente a través de la API de consulta. En esta sección, puede encontrar consejos de solución de problemas e información acerca de la creación y autenticación de solicitudes, así como de la gestión de respuestas.

- [Referencia de la API de Amazon ElastiCache](#)

Este documento independiente abarca todas las operaciones de la API de ElastiCache, incluida la sintaxis, además de ejemplos.

AWS Regiones y zonas de disponibilidad

Los recursos de informática en la nube de Amazon están alojados en instalaciones de centros de datos con alta disponibilidad, en diferentes zonas del mundo (por ejemplo, Norteamérica, Europa o Asia). Cada ubicación del centro de datos se denomina AWS región.

Cada AWS región contiene varias ubicaciones distintas denominadas zonas de disponibilidad o AZ. Cada zona de disponibilidad está diseñada para quedar aislada en caso de error en otras zonas de disponibilidad. Cada una está diseñada para proporcionar una conectividad de red económica y de baja latencia con otras zonas de disponibilidad de la misma AWS región. Al lanzar instancias en distintas zonas de disponibilidad, puede proteger sus aplicaciones de los errores que se produzcan en una única ubicación. Para obtener más información, consulte [Elección de regiones y zonas de disponibilidad](#).

Puede crear un clúster en varias zonas de disponibilidad, lo que recibe el nombre de implementación Multi-AZ. Cuando elige esta opción, Amazon aprovisiona y mantiene una instancia de nodos secundaria en espera en una zona de disponibilidad diferente de forma automática. Su instancia de nodo primario se replica de forma asincrónica en las zonas de disponibilidad para la instancia secundaria. Este enfoque ayuda a proporcionar redundancia de datos y soporte de conmutación por error, elimina los bloqueos de E/S y minimiza los picos de latencia durante las copias de seguridad del sistema. Para obtener más información, consulte [Minimizar el tiempo de inactividad en ElastiCache \(Redis OSS\) con Multi-AZ](#).

Casos de ElastiCache uso comunes y cómo ElastiCache pueden ayudar

Tanto si se trata de la publicación de las últimas noticias, de una tabla de clasificación Top 10, de un catálogo de productos o de la venta de entradas para un espectáculo, la velocidad es un factor clave a tener en cuenta. El éxito de su sitio web y de su negocio depende considerablemente de la rapidez con la que se entrega el contenido.

En el artículo del New York Times “[For Impatient Web Users, an Eye Blink Is Just Too Long to Wait](#)” (Para un usuario impaciente de Internet, un abrir y cerrar de ojos ya es esperar demasiado), se observó que los usuarios pueden percibir una diferencia de 250 milisegundos (1/4 de segundo) entre los sitios competidores. Los usuarios suelen descartar el sitio más lento en favor del más rápido. Las pruebas realizadas en Amazon, citadas en [How Webpage Load Time Is Related to Visitor Loss](#) (Relación entre el tiempo de carga de la página web y la pérdida de visitantes), revelaron que por cada incremento de 100 ms (1/10 de segundo) de tiempo de carga, las ventas disminuían un 1 %.

Si alguien necesita tener acceso a estos datos, puede entregar esos datos mucho más rápido si se almacenan en caché. Eso es cierto, independientemente de si se trata de una página web o de un informe que impulsa las decisiones empresariales. ¿Puede su negocio permitirse no almacenar en caché sus páginas web con el fin de ofrecerlas con la menor latencia posible?

Puede parecer intuitivamente evidente que desee almacenar en caché los elementos más solicitados. Pero ¿por qué no en almacenar en la caché los elementos solicitados con menos frecuencia? Incluso la consulta a la base de datos o la API llamada remota más optimizadas son notablemente más lentas que la recuperación de una clave simple de una caché en memoria. El concepto de notablemente más lento suele mandar a los clientes a otro sitio.

Los siguientes ejemplos ilustran algunas de las formas en que el uso ElastiCache puede mejorar el rendimiento general de la aplicación.

Temas

- [Almacenamiento de datos en memoria](#)
- [Tablas de clasificación de juegos \(conjuntos ordenados en RedisOSS\)](#)
- [Mensajería \(Redis Pub/SubOSS\)](#)
- [Datos de recomendación \(hashes de Redis\) OSS](#)
- [Otros usos de Redis OSS](#)

- [ElastiCache Testimonios de clientes](#)

Almacenamiento de datos en memoria

El propósito principal del almacenamiento en memoria de pares clave-valor es proporcionar un acceso ultrarrápido (latencia de milisegundos) y económico a las copias de datos. La mayoría de almacenes de datos tienen áreas de datos a los que se accede con frecuencia, pero que raramente se actualizan. Además, la consulta de una base de datos siempre será más lenta y más cara que la localización de una clave en una caché de par clave-valor. Algunas consultas de base de datos son especialmente costosas de realizar. Por ejemplo, las consultas que implican uniones entre varias tablas o las consultas con cálculos intensivos. Al almacenar en caché resultados de una consulta de este tipo, solo pagará el precio de la consulta una vez. A continuación, puede recuperar rápidamente los datos varias veces sin tener que volver a ejecutar la consulta.

¿Qué debo almacenar en caché?

A la hora de decidir qué datos almacenar en caché, tenga en cuenta estos factores:

Velocidad y gastos: siempre es más lento y más caro adquirir los datos de una base de datos que de una caché. Algunas consultas de base de datos son intrínsecamente más lentas y más caras que otras. Por ejemplo, las consultas que realizan uniones en varias tablas son mucho más lentas y más caras que las consultas sencillas en una única tabla. Si los datos interesantes que se desean adquirir requieren una consulta lenta y cara, son un candidato para el almacenamiento en caché. Si la adquisición de los datos requiere una consulta relativamente rápida y sencilla, puede ser un candidato para el almacenamiento en caché, en función de otros factores.

Patrón de datos y acceso: determinar qué almacenar en caché también implica comprender los datos en sí y sus patrones de acceso. Por ejemplo, no tiene sentido almacenar en caché los datos a los que se accede raramente o que cambian rápidamente. Para que el almacenamiento en caché proporcione un beneficio real, los datos deben ser relativamente estáticos y el acceso a estos debería ser frecuente. Un ejemplo es un perfil personal en un sitio de redes sociales. Por otro lado, no deseará almacenar en caché datos si dicho almacenamiento no supone ventajas de velocidad o costos. Por ejemplo, no tiene sentido almacenar en caché páginas web que devuelven los resultados de búsqueda, ya que estas consultas y sus resultados suelen ser únicos.

Estancamiento: por definición, los datos almacenados en caché son datos obsoletos. Incluso si en ciertas circunstancias no se encuentran obsoletos, siempre deben considerarse y tratarse como

obsoletos. Para determinar si sus datos son un candidato para el almacenamiento en caché, debe evaluar la tolerancia de su aplicación a los datos obsoletos.

Es posible que la aplicación sea capaz de tolerar datos obsoletos en un contexto, pero no en otro. Por ejemplo, suponga que su sitio web sirve a un precio bursátil. Sus clientes podrían aceptar cierto estancamiento con una exención de responsabilidad de que los precios podrían tener un retraso de *n* minutos. Sin embargo, si presenta ese precio a un agente que realiza una venta o una compra, deseará tener datos en tiempo real.

Considere la posibilidad de almacenar en caché los datos si:

- Resultan lentos o caros de adquirir en comparación con la recuperación en caché.
- Los usuarios acceden a sus datos con frecuencia.
- Sus datos se mantienen relativamente iguales, o si cambian con rapidez, el estancamiento no es un problema importante.

Para más información, consulte los siguientes temas:

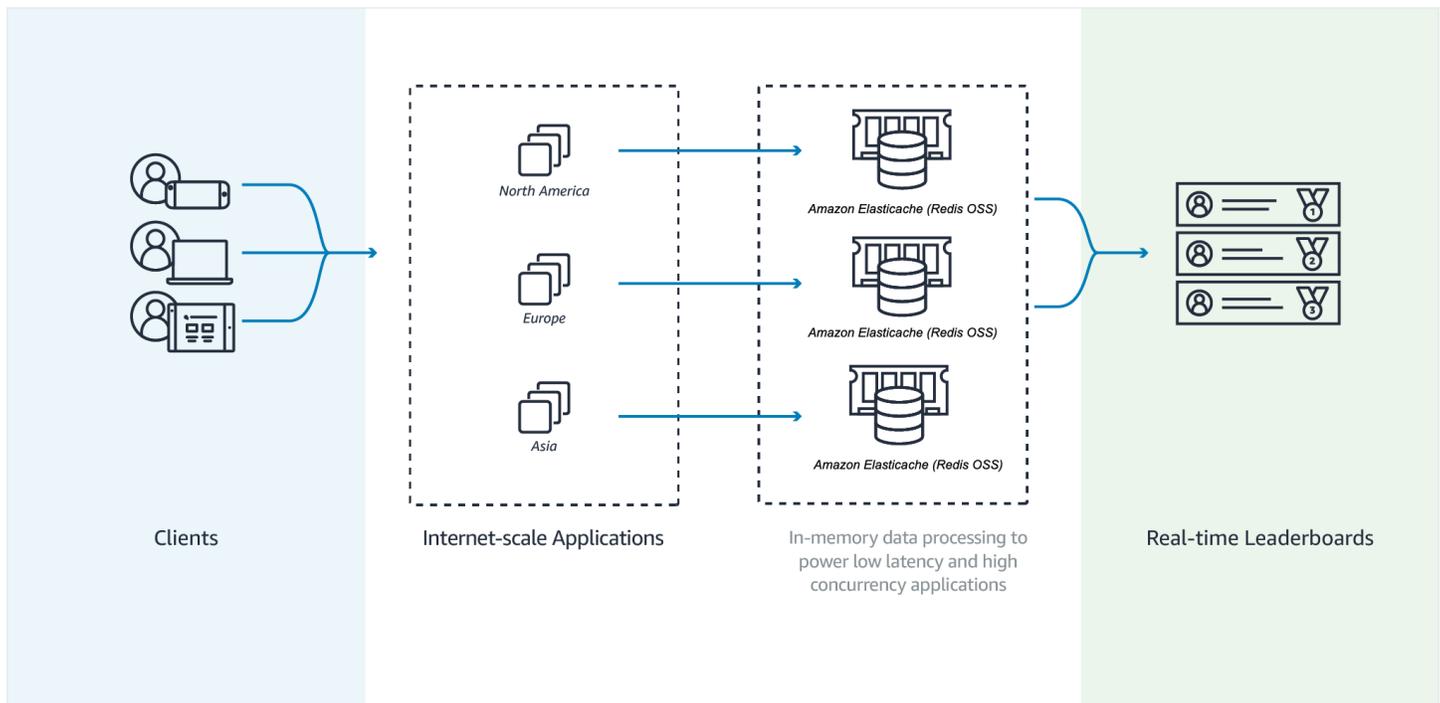
- [Estrategias de almacenamiento en caché](#) en la guía del usuario de ElastiCache (Redis) OSS

Tablas de clasificación de juegos (conjuntos ordenados en RedisOSS)

Los conjuntos OSS ordenados en Redis trasladan la complejidad computacional de las tablas de clasificación de la aplicación al clúster de Redis. OSS

Las tablas de clasificación, como las de 10 mejores puntuaciones de un juego, son complejas en términos informáticos. Esto es especialmente cierto cuando hay un gran número de jugadores simultáneos y las puntuaciones cambian de forma constante. Los conjuntos OSS ordenados de Redis garantizan tanto la unicidad como el orden de los elementos. Al usar los conjuntos OSS ordenados de Redis, cada vez que se agrega un nuevo elemento al conjunto ordenado, se vuelve a clasificar en tiempo real. A continuación, se agrega al conjunto en el orden numérico correcto.

En el siguiente diagrama, puedes ver cómo funciona una tabla de clasificación de juegos ElastiCache (de RedisOSS).



Example Tabla de clasificación de Redis OSS

En este ejemplo, cuatro jugadores y sus puntuaciones se introducen en una lista ordenada mediante ZADD. El comando ZREVRANGEBYSCORE muestra a los jugadores por su puntuación, de más alta a más baja. A continuación, ZADD se utiliza para actualizar la puntuación de June, sobrescribiendo la entrada existente. Por último, ZREVRANGEBYSCORE muestra a los jugadores por su puntuación, de más alta a más baja. La lista muestra que June ha ascendido en la clasificación.

```
ZADD leaderboard 132 Robert
ZADD leaderboard 231 Sandra
ZADD leaderboard 32 June
ZADD leaderboard 381 Adam

ZREVRANGEBYSCORE leaderboard +inf -inf
1) Adam
2) Sandra
3) Robert
4) June

ZADD leaderboard 232 June

ZREVRANGEBYSCORE leaderboard +inf -inf
1) Adam
2) June
```

- 3) Sandra
- 4) Robert

El siguiente comando permite a June conocer dónde se sitúa entre todos los jugadores. Debido a que la clasificación se basa en cero, ZREVRANK arroja un 1 para June, que se encuentra en segunda posición.

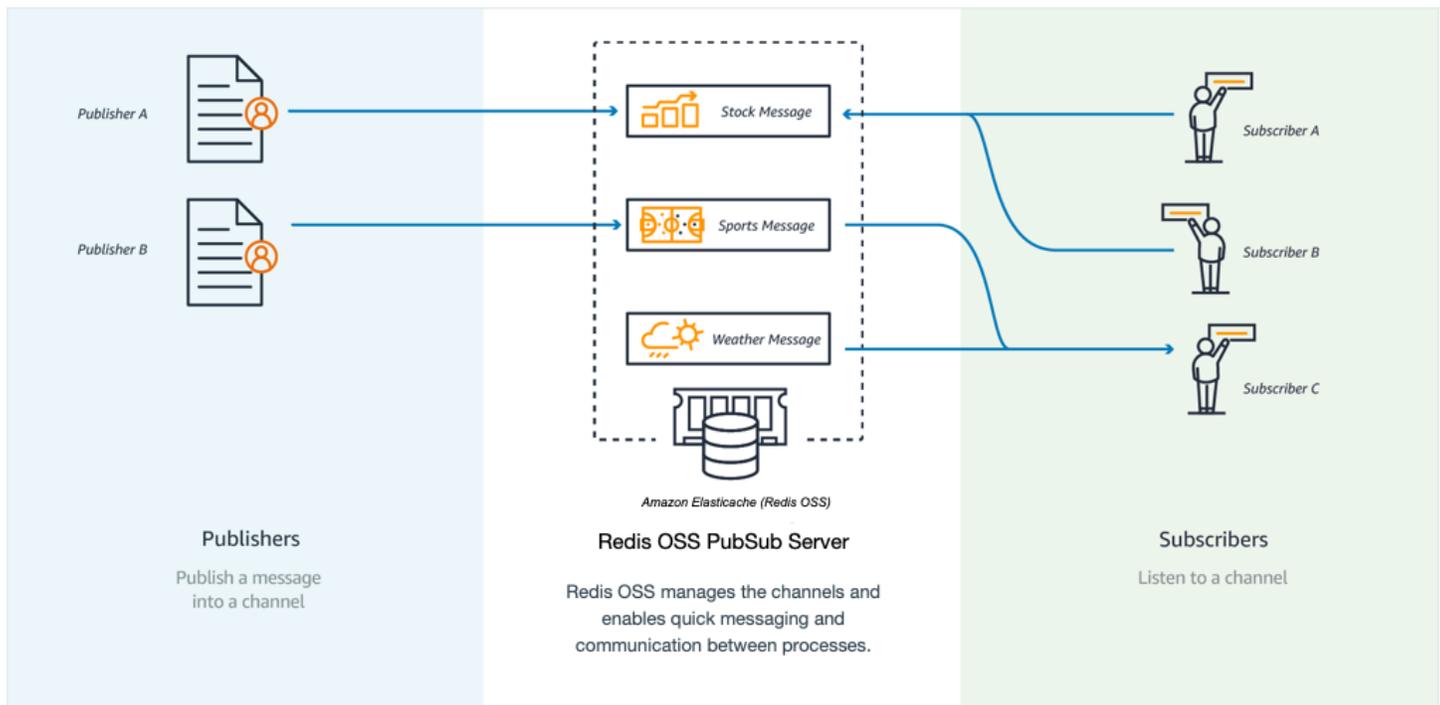
```
ZREVRANK leaderboard June  
1
```

Para obtener más información, consulte la [OSSdocumentación de Redis](#) sobre conjuntos ordenados.

Mensajería (Redis Pub/SubOSS)

Al enviar un mensaje de correo electrónico, lo envía a uno o a varios destinatarios especificados. En el paradigma de publicación/envío, envía un mensaje a un canal específico sin saber quién lo recibirá y si lo recibirá. Las personas que reciben el mensaje son aquellas que se encuentran suscritas al canal. Por ejemplo, suponga que se suscribe al canal news.sports.golf. Usted y todos los demás que se hayan suscrito al canal news.sports.golf recibirán todos los mensajes que se publiquen en news.sports.golf.

La funcionalidad OSS pub/sub de Redis no tiene relación con ningún espacio clave. Por lo tanto, no interferirá en ningún nivel. En el siguiente diagrama, puede encontrar una ilustración de los mensajes ElastiCache (de OSS Redis).



Suscripción

Para recibir mensajes en un canal, debe suscribirse a dicho canal. Puede suscribirse a un solo canal, a varios canales especificados o a todos los canales que coincidan con un patrón. Para cancelar una suscripción, debe cancelar la suscripción al canal especificado en el momento de la suscripción. O bien, si se ha suscrito mediante la coincidencia de patrones, anula la suscripción con el mismo patrón que utilizó antes.

Example - Suscripción a un único canal

Para suscribirse a un solo canal, utilice el SUBSCRIBE comando que especifica el canal al que desea suscribirse. En el siguiente ejemplo, un cliente se suscribe al canal `news.sports.golf`.

```
SUBSCRIBE news.sports.golf
```

Transcurrido un tiempo, el cliente cancela su suscripción al canal mediante el UNSUBSCRIBE comando que especifica el canal del que quiere darse de baja.

```
UNSUBSCRIBE news.sports.golf
```

Example - Suscripciones a varios canales especificados

Para suscribirse a varios canales específicos, enumere los canales con el SUBSCRIBE comando. En el siguiente ejemplo, un cliente se suscribe a los canales news.sports.golf, news.sports.soccer y news.sports.skiing

```
SUBSCRIBE news.sports.golf news.sports.soccer news.sports.skiing
```

Para cancelar una suscripción a un canal específico, usa el UNSUBSCRIBE comando y especifica el canal del que deseas cancelar la suscripción.

```
UNSUBSCRIBE news.sports.golf
```

Para cancelar las suscripciones a varios canales, usa el UNSUBSCRIBE comando y especifica los canales de los que deseas cancelar la suscripción.

```
UNSUBSCRIBE news.sports.golf news.sports.soccer
```

Para cancelar todas las suscripciones, utilice UNSUBSCRIBE y especifique cada canal. O utilice UNSUBSCRIBE y no especifique un canal.

```
UNSUBSCRIBE news.sports.golf news.sports.soccer news.sports.skiing
```

o

```
UNSUBSCRIBE
```

Example - Suscripciones mediante la coincidencia de patrones

Los clientes pueden suscribirse a todos los canales que coincidan con un patrón mediante el PSUBSCRIBE comando.

En el siguiente ejemplo, un cliente se suscribe a todos los canales de deportes. No enumera todos los canales de deportes de manera individual, como se haría con SUBSCRIBE. En su lugar, con el comando PSUBSCRIBE, se utiliza la coincidencia de patrones.

```
PSUBSCRIBE news.sports.*
```

Example Cancelación de suscripciones

Para cancelar las suscripciones a estos canales, utilice el comando PUNSUBSCRIBE.

```
PUNSUBSCRIBE news.sports.*
```

Important

La cadena de canales enviada a un SUBSCRIBE comando [P] y al UNSUBSCRIBE comando [P] debe coincidir. No puede PSUBSCRIBE a news.* y PUNSUBSCRIBE de news.sports.* o UNSUBSCRIBE de news.sports.golf.

Publicación

Para enviar un mensaje a todos los suscriptores de un canal, utilice el comando PUBLISH y especifique el canal y el mensaje. En el siguiente ejemplo, se publica el mensaje: "It's Saturday and sunny. I'm headed to the links" (Es sábado y está soleado. Vayamos directamente a los enlaces). en el canal news.sports.golf.

```
PUBLISH news.sports.golf "It's Saturday and sunny. I'm headed to the links."
```

Un cliente no puede publicar en un canal al que está suscrito.

Para obtener más información, consulte [Pub/Sub](#) en la documentación de OSS Redis.

Datos de recomendación (hashes de Redis) OSS

El uso de Redis INCR o DECR en Redis OSS simplifica la compilación de recomendaciones. Cada vez que un usuario da un "me gusta" a un producto, aumenta un contador item:productID:like. Cada vez que un usuario da un "no me gusta" a un producto, aumenta un contador item:productID:dislike. Con los OSS hashes de Redis, también puede mantener una lista de todas las personas a las que les ha gustado o no un producto.

Example - Me gusta y no me gusta

```
INCR item:38923:likes  
HSET item:38923:ratings Susan 1  
INCR item:38923:dislikes
```

```
HSET item:38923:ratings Tommy -1
```

Otros usos de Redis OSS

La entrada del blog [How to take advantage of Redis OSS just adding it to your stack](#), de Salvatore Sanfilippo, analiza una serie de problemas comunes relacionados con las bases de datos y cómo pueden resolverse fácilmente con Redis. OSS Este método elimina la carga de la base de datos y mejora el rendimiento.

ElastiCache Testimonios de clientes

Para obtener más información sobre cómo empresas como AirbnbPBS, Esri y otras utilizan Amazon ElastiCache para hacer crecer sus negocios con una mejor experiencia de cliente, consulta [Cómo utilizan Amazon ElastiCache otros usuarios](#).

También puedes ver los [vídeos tutoriales](#) para ver otros casos de uso de los ElastiCache clientes.

Introducción a Amazon ElastiCache (Redis OSS)

Utilice el tutorial práctico de esta sección como ayuda para empezar y obtener más información sobre ElastiCache (Redis OSS).

Temas

- [Configuración](#)
- [Paso 1: Crear una caché](#)
- [Paso 2: Lectura/escritura de datos en la caché](#)
- [Paso 3 \(opcional\): Limpieza](#)
- [Sigüientes pasos](#)
- [Introducción a ElastiCache y a los SDK de AWS](#)
- [Tutorial: Configuración de una función de Lambda para acceder a Amazon ElastiCache en una Amazon VPC](#)

Configuración

Para configurar ElastiCache:

Temas

- [Inscríbbase en una Cuenta de AWS](#)
- [Creación de un usuario con acceso administrativo](#)
- [Concesión de acceso programático](#)
- [Configure sus permisos \(solo para ElastiCache usuarios nuevos\)](#)
- [Configurar EC2](#)
- [Concede acceso a la red desde un grupo VPC de seguridad de Amazon a tu caché](#)
- [Descarga y configuración de redis-cli](#)

Inscríbbase en una Cuenta de AWS

Si no tiene una Cuenta de AWS, complete los siguientes pasos para crearlo.

Para suscribirte a una Cuenta de AWS

1. Abrir <https://portal.aws.amazon.com/billing/registro>.
2. Siga las instrucciones que se le indiquen.

Parte del procedimiento de registro consiste en recibir una llamada telefónica e indicar un código de verificación en el teclado del teléfono.

Cuando te registras en un Cuenta de AWS, Usuario raíz de la cuenta de AWS se crea un. El usuario raíz tendrá acceso a todos los Servicios de AWS y recursos de esa cuenta. Como práctica recomendada de seguridad, asigne acceso administrativo a un usuario y utilice únicamente el usuario raíz para realizar [tareas que requieren acceso de usuario raíz](#).

AWS te envía un correo electrónico de confirmación una vez finalizado el proceso de registro. Puede ver la actividad de la cuenta y administrar la cuenta en cualquier momento entrando en <https://aws.amazon.com/> y seleccionando Mi cuenta.

Creación de un usuario con acceso administrativo

Después de crear un usuario administrativo Cuenta de AWS, asegúrelo Usuario raíz de la cuenta de AWS AWS IAM Identity Center, habilite y cree un usuario administrativo para no usar el usuario root en las tareas diarias.

Proteja su Usuario raíz de la cuenta de AWS

1. Inicie sesión [AWS Management Console](#) como propietario de la cuenta seleccionando el usuario root e introduciendo su dirección de Cuenta de AWS correo electrónico. En la siguiente página, escriba su contraseña.

Para obtener ayuda para iniciar sesión con el usuario raíz, consulte [Iniciar sesión como usuario raíz](#) en la Guía del usuario de AWS Sign-In .

2. Activa la autenticación multifactorial (MFA) para tu usuario root.

Para obtener instrucciones, consulte [Habilitar un MFA dispositivo virtual para el usuario Cuenta de AWS root \(consola\)](#) en la Guía del IAM usuario.

Creación de un usuario con acceso administrativo

1. Habilite IAM Identity Center.

Consulte las instrucciones en [Activar AWS IAM Identity Center](#) en la Guía del usuario de AWS IAM Identity Center .

2. En IAM Identity Center, conceda acceso administrativo a un usuario.

Para ver un tutorial sobre cómo usar el Directorio de IAM Identity Center como fuente de identidad, consulte [Configurar el acceso de los usuarios con la configuración predeterminada Directorio de IAM Identity Center](#) en la Guía del AWS IAM Identity Center usuario.

Iniciar sesión como usuario con acceso de administrador

- Para iniciar sesión con su usuario de IAM Identity Center, utilice el inicio de sesión URL que se envió a su dirección de correo electrónico cuando creó el usuario de IAM Identity Center.

Para obtener ayuda para iniciar sesión con un usuario de IAM Identity Center, consulte [Iniciar sesión en el portal de AWS acceso](#) en la Guía del AWS Sign-In usuario.

Concesión de acceso a usuarios adicionales

1. En IAM Identity Center, cree un conjunto de permisos que siga la práctica recomendada de aplicar permisos con privilegios mínimos.

Para conocer las instrucciones, consulte [Create a permission set](#) en la Guía del usuario de AWS IAM Identity Center .

2. Asigne usuarios a un grupo y, a continuación, asigne el acceso de inicio de sesión único al grupo.

Para conocer las instrucciones, consulte [Add groups](#) en la Guía del usuario de AWS IAM Identity Center .

Concesión de acceso programático

Los usuarios necesitan acceso programático si quieren interactuar con personas AWS ajenas a AWS Management Console La forma de conceder el acceso programático depende del tipo de usuario que acceda. AWS

Para conceder acceso programático a los usuarios, elija una de las siguientes opciones.

¿Qué usuario necesita acceso programático?	Para	Mediante
Identidad del personal (Los usuarios se administran en IAM Identity Center)	Utilice credenciales temporales para firmar las solicitudes programáticas dirigidas al AWS CLI AWS SDKs, o AWS APIs.	Siga las instrucciones de la interfaz que desea utilizar: <ul style="list-style-type: none"> • Para ello AWS CLI, consulte Configuración del AWS CLI uso AWS IAM Identity Center en la Guía del AWS Command Line Interface usuario. • Para AWS SDKs ver las herramientas y AWS APIs, consulte la autenticación de IAM Identity Center en la Guía de referencia de herramientas AWS SDKs y herramientas.
IAM	Utilice credenciales temporales para firmar las solicitudes programáticas dirigidas al AWS CLI AWS SDKs, o AWS APIs.	Siga las instrucciones de Uso de credenciales temporales con AWS recursos de la Guía del IAM usuario.
IAM	(No recomendado) Utilice credenciales de larga duración para firmar las solicitudes programáticas dirigidas al AWS CLI, AWS SDKs, o AWS APIs.	Siga las instrucciones de la interfaz que desea utilizar: <ul style="list-style-type: none"> • Para ello AWS CLI, consulte Autenticación con credenciales IAM de usuario en la Guía del AWS Command Line Interface usuario. • Para obtener AWS SDKs información sobre las herramientas, consulte

¿Qué usuario necesita acceso programático?	Para	Mediante
		<p>Autenticarse con credenciales de larga duración en la Guía de referencia de herramientas AWS SDKs y herramientas.</p> <ul style="list-style-type: none"> • Para ello AWS APIs, consulte Administrar las claves de acceso de IAM los usuarios en la Guía del IAM usuario.

Temas relacionados:

- [Contenido de IAM](#) la Guía IAM del usuario.
- [AWS Las credenciales de seguridad](#) como referencia AWS general.

Configure sus permisos (solo para ElastiCache usuarios nuevos)

Para dar acceso, agregue permisos a los usuarios, grupos o roles:

- Usuarios y grupos en AWS IAM Identity Center:

Cree un conjunto de permisos. Siga las instrucciones de [Creación de un conjunto de permisos](#) en la Guía del usuario de AWS IAM Identity Center .

- Usuarios gestionados IAM a través de un proveedor de identidad:

Cree un rol para la federación de identidades. Siga las instrucciones de la Guía del IAM usuario sobre cómo [crear un rol para un proveedor de identidades externo \(federación\)](#).

- IAMusuarios:

- Cree un rol que el usuario pueda aceptar. Siga las instrucciones de la Guía del [IAMusuario sobre cómo crear un rol para un IAM usuario](#).

- (No recomendado) Adjunte una política directamente a un usuario o añada un usuario a un grupo de usuarios. Siga las instrucciones de [Añadir permisos a un usuario \(consola\)](#) de la Guía del IAM usuario.

Amazon ElastiCache crea y utiliza funciones vinculadas a servicios para aprovisionar recursos y acceder a otros AWS recursos y servicios en tu nombre. ElastiCache Para crear un rol vinculado a un servicio para usted, utilice la AWS política administrada denominada AmazonElastiCacheFullAccess Esta función ya está aprovisionada con los permisos que el servicio requiere para crear un rol vinculado a un servicio en su nombre.

Si lo prefiere, puede no utilizar la política predeterminada, sino una administrada de forma personalizada. En este caso, asegúrese de tener permisos para llamar `iam:createServiceLinkedRole` o de haber creado el rol vinculado al ElastiCache servicio.

Para más información, consulte los siguientes temas:

- [Crear una nueva política \(\) IAM](#)
- [Políticas administradas por AWS para Amazon ElastiCache](#)
- [Uso de roles vinculados a servicios para Amazon ElastiCache](#)

Configurar EC2

Necesitarás configurar una EC2 instancia desde la que te conectarás a tu caché.

- Si aún no tienes una EC2 instancia, consulta cómo configurarla EC2 aquí: Cómo [empezar EC2](#).
- La EC2 instancia debe estar en la misma ubicación VPC y tener la misma configuración de grupo de seguridad que la caché. De forma predeterminada, Amazon ElastiCache crea una caché en su configuración predeterminada VPC y utiliza el grupo de seguridad predeterminado. Para seguir este tutorial, asegúrate de que tu EC2 instancia esté en el grupo de seguridad predeterminado VPC y tenga el grupo de seguridad predeterminado.

Concede acceso a la red desde un grupo VPC de seguridad de Amazon a tu caché

ElastiCache Los clústeres de diseño propio utilizan el puerto 6379 para los OSS comandos de Redis, y los clústeres ElastiCache sin servidor utilizan tanto el puerto 6379 como el puerto 6380. Para poder

conectar y ejecutar correctamente los OSS comandos de Redis desde su EC2 instancia, su grupo de seguridad debe permitir el acceso a estos puertos según sea necesario.

1. Inicia sesión en la [EC2consola de Amazon AWS Command Line Interface](#) y ábrela.
2. En el panel de navegación, en Network & Security (Redes y seguridad), seleccione Security Groups(Grupos de seguridad).
3. En la lista de grupos de seguridad, elige el grupo de seguridad para tu AmazonVPC. A menos que haya creado un grupo de seguridad para ElastiCache su uso, este grupo de seguridad se denominará predeterminado.
4. Elija la pestaña Entrada y haga lo siguiente:
 - a. Elija Edit (Editar).
 - b. Seleccione Agregar regla.
 - c. En la columna Tipo, elija TCPRegla personalizada.
 - d. En el cuadro Rango de puertos, escriba 6379.
 - e. En el cuadro Fuente, selecciona Cualquier lugar que tenga el rango de puertos (0.0.0.0/0) para que cualquier EC2 instancia de Amazon que lance en tu Amazon VPC pueda conectarse a tu caché.
 - f. Si utilizas un sistema ElastiCache sin servidor, añade otra regla. Para ello, selecciona Añadir regla.
 - g. En la columna Tipo, selecciona TCP Regla personalizada.
 - h. En el cuadro Rango de puertos, escriba 6380.
 - i. En el cuadro Fuente, selecciona Cualquier lugar que tenga el rango de puertos (0.0.0.0/0) para que cualquier EC2 instancia de Amazon que lance en tu Amazon VPC pueda conectarse a tu caché.
 - j. Seleccione Save.

Descarga y configuración de redis-cli

1. Conéctate a tu EC2 instancia de Amazon mediante la utilidad de conexión que prefieras. Para obtener instrucciones sobre cómo conectarse a una EC2 instancia de Amazon, consulta la [Guía de EC2 introducción de Amazon](#).
2. Descargue e instale la utilidad redis-cli mediante el comando adecuado para la configuración.

Amazon Linux 2023

```
sudo yum install redis6 -y
```

Amazon Linux 2

```
sudo amazon-linux-extras install epel -y
sudo yum install gcc jemalloc-devel openssl-devel tcl tcl-devel -y
wget https://download.redis.io/releases/redis-7.2.5.tar.gz
tar xvzf redis-7.2.5.tar.gz
cd redis-7.2.5
make BUILD_TLS=yes
```

Note

- Al instalar el paquete redis6, se instalará redis6-cli con soporte de cifrado predeterminado.
- Es importante contar con soporte de compilación para TLS instalar redis-cli. ElastiCache Solo se puede acceder a Serverless cuando está activado. TLS
- Si el clúster al que se conecta no está cifrado, no necesita la opción Build_TLS=yes.

Paso 1: Crear una caché

En este paso, crearás una nueva caché en Amazon ElastiCache.

AWS Management Console

Para crear una caché nueva mediante la ElastiCache consola:

1. Inicie sesión en <https://console.aws.amazon.com/connect/> **AWS Management Console y ábralo.**
2. En el panel de navegación de la parte izquierda de la consola, selecciona Cachés OSS de Redis.
3. En la parte derecha de la consola, seleccione Crear caché de Redis OSS
4. En Configuración de caché, introduzca un Nombre. Si lo desea, puede especificar una descripción para la caché.
5. Deje seleccionado el ajuste predeterminado.

- Haga clic en Crear para crear la caché.
- Cuando la caché se encuentre en estado “ACTIVO”, podrá empezar a usarla para leer y escribir.

AWS CLI

El siguiente AWS CLI ejemplo crea una nueva caché mediante create-serverless-cache.

Linux

```
aws elasticache create-serverless-cache \  
  --serverless-cache-name CacheName \  
  --engine redis
```

Windows

```
aws elasticache create-serverless-cache ^  
  --serverless-cache-name CacheName ^  
  --engine redis
```

Como puede ver, el valor del campo Estado es CREATING.

Para comprobar que se ElastiCache ha terminado de crear la caché, utilice el describe-serverless-caches comando.

Linux

```
aws elasticache describe-serverless-caches --serverless-cache-name CacheName
```

Windows

```
aws elasticache describe-serverless-caches --serverless-cache-name CacheName
```

Después de crear la caché, continúe con [Paso 2: Lectura/escritura de datos en la caché](#).

Paso 2: Lectura/escritura de datos en la caché

En esta sección se da por sentado que ha creado una instancia de Amazon EC2 y que se puede conectar a ella. Para obtener instrucciones al respecto, consulte la [Guía de introducción a Amazon EC2](#).

En esta sección, también se supone que ha configurado los ajustes de acceso a la VPC y del grupo de seguridad para la instancia EC2 desde la que se conecta a la memoria caché y que ha configurado redis-cli en la instancia de EC2. Para obtener más información sobre este paso, consulte [Configuración](#).

Buscar el punto de conexión de la caché

AWS Management Console

Para encontrar el punto final de la caché mediante la ElastiCache consola:

1. Inicia sesión en la ElastiCache consola de Amazon AWS Management Console y ábrela en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación de la parte izquierda de la consola, selecciona Cachés OSS de Redis.
3. En la parte derecha de la consola, haga clic en el nombre de la caché que acaba de crear.
4. En los Detalles de la caché, busque y copie el punto de conexión de la caché.

AWS CLI

En el siguiente AWS CLI ejemplo, se muestra cómo encontrar el punto final de la nueva caché mediante el describe-serverless-caches comando. Cuando haya ejecutado el comando, busque el campo “Punto de conexión”.

Linux

```
aws elasticache describe-serverless-caches \  
  --serverless-cache-name CacheName
```

Windows

```
aws elasticache describe-serverless-caches ^  
  --serverless-cache-name CacheName
```

Conéctese a su caché OSS de Redis (Linux)

Ahora que tiene el punto de conexión que necesita, puede iniciar sesión en una instancia EC2 y conectarse a la caché. En el siguiente ejemplo, utilice la utilidad redis-cli para conectarse a un

clúster. El siguiente comando se conecta a una caché (nota: sustituya cache-endpoint por el punto de conexión que recuperó en el paso anterior).

```
src/redis-cli -h cache-endpoint --tls -p 6379
set a "hello"          // Set key "a" with a string value and no expiration
OK
get a                  // Get value for key "a"
"hello"
```

Conéctese a su caché OSS de Redis (Windows)

Ahora que tiene el punto de conexión que necesita, puede iniciar sesión en una instancia EC2 y conectarse a la caché. En el siguiente ejemplo, utilice la utilidad redis-cli para conectarse a un clúster. El siguiente comando se conecta a una caché. Abra la línea de comandos, vaya al directorio OSS de Redis y ejecute el comando (nota: sustituya Cache_Endpoint por el punto final que recuperó en el paso anterior).

```
c:\Redis>redis-cli -h Redis_Cluster_Endpoint --tls -p 6379
set a "hello"          // Set key "a" with a string value and no expiration
OK
get a                  // Get value for key "a"
"hello"
```

Ahora puede pasar a [Paso 3 \(opcional\): Limpieza](#).

Paso 3 (opcional): Limpieza

Si ya no necesitas la ElastiCache caché de Amazon que creaste, puedes eliminarla. De este modo, no se le cobrará por recursos que no vaya a utilizar. Puedes usar la ElastiCache consola AWS CLI, la API o la ElastiCache API para eliminar la caché.

AWS Management Console

Para eliminar la caché con la consola:

1. Inicia sesión en la ElastiCache consola de Amazon AWS Management Console y ábrela en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación de la parte izquierda de la consola, selecciona Redis OSS Caches.
3. Elija el botón de opción situado al lado de la caché que desea eliminar.

4. Seleccione Acciones, en la parte superior derecha, y Eliminar.
5. Si lo desea, puede tomar una última instantánea antes de eliminar la memoria caché.
6. En la pantalla de confirmación Eliminar, vuelva a introducir el nombre de la caché y seleccione Eliminar, para eliminar el clúster, o Cancelar, para conservarlo.

En cuanto la caché pase al estado DELETING, dejará de generar gastos.

AWS CLI

En el siguiente AWS CLI ejemplo, se elimina una memoria caché mediante el comando. `delete-serverless-cache`

Linux

```
aws elasticache delete-serverless-cache \  
  --serverless-cache-name CacheName
```

Windows

```
aws elasticache delete-serverless-cache ^  
  --serverless-cache-name CacheName
```

Como puede ver, el valor del campo Estado es DELETING.

Ahora puede pasar a [Siguientes pasos](#).

Siguientes pasos

Para obtener más información sobre ElastiCache, consulte las siguientes páginas:

- [Trabajando con ElastiCache](#)
- [Escalado ElastiCache \(Redis OSS\)](#)
- [Registro y Supervisión en Amazon ElastiCache](#)
- [ElastiCache mejores prácticas y estrategias de almacenamiento en caché](#)
- [Instantánea y restauración](#)
- [SNSMonitorización de ElastiCache eventos por parte de Amazon](#)

Introducción a ElastiCache y a los SDK de AWS

Esta sección contiene tutoriales prácticos para ayudar a entender Amazon ElastiCache. Recomendamos que realice alguno de los tutoriales específicos de los distintos lenguajes.

Note

Los SDK de AWS están disponibles en una amplia variedad de idiomas. Para obtener una lista completa, consulte [Herramientas para Amazon Web Services](#).

Python y ElastiCache

En este tutorial, utilizarás el AWS SDK para Python (Boto3) para escribir programas sencillos que realicen las siguientes operaciones: ElastiCache

- Crear ElastiCache clústeres (modo de clúster activado y modo de clúster desactivado)
- Compruebe si existen usuarios o grupos de usuarios; de lo contrario, créelos (solo en Redis OSS 6.0 en adelante)
- Connect to ElastiCache
- Realice operaciones tales como configurar y obtener cadenas, leer y escribir en flujos, y publicar y suscribirse desde el canal Pub/Sub.

A medida que avance en este tutorial, puede consultar la documentación del AWS SDK para Python (Boto). La siguiente sección es específica para ElastiCache: cliente de [ElastiCache bajo nivel](#)

Temas

- [Requisitos previos del tutorial](#)
- [Creación de ElastiCache clústeres y usuarios](#)
- [Conectarse a ElastiCache](#)
- [Ejemplos de uso](#)

Requisitos previos del tutorial

- Configura una clave de AWS acceso para usar los AWS SDK. Para obtener más información, consulte [Configuración](#).

- Instale Python 3.0 o una versión posterior. Para obtener más información, consulte <https://www.python.org/downloads>. Para obtener instrucciones, consulte [Quickstart](#) en la documentación de Boto 3.

Creación de ElastiCache clústeres y usuarios

Los siguientes ejemplos utilizan el SDK boto3 para las operaciones de ElastiCache administración (creación de clústeres o usuarios) y redis-py-cluster redis-py/ para el manejo de datos.

Temas

- [Crear un clúster en modo de clúster deshabilitado](#)
- [Crear un clúster en modo de clúster deshabilitado con TLS y RBAC](#)
- [Crear un clúster en modo de clúster habilitado](#)
- [Crear un clúster en modo de clúster habilitado con TLS y RBAC](#)
- [Verificar si los usuarios o grupos de usuarios existen, de lo contrario, es necesario crearlos](#)

Crear un clúster en modo de clúster deshabilitado

Copie el siguiente programa y péguelo en un archivo denominado .py.

CreateClusterModeDisabledCluster

```
import boto3
import logging

logging.basicConfig(level=logging.INFO)
client = boto3.client('elasticache')

def
create_cluster_mode_disabled(CacheNodeType='cache.t3.small', EngineVersion='6.0', NumCacheClusterNodes=1,
                             cache_cluster_id=None, ReplicationGroupId=None):
    """Creates an ElastiCache Cluster with cluster mode disabled

    Returns a dictionary with the API response

    :param CacheNodeType: Node type used on the cluster. If not specified,
    cache.t3.small will be used
    Refer to https://docs.aws.amazon.com/AmazonElastiCache/latest/red-ug/
    CacheNodes.SupportedTypes.html for supported node types
```

```

:param EngineVersion: Engine version to be used. If not specified, latest will be
used.
:param NumCacheClusters: Number of nodes in the cluster. Minimum 1 (just a primary
node) and maximum 6 (1 primary and 5 replicas).
If not specified, cluster will be created with 1 primary and 1 replica.
:param ReplicationGroupDescription: Description for the cluster.
:param ReplicationGroupId: Name for the cluster
:return: dictionary with the API results

"""
if not ReplicationGroupId:
    return 'ReplicationGroupId parameter is required'

response = client.create_replication_group(
    AutomaticFailoverEnabled=True,
    CacheNodeType=CacheNodeType,
    Engine='redis',
    EngineVersion=EngineVersion,
    NumCacheClusters=NumCacheClusters,
    ReplicationGroupDescription=ReplicationGroupDescription,
    ReplicationGroupId=ReplicationGroupId,
    SnapshotRetentionLimit=30,
)
return response

if __name__ == '__main__':

    # Creates an ElastiCache Cluster mode disabled cluster, based on cache.m6g.large
nodes, Redis OSS 6, one primary and two replicas
    elasticacheResponse = create_cluster_mode_disabled(
        #CacheNodeType='cache.m6g.large',
        EngineVersion='6.0',
        NumCacheClusters=3,
        ReplicationGroupDescription='Redis cluster mode disabled with replicas',
        ReplicationGroupId='redis202104053'
    )

    logging.info(elasticacheResponse)

```

Para ejecutar el programa, introduzca el siguiente comando:

```
python CreateClusterModeDisabledCluster.py
```

Para obtener más información, consulte [Administración de clústeres](#).

Crear un clúster en modo de clúster deshabilitado con TLS y RBAC

Para garantizar la seguridad, puede utilizar Transport Layer Security (TLS) y el control de acceso basado en roles (RBAC) al crear un clúster en modo de clúster deshabilitado. A diferencia de Redis OSS AUTH, en el que todos los clientes autenticados tienen acceso total a los grupos de replicación si su token está autenticado, RBAC le permite controlar el acceso al clúster a través de los grupos de usuarios. Estos grupos de usuarios se han diseñado como una forma de organizar el acceso a los grupos de reproducción. Para obtener más información, consulte [Control de acceso basado en roles \(RBAC\)](#).

Copie el siguiente programa y péguelo en un archivo denominado RBAC.py.

ClusterModeDisabledWith

```
import boto3
import logging

logging.basicConfig(level=logging.INFO)
client = boto3.client('elasticache')

def
    create_cluster_mode_disabled_rbac(CacheNodeType='cache.t3.small',EngineVersion='6.0',NumCacheC
    cache cluster',ReplicationGroupId=None, UserGroupIds=None,
    SecurityGroupIds=None,CacheSubnetGroupName=None):
    """Creates an ElastiCache Cluster with cluster mode disabled and RBAC

    Returns a dictionary with the API response

    :param CacheNodeType: Node type used on the cluster. If not specified,
    cache.t3.small will be used
    Refer to https://docs.aws.amazon.com/AmazonElastiCache/latest/red-ug/
    CacheNodes.SupportedTypes.html for supported node types
    :param EngineVersion: Engine version to be used. If not specified, latest will be
    used.
    :param NumCacheClusters: Number of nodes in the cluster. Minimum 1 (just a primary
    node) and maximun 6 (1 primary and 5 replicas).
    If not specified, cluster will be created with 1 primary and 1 replica.
    :param ReplicationGroupDescription: Description for the cluster.
    :param ReplicationGroupId: Mandatory name for the cluster.
    :param UserGroupIds: The ID of the user group to be assigned to the cluster.
    :param SecurityGroupIds: List of security groups to be assigned. If not defined,
    default will be used
```

```

:param CacheSubnetGroupName: subnet group where the cluster will be placed. If not
defined, default will be used.
:return: dictionary with the API results

"""
if not ReplicationGroupId:
    return {'Error': 'ReplicationGroupId parameter is required'}
elif not isinstance(UserGroupIds,(list)):
    return {'Error': 'UserGroupIds parameter is required and must be a list'}

params={'AutomaticFailoverEnabled': True,
        'CacheNodeType': CacheNodeType,
        'Engine': 'redis',
        'EngineVersion': EngineVersion,
        'NumCacheClusters': NumCacheClusters,
        'ReplicationGroupDescription': ReplicationGroupDescription,
        'ReplicationGroupId': ReplicationGroupId,
        'SnapshotRetentionLimit': 30,
        'TransitEncryptionEnabled': True,
        'UserGroupIds':UserGroupIds
        }

# defaults will be used if CacheSubnetGroupName or SecurityGroups are not explicit.
if isinstance(SecurityGroupIds,(list)):
    params.update({'SecurityGroupIds':SecurityGroupIds})
if CacheSubnetGroupName:
    params.update({'CacheSubnetGroupName':CacheSubnetGroupName})

response = client.create_replication_group(**params)
return response

if __name__ == '__main__':

    # Creates an ElastiCache Cluster mode disabled cluster, based on cache.m6g.large
nodes, Redis OSS 6, one primary and two replicas.
    # Assigns the existent user group "mygroup" for RBAC authentication

response=create_cluster_mode_disabled_rbac(
    CacheNodeType='cache.m6g.large',
    EngineVersion='6.0',
    NumCacheClusters=3,
    ReplicationGroupDescription='Redis OSS cluster mode disabled with replicas',
    ReplicationGroupId='redis202104',
    UserGroupIds=[

```

```

        'mygroup'
    ],
    SecurityGroupIds=[
        'sg-7cc73803'
    ],
    CacheSubnetGroupName='default'
)

logging.info(response)

```

Para ejecutar el programa, introduzca el siguiente comando:

```
python ClusterModeDisabledWithRBAC.py
```

Para obtener más información, consulte [Administración de clústeres](#).

Crear un clúster en modo de clúster habilitado

Copie el siguiente programa y péguelo en un archivo denominado ClusterModeEnabled.py.

```

import boto3
import logging

logging.basicConfig(level=logging.INFO)
client = boto3.client('elasticache')

def
    create_cluster_mode_enabled(CacheNodeType='cache.t3.small',EngineVersion='6.0',NumNodeGroups=1
    ReplicationGroupDescription='Sample cache with cluster mode
    enabled',ReplicationGroupId=None):
    """Creates an ElastiCache Cluster with cluster mode enabled

    Returns a dictionary with the API response

    :param CacheNodeType: Node type used on the cluster. If not specified,
    cache.t3.small will be used
    Refer to https://docs.aws.amazon.com/AmazonElastiCache/latest/red-ug/
    CacheNodes.SupportedTypes.html for supported node types
    :param EngineVersion: Engine version to be used. If not specified, latest will be
    used.
    :param NumNodeGroups: Number of shards in the cluster. Minimum 1 and maximun 90.
    If not specified, cluster will be created with 1 shard.
    :param ReplicasPerNodeGroup: Number of replicas per shard. If not specified 1
    replica per shard will be created.

```

```

:param ReplicationGroupDescription: Description for the cluster.
:param ReplicationGroupId: Name for the cluster
:return: dictionary with the API results

"""
if not ReplicationGroupId:
    return 'ReplicationGroupId parameter is required'

response = client.create_replication_group(
    AutomaticFailoverEnabled=True,
    CacheNodeType=CacheNodeType,
    Engine='redis',
    EngineVersion=EngineVersion,
    ReplicationGroupDescription=ReplicationGroupDescription,
    ReplicationGroupId=ReplicationGroupId,
    # Creates a cluster mode enabled cluster with 1 shard(NumNodeGroups), 1 primary
node (implicit) and 2 replicas (replicasPerNodeGroup)
    NumNodeGroups=NumNodeGroups,
    ReplicasPerNodeGroup=ReplicasPerNodeGroup,
    CacheParameterGroupName='default.redis6.0.cluster.on'
)

return response

# Creates a cluster mode enabled
response = create_cluster_mode_enabled(
    CacheNodeType='cache.m6g.large',
    EngineVersion='6.0',
    ReplicationGroupDescription='Redis OSS cluster mode enabled with replicas',
    ReplicationGroupId='redis20210',
    # Creates a cluster mode enabled cluster with 1 shard(NumNodeGroups), 1 primary
(implicit) and 2 replicas (replicasPerNodeGroup)
    NumNodeGroups=2,
    ReplicasPerNodeGroup=1,
)

logging.info(response)

```

Para ejecutar el programa, introduzca el siguiente comando:

```
python ClusterModeEnabled.py
```

Para obtener más información, consulte [Administración de clústeres](#).

Crear un clúster en modo de clúster habilitado con TLS y RBAC

Para garantizar la seguridad, puede utilizar Transport Layer Security (TLS) y el control de acceso basado en roles (RBAC) al crear un clúster en modo de clúster habilitado. A diferencia de Redis OSS AUTH, en el que todos los clientes autenticados tienen acceso total a los grupos de replicación si su token está autenticado, RBAC le permite controlar el acceso al clúster a través de los grupos de usuarios. Estos grupos de usuarios se han diseñado como una forma de organizar el acceso a los grupos de reproducción. Para obtener más información, consulte [Control de acceso basado en roles \(RBAC\)](#).

Copie el siguiente programa y péguelo en un archivo denominado RBAC.py.

ClusterModeEnabledWith

```
import boto3
import logging

logging.basicConfig(level=logging.INFO)
client = boto3.client('elasticache')

def
  create_cluster_mode_enabled(CacheNodeType='cache.t3.small',EngineVersion='6.0',NumNodeGroups=1,
  ReplicationGroupDescription='Sample cache with cluster
  mode enabled',ReplicationGroupId=None,UserGroupIds=None,
  SecurityGroupIds=None,CacheSubnetGroupName=None,CacheParameterGroupName='default.redis6.0.clus
  """"Creates an ElastiCache Cluster with cluster mode enabled and RBAC

  Returns a dictionary with the API response

  :param CacheNodeType: Node type used on the cluster. If not specified,
  cache.t3.small will be used
  Refer to https://docs.aws.amazon.com/AmazonElastiCache/latest/red-ug/
  CacheNodes.SupportedTypes.html for supported node types
  :param EngineVersion: Engine version to be used. If not specified, latest will be
  used.
  :param NumNodeGroups: Number of shards in the cluster. Minimum 1 and maximun 90.
  If not specified, cluster will be created with 1 shard.
  :param ReplicasPerNodeGroup: Number of replicas per shard. If not specified 1
  replica per shard will be created.
  :param ReplicationGroupDescription: Description for the cluster.
  :param ReplicationGroupId: Name for the cluster.
  :param CacheParameterGroupName: Parameter group to be used. Must be compatible with
  the engine version and cluster mode enabled.
```

```

:return: dictionary with the API results

"""
if not ReplicationGroupId:
    return 'ReplicationGroupId parameter is required'
elif not isinstance(UserGroupIds,(list)):
    return {'Error': 'UserGroupIds parameter is required and must be a list'}

params={'AutomaticFailoverEnabled': True,
        'CacheNodeType': CacheNodeType,
        'Engine': 'redis',
        'EngineVersion': EngineVersion,
        'ReplicationGroupDescription': ReplicationGroupDescription,
        'ReplicationGroupId': ReplicationGroupId,
        'SnapshotRetentionLimit': 30,
        'TransitEncryptionEnabled': True,
        'UserGroupIds':UserGroupIds,
        'NumNodeGroups': NumNodeGroups,
        'ReplicasPerNodeGroup': ReplicasPerNodeGroup,
        'CacheParameterGroupName': CacheParameterGroupName
    }

# defaults will be used if CacheSubnetGroupName or SecurityGroups are not explicit.
if isinstance(SecurityGroupIds,(list)):
    params.update({'SecurityGroupIds':SecurityGroupIds})
if CacheSubnetGroupName:
    params.update({'CacheSubnetGroupName':CacheSubnetGroupName})

response = client.create_replication_group(**params)
return response

if __name__ == '__main__':
    # Creates a cluster mode enabled cluster
    response = create_cluster_mode_enabled(
        CacheNodeType='cache.m6g.large',
        EngineVersion='6.0',
        ReplicationGroupDescription='Redis OSS cluster mode enabled with replicas',
        ReplicationGroupId='redis2021',
        # Creates a cluster mode enabled cluster with 1 shard(NumNodeGroups), 1 primary
        (implicit) and 2 replicas (replicasPerNodeGroup)
        NumNodeGroups=2,
        ReplicasPerNodeGroup=1,
        UserGroupIds=[
            'mygroup'

```

```
    ],
    SecurityGroupIds=[
        'sg-7cc73803'
    ],
    CacheSubnetGroupName='default'

)

logging.info(response)
```

Para ejecutar el programa, introduzca el siguiente comando:

```
python ClusterModeEnabledWithRBAC.py
```

Para obtener más información, consulte [Administración de clústeres](#).

Verificar si los usuarios o grupos de usuarios existen, de lo contrario, es necesario crearlos

Con el RBAC, puede crear usuarios y asignarles permisos específicos mediante una cadena de acceso. Los usuarios se asignan a grupos de usuarios alineados con una función específica (administradores, recursos humanos) que, a continuación, se despliegan en uno o más grupos de replicación ElastiCache (Redis OSS). De este modo, puede establecer límites de seguridad entre los clientes que utilizan el mismo grupo o grupos de replicación de Redis OSS e impedir que los clientes accedan a los datos de los demás. Para obtener más información, consulte [Control de acceso basado en roles \(RBAC\)](#).

Copie el siguiente programa y péguelo en un archivo denominado UserAndUserGroups.py. Actualice el mecanismo de suministro de credenciales. Las credenciales de este ejemplo se muestran como reemplazables y se les asigna un elemento no declarado. Evite codificar las credenciales de forma rígida.

```
import boto3
import logging

logging.basicConfig(level=logging.INFO)
client = boto3.client('elasticache')

def check_user_exists(UserId):
    """Checks if UserId exists

    Returns True if UserId exists, otherwise False
    :param UserId: ElastiCache User ID
```

```

:return: True|False
"""
try:
    response = client.describe_users(
        UserId=UserId,
    )
    if response['Users'][0]['UserId'].lower() == UserId.lower():
        return True
except Exception as e:
    if e.response['Error']['Code'] == 'UserNotFound':
        logging.info(e.response['Error'])
        return False
    else:
        raise

def check_group_exists(UserGroupId):
    """Checks if UserGroupID exists

    Returns True if Group ID exists, otherwise False
    :param UserGroupId: ElastiCache User ID
    :return: True|False
    """

    try:
        response = client.describe_user_groups(
            UserGroupId=UserGroupId
        )
        if response['UserGroups'][0]['UserGroupId'].lower() == UserGroupId.lower():
            return True
    except Exception as e:
        if e.response['Error']['Code'] == 'UserGroupNotFound':
            logging.info(e.response['Error'])
            return False
        else:
            raise

def create_user(UserId=None,UserName=None>Password=None,AccessString=None):
    """Creates a new user

    Returns the ARN for the newly created user or the error message
    :param UserId: ElastiCache user ID. User IDs must be unique
    :param UserName: ElastiCache user name. ElastiCache allows multiple users with the
    same name as long as the associated user ID is unique.
    :param Password: Password for user. Must have at least 16 chars.

```

```

:param AccessString: Access string with the permissions for the user. For
details refer to https://docs.aws.amazon.com/AmazonElastiCache/latest/red-ug/
Clusters.RBAC.html#Access-string
:return: user ARN
"""
try:
    response = client.create_user(
        UserId=UserId,
        UserName=UserName,
        Engine='Redis',
        Passwords=[Password],
        AccessString=AccessString,
        NoPasswordRequired=False
    )
    return response['ARN']
except Exception as e:
    logging.info(e.response['Error'])
    return e.response['Error']

def create_group(UserGroupId=None, UserIds=None):
    """Creates a new group.
    A default user is required (mandatory) and should be specified in the UserIds list

    Return: Group ARN
    :param UserIds: List with user IDs to be associated with the new group. A default
user is required
    :param UserGroupId: The ID (name) for the group
    :return: Group ARN
    """
    try:
        response = client.create_user_group(
            UserGroupId=UserGroupId,
            Engine='Redis',
            UserIds=UserIds
        )
        return response['ARN']
    except Exception as e:
        logging.info(e.response['Error'])

if __name__ == '__main__':

    groupName='mygroup2'
    userName = 'myuser2'

```

```

    userId=groupName+'-'+userName

    # Creates a new user if the user ID does not exist.
    for tmpUserId,tmpUserName in [ (userId,userName), (groupName+'-
default','default')]:
        if not check_user_exists(tmpUserId):
            response=create_user(UserId=tmpUserId,
UserName=EXAMPLE,Password=EXAMPLE,AccessString='on ~* +@all')
            logging.info(response)
        # assigns the new user ID to the user group
    if not check_group_exists(groupName):
        UserIds = [ userId , groupName+'-default']
        response=create_group(UserGroupId=groupName,UserIds=UserIds)
        logging.info(response)

```

Para ejecutar el programa, introduzca el siguiente comando:

```
python UserAndUserGroups.py
```

Conectarse a ElastiCache

Los siguientes ejemplos utilizan el cliente OSS de Redis para conectarse a ElastiCache.

Temas

- [Conexión a un clúster en modo de clúster deshabilitado](#)
- [Conexión a un clúster en modo de clúster habilitado](#)

Conexión a un clúster en modo de clúster deshabilitado

Copie el siguiente programa y péguelo en un archivo denominado ConnectClusterModeDisabled.py. Actualice el mecanismo de suministro de credenciales. Las credenciales de este ejemplo se muestran como reemplazables y se les asigna un elemento no declarado. Evite codificar las credenciales de forma rígida.

```

from redis import Redis
import logging

logging.basicConfig(level=logging.INFO)
redis = Redis(host='primary.xxx.yyyyyy.zzz1.cache.amazonaws.com', port=6379,
    decode_responses=True, ssl=True, username=example, password=EXAMPLE)

```

```
if redis.ping():
    logging.info("Connected to Redis")
```

Para ejecutar el programa, introduzca el siguiente comando:

```
python ConnectClusterModeDisabled.py
```

Conexión a un clúster en modo de clúster habilitado

Copie el siguiente programa y péguelo en un archivo denominado ConnectClusterModeEnabled.py.

```
from rediscluster import RedisCluster
import logging

logging.basicConfig(level=logging.INFO)
redis = RedisCluster(startup_nodes=[{"host":
    "xxx.yyy.clustercfg.zzz1.cache.amazonaws.com", "port": "6379"}],
    decode_responses=True, skip_full_coverage_check=True)

if redis.ping():
    logging.info("Connected to Redis")
```

Para ejecutar el programa, introduzca el siguiente comando:

```
python ConnectClusterModeEnabled.py
```

Ejemplos de uso

En los siguientes ejemplos se utiliza el SDK de Boto3 para que ElastiCache funcione con ElastiCache.

Temas

- [Establecer y obtener cadenas](#)
- [Establecer y obtener un hash con varios elementos](#)
- [Publicar \(escribir\) y suscribirse \(leer\) desde un canal Pub/Sub](#)
- [Escribir y leer a partir de un flujo](#)

Establecer y obtener cadenas

Copie el siguiente programa y péguelo en un archivo denominado SetAndGetStrings.py.

```
import time
import logging
logging.basicConfig(level=logging.INFO,format='%(asctime)s: %(message)s')

keyName='mykey'
currTime=time.ctime(time.time())

# Set the key 'mykey' with the current date and time as value.
# The Key will expire and removed from cache in 60 seconds.
redis.set(keyName, currTime, ex=60)

# Sleep just for better illustration of TTL (expiration) value
time.sleep(5)

# Retrieve the key value and current TTL
keyValue=redis.get(keyName)
keyTTL=redis.ttl(keyName)

logging.info("Key {} was set at {} and has {} seconds until expired".format(keyName,
keyValue, keyTTL))
```

Para ejecutar el programa, introduzca el siguiente comando:

```
python SetAndGetStrings.py
```

Establecer y obtener un hash con varios elementos

Copie el siguiente programa y péguelo en un archivo denominado SetAndGetHash.py.

```
import logging
import time

logging.basicConfig(level=logging.INFO,format='%(asctime)s: %(message)s')

keyName='mykey'
keyValues={'datetime': time.ctime(time.time()), 'epochtime': time.time()}

# Set the hash 'mykey' with the current date and time in human readable format
# (datetime field) and epoch number (epochtime field).
redis.hset(keyName, mapping=keyValues)

# Set the key to expire and removed from cache in 60 seconds.
redis.expire(keyName, 60)
```

```
# Sleep just for better illustration of TTL (expiration) value
time.sleep(5)

# Retrieves all the fields and current TTL
keyValues=redis.hgetall(keyName)
keyTTL=redis.ttl(keyName)

logging.info("Key {} was set at {} and has {} seconds until expired".format(keyName,
keyValues, keyTTL))
```

Para ejecutar el programa, introduzca el siguiente comando:

```
python SetAndGetHash.py
```

Publicar (escribir) y suscribirse (leer) desde un canal Pub/Sub

Copie el siguiente programa y péguelo en un archivo denominado PubAndSub.py.

```
import logging
import time

def handlerFunction(message):
    """Prints message got from PubSub channel to the log output

    Return None
    :param message: message to log
    """
    logging.info(message)

logging.basicConfig(level=logging.INFO)
redis = Redis(host="redis202104053.tihewd.ng.0001.use1.cache.amazonaws.com", port=6379,
decode_responses=True)

# Creates the subscriber connection on "mychannel"
subscriber = redis.pubsub()
subscriber.subscribe(**{'mychannel': handlerFunction})

# Creates a new thread to watch for messages while the main process continues with its
routines
thread = subscriber.run_in_thread(sleep_time=0.01)

# Creates publisher connection on "mychannel"
```

```
redis.publish('mychannel', 'My message')

# Publishes several messages. Subscriber thread will read and print on log.
while True:
    redis.publish('mychannel',time.ctime(time.time()))
    time.sleep(1)
```

Para ejecutar el programa, introduzca el siguiente comando:

```
python PubAndSub.py
```

Escribir y leer a partir de un flujo

Copie el siguiente programa y péguelo en un archivo denominado ReadWriteStream.py.

```
from redis import Redis
import redis.exceptions as exceptions
import logging
import time
import threading

logging.basicConfig(level=logging.INFO)

def writeMessage(streamName):
    """Starts a loop writting the current time and thread name to 'streamName'

    :param streamName: Stream (key) name to write messages.
    """
    fieldsDict={'writerId':threading.currentThread().getName(),'myvalue':None}
    while True:
        fieldsDict['myvalue'] = time.ctime(time.time())
        redis.xadd(streamName,fieldsDict)
        time.sleep(1)

def readMessage(groupName=None,streamName=None):
    """Starts a loop reading from 'streamName'
    Multiple threads will read from the same stream consumer group. Consumer group is
    used to coordinate data distribution.
    Once a thread acknowleges the message, it won't be provided again. If message
    wasn't acknowledged, it can be served to another thread.

    :param groupName: stream group were multiple threads will read.
    :param streamName: Stream (key) name where messages will be read.
    """
```

```

readerID=threading.currentThread().getName()
while True:
    try:
        # Check if the stream has any message
        if redis.xlen(streamName)>0:
            # Check if if the messages are new (not acknowledged) or not (already
processed)
            streamData=redis.xreadgroup(groupName,readerID,
{streamName:'>'},count=1)
            if len(streamData) > 0:
                msgId,message = streamData[0][1][0]
                logging.info("{}: Got {} from ID
{}".format(readerID,message,msgId))
                #Do some processing here. If the message has been processed
sucessfully, acknowledge it and (optional) delete the message.
                redis.xack(streamName,groupName,msgId)
                logging.info("Stream message ID {} read and processed successfully
by {}".format(msgId,readerID))
                redis.xdel(streamName,msgId)
            else:
                pass
        except:
            raise

        time.sleep(0.5)

# Creates the stream 'mystream' and consumer group 'myworkergroup' where multiple
threads will write/read.
try:
    redis.xgroup_create('mystream','myworkergroup',mkstream=True)
except exceptions.ResponseError as e:
    logging.info("Consumer group already exists. Will continue despite the error:
{}".format(e))
except:
    raise

# Starts 5 writer threads.
for writer_no in range(5):
    writerThread = threading.Thread(target=writeMessage, name='writer-'+str(writer_no),
args=('mystream',),daemon=True)
    writerThread.start()

# Starts 10 reader threads

```

```
for reader_no in range(10):
    readerThread = threading.Thread(target=readMessage, name='reader-'+str(reader_no),
    args=('myworkergroup', 'mystream', ), daemon=True)
    readerThread.daemon = True
    readerThread.start()

# Keep the code running for 30 seconds
time.sleep(30)
```

Para ejecutar el programa, introduzca el siguiente comando:

```
python ReadWriteStream.py
```

Tutorial: Configuración de una función de Lambda para acceder a Amazon ElastiCache en una Amazon VPC

En este tutorial, puede aprender a crear una caché ElastiCache sin servidor, crear una función Lambda, probar la función Lambda y, opcionalmente, limpiarla después.

Temas

- [Paso 1: Crear una memoria caché sin servidor](#)
- [Paso 2: Crear una función de Lambda](#)
- [Paso 3: comprobación de la función de Lambda](#)
- [Paso 4: limpiar \(opcional\)](#)

Paso 1: Crear una memoria caché sin servidor

Para crear una caché sin servidor, sigue estos pasos.

Temas

- [Paso 1.1: Crea una caché sin servidor](#)
- [Paso 1.2: Copiar el punto final de la caché sin servidor](#)
- [Paso 1.3: Crear un rol de IAM](#)
- [Paso 1.4: Crear una memoria caché sin servidor](#)

Paso 1.1: Crea una caché sin servidor

En este paso, crea una caché sin servidor en la Amazon VPC predeterminada de la región us-east-1 de su cuenta mediante la (CLI). AWS Command Line Interface Para obtener información sobre cómo crear una caché sin servidor mediante la ElastiCache consola o la API, consulte. [Paso 1: Crear una caché](#)

```
aws elasticache create-serverless-cache \  
  --serverless-cache-name cache-01 \  
  --description "ElastiCache IAM auth application" \  
  --engine redis
```

Como puede ver, el valor del campo Estado es CREATING. Terminar de crear la caché puede tardar un minuto. ElastiCache

Paso 1.2: Copiar el punto final de la caché sin servidor

Compruebe que ElastiCache (Redis OSS) haya terminado de crear la caché con el `describe-serverless-caches` comando.

```
aws elasticache describe-serverless-caches \  
  --serverless-cache-name cache-01
```

Copie la dirección del punto de conexión que aparece en el resultado. Necesitará esta dirección cuando cree el paquete de implementación para la función de Lambda.

Paso 1.3: Crear un rol de IAM

1. Cree un documento de política de confianza de IAM, como se muestra a continuación, para el rol que permita a la cuenta asumir el nuevo rol. Guarde la política en un archivo denominado `trust-policy.json`.

```
{  
  "Version": "2012-10-17",  
  "Statement": [{  
    "Effect": "Allow",  
    "Principal": { "AWS": "arn:aws:iam::123456789012:root" },  
    "Action": "sts:AssumeRole"  
  }],  
}
```

```
{
  "Effect": "Allow",
  "Principal": {
    "Service": "lambda.amazonaws.com"
  },
  "Action": "sts:AssumeRole"
}]
}
```

2. Cree un documento de política de IAM, como se muestra a continuación. Guarde la política en un archivo denominado `policy.json`.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect" : "Allow",
      "Action" : [
        "elasticache:Connect"
      ],
      "Resource" : [
        "arn:aws:elasticache:us-east-1:123456789012:serverlesscache:cache-01",
        "arn:aws:elasticache:us-east-1:123456789012:user:iam-user-01"
      ]
    }
  ]
}
```

3. Crear un rol de IAM.

```
aws iam create-role \
--role-name "elasticache-iam-auth-app" \
--assume-role-policy-document file://trust-policy.json
```

4. Cree la política de IAM.

```
aws iam create-policy \
--policy-name "elasticache-allow-all" \
--policy-document file://policy.json
```

5. Adjunte la política de IAM al rol.

```
aws iam attach-role-policy \
```

```
--role-name "elasticache-iam-auth-app" \  
--policy-arn "arn:aws:iam::123456789012:policy/elasticache-allow-all"
```

Paso 1.4: Crear una memoria caché sin servidor

1. Cree un nuevo usuario predeterminado.

```
aws elasticache create-user \  
  --user-name default \  
  --user-id default-user-disabled \  
  --engine redis \  
  --authentication-mode Type=no-password-required \  
  --access-string "off +get ~keys*"
```

2. Cree un nuevo usuario habilitado para IAM.

```
aws elasticache create-user \  
  --user-name iam-user-01 \  
  --user-id iam-user-01 \  
  --authentication-mode Type=iam \  
  --engine redis \  
  --access-string "on ~* +@all"
```

3. Cree un grupo de usuarios y asocie al usuario.

```
aws elasticache create-user-group \  
  --user-group-id iam-user-group-01 \  
  --engine redis \  
  --user-ids default-user-disabled iam-user-01  
  
aws elasticache modify-serverless-cache \  
  --serverless-cache-name cache-01 \  
  --user-group-id iam-user-group-01
```

Paso 2: Crear una función de Lambda

Para crear una función Lambda, siga estos pasos.

Temas

- [Paso 2.1: Crear una función Lambda](#)

- [Paso 2.2: crear el rol de IAM \(rol de ejecución\)](#)
- [Paso 2.3: cargar el paquete de despliegue \(crear la función de Lambda\)](#)

Paso 2.1: Crear una función Lambda

En este tutorial, proporcionamos código de ejemplo en Python para su función Lambda.

Python

El siguiente ejemplo de código Python lee y escribe un elemento en la ElastiCache caché. Copie el código y guárdelo en un archivo con el nombre `app.py`. Asegúrese de reemplazar el `elasticache_endpoint` valor del código por la dirección de punto final que copió en el paso 1.2.

```
from typing import Tuple, Union
from urllib.parse import ParseResult, urlencode, urlunparse

import boto3.session
import redis
from boto3.model import ServiceId
from boto3.signers import RequestSigner
from cachetools import TTLCache, cached
import uuid

class ElastiCacheIAMProvider(redis.CredentialProvider):
    def __init__(self, user, cache_name, is_serverless=False, region="us-east-1"):
        self.user = user
        self.cache_name = cache_name
        self.is_serverless = is_serverless
        self.region = region

        session = boto3.session.get_session()
        self.request_signer = RequestSigner(
            ServiceId("elasticache"),
            self.region,
            "elasticache",
            "v4",
            session.get_credentials(),
            session.get_component("event_emitter"),
        )

        # Generated IAM tokens are valid for 15 minutes
        @cached(cache=TTLCache(maxsize=128, ttl=900))
```

```

def get_credentials(self) -> Union[Tuple[str], Tuple[str, str]]:
    query_params = {"Action": "connect", "User": self.user}
    if self.is_serverless:
        query_params["ResourceType"] = "ServerlessCache"
    url = urlunparse(
        ParseResult(
            scheme="https",
            netloc=self.cache_name,
            path="/",
            query=urlencode(query_params),
            params="",
            fragment="",
        )
    )
    signed_url = self.request_signer.generate_presigned_url(
        {"method": "GET", "url": url, "body": {}, "headers": {}, "context": {}},
        operation_name="connect",
        expires_in=900,
        region_name=self.region,
    )
    # RequestSigner only seems to work if the URL has a protocol, but
    # Elasticache only accepts the URL without a protocol
    # So strip it off the signed URL before returning
    return (self.user, signed_url.removeprefix("https://"))

def lambda_handler(event, context):
    username = "iam-user-01" # replace with your user id
    cache_name = "cache-01" # replace with your cache name
    elasticache_endpoint = "cache-01-xxxxx.serverless.us1.cache.amazonaws.com" #
    replace with your cache endpoint
    creds_provider = ElastiCacheIAMProvider(user=username, cache_name=cache_name,
    is_serverless=True)
    redis_client = redis.Redis(host=elasticache_endpoint, port=6379,
    credential_provider=creds_provider, ssl=True, ssl_cert_reqs="none")

    key='uuid'
    # create a random UUID - this will be the sample element we add to the cache
    uuid_in = uuid.uuid4().hex
    redis_client.set(key, uuid_in)
    result = redis_client.get(key)
    decoded_result = result.decode("utf-8")
    # check the retrieved item matches the item added to the cache and print
    # the results
    if decoded_result == uuid_in:

```

```
    print(f"Success: Inserted {uuid_in}. Fetched {decoded_result} from Redis OSS.")
    else:
        raise Exception(f"Bad value retrieved. Expected {uuid_in}, got
{decoded_result}")

    return "Fetched value from Redis"
```

Este código usa la biblioteca redis-py de Python para colocar elementos en la caché y recuperarlos. Este código usa cachetools para almacenar en caché los tokens de autenticación de IAM generados durante 15 minutos. Para crear un paquete de despliegue que contenga redis-py y cachetools, lleve a cabo los siguientes pasos.

En el directorio del proyecto que contiene el archivo de código fuente app.py, cree un paquete de carpetas en el que instalar las bibliotecas redis-py y cachetools.

```
mkdir package
```

Instala redis-py y cachetools usando pip.

```
pip install --target ./package redis
pip install --target ./package cachetools
```

Cree un archivo.zip que contenga las bibliotecas redis-py y cachetools. En Linux y macOS, ejecute el siguiente comando. En Windows, utilice la utilidad zip que prefiera para crear un archivo.zip con las bibliotecas redis-py y cachetools en la raíz.

```
cd package
zip -r ../my_deployment_package.zip .
```

Añada el código de función al archivo .zip. En Linux y macOS, ejecute el siguiente comando. En Windows, usa la utilidad zip que prefieras para agregar app.py a la raíz del archivo.zip.

```
cd ..
zip my_deployment_package.zip app.py
```

Paso 2.2: crear el rol de IAM (rol de ejecución)

Adjunte la política AWS administrada nombrada AWSLambdaVPCLambdaAccessExecutionRole al rol.

```
aws iam attach-role-policy \  
  --role-name "elasticache-iam-auth-app" \  
  --policy-arn "arn:aws:iam::aws:policy/service-role/AWSLambdaVPCLambdaAccessExecutionRole"
```

Paso 2.3: cargar el paquete de despliegue (crear la función de Lambda)

En este paso, se crea la función Lambda (AccessRedis) mediante el comando AWS CLI `create-function`.

Desde el directorio del proyecto que contiene el archivo.zip del paquete de despliegue, ejecute el siguiente comando de la `create-function` CLI de Lambda.

Para la opción de rol, utilice el ARN del rol de ejecución que creó en el paso 2.2. Para la configuración `vpc-config`, introduzca listas separadas por comas de las subredes de la VPC predeterminada y el ID del grupo de seguridad de la VPC predeterminada. Puede encontrar estos valores en la Consola de Amazon VPC. Para buscar las subredes de su VPC predeterminadas, elija `Sus VPC` y, a continuación, elija la VPC predeterminada AWS de su cuenta. Para buscar el grupo de seguridad de esta VPC, vaya a `Seguridad` y elija `Grupos de seguridad`. Compruebe que ha seleccionado la región `us-east-1`.

```
aws lambda create-function \  
  --function-name AccessRedis \  
  --region us-east-1 \  
  --zip-file fileb://my_deployment_package.zip \  
  --role arn:aws:iam::123456789012:role/elasticache-iam-auth-app \  
  --handler app.lambda_handler \  
  --runtime python3.12 \  
  --timeout 30 \  
  --vpc-config SubnetIds=comma-separated-vpc-subnet-ids,SecurityGroupIds=default-security-group-id
```

Paso 3: comprobación de la función de Lambda

En este paso, se invoca la función Lambda manualmente mediante el comando `invoke`. Cuando se ejecuta la función Lambda, genera un UUID y lo escribe en la ElastiCache caché que especificó en el código Lambda. A continuación, la función de Lambda recupera el elemento de la caché.

1. Invoque la función Lambda `AccessRedis ()` mediante AWS Lambda el comando `invoke`.

```
aws lambda invoke \  
  --function-name AccessRedis \  
  --payload '{"key": "my_key", "value": "my_value"}'
```

```
--function-name AccessRedis \  
--region us-east-1 \  
output.txt
```

2. Compruebe que la función de Lambda se ha ejecutado correctamente del modo siguiente:
 - Revise el archivo output.txt.
 - Verifique los resultados en los CloudWatch registros abriendo la CloudWatch consola y eligiendo el grupo de registros para su función (/aws/lambda/). AccessRedis El flujo de registro debería contener una salida similar a lo siguiente:

```
Success: Inserted 826e70c5f4d2478c8c18027125a3e01e. Fetched  
826e70c5f4d2478c8c18027125a3e01e from Redis OSS.
```

- Revise los resultados en la consola. AWS Lambda

Paso 4: limpiar (opcional)

Para limpiar, sigue estos pasos.

Temas

- [Paso 4.1: Eliminar la función Lambda](#)
- [Paso 4.2: Eliminar la caché sin servidor](#)
- [Paso 4.3: Eliminar el rol y las políticas de IAM](#)

Paso 4.1: Eliminar la función Lambda

```
aws lambda delete-function \  
--function-name AccessRedis
```

Paso 4.2: Eliminar la caché sin servidor

Elimine la memoria caché.

```
aws elasticache delete-serverless-cache \  
--serverless-cache-name cache-01
```

Elimine los usuarios y el grupo de usuarios.

```
aws elasticache delete-user \  
  --user-id default-user-disabled  
  
aws elasticache delete-user \  
  --user-id iam-user-01  
  
aws elasticache delete-user-group \  
  --user-group-id iam-user-group-01
```

Paso 4.3: Eliminar el rol y las políticas de IAM

```
aws iam detach-role-policy \  
  --role-name "elasticache-iam-auth-app" \  
  --policy-arn "arn:aws:iam::123456789012:policy/elasticache-allow-all"  
  
aws iam detach-role-policy \  
  --role-name "elasticache-iam-auth-app" \  
  --policy-arn "arn:aws:iam::aws:policy/service-role/AWSLambdaVPCLambdaAccessExecutionRole"  
  
aws iam delete-role \  
  --role-name "elasticache-iam-auth-app"  
  
aws iam delete-policy \  
  --policy-arn "arn:aws:iam::123456789012:policy/elasticache-allow-all"
```

Diseñar y administrar su propio clúster ElastiCache (Redis OSS)

Si necesita un control detallado sobre su ElastiCache clúster, puede optar por diseñar su propio clúster. ElastiCache le permite operar un clúster basado en nodos al elegir el tipo de nodo, la cantidad de nodos y la ubicación de los nodos en las zonas de disponibilidad del clúster. AWS Como ElastiCache se trata de un servicio totalmente gestionado, gestiona automáticamente el aprovisionamiento del hardware, la supervisión, la sustitución de nodos y la aplicación de parches de software para el clúster.

Para obtener información sobre la configuración, consulte [Configuración](#). Para obtener más información sobre la administración, la actualización o la eliminación de nodos o clústeres, consulte [Administración de nodos](#). Para obtener una descripción general de los componentes principales de una ElastiCache implementación de Amazon cuando diseñas tu propio ElastiCache clúster, consulta estos [conceptos clave](#).

Temas

- [ElastiCache componentes y características](#)
- [ElastiCache Terminología \(Redis OSS\)](#)
- [Diseño de su propio clúster](#)
- [Administración de nodos](#)
- [Administración de clústeres](#)
- [Comparación de las cachés autodiseñadas de Memcached y Redis OSS](#)
- [Migración en línea a ElastiCache](#)
- [Elección de regiones y zonas de disponibilidad](#)

ElastiCache componentes y características

A continuación, encontrarás un resumen de los principales componentes de una ElastiCache implementación de Amazon.

Temas

- [ElastiCache nodos](#)
- [ElastiCache Fragmentos \(Redis OSS\)](#)

- [ElastiCache Clústeres \(Redis OSS\)](#)
- [ElastiCache replicación](#)
- [AWS Regiones y zonas de disponibilidad](#)
- [ElastiCache puntos finales](#)
- [ElastiCache grupos de parámetros](#)
- [ElastiCache seguridad](#)
- [ElastiCache grupos de subredes](#)
- [ElastiCache \(Redis OSS\) copias de seguridad](#)
- [ElastiCache eventos](#)

ElastiCache nodos

Un nodo es el componente básico más pequeño de una ElastiCache implementación. Un nodo puede existir de forma aislada o con alguna relación con otros nodos.

Un nodo es una partición de tamaño fijo de RAM segura conectada a la red. Cada nodo ejecuta una instancia del motor y la versión elegidos al crear el clúster. Si es necesario, puede escalar los nodos de un clúster para ampliar o reducir a un tipo de instancia diferente. Para obtener más información, consulte [Escalado ElastiCache \(Redis OSS\)](#).

Todos los nodos contenidos en un clúster son del mismo tipo de instancia y ejecutan el mismo motor de caché. Cada nodo de caché tiene su propio puerto y nombre de servicio de nombres de dominio (DNS). Se admiten varios tipos de nodos de caché, cada uno de los cuales tiene una cantidad diferente de memoria asociada. Para ver una lista de los tipos de instancia de nodos admitidos, consulte [Tipos de nodos compatibles](#).

Puede comprar nodos de forma pay-as-you-go individual, pagando solo por el uso de un nodo. También puede adquirir nodos reservados a una tarifa por hora más reducida. Si la tasa de uso es alta, adquirir nodos reservados podría ayudar a que ahorre dinero. Supongamos que el clúster está casi siempre en uso, por lo que de vez en cuando agregue nodos para gestionar los picos de uso. En este caso, puede adquirir una serie de nodos reservados para ejecutar la mayor parte del tiempo. A continuación, podrá comprar pay-as-you-go nodos para las veces que necesite añadir nodos de vez en cuando. Para obtener más información sobre los nodos reservados, consulte [Nodos reservados de ElastiCache](#).

Para obtener más información sobre los nodos, consulte [Administración de nodos](#).

ElastiCache Fragmentos (Redis OSS)

Una partición OSS de Redis (denominada grupo de nodos en la API y la CLI) es una agrupación de uno a seis nodos relacionados. Un clúster de OSS de Redis con el modo de clúster activado siempre tiene al menos un fragmento.

La fragmentación es un método de particionamiento de bases de datos que separa las bases de datos grandes en partes más pequeñas, rápidas y fáciles de gestionar denominadas fragmentos de datos. Esto puede aumentar la eficiencia de la base de datos al distribuir las operaciones en varias secciones independientes. El uso de fragmentos puede ofrecer muchos beneficios, como la mejora del rendimiento, la escalabilidad y la rentabilidad.

Los clústeres OSS de Redis con el modo de clúster activado pueden tener hasta 500 fragmentos, y los datos se dividen entre ellos. El límite de nodos o fragmentos se puede aumentar hasta un máximo de 500 por clúster si la versión del motor OSS de Redis es 5.0.6 o superior. Por ejemplo, puede elegir configurar un clúster de 500 nodos que oscila entre 83 particiones (uno primario y 5 réplicas por partición) y 500 particiones (único primario y sin réplicas). Asegúrese de que hay suficientes direcciones IP disponibles para acomodar el aumento. Algunos problemas comunes incluyen que las subredes del grupo de subredes tienen un rango CIDR demasiado pequeño o que otros clústeres comparten y utilizan considerablemente las subredes. Para obtener más información, consulte [Creación de un grupo de subredes](#). Para las versiones inferiores a 5.0.6, el límite es de 250 por clúster.

Para solicitar un aumento del límite, consulte [AWS Service Limits](#) y elija el tipo de límite Nodes per cluster per instance type (Nodos por clúster por tipo de instancias).

Una partición de varios nodos implementa la reproducción al tener un nodo primario de lectura/escritura y 1 a 5 nodos de réplica. Para obtener más información, consulte [Alta disponibilidad a través de grupos de reproducción](#).

Para obtener más información acerca de las particiones, consulte [Trabajar con particiones](#).

ElastiCache Clústeres (Redis OSS)

Un clúster OSS de Redis es una agrupación lógica de uno o más. [ElastiCache Fragmentos \(Redis OSS\)](#) Los datos se dividen entre los fragmentos de un clúster de OSS de Redis que tiene el modo de clúster activado.

Muchas ElastiCache operaciones están dirigidas a los clústeres:

- creación de un clúster
- Modificación de un clúster
- Uso de instantáneas de un clúster (todas las versiones de Redis)
- Eliminación de un clúster
- Visualización de elementos de un clúster
- Adición o eliminación de etiquetas de asignación de costos en un clúster

Para obtener información más detallada, consulte los siguientes temas relacionados:

- [Administración de clústeres](#) y [Administración de nodos](#)

Información acerca de los clústeres, nodos, y operaciones relacionadas.

- [AWS límites de servicio: Amazon ElastiCache](#)

Información sobre ElastiCache los límites, como el número máximo de nodos o clústeres. Para superar algunos de estos límites, puedes realizar una solicitud mediante el [formulario de solicitud de nodos de ElastiCache caché de Amazon](#).

- [Mitigación de errores](#)

Información sobre la mejora de la tolerancia a errores de los clústeres y grupos de reproducción.

Configuraciones típicas de clúster

A continuación se muestran configuraciones típicas de clúster.

Clústeres de OSS de Redis

Los clústeres de Redis OSS con el modo de clúster desactivado siempre contienen solo un fragmento (en la API y la CLI, un grupo de nodos). Un fragmento OSS de Redis contiene de uno a seis nodos. Si hay más de un nodo en una partición, esta admite la reproducción. En este caso, un nodo es el nodo primario de lectura/escritura y los demás son nodos de réplica de solo lectura.

Para mejorar la tolerancia a los errores, recomendamos tener al menos dos nodos en un clúster OSS de Redis y habilitar Multi-AZ. Para obtener más información, consulte [Mitigación de errores](#).

A medida que cambie la demanda de su clúster de OSS de Redis, puede ampliarlo o reducirlo. Para ello, mueva el clúster a un tipo de instancia de nodo diferente. Si su aplicación es de lectura

intensiva, le recomendamos añadir réplicas de solo lectura al clúster de Redis OSS. Al hacerlo, puede distribuir las lecturas entre un número más apropiado de nodos.

También puede utilizar la organización de datos en niveles. Los datos a los que se accede con más frecuencia se almacenan en la memoria y los datos a los que se accede con menos frecuencia se almacenan en el disco. La ventaja de utilizar la organización de datos en niveles es que disminuye las necesidades de memoria. Para obtener más información, consulte [Organización de datos en niveles](#).

ElastiCache admite cambiar el tipo de nodo de un clúster OSS de Redis a un tipo de nodo más grande de forma dinámica. Para obtener información acerca del escalado mediante ampliación o reducción, consulte [Escalado de clústeres de un solo nodo para Redis OSS \(modo de clúster desactivado\)](#) o [Escalado de clústeres de Redis OSS \(modo de clúster desactivado\) con nodos de réplica](#).

ElastiCache replicación

La reproducción se implementa al agrupar de dos a seis nodos en una partición (denominada grupo de nodos en la API y la CLI). Uno de estos nodos es el nodo principal de lectura/escritura. Todos los demás nodos son nodos de réplicas de solo lectura.

Cada nodo de réplica de lectura mantiene una copia de los datos del nodo principal. Los nodos de réplica usan mecanismos de reproducción asíncronos para mantener la sincronización con el nodo principal. Las aplicaciones pueden leer desde cualquiera de los nodos del clúster, pero solo pueden escribir en los nodos principales. Las réplicas de lectura mejoran la escalabilidad al distribuir las lecturas entre varios puntos de enlace. Las réplicas de lectura también mejoran la tolerancia a errores al mantener varias copias de los datos. Ubicar réplicas de lectura en múltiples zonas de disponibilidad mejora aún más la tolerancia a errores. Para obtener más información sobre la tolerancia a errores, consulte [Mitigación de errores](#).

Los clústeres OSS de Redis admiten un fragmento (en la API y la CLI, denominado grupo de nodos).

La reproducción desde la perspectiva de la API y la CLI utiliza una terminología diferente para mantener la compatibilidad con las versiones anteriores, pero los resultados son los mismos. En la siguiente tabla se muestran los términos de la API y la CLI para la implementación de la reproducción.

Comparación de la replicación: Redis OSS (modo de clúster desactivado) y Redis OSS (modo de clúster activado) --> clúster de Redis OSS con modo de clúster activado y clúster de Redis OSS con modo de clúster desactivado

En la siguiente tabla, puede encontrar una comparación de las características de los grupos de replicación de Redis OSS (modo de clúster desactivado) y Redis OSS (modo de clúster habilitado).

	Clúster OSS de Redis con el modo de clúster desactivado	Clúster OSS de Redis con el modo de clúster activado
Particiones (grupos de nodos)	1	1 a 500
Réplicas para cada partición (grupo de nodos)	0 a 5	0 a 5
Particiones de datos	No	Sí
Agregar o eliminar réplicas	Sí	Sí
Agregar o eliminar grupos de nodos	No	Sí
Compatible con el escalado vertical	Sí	Sí
Compatible con las actualizaciones del motor	Sí	Sí
Promociona la réplica a principal	Sí	Automático
Multi-AZ	Opcional	Obligatoria
Copia de seguridad y restauración	Sí	Sí

Notas:

Si cualquier primario no tiene réplicas y falla, se perderán todos los datos de ese primario.

Puede utilizar la copia de seguridad y la restauración para migrar a Redis OSS (modo de clúster activado).

	Clúster OSS de Redis con el modo de clúster desactivado	Clúster OSS de Redis con el modo de clúster activado
	Puede utilizar las funciones de copia de seguridad y restauración para cambiar el tamaño del clúster de Redis OSS (modo de clúster activado).	

Todas las particiones (grupos de nodos en la API y en la CLI) y todos los nodos deben residir en la misma región de AWS . Sin embargo, puede aprovisionar los nodos individuales en varias zonas de disponibilidad dentro de esa AWS región.

Las réplicas de lectura evitan las posibles pérdidas de datos, ya que los datos se replican a través de dos o varios nodos: el principal y una o varias réplicas de lectura. Para una mayor fiabilidad y una recuperación más rápida, recomendamos crear una o varias réplicas de lectura en distintas zonas de disponibilidad.

También puede aprovechar los almacenes de datos Globales. Al utilizar la función OSS Global Datastore for Redis, puede trabajar con una replicación totalmente gestionada, rápida, fiable y segura en todas las regiones. AWS Con esta función, puede crear clústeres de réplicas de lectura entre regiones para permitir las lecturas de baja latencia y ElastiCache la recuperación ante desastres en todas las regiones. AWS Para obtener más información, consulte [Replicación entre AWS regiones mediante almacenes de datos globales](#).

Replicación: límites y exclusiones

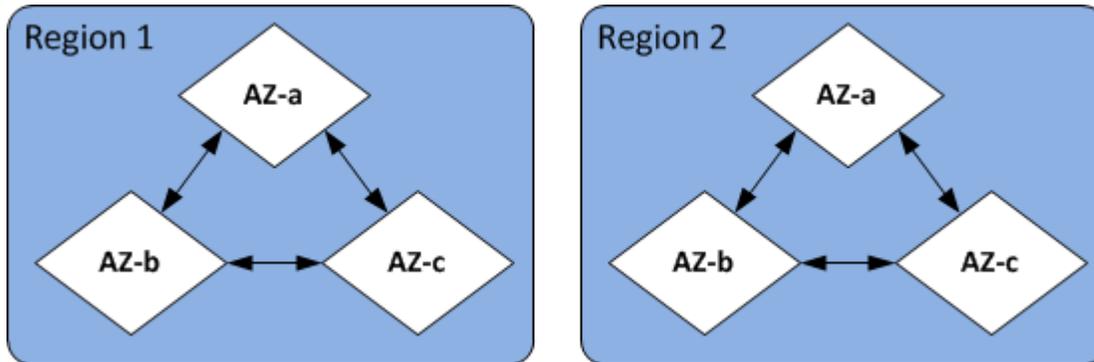
- Multi-AZ no se admite en los tipos de nodo T1.

AWS Regiones y zonas de disponibilidad

Amazon ElastiCache está disponible en varias AWS regiones de todo el mundo. De este modo, puede lanzar ElastiCache clústeres en las ubicaciones que mejor se adapten a las necesidades de su empresa. Por ejemplo, puede lanzarlos en la AWS región más cercana a sus clientes o para cumplir con ciertos requisitos legales.

De forma predeterminada, los AWS SDK AWS CLI, la ElastiCache API y la ElastiCache consola hacen referencia a la región EE.UU. Oeste (Oregón). A medida que ElastiCache se amplía la disponibilidad a nuevas AWS regiones, también están disponibles nuevos puntos de conexión para estas AWS regiones. Puede utilizarlos en sus solicitudes HTTP, los AWS SDK y ElastiCache la AWS CLI consola.

Cada AWS región está diseñada para estar completamente aislada de las demás AWS regiones. Dentro de cada una de ellas hay varias zonas de disponibilidad. Al lanzar los nodos en zonas de disponibilidad diferentes, puede lograr la máxima tolerancia a errores. Para obtener más información sobre AWS las regiones y las zonas de disponibilidad, consulte [Elección de regiones y zonas de disponibilidad](#). En el siguiente diagrama, puede ver una vista general del funcionamiento de AWS las regiones y las zonas de disponibilidad.



Para obtener información sobre AWS las regiones compatibles ElastiCache y sus puntos de conexión, consulte [Regiones y puntos de enlace admitidos](#).

ElastiCache puntos finales

Un punto final es la dirección única que utiliza su aplicación para conectarse a un ElastiCache nodo o clúster.

Terminales de un solo nodo para Redis OSS con el modo de clúster desactivado

El punto final de un clúster de Redis OSS de un solo nodo se utiliza para conectarse al clúster tanto para lecturas como para escrituras.

Terminales de varios nodos para Redis OSS con el modo de clúster desactivado

Un clúster de Redis OSS de varios nodos con el modo de clúster desactivado tiene dos tipos de puntos finales. El punto de conexión principal siempre se conecta al nodo principal en el clúster, incluso si el nodo específico de la función principal cambia. Utilice el punto de conexión principal para todas las escrituras en el clúster.

Use el punto de conexión de lector para dividir de manera uniforme las conexiones entrantes al punto de enlace entre todas las réplicas de lectura. Use los puntos de conexión del nodo para las operaciones de lectura (en la API/CLI, estas se denominan puntos de conexión de lectura).

Terminales Redis OSS (modo de clúster activado)

Un clúster OSS de Redis con el modo de clúster activado tiene un único punto final de configuración. Al conectarse al punto de conexión de configuración, la aplicación puede detectar los puntos de conexión principal y de lectura de cada partición del clúster.

Para obtener más información, consulte [Búsqueda de puntos de conexión](#).

ElastiCache grupos de parámetros

Los grupos de parámetros de caché son una forma sencilla de administrar la configuración de tiempo de ejecución para el software del motor compatible. Los parámetros se utilizan para controlar el uso de la memoria, las políticas de expulsión, los tamaños de elementos y mucho más. Un grupo de ElastiCache parámetros es una colección con nombre de parámetros específicos del motor que se puede aplicar a un clúster. De este modo, se asegura de que todos los nodos de ese clúster estén configurados exactamente de la misma forma.

Para obtener una lista de los parámetros compatibles, sus valores predeterminados y cuáles se pueden modificar, consulte [DescribeEngineDefaultParameters](#) (CLI: [describe-engine-default-parameters](#)).

Para obtener información más detallada sobre los grupos de ElastiCache parámetros, consulte [Configuración de los parámetros de motor mediante los grupos de parámetros](#)

ElastiCache seguridad

Para mejorar la seguridad, el acceso a los nodos ElastiCache (Redis OSS) está restringido a las aplicaciones que se ejecutan en las instancias de Amazon EC2 que usted permita. Puede controlar las instancias de Amazon EC2 que pueden acceder al clúster mediante los grupos de seguridad.

De forma predeterminada, todos los ElastiCache clústeres nuevos se lanzan en un entorno de Amazon Virtual Private Cloud (Amazon VPC). Puede utilizar grupos de subredes para conceder acceso al clúster desde las instancias de Amazon EC2 que se ejecutan en subredes concretas.

Además de restringir el acceso a los nodos, ElastiCache admite el TLS y el cifrado local para los nodos que ejecutan versiones específicas de ElastiCache. Para más información, consulte los siguientes temas:

- [Seguridad de datos en Amazon ElastiCache](#)
- [Autenticación con el comando REDIS OSS AUTH](#)

ElastiCache grupos de subredes

Un grupo de subredes es una colección de subredes (que suelen ser privadas) que se puede designar para los clústeres que se ejecuten en un entorno de Amazon VPC.

Si crea un clúster en una Amazon VPC, debe especificar un grupo de subredes de caché. ElastiCache usa ese grupo de subredes de caché para elegir una subred y las direcciones IP dentro de esa subred para asociarlas a los nodos de caché.

Para obtener más información sobre el uso de los grupos de subredes de caché en entornos de Amazon VPC, consulte lo siguiente:

- [Seguridad de ElastiCache y Amazon VPC](#)
- [Paso 3: autorizar acceso al clúster](#)
- [Subredes y grupos de subredes](#)

ElastiCache (Redis OSS) copias de seguridad

Una copia de seguridad es una point-in-time copia de un clúster de Redis OSS. Dichas copias de seguridad pueden usarse para restaurar un clúster existente o para propagar datos en un nuevo clúster. Las copias de seguridad se componen de todos los datos en un clúster, además de algunos metadatos.

Según la versión de Redis OSS que se ejecute en el clúster, el proceso de copia de seguridad requiere diferentes cantidades de memoria reservada para funcionar correctamente. Para más información, consulte los siguientes temas:

- [Instantánea y restauración](#)
- [Cómo se implementan la sincronización y la copia de seguridad](#)
- [El impacto en el rendimiento de las copias de seguridad de los clústeres de autodiseño](#)
- [Asegurarse de tener suficiente memoria para crear una instantánea de Redis OSS](#)

ElastiCache eventos

Cuando se producen eventos importantes en un clúster de caché, ElastiCache envía una notificación a un tema específico de Amazon SNS. Algunos eventos significativos pueden ser, por ejemplo, errores al agregar un nodo, adiciones de nodos correctas, cambios en los grupos de seguridad, entre

otros. Al monitorear los eventos clave podrá conocer el estado actual de los clústeres y, según el evento, adoptar medidas correctivas.

Para obtener más información sobre ElastiCache los eventos, consulte [SNSMonitorización de ElastiCache eventos por parte de Amazon](#).

ElastiCache Terminología (Redis OSS)

En octubre de 2016, Amazon ElastiCache lanzó el soporte para Redis OSS 3.2. En ese momento, añadimos la posibilidad de particionar los datos en hasta 500 fragmentos (denominados grupos de nodos en la ElastiCache API y). AWS CLI Para mantener la compatibilidad con las versiones anteriores, ampliamos las operaciones de la API de la versión 2015-02-02 para incluir la nueva funcionalidad de Redis OSS.

Al mismo tiempo, empezamos a utilizar en la ElastiCache consola la terminología que se utiliza en esta nueva funcionalidad y que es común en todo el sector. Estos cambios implican que, en algunos aspectos, la terminología usada en la API y la CLI puede ser diferente de la terminología empleada en la consola. La siguiente lista identifica términos que pueden variar entre la API y la CLI y la consola.

Clúster de caché o nodo frente a nodo

Existe una one-to-one relación entre un nodo y un clúster de caché cuando no hay nodos de réplica. Por lo tanto, la ElastiCache consola solía utilizar los términos indistintamente. A partir de ahora, la consola utiliza el término nodo para todo. La única excepción es el botón Create Cluster, que lanza el proceso para crear un clúster con o sin nodos de réplica.

La ElastiCache API y AWS CLI seguimos utilizando los términos como lo han hecho en el pasado.

Clúster frente a grupo de reproducción

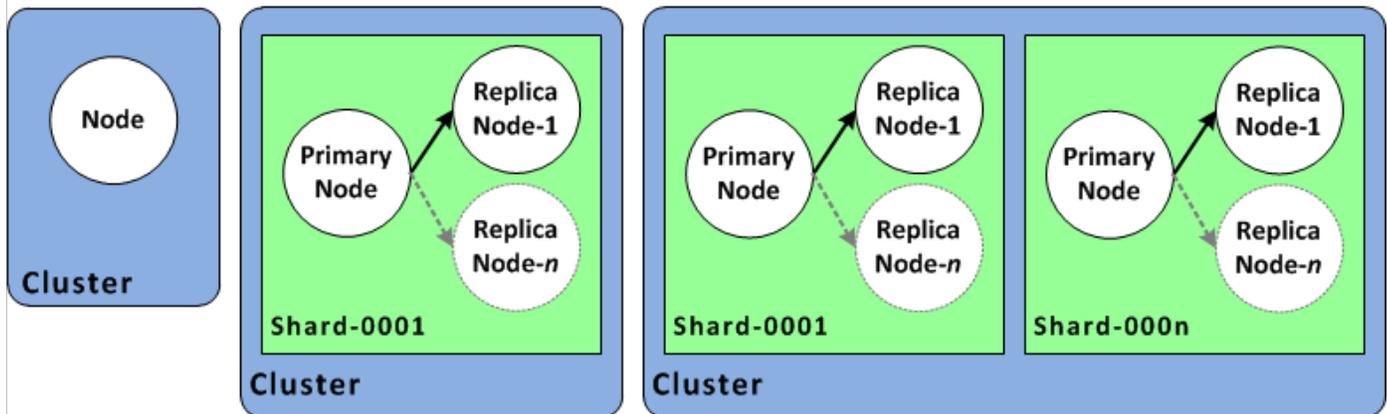
La consola ahora usa el término clúster para todos los clústeres ElastiCache (Redis OSS). La consola utiliza el término clúster en todas estas circunstancias:

- Cuando el clúster es un clúster de Redis OSS de un solo nodo.
- Cuando el clúster es un clúster OSS (modo de clúster desactivado) de Redis que admite la replicación dentro de un único fragmento (en la API y la CLI, denominado grupo de nodos).
- Cuando el clúster es un clúster OSS de Redis (modo de clúster habilitado) que admite la replicación entre 1 y 90 fragmentos o hasta 500 con una solicitud de aumento de límite. Para solicitar un aumento del límite, consulte [AWS Service Limits](#) y elija el tipo de límite Nodes per cluster per instance type (Nodos por clúster por tipo de instancias).

Para obtener más información acerca de los grupos de reproducción, consulte [Alta disponibilidad a través de grupos de reproducción](#).

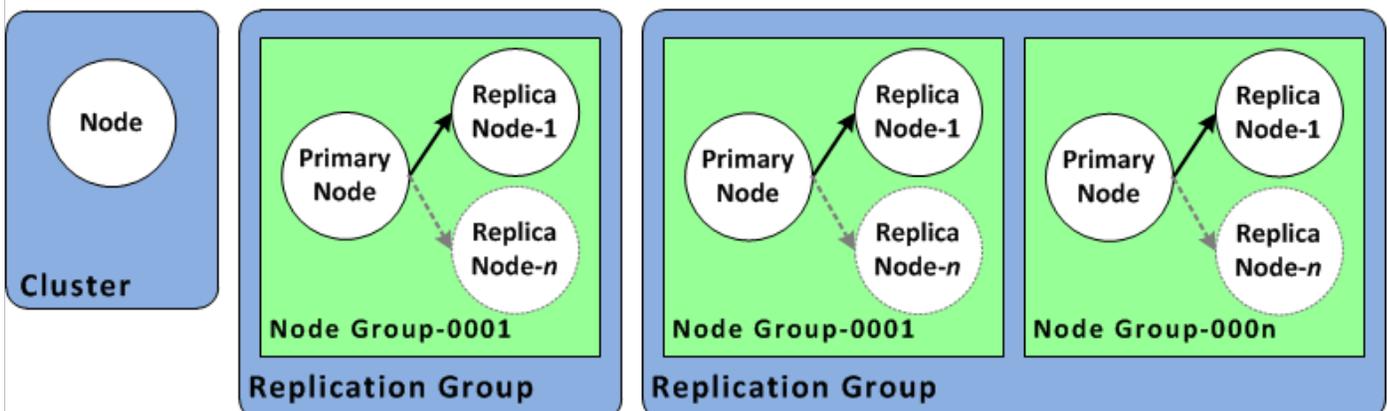
El siguiente diagrama ilustra las distintas topologías de los clústeres ElastiCache (Redis OSS) desde la perspectiva de la consola.

ElastiCache (Redis OSS): Console View



La ElastiCache API y AWS CLI las operaciones siguen distinguiendo los clústeres de un solo nodo ElastiCache (Redis OSS) de los grupos de replicación de varios nodos. El siguiente diagrama ilustra las distintas topologías ElastiCache (Redis OSS) desde la ElastiCache API y desde su perspectiva. AWS CLI

ElastiCache (Redis OSS): API/CLI View



Grupo de reproducción frente al almacén de datos global

Un almacén de datos global es una colección de uno o varios clústeres que se replican entre sí entre regiones, mientras que un grupo de reproducción replica datos entre un clúster (clúster modo habilitado) con varias particiones. Un almacén de datos global consta de lo siguiente:

- **Clúster principal (activo):** un clúster principal acepta escrituras que se replican en todos los clústeres dentro del almacén de datos global. Un clúster principal también acepta además solicitudes de lectura.

- **Clúster secundario (pasivo):** un clúster secundario solo acepta solicitudes de lectura y replica las actualizaciones de datos a partir de un clúster principal. Un clúster secundario debe estar en una AWS región diferente a la del clúster principal.

Para obtener información sobre almacenes de datos globales, consulte [Replicación entre AWS regiones mediante almacenes de datos globales](#).

Diseño de su propio clúster

A continuación, se indican las acciones únicas que debe realizar para empezar a diseñar el ElastiCache clúster.

Temas

- [Configuración de un grupo de subredes](#)
- [Paso 1: crear un grupo de subredes](#)
- [Paso 2: crear un clúster](#)
- [Paso 3: autorizar acceso al clúster](#)
- [Paso 4: conectarse al nodo de un clúster](#)
- [Paso 5: eliminar un clúster](#)
- [ElastiCache tutoriales y vídeos](#)
- [¿Qué tengo que hacer ahora?](#)

Configuración de un grupo de subredes

Antes de crear el clúster, asegúrese de crear un grupo de subredes. Un grupo de subredes de caché es una colección de subredes que puede desear designar para sus clústeres de caché en una VPC. Al lanzar un clúster de caché en una VPC, tiene que seleccionar un grupo de subredes de caché. A continuación, ElastiCache utiliza ese grupo de subredes de caché para asignar direcciones IP dentro de esa subred a cada nodo de caché del clúster.

Cuando cree un nuevo grupo de subredes, tenga en cuenta el número de direcciones IP disponibles. Si la subred tiene pocas direcciones IP libres, el número de nodos que podrá agregar al clúster será limitado. Para solucionar este problema, puede asignar una o varias subredes a un grupo de subredes para, de este modo, disponer de suficientes direcciones IP en la zona de disponibilidad de su clúster. Hecho esto, podrá agregar más nodos a su clúster.

Para obtener más información sobre la configuración ElastiCache , consulte. [Configuración](#)

Paso 1: crear un grupo de subredes

En los siguientes procedimientos, se muestra cómo crear un grupo de subredes denominado `mysubnetgroup` (consola) y la AWS CLI.

Creación de un grupo de subredes (consola)

En el siguiente procedimiento, se muestra cómo crear un grupo de subredes (consola).

Para crear un grupo de subredes (consola)

1. Inicie sesión en la AWS Management Console y abra la consola de ElastiCache en <https://console.aws.amazon.com/elasticache/>.
2. En la lista de navegación, elija Subnet Groups (Grupos de subredes).
3. Elija Create Subnet Group (Crear grupo de subredes).
4. En el asistente Create Subnet Group (Crear grupo de subredes), haga lo siguiente. Una vez que la configuración sea la deseada, elija Yes, Create (Sí, crear).
 - a. En el cuadro Name (Nombre), escriba un nombre para el grupo de subredes.
 - b. En el cuadro Description (Descripción), escriba la descripción del grupo de subredes.
 - c. En el cuadro de VPC ID (ID de la VPC), elija la Amazon VPC que creó anteriormente.
 - d. En las listas de Availability Zone (Zona de disponibilidad) y Subnet ID (ID de la subred), elija la zona de disponibilidad o [zona local](#) y el ID de su subred privada y, a continuación, elija Add (Agregar).

Subnet group settings

A subnet group is a collection of subnets (typically private). Designate a subnet group for your clusters running in an Amazon Virtual Private Cloud (VPC) environment.

Name

The name is required, can have up to 255 characters, and must begin with a letter. It should not end with a hyphen or contain two consecutive hyphens. Valid characters: A-Z, a-z, 0-9, and - (hyphen).

Description - optional

VPC ID

The identifier for the VPC environment where your cluster is to run.

 ▼ Create VPC [↗](#)

ⓘ For Multi-AZ high availability mode, choose IDs for at least two subnets from two Availability Zones in the table below.

Selected subnets (6) Manage

Availability Zone ▲	Subnet ID ▼	Outpost ID ▼	CIDR block ▼
us-east-1a	subnet-██████████		172.31.16.0/20
us-east-1b	subnet-██████████		172.31.32.0/20
us-east-1c	subnet-██████████		172.31.0.0/20
us-east-1d	subnet-██████████		172.31.80.0/20

5. En el mensaje de confirmación que aparece, elija Close (Cerrar).

El grupo de subredes nuevo aparecerá en la lista Subnet Groups (Grupos de subredes) de la consola de ElastiCache. En la parte inferior de la ventana, podrá elegir el grupo de subredes para ver detalles tales como todas las subredes asociadas al grupo.

Crear un grupo de subredes (AWS CLI)

En el símbolo del sistema, utilice el comando `create-cache-subnet-group` para crear un grupo de subredes.

Para Linux, macOS o Unix:

```
aws elasticache create-cache-subnet-group \
  --cache-subnet-group-name mysubnetgroup \
  --cache-subnet-group-description "Testing" \
```

```
--subnet-ids subnet-53df9c3a
```

Para Windows:

```
aws elasticache create-cache-subnet-group ^  
  --cache-subnet-group-name mysubnetgroup ^  
  --cache-subnet-group-description "Testing" ^  
  --subnet-ids subnet-53df9c3a
```

Este comando debería producir un resultado similar al siguiente:

```
{  
  "CacheSubnetGroup": {  
    "VpcId": "vpc-37c3cd17",  
    "CacheSubnetGroupDescription": "Testing",  
    "Subnets": [  
      {  
        "SubnetIdentifier": "subnet-53df9c3a",  
        "SubnetAvailabilityZone": {  
          "Name": "us-west-2a"  
        }  
      }  
    ],  
    "CacheSubnetGroupName": "mysubnetgroup"  
  }  
}
```

Para obtener más información, consulte el tema de la AWS CLI [create-cache-subnet-group](#).

Paso 2: crear un clúster

Antes de crear un clúster para su uso en producción, obviamente debe considerar cómo configurará el clúster a fin de satisfacer las necesidades del negocio. Estos problemas se abordan en la sección [Preparación de un clúster](#). A los efectos de este ejercicio de introducción, creará un clúster con el modo de clúster deshabilitado y podrá aceptar los valores de configuración predeterminados donde se apliquen.

El clúster que crea se ejecutará en un entorno real, no en uno de pruebas. Se le cobrarán las tarifas de ElastiCache uso estándar de la instancia hasta que la elimine. Los cargos totales serán mínimos (normalmente menos de un dólar) si completa el ejercicio descrito aquí de una vez y elimina el

clúster al finalizar. Para obtener más información sobre las tasas de ElastiCache uso, consulta [Amazon ElastiCache](#).

El clúster se lanza en una nube privada virtual (VPC) en función del servicio de Amazon VPC.

Creación de un clúster de Redis OSS (modo de clúster desactivado) (consola)

Para crear un clúster de Redis OSS (modo de clúster desactivado) mediante la consola ElastiCache

1. Inicia sesión en la ElastiCache consola de Amazon AWS Management Console y ábrela en <https://console.aws.amazon.com/elasticache/>.
2. En la lista de la esquina superior derecha, selecciona la AWS región en la que quieres lanzar este clúster.
3. En el panel de navegación, seleccione Get started (Introducción).
4. Elija Create VPC (Crear una VPC) y siga los pasos que se detallan en [Creating a Virtual Private Cloud \(VPC\)](#) (Creación de una nube privada virtual).
5. En la página del ElastiCache panel de control, elija la memoria caché de OSS de Redis y, a continuación, elija Crear memoria caché de OSS de Redis.
6. En Cluster settings (Configuración de clúster), realice la siguiente operación:
 - a. Elija Configure and create a new cluster (Configurar y crear un nuevo clúster).
 - b. Para Cluster mode (Modo clúster), elige Disabled (Desactivado).
 - c. Para Cluster info (Información del clúster) introduzca un valor para Name (Nombre).
 - d. (Opcional) Introduzca un valor para Description (Descripción).
7. En Location (Ubicación):

AWS Cloud

1. Para Nube de AWS , le recomendamos que acepte la configuración predeterminada de Multi-AZ y Conmutación por error automática. Para obtener más información, consulte [Minimizar el tiempo de inactividad en ElastiCache \(Redis OSS\)](#) con Multi-AZ.
2. En Cluster settings (Configuración del clúster)
 - a. Para Engine version (Versión del motor), elija una versión disponible.
 - b. En Port (Puerto), utilice el puerto predeterminado, 6379. Si tiene un motivo para utilizar un puerto diferente, escriba el número de puerto.

- c. En Grupo de parámetros, elija un grupo de parámetros o cree uno nuevo. Los grupos de parámetros controlan los parámetros de tiempo de ejecución de su clúster. Para obtener más información acerca de los grupos de parámetros, consulte [Parámetros específicos de Redis OSS](#) y [Creación de un grupo de parámetros](#).

 Note

Cuando se selecciona un grupo de parámetros para definir los valores de configuración del motor, dicho grupo de parámetros se aplica a todos los clústeres del almacén de datos global. En la página Parameter Groups (Grupos de parámetros), el atributo Global sí/no, indica si un grupo de parámetros forma parte de un almacén de datos global.

- d. Para Node type, elija la flecha hacia abajo (▼).

En el cuadro de diálogo Cambiar tipo de nodo, elija un valor para Familia de instancias para el tipo de nodo que desea. A continuación, elija el tipo de nodo que desea utilizar para este clúster y, a continuación, elija Guardar.

Para obtener más información, consulte [Elección del tamaño del nodo](#).

Si elige un tipo de nodo r6gd, la organización de datos en niveles se activará automáticamente. Para obtener más información, consulte [Organización de datos en niveles](#).

- e. Para Number of replicas (Número de réplicas), elija el número de réplicas de lectura que desee. Si habilitó Multi-AZ, el número debe estar entre 1 y 5.
3. En Connectivity (Conectividad)
 - a. Para el Network type (Tipo de red), elija las versiones IP que admitirá este clúster.
 - b. Para los grupos de subredes, elija la subred que desee aplicar a este clúster. ElastiCache usa ese grupo de subredes para elegir una subred y las direcciones IP dentro de esa subred para asociarlas a los nodos. ElastiCache Los clústeres requieren una subred de doble pila con direcciones IPv4 e IPv6 asignadas para funcionar en modo de doble pila y una subred solo de IPv6 para que funcione solo de IPv6.

Al crear un nuevo grupo de subredes, escriba el VPC ID (ID de VPC) al que pertenece.

Para obtener más información, consulte:

- [Elección de un tipo de red.](#)
- [Cree una subred en la VPC.](#)

Si es [Uso de zonas locales con ElastiCache](#), debe crear o elegir una subred que se encuentre en la zona local.

Para obtener más información, consulte [Subredes y grupos de subredes.](#)

4. En Availability zone placements Ubicaciones de las zonas de disponibilidad), tiene dos opciones:

- ElastiCache Sin preferencia: elige la zona de disponibilidad.
- Specify availability zones (Especificar zonas de disponibilidad): especifique la zona de disponibilidad para cada clúster.

Si decide especificar las zonas de disponibilidad, para cada clúster en cada fragmento, elija la zona de disponibilidad de la lista.

Para obtener más información, consulte [Elección de regiones y zonas de disponibilidad.](#)

5. Elija Next (Siguiente).

6. En la configuración avanzada de Redis OSS

- Para Security (Seguridad):
 - i. Para cifrar sus datos, tiene las siguientes opciones:
 - Encryption at rest (Cifrado en reposo): permite el cifrado de los datos almacenados en el disco. Para obtener más información, consulte [Cifrado en reposo.](#)

Note

Tiene la opción de proporcionar una clave de cifrado diferente; para ello, seleccione la clave AWS KMS gestionada por el cliente y elija la clave. Para obtener más información, consulte [Uso de claves de AWS KMS administradas por el cliente](#).

- Encryption in-transit (Cifrado en tránsito): permite el cifrado de datos del cable. Para obtener más información, consulte [Cifrado en tránsito](#). Para la versión 6.0 y superior del motor OSS de Redis, si habilita el cifrado en tránsito, se le pedirá que especifique una de las siguientes opciones de control de acceso:
 - No Access Control (Sin control de acceso): este es el valor predeterminado. Esto indica que no hay restricciones en el acceso de los usuarios al clúster.
 - User Group Access Control List (Lista de control de acceso del grupo de usuarios): seleccione un grupo de usuarios con un conjunto definido de usuarios que pueden acceder al clúster. Para obtener más información, consulte [Administración de grupos de usuarios con la consola y la CLI](#).
 - Usuario predeterminado de AUTH de Redis OSS: mecanismo de autenticación para el servidor OSS de Redis. Para obtener más información, consulte [Redis OSS AUTH](#).
- Redis OSS AUTH: un mecanismo de autenticación para el servidor Redis OSS. Para obtener más información, consulte [Redis OSS AUTH](#).

Note

Para las versiones de Redis OSS desde la 3.2.6 en adelante, excepto la versión 3.2.10, la única opción es Redis OSS AUTH.

- ii. En Grupos de seguridad, elija los grupos de seguridad que desea para este clúster. Un grupo de seguridad actúa como un firewall para controlar el acceso de red al clúster. Puede utilizar el grupo de seguridad predeterminado para la VPC o crear uno nuevo.

Para obtener más información sobre los grupos de seguridad, consulte [Grupos de seguridad de su VPC](#) en la Guía del usuario de Amazon VPC.

7. En las copias de seguridad automáticas programadas regularmente, seleccione Enable automatic backups (Habilitar copias de seguridad automáticas) y, a continuación, escriba el número de días que desea conservar cada copia de seguridad automática antes de que se elimine automáticamente. Si no quiere tener copias de seguridad automáticas programadas regularmente, desactive la casilla de verificación Enable automatic backups (Habilitar copias de seguridad automáticas). En cualquier caso, siempre tendrá la opción de crear copias de seguridad manuales.

Para obtener más información sobre la copia de seguridad y la restauración de Redis OSS, consulte. [Instantánea y restauración](#)

8. (Opcional) Especifique un periodo de mantenimiento. El período de mantenimiento es el tiempo, que suele durar una hora cada semana, en el que se ElastiCache programa el mantenimiento del sistema del clúster. Puede ElastiCache elegir el día y la hora de su período de mantenimiento (sin preferencia) o puede elegir el día, la hora y la duración usted mismo (especifique el período de mantenimiento). Si elige Specify maintenance window, elija Start day, Start time y Duration (en horas) de las listas para el periodo de mantenimiento. Todas las horas se indican en UCT.

Para obtener más información, consulte [Administración del mantenimiento](#).

9. (Opcional) Para Logs (Registros):
 - En Log format (Formato de registro), elija Text o JSON.
 - En Tipo de destino, elija CloudWatch Logs o Kinesis Firehose.
 - En Destino del registro, elija Crear nuevo e introduzca el nombre del grupo de CloudWatch registros de Logs o el nombre de la transmisión de Firehose, o bien elija Seleccionar existente y, a continuación, elija el nombre del grupo de CloudWatch registros de Logs o el nombre de la transmisión de Firehose,
10. En el caso de las etiquetas, para ayudarte a gestionar tus clústeres y otros ElastiCache recursos, puedes asignar tus propios metadatos a cada recurso en forma de etiquetas. Para obtener más información, consulte [Etiquetado de los recursos de ElastiCache](#).
11. Elija Siguiente.
12. Revise todas las entradas y opciones y, a continuación, realice todos los cambios necesarios. Cuando haya terminado, elija Create (Crear).

On premises

1. Para On premises (En las instalaciones), le recomendamos que deje Auto-failover (Conmutación por error automática) habilitada. Para obtener más información, consulte [Minimizar el tiempo de inactividad en ElastiCache \(Redis OSS\) con Multi-AZ](#)
2. Para terminar de crear el clúster, siga los pasos que se indican en [Using Outposts](#) (Uso de Outposts).

En cuanto el estado de su clúster sea available (disponible), podrá conceder a Amazon EC2 acceso a este, así como conectarse a él y comenzar a utilizarlo. Para obtener más información, consulte [Paso 3: autorizar acceso al clúster](#) y [Paso 4: conectarse al nodo de un clúster](#).

Important

Una vez que su clúster esté disponible, se cobrará por cada hora u hora parcial que el clúster esté activo, incluso si no lo está utilizando de forma activa. Para dejar de incurrir en cargos por este clúster, debe eliminarlo. Consulte [Eliminación de un clúster](#).

Creación de un clúster de Redis OSS (modo de clúster desactivado) (AWS CLI)

Example

El siguiente código CLI crea un clúster de caché de Redis OSS (modo de clúster desactivado) sin réplicas.

Para Linux, macOS o Unix:

```
aws elasticache create-cache-cluster \  
--cache-cluster-id my-cluster \  
--cache-node-type cache.r4.large \  
--engine redis \  
--num-cache-nodes 1 \  
--snapshot-arns arn:aws:s3:::my_bucket/snapshot.rdb
```

Para Windows:

```
aws elasticache create-cache-cluster ^
```

```
--cache-cluster-id my-cluster ^  
--cache-node-type cache.r4.large ^  
--engine redis ^  
--num-cache-nodes 1 ^  
--snapshot-arns arn:aws:s3:::my_bucket/snapshot.rdb
```

Para trabajar con el modo de clúster habilitado, consulte los temas siguientes:

- Para utilizar la consola, consulte [Creación de un clúster de Redis OSS \(modo de clúster activado\) \(consola\)](#).
- Para usar el AWS CLI, consulte. [Crear un clúster de Redis OSS \(modo de clúster habilitado\) \(AWS CLI\)](#)

Paso 3: autorizar acceso al clúster

En esta sección se presupone que se encuentra familiarizado con los procesos de lanzar instancias de Amazon EC2 y conectarse a estas. Para obtener más información, consulte la [Guía de introducción de Amazon EC2](#).

Todos los ElastiCache clústeres están diseñados para que se pueda acceder a ellos desde una instancia de Amazon EC2. El escenario más común es acceder a un ElastiCache clúster desde una instancia de Amazon EC2 en la misma Amazon Virtual Private Cloud (Amazon VPC), que será el caso de este ejercicio.

De forma predeterminada, el acceso de red al clúster está limitado a la cuenta que se utilizó para crearlo. Antes de poder conectarse a un clúster desde una instancia EC2, debe autorizar a la instancia EC2 el acceso al clúster. Los pasos necesarios dependerán de si ha lanzado el clúster en EC2-VPC o EC2-Classic.

El caso de uso más común es cuando una aplicación implementada en una instancia EC2 debe conectarse a un clúster en la misma VPC. La forma más sencilla de administrar el acceso entre instancias EC2 y clústeres de la misma VPC es realizar lo siguiente:

1. Cree un grupo de seguridad de VPC para su clúster. Este grupo de seguridad se puede utilizar para restringir el acceso a las instancias de clúster. Por ejemplo, puede crear una regla personalizada para este grupo de seguridad que permita el acceso mediante TCP utilizando el puerto que asignó al clúster de base de datos cuando lo creó y una dirección IP que se utilizará para obtener acceso al clúster.

El puerto predeterminado para los clústeres de OSS y los grupos de replicación de Redis es. 6379

Important

Los grupos ElastiCache de seguridad de Amazon solo se aplican a los clústeres que no se ejecutan en un entorno de Amazon Virtual Private Cloud (VPC). Si la ejecución tiene lugar en una Amazon Virtual Private Cloud, Security Groups (Grupos de seguridad) no se encontrará disponible en el panel de navegación de la consola.

Si ejecuta sus ElastiCache nodos en una Amazon VPC, controla el acceso a sus clústeres con grupos de seguridad de Amazon VPC, que son distintos de los grupos de

seguridad. ElastiCache Para obtener más información sobre el uso ElastiCache en una Amazon VPC, consulte [Seguridad de ElastiCache y Amazon VPC](#)

2. Cree un grupo de seguridad de VPC para sus instancias EC2 (servidores web y de aplicaciones). Si es necesario, este grupo de seguridad puede permitir el acceso a la instancia EC2 desde Internet a través de la tabla de enrutamiento de la VPC. Por ejemplo, puede establecer reglas en este grupo de seguridad para permitir el acceso mediante TCP a la instancia EC2 a través del puerto 22.
3. Cree reglas personalizadas en el grupo de seguridad para su clúster que permitan las conexiones desde el grupo de seguridad que creó para las instancias EC2. Esto permitirá a cualquier miembro del grupo de seguridad obtener acceso a los clústeres.

 Note

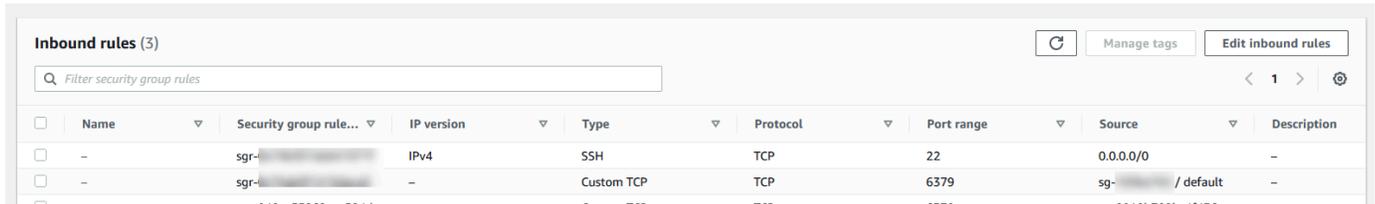
Si planea utilizar [Local Zones](#), asegúrese de que las ha habilitado. Cuando crea un grupo de subredes en esa zona local, la VPC se amplía a dicha zona local y la VPC tratará la subred como cualquier subred de cualquier otra zona de disponibilidad. Todas las gateways y tablas de enrutamiento relevantes se ajustarán de forma automática.

Para crear una regla en un grupo de seguridad de VPC que permita establecer conexiones desde otro grupo de seguridad

1. [Inicie sesión en la consola AWS de administración y abra la consola de Amazon VPC en https://console.aws.amazon.com/vpc.](https://console.aws.amazon.com/vpc)
2. En el panel de navegación, elija Security Groups (Grupos de seguridad).
3. Seleccione o cree un grupo de seguridad que utilizará para sus instancias de clúster. En Inbound Rules (Reglas de entrada), seleccione Edit Inbound Rules (Editar reglas de entrada) y, a continuación, seleccione Add Rule (Agregar regla). Este grupo de seguridad permitirá el acceso a los miembros de otro grupo de seguridad.
4. En Type (Tipo), elija Custom TCP Rule (Personalizar regla de TCP).
 - a. En Port Range (Rango de puerto), especifique el puerto que utilizó al crear su clúster.

El puerto predeterminado para los clústeres y grupos de replicación de Redis OSS es. 6379

- b. En el cuadro Source (Fuente), comience a escribir el ID del grupo de seguridad. Desde la lista, seleccione el grupo de seguridad que utilizará para sus instancias de Amazon EC2.
5. Cuando haya terminado, elija Save (Guardar).



Una vez que haya habilitado el acceso, se encontrará listo para conectarse al nodo, como se describe en la siguiente sección.

Para obtener información sobre cómo acceder a su ElastiCache clúster desde una Amazon VPC diferente, una AWS región diferente o incluso su red corporativa, consulte lo siguiente:

- [Patrones de acceso para acceder a una ElastiCache caché en una Amazon VPC](#)
- [Acceder a ElastiCache los recursos desde el exterior AWS](#)

Paso 4: conectarse al nodo de un clúster

Antes de continuar, realice el [Paso 3: autorizar acceso al clúster](#).

En esta sección se da por sentado que ha creado una instancia de Amazon EC2 y que se puede conectar a ella. Para obtener instrucciones al respecto, consulte la [Guía de introducción a Amazon EC2](#).

Una instancia de Amazon EC2 se puede conectar a un nodo de clúster solo si lo ha autorizado.

Encontrar los puntos de enlace de los nodos

Una vez que el clúster se encuentra en estado disponible y ha autorizado el acceso a él, puede iniciar sesión en una instancia de Amazon EC2 y conectarse al clúster. Para ello, primero debe determinar el punto de conexión.

Búsqueda de los puntos finales de un clúster de Redis OSS (modo de clúster desactivado) (consola)

Si un clúster de Redis OSS (modo de clúster desactivado) tiene un solo nodo, el punto final del nodo se utiliza tanto para las lecturas como para las escrituras. Si un clúster tiene varios nodos, hay tres tipos de puntos de conexión: el punto de conexión primario, el punto de conexión de lectura y los puntos de conexión de nodo.

El punto de conexión principal es un nombre de DNS que siempre se resuelve en el nodo principal del clúster. El punto de conexión principal es inmune a los cambios en el clúster, como la promoción de una réplica de lectura a la función principal. Para una actividad de escritura, recomendamos que sus aplicaciones se conecten al punto de conexión principal.

Un punto final del lector dividirá en partes iguales las conexiones entrantes al punto final entre todas las réplicas de lectura de un clúster ElastiCache (Redis OSS). Factores adicionales como cuando la aplicación crea las conexiones o cómo la aplicación (re)utiliza las conexiones determinará la distribución del tráfico. Los puntos de enlace de lector se mantienen actualizados con los cambios del clúster en tiempo real cuando se añaden o eliminan réplicas. Puede colocar las múltiples réplicas de lectura de su clúster ElastiCache (Redis OSS) en diferentes zonas de AWS disponibilidad (AZ) para garantizar una alta disponibilidad de los puntos finales de los lectores.

Note

Un punto de conexión del lector no es un equilibrador de carga. Se trata de un registro de DNS que se resolverá en una dirección IP de uno de los nodos de réplica en forma de turno rotativo.

Para la actividad de lectura, las aplicaciones pueden conectarse también a cualquier nodo del clúster. A diferencia del punto de conexión principal, los puntos de conexión de nodo se resuelven en puntos de enlace específicos. Si realiza un cambio en su clúster, como añadir o eliminar una réplica, debe actualizar los puntos de enlace del nodo en su aplicación.

Para encontrar los puntos finales de un clúster de Redis OSS (modo de clúster desactivado)

1. [Inicie sesión AWS Management Console y abra la ElastiCache consola en https://console.aws.amazon.com/elasticache/.](https://console.aws.amazon.com/elasticache/)
2. En el panel de navegación, elija las cachés OSS de Redis.

Aparecerá la pantalla de clústeres con una lista que incluirá las cachés sin servidor de Redis OSS, los clústeres de Redis OSS (modo de clúster desactivado) y Redis OSS (modo de clúster activado) existentes. Elija el clúster que ha creado en la sección [Creación de un clúster de Redis OSS \(modo de clúster desactivado\) \(consola\)](#).

3. Para buscar los puntos de conexión principal o de lector, elija el nombre del clúster (no el botón de opción).

▼ Cluster details			
Cluster name [redacted]	Description [redacted]	Node type cache.r6g.large	Status Available
Engine Redis OSS	Engine version 6.0.5	Global datastore -	Global datastore role -
Update status Update available	Cluster mode Off	Shards 1	Number of nodes 3
Data tiering Disabled	Multi-AZ Enabled	Auto-failover Enabled	Encryption in transit Disabled
Encryption at rest Disabled	Parameter group default.redis6.x	Outpost ARN -	Configuration endpoint -
Primary endpoint [redacted]-encrypted.llru6f.ng.0001.use1.cache.amazonaws.com:6379	Reader endpoint [redacted]-encrypted-ro.llru6f.ng.0001.use1.cache.amazonaws.com:6379	ARN [redacted]	

Puntos finales principales y de lectura de un clúster de Redis OSS (modo de clúster desactivado)

Si el clúster tiene solo un nodo, quiere decir que no hay punto de conexión principal y, por lo tanto, puede continuar con el paso siguiente.

4. Si el clúster OSS de Redis (modo de clúster desactivado) tiene nodos de réplica, puede encontrar los puntos finales de los nodos de réplica del clúster seleccionando el nombre del clúster y, a continuación, seleccionando la pestaña Nodos.

Aparecerá la pantalla de nodos con los nodos del clúster, principal y réplicas; cada uno con su punto de conexión.

<input type="checkbox"/>	Node Name	Status	Current Role	Port	Endpoint
<input type="checkbox"/>	test-no-001	available	primary	6379	test-no-001.usw2.cache.amazonaws.com
<input type="checkbox"/>	test-no-002	available	replica	6379	test-no-002.usw2.cache.amazonaws.com
<input type="checkbox"/>	test-no-003	available	replica	6379	test-no-003.usw2.cache.amazonaws.com

Puntos de enlace de nodo de un clúster de Redis OSS (modo de clúster desactivado)

5. Para copiar un punto de conexión al portapapeles:
 - a. Un punto de conexión cada vez, encuentre el punto de conexión que desea copiar.
 - b. Elija el icono de copia directamente delante del punto de conexión.

El punto de conexión se ha copiado ahora en el portapapeles. Para obtener información sobre el uso del punto de conexión a fin de conectarse a un nodo, consulte [Conexión a los nodos](#).

Un punto final principal de Redis OSS (modo de clúster desactivado) tiene un aspecto similar al siguiente. Existe una diferencia dependiendo de si está habilitado el cifrado en tránsito o no lo está.

Cifrado en tránsito no habilitado

```
clusterName.xxxxxx.nodeId.regionAndAz.cache.amazonaws.com:port
```

```
redis-01.7abc2d.0001.usw2.cache.amazonaws.com:6379
```

Cifrado en tránsito habilitado

```
master.clusterName.xxxxxx.regionAndAz.cache.amazonaws.com:port
```

```
master.ncit.ameaqx.use1.cache.amazonaws.com:6379
```

Para buscar puntos de enlace, consulte el tema correspondiente sobre el motor y el tipo de clúster que ejecuta.

- [Búsqueda de puntos de conexión](#)
- [Búsqueda de puntos finales para un clúster de Redis OSS \(modo de clúster activado\) \(consola\)](#): necesita el punto de conexión de configuración del clúster.
- [Búsqueda de puntos de conexión \(AWS CLI\)](#)
- [Búsqueda de puntos finales \(\) ElastiCache API](#)

Conectarse a un clúster o grupo de replicación de Redis OSS (Linux)

Ahora que tiene el punto de conexión que necesita, puede iniciar sesión en una instancia EC2 y conectarse al clúster o grupo de reproducción. En el siguiente ejemplo, utilice la utilidad redis-cli para conectarse a un clúster. La última versión de redis-cli también admite SSL/TLS a fin de conectar clústeres habilitados para cifrado y autenticación.

En el siguiente ejemplo se utilizan instancias de Amazon EC2 que ejecutan Amazon Linux y Amazon Linux 2. Para obtener más detalles sobre la instalación y compilación de redis-cli con otras distribuciones de Linux, consulte la documentación del sistema operativo específico.

Note

En este proceso se cubre la prueba de una conexión mediante la utilidad redis-cli solo para uso no planificado. [Para obtener una lista de los clientes de Redis OSS compatibles, consulte la documentación de Redis OSS.](#) Para ver ejemplos del uso de los AWS SDK con ElastiCache, consulte. [Introducción a ElastiCache y a los SDK de AWS](#)

Conexión a un clúster sin cifrar en modo de clúster deshabilitado

1. Ejecute el siguiente comando para conectarse al clúster y reemplace *punto de conexión principal* y *número de puerto* por el punto de conexión del clúster y el número de puerto. (El puerto predeterminado para Redis OSS es 6379).

```
src/redis-cli -h primary-endpoint -p port number
```

El resultado en una línea de comandos de Redis OSS es similar al siguiente:

```
primary-endpoint:port number
```

2. Ahora puede ejecutar los comandos de Redis OSS.

```
set x Hello
OK

get x
"Hello"
```

Conexión a un clúster en modo de clúster sin cifrar habilitado

1. Ejecute el siguiente comando para conectarse al clúster y reemplace *punto de conexión de configuración* y *número de puerto* por el punto de conexión del clúster y el número de puerto. (El puerto predeterminado para Redis OSS es 6379).

```
src/redis-cli -h configuration-endpoint -c -p port number
```

 Note

En el comando anterior, la opción `-c` habilita el modo clúster en función de las [redirecciones -ASK y -MOVED](#).

El resultado en una línea de comandos de Redis OSS es similar al siguiente:

```
configuration-endpoint:port number
```

2. Ahora puede ejecutar los comandos de Redis OSS. Tenga en cuenta que la redirección se produce porque la habilitó mediante la opción `-c`. Si la redirección no se encuentra habilitada, el comando devuelve el error MOVED (MOVIDO). Para obtener más información sobre el error MOVED, consulte la especificación del [clúster OSS de Redis](#).

```
set x Hi
-> Redirected to slot [16287] located at 172.31.28.122:6379
OK
set y Hello
OK
get y
"Hello"
set z Bye
-> Redirected to slot [8157] located at 172.31.9.201:6379
OK
get z
"Bye"
get x
-> Redirected to slot [16287] located at 172.31.28.122:6379
"Hi"
```

Conexión a un clúster habilitado para cifrado y autenticación

De forma predeterminada, redis-cli utiliza una conexión TCP no cifrada cuando se conecta a Redis OSS. La opción BUILD_TLS=yes habilita SSL/TLS en el momento de la compilación de redis-cli, como se muestra en la sección [Descarga y configuración de redis-cli](#). La habilitación de AUTH es opcional. Sin embargo, debe habilitar el cifrado en tránsito para habilitar AUTH. Para obtener más información sobre el ElastiCache cifrado y la autenticación, consulte. [ElastiCache cifrado en tránsito \(\) TLS](#)

Note

Puede utilizar la opción `--tls` con redis-cli para conectarse a clústeres cifrados en los modos clúster habilitado y deshabilitado. Si un clúster tiene un conjunto de tokens AUTH, puede utilizar la opción `-a` para proporcionar una contraseña de AUTH.

En los siguientes ejemplos, asegúrese de reemplazar *punto de conexión de clúster* y *número de puerto* por el punto de conexión del clúster y el número de puerto. (El puerto predeterminado para Redis OSS es 6379.)

Conectarse a clústeres cifrados en modo de clúster deshabilitado

El siguiente ejemplo se conecta a un clúster habilitado para cifrado y autenticación:

```
src/redis-cli -h cluster-endpoint --tls -a your-password -p port number
```

El siguiente ejemplo se conecta a un clúster que solo tiene habilitado el cifrado:

```
src/redis-cli -h cluster-endpoint --tls -p port number
```

Conectarse a clústeres cifrados en modo de clúster habilitado

El siguiente ejemplo se conecta a un clúster habilitado para cifrado y autenticación:

```
src/redis-cli -c -h cluster-endpoint --tls -a your-password -p port number
```

El siguiente ejemplo se conecta a un clúster que solo tiene habilitado el cifrado:

```
src/redis-cli -c -h cluster-endpoint --tls -p port number
```

Tras conectarse al clúster, puede ejecutar los comandos de Redis OSS, como se muestra en los ejemplos anteriores, para los clústeres no cifrados.

Alternativa a redis-cli

Si el clúster no se encuentra en el modo de clúster habilitado y necesita realizar una conexión con el clúster para una prueba corta, pero sin pasar por la compilación de redis-cli, puede utilizar telnet u openssl. En los siguientes comandos de ejemplo, asegúrese de reemplazar *punto de conexión de clúster* y *número de puerto* por el punto de conexión del clúster y el número de puerto. (El puerto predeterminado para Redis OSS es 6379).

El siguiente ejemplo se conecta a un clúster en modo de clúster deshabilitado que se encuentra habilitado para el cifrado o la autenticación:

```
openssl s_client -connect cluster-endpoint:port number
```

Si el clúster tiene una contraseña establecida, primero conéctese al clúster. Después de conectarse, autentique el clúster con el siguiente comando y, a continuación, presione la tecla Enter. En el siguiente ejemplo, reemplace *your-password* con la contraseña del clúster.

```
Auth your-password
```

El siguiente ejemplo se conecta a un clúster en modo de clúster deshabilitado que no cuenta con el cifrado o la autenticación habilitados:

```
telnet cluster-endpoint port number
```

Conectarse a un clúster o grupo de replicación de Redis OSS (Windows)

Para conectarse al clúster de OSS de Redis desde una instancia de EC2 de Windows mediante la CLI de Redis OSS, debe descargar el paquete redis-cli y usar redis-cli.exe para conectarse al clúster de OSS de Redis desde una instancia de EC2 de Windows.

En el siguiente ejemplo, se utiliza la utilidad redis-cli para conectarse a un clúster que no esté habilitado para el cifrado y que ejecute Redis OSS. Para obtener más información sobre Redis OSS y los comandos de Redis OSS disponibles, consulte los comandos de Redis OSS en el sitio web de [Redis OSS](#).

Para conectarse a un clúster de OSS de Redis que no esté habilitado para el cifrado mediante redis-cli

1. Conéctese a la instancia de Amazon EC2 mediante la utilidad de conexión que prefiera. Para obtener instrucciones sobre cómo conectarse a una instancia de Amazon EC2, consulte la [Guía de introducción de Amazon EC2](#).
2. Copie y pegue el enlace <https://github.com/microsoftarchive/redis/releases/download/win-3.0.504/Redis-x64-3.0.504.zip> en un navegador de Internet para descargar el archivo zip del cliente OSS de Redis de la versión disponible en GitHub <https://github.com/microsoftarchive/redis/releases/tag/win-3.0.504>

Extraiga el archivo zip en la carpeta/ruta deseada.

Abra la línea de comandos, vaya al directorio OSS de Redis y ejecute el comando. c :

```
\Redis>redis-cli -h Redis_Cluster_Endpoint -p 6379
```

Por ejemplo:

```
c:\Redis>redis-cli -h cmd.xxxxxxxx.ng.0001.usw2.cache.amazonaws.com -p 6379
```

3. Ejecute los comandos de Redis OSS.

Ahora está conectado al clúster y puede ejecutar comandos de Redis OSS como los siguientes.

```
set a "hello"           // Set key "a" with a string value and no expiration
OK
get a                   // Get value for key "a"
"hello"
get b                   // Get value for key "b" results in miss
(nil)
set b "Good-bye" EX 5  // Set key "b" with a string value and a 5 second expiration
"Good-bye"
get b                   // Get value for key "b"
"Good-bye"

                        // wait >= 5 seconds

get b
(nil)                   // key has expired, nothing returned
quit                    // Exit from redis-cli
```

Paso 5: eliminar un clúster

Siempre que un clúster tenga el estado available, se cobrará por él, independientemente de si lo esté usando de forma activa o no. Para que dejen de devengarse cargos, elimine el clúster.

Warning

- Al eliminar un clúster ElastiCache (RedisOSS), se conservan las instantáneas manuales. También puede crear una instantánea final antes de eliminar el clúster. Por el contrario, las instantáneas de caché automáticas no se conservan. Para obtener más información, consulte [Instantánea y restauración](#).
- `CreateSnapshotse` necesita permiso para crear una instantánea final. Sin este permiso, la API llamada fallará con una `Access Denied` excepción.

Usando el AWS Management Console

El siguiente procedimiento elimina un único clúster de su implementación. Para eliminar varios clústeres, repita el procedimiento por cada clúster que desee eliminar. No es necesario esperar a un clúster para terminar de eliminarlo antes de empezar el procedimiento para eliminar otro clúster.

Para eliminar un clúster

1. Inicia sesión en la ElastiCache consola de Amazon AWS Management Console y ábrela en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de control de la ElastiCache consola, selecciona RedisOSS.

Aparece una lista de todas las cachés que ejecutan OSS Redis.

3. Para elegir el clúster que se va a eliminar, elija el nombre del clúster de la lista de clústeres. En este caso, el nombre del clúster que creó en [Paso 2: crear un clúster](#).

Important

Solo puede eliminar un clúster a la vez desde la ElastiCache consola. Al elegir varios clústeres, se deshabilita la operación de eliminación.

4. En Actions (Acciones), seleccione Delete (Eliminar).
5. En la pantalla de confirmación Eliminar clúster, escriba el nombre del clúster y elija Copia de seguridad final. A continuación, seleccione Eliminar para eliminar el clúster o Cancelar para conservarlo.

Si elige Delete, el estado del clúster cambia a deleting.

En cuanto el clúster desaparezca de la lista de clústeres, dejará de incurrir en gastos.

Usando el AWS CLI

El código siguiente elimina el clúster de caché `my-cluster`. En este caso, sustituya `my-cluster` con el nombre del clúster que creó en [Paso 2: crear un clúster](#).

```
aws elasticache delete-cache-cluster --cache-cluster-id my-cluster
```

La `delete-cache-cluster` CLI acción solo elimina un clúster de caché. Para eliminar varios clústeres de caché, llame a `delete-cache-cluster` por cada clúster de caché que desee eliminar. No es necesario esperar a un clúster de caché para terminar de eliminarlo antes de eliminar otro.

Para Linux, macOS o Unix:

```
aws elasticache delete-cache-cluster \
```

```
--cache-cluster-id my-cluster \  
--region us-east-2
```

Para Windows:

```
aws elasticache delete-cache-cluster ^  
--cache-cluster-id my-cluster ^  
--region us-east-2
```

Para obtener más información, consulte AWS CLI el ElastiCache tema [delete-cache-cluster](#)

ElastiCache tutoriales y vídeos

Los siguientes tutoriales abordan tareas de interés para el ElastiCache usuario de Amazon.

- [ElastiCache Vídeos](#)
- [Tutorial: Configuración de una función Lambda para acceder a Amazon ElastiCache en una Amazon VPC](#)

ElastiCache Vídeos

A continuación, encontrarás vídeos que te ayudarán a aprender ElastiCache conceptos básicos y avanzados de Amazon. Para obtener información sobre la AWS formación, consulta [AWS Formación y certificación](#).

Temas

- [Videos introductorios](#)
- [Videos avanzados](#)

Videos introductorios

Los siguientes vídeos te presentan Amazon ElastiCache.

Temas

- [AWS re:Invent 2020: Novedades de Amazon ElastiCache](#)
- [AWS re:Invent 2019: Novedades de Amazon ElastiCache](#)
- [AWS re:Invent 2017: Novedades de Amazon ElastiCache](#)
- [DAT204: Creación de aplicaciones escalables en servicios AWS NoSQL \(re:Invent 2015\)](#)
- [DAT207: aceleración del rendimiento de las aplicaciones con Amazon ElastiCache \(re:Invent 2013\)AWS](#)

AWS re:Invent 2020: Novedades de Amazon ElastiCache

[AWS re:Invent 2020: Novedades de Amazon ElastiCache](#)

AWS re:Invent 2019: Novedades de Amazon ElastiCache

[AWS re:Invent 2019: Novedades de Amazon ElastiCache](#)

AWS re:Invent 2017: Novedades de Amazon ElastiCache

[AWS re:Invent 2017: Novedades de Amazon ElastiCache](#)

DAT204: Creación de aplicaciones escalables en servicios AWS NoSQL (re:Invent 2015)

En esta sesión, analizaremos las ventajas de las bases de datos NoSQL y haremos un recorrido por los principales servicios NoSQL que ofrecen Amazon DynamoDB y AWS Amazon. ElastiCache

Luego, escuchamos a dos clientes líderes, Expedia y Mapbox, hablar sobre sus casos de uso y desafíos arquitectónicos, y cómo los abordaron utilizando los servicios AWS NoSQL, incluidos los patrones de diseño y las mejores prácticas. Terminará esta sesión con una mejor comprensión de NoSQL y sus potentes capacidades, listo para abordar los desafíos de su base de datos con confianza.

[DAT204: Creación de aplicaciones escalables en servicios AWS NoSQL \(re:Invent 2015\)](#)

DAT207: aceleración del rendimiento de las aplicaciones con Amazon ElastiCache (re:Invent 2013)AWS

En este vídeo, descubre cómo puedes usar Amazon ElastiCache para implementar fácilmente un sistema de almacenamiento en caché en memoria para acelerar el rendimiento de tus aplicaciones. Le mostramos cómo usar Amazon ElastiCache para mejorar la latencia de sus aplicaciones y reducir la carga en los servidores de sus bases de datos. Asimismo, mostraremos cómo crear una capa de almacenamiento en caché que resulte fácil de administrar y escalar a medida que crezca su aplicación. Durante esta sesión, analizamos varios escenarios y casos de uso que pueden resultar beneficiosos al habilitar el almacenamiento en caché, y analizamos las funciones que ofrece Amazon ElastiCache.

[DAT207: Aceleración del rendimiento de las aplicaciones con Amazon ElastiCache \(re:Invent 2013\)](#)

Videos avanzados

Los siguientes vídeos tratan ElastiCache temas más avanzados de Amazon.

Temas

- [Diseña para triunfar con las ElastiCache mejores prácticas de Amazon \(re:Invent 2020\)](#)
- [Potencia tus aplicaciones en tiempo real con Amazon ElastiCache \(re:Invent 2019\)](#)
- [Prácticas recomendadas: migración de clústeres de Redis OSS de Amazon EC2 ElastiCache a \(re:Invent 2019\)](#)
- [Cómo escalar una plataforma de deportes de fantasía con Amazon ElastiCache y Amazon Aurora STP11 \(re:Invent 2018\)](#)
- [Redis OSS fiable y escalable en la nube con Amazon ElastiCache \(re:Invent 2018\)](#)
- [ElastiCache Análisis profundo: patrones de diseño para almacenes de datos en memoria \(re:Invent 2018\)](#)
- [DAT305: Amazon ElastiCache Deep Dive \(re:Invent 2017\)](#)
- [DAT306: Amazon ElastiCache Deep Dive \(re:Invent 2016\)](#)

- [DAT317: Cómo utiliza IFTTT \(Redis OSS\) para predecir eventos ElastiCache \(re:Invent 2016\)](#)
- [DAT407: Amazon ElastiCache Deep Dive \(re:Invent 2015\)](#)
- [SDD402: Amazon ElastiCache Deep Dive \(re:Invent 2014\)](#)
- [DAT307: profundiza en la ElastiCache arquitectura y los patrones de diseño de Amazon \(re:Invent 2013\)](#)

[Diseña para triunfar con las ElastiCache mejores prácticas de Amazon \(re:Invent 2020\)](#)

Con el crecimiento explosivo de las aplicaciones en tiempo real fundamentales para la empresa basadas en Redis OSS, la disponibilidad, la escalabilidad y la seguridad se han convertido en las principales consideraciones. Conozca las mejores prácticas para preparar Amazon ElastiCache para el éxito con el escalado en línea, la alta disponibilidad en las implementaciones Multi-AZ y las configuraciones de seguridad.

[Diseña para triunfar con las ElastiCache mejores prácticas de Amazon \(re:Invent 2020\)](#)

[Potencia tus aplicaciones en tiempo real con Amazon ElastiCache \(re:Invent 2019\)](#)

Con el rápido crecimiento de la adopción de la nube y los escenarios nuevos que habilita, las aplicaciones necesitan latencia de microsegundos y un alto rendimiento para soportar millones de solicitudes por segundo. Tradicionalmente, los desarrolladores han confiado en hardware especializado y soluciones alternativas, como bases de datos con base en disco combinadas con técnicas de reducción de datos, a fin de administrar datos para aplicaciones en tiempo real. Estos enfoques pueden ser costosos y no se pueden escalar. Descubra cómo puede aumentar el rendimiento de las aplicaciones en tiempo real mediante el uso de Amazon, integrado en memoria y totalmente gestionado, ElastiCache para obtener un rendimiento extremo, una alta escalabilidad, disponibilidad y seguridad.

[Potencia tus aplicaciones en tiempo real con Amazon ElastiCache \(re:Invent 2019:\)](#)

[Prácticas recomendadas: migración de clústeres de Redis OSS de Amazon EC2 ElastiCache a \(re:Invent 2019\)](#)

Administrar los clústeres de Redis OSS por su cuenta puede resultar difícil. Debe aprovisionar hardware, aplicar parches de software, realizar copias de seguridad de los datos y monitorear las cargas de trabajo de forma constante. Con la nueva función de migración en línea para Amazon ElastiCache, ahora puede mover fácilmente sus datos desde Redis OSS autohospedado en Amazon EC2 a Amazon totalmente gestionado, con el modo de ElastiCache clúster desactivado. En esta

sesión, conocerá la nueva herramienta de migración en línea, verá una demostración y, lo que es más importante, aprenderá las mejores prácticas para una migración fluida a Amazon ElastiCache.

[Prácticas recomendadas: migración de clústeres de Redis OSS de Amazon EC2 ElastiCache a \(re:Invent 2019\)](#)

[Cómo escalar una plataforma de deportes de fantasía con Amazon ElastiCache y Amazon Aurora STP11 \(re:Invent 2018\)](#)

Dream11 es la empresa emergente líder en tecnología deportiva de la India. Tiene una base creciente de más de 40 millones de usuarios que practican varios deportes, incluido cricket de fantasía, fútbol y baloncesto, y actualmente asiste a un millón de usuarios simultáneos, que producen tres millones de solicitudes por minuto en un tiempo de respuesta de 50 milisegundos. En esta charla, el director de tecnología de Dream11, Amit Sharma, explica cómo la empresa utiliza Amazon Aurora y Amazon ElastiCache para gestionar el tráfico flash, que puede triplicarse en un período de respuesta de 30 segundos. Sharma también habla sobre la escalabilidad de las transacciones sin bloquear y comparte los pasos para gestionar el tráfico rápido, con lo que asiste a cinco millones de usuarios activos diarios. Título completo: AWS re:Invent 2018: escalando una plataforma de deportes de fantasía con Amazon ElastiCache y Amazon Aurora (STP11)

[Cómo escalar una plataforma de deportes de fantasía con Amazon ElastiCache y Amazon Aurora STP11 \(re:Invent 2018\)](#)

[Redis OSS fiable y escalable en la nube con Amazon ElastiCache \(re:Invent 2018\)](#)

Esta sesión trata sobre las funciones y mejoras de nuestro servicio compatible con Redis OSS, Amazon ElastiCache (Redis OSS). Abordamos las características clave, como Redis OSS 5, las mejoras de escalabilidad y rendimiento, la seguridad y el cumplimiento, y mucho más. También analizamos las próximas características y los casos prácticos de clientes.

[Redis OSS fiable y escalable en la nube con Amazon ElastiCache \(re:Invent 2018\)](#)

[ElastiCache Análisis profundo: patrones de diseño para almacenes de datos en memoria \(re:Invent 2018\)](#)

En esta sesión, ofrecemos un vistazo entre bastidores para aprender sobre el diseño y la arquitectura de Amazon ElastiCache. Conozca los patrones de diseño habituales de nuestras ofertas de Redis OSS y Memcached y descubra cómo los clientes los utilizan para el procesamiento de datos en memoria a fin de reducir la latencia y mejorar el rendimiento de las aplicaciones. Revisamos las ElastiCache mejores prácticas, los patrones de diseño y los antipatrones.

[ElastiCache Análisis profundo: patrones de diseño para almacenes de datos en memoria \(re:Invent 2018\)](#)

DAT305: Amazon ElastiCache Deep Dive (re:Invent 2017)

Eche un vistazo entre bastidores para obtener información sobre ElastiCache el diseño y la arquitectura de Amazon. Conozca los patrones de diseño habituales de nuestras ofertas OSS de Memcached y Redis y descubra cómo los han utilizado los clientes en las operaciones en memoria para reducir la latencia y mejorar el rendimiento de las aplicaciones. En este vídeo, analizamos las ElastiCache mejores prácticas, los patrones de diseño y los antipatrones.

El video introduce lo siguiente:

- ElastiCache (Redis OSS): refragmentación en línea
- ElastiCache seguridad y cifrado
- ElastiCache (Redis OSS) versión 3.2.10

[DAT305: Amazon ElastiCache Deep Dive \(re:Invent 2017\)](#)

DAT306: Amazon ElastiCache Deep Dive (re:Invent 2016)

Eche un vistazo entre bastidores para obtener información sobre ElastiCache el diseño y la arquitectura de Amazon. Conozca los patrones de diseño habituales de nuestras ofertas OSS de Memcached y Redis y descubra cómo los han utilizado los clientes en las operaciones en memoria para reducir la latencia y mejorar el rendimiento de las aplicaciones. Durante esta sesión, analizamos las ElastiCache mejores prácticas, los patrones de diseño y los antipatrones.

[DAT306: Amazon ElastiCache Deep Dive \(re:Invent 2016\)](#)

DAT317: Cómo utiliza IFTTT (Redis OSS) para predecir eventos ElastiCache (re:Invent 2016)

IFTTT es un servicio gratuito que permite a las personas hacer mucho más con sus servicios favoritos, desde automatizar tareas sencillas hasta transformar el modo en el que controlar su hogar e interactuar con este. IFTTT utiliza ElastiCache (Redis OSS) para almacenar el historial de transacciones y las predicciones de programación, así como los índices de los documentos de registro en Amazon S3. Vea esta sesión para aprender cómo la potencia de programación de Lua y los tipos de datos de Redis OSS permitieron a las personas lograr algo que no habrían podido lograr en ningún otro lugar.

[DAT317: cómo IFTTT utiliza ElastiCache \(Redis OSS\) para predecir eventos \(re:Invent 2016\)](#)

DAT407: Amazon ElastiCache Deep Dive (re:Invent 2015)

Eche un vistazo entre bastidores para obtener información sobre el diseño y ElastiCache la arquitectura de Amazon. Conozca los patrones de diseño habituales de nuestras ofertas de OSS de Memcached y Redis y descubra cómo los clientes los han utilizado para las operaciones en memoria y han conseguido mejorar la latencia y el rendimiento de las aplicaciones. Durante esta sesión, revisamos las mejores prácticas, los patrones de diseño y los antipatrones relacionados con Amazon ElastiCache.

[DAT407: Amazon ElastiCache Deep Dive \(re:Invent 2015\)](#)

SDD402: Amazon ElastiCache Deep Dive (re:Invent 2014)

En este vídeo, examinamos los casos de uso más habituales del almacenamiento en caché: los motores OSS de Memcached y Redis, los patrones que le ayudan a determinar qué motor se adapta mejor a sus necesidades, el uso de hash coherente y mucho más como medio para crear aplicaciones rápidas y escalables. Frank Wiebe, científico principal de Adobe, detalla cómo Adobe utiliza Amazon ElastiCache para mejorar la experiencia del cliente y ampliar su negocio.

[DAT402: Amazon ElastiCache Deep Dive \(re:Invent 2014\)](#)

DAT307: profundiza en la ElastiCache arquitectura y los patrones de diseño de Amazon (re:Invent 2013)

En este video, examinaremos el almacenamiento en caché, estrategias de almacenamiento en caché, escalado ascendente y monitoreo. También comparamos los motores OSS de Memcached y Redis. Durante esta sesión, también revisamos las mejores prácticas y los patrones de diseño relacionados con Amazon ElastiCache.

[DAT307: Profundice en la ElastiCache arquitectura y los patrones de diseño de Amazon \(AWS re:Invent 2013\)](#)

¿Qué tengo que hacer ahora?

Ahora que ha probado el ejercicio de introducción, puede explorar las siguientes secciones para obtener más información acerca de ElastiCache y las herramientas disponibles:

- [Introducción a AWS](#)
- [Herramientas para Amazon Web Services](#)
- [AWS Interfaz de línea de comandos](#)
- [Referencia de la API de Amazon ElastiCache](#)

Después de completar el ejercicio de introducción, puede leer estas secciones para obtener más información acerca de la administración de ElastiCache:

- [Elección del tamaño del nodo](#)

Desea que la caché sea lo suficientemente grande como para alojar todos los datos que necesite almacenar en caché. Al mismo tiempo, no desea pagar más caché de la que necesita. Utilice este tema para elegir el mejor tamaño de nodo.

- [ElastiCache mejores prácticas y estrategias de almacenamiento en caché](#)

Explica cómo identificar y abordar los problemas que pueden influir en la eficiencia del clúster.

Administración de nodos

Un nodo es el componente más pequeño de una ElastiCache implementación de Amazon. Se trata de un fragmento de tamaño fijo de RAM segura conectada a la red. Cada nodo ejecuta el motor que se eligió cuando se creó o se modificó por última vez el clúster o el grupo de reproducción. Cada nodo tiene su propio puerto y nombre de servicio de nombres de dominio (DNS). Se admiten varios tipos de ElastiCache nodos, cada uno con cantidades variables de memoria asociada y potencia computacional.

En términos generales, debido a su compatibilidad con la fragmentación, las implementaciones de Redis OSS (modo de clúster habilitado) tienen varios nodos más pequeños. Por el contrario, las implementaciones de Redis OSS (modo de clúster desactivado) tienen menos nodos y son más grandes en un clúster. Consulte [Elección del tamaño del nodo](#) para ver una explicación más detallada sobre el tamaño de nodo que debe utilizarse.

Temas

- [Visualización ElastiCache del estado de los nodos](#)
- [Nodos y fragmentos de Redis OSS](#)
- [Conexión a los nodos](#)
- [Tipos de nodos compatibles](#)
- [Reinicio de nodos \(solo modo de clúster deshabilitado\)](#)
- [Sustitución de nodos](#)
- [Nodos reservados de ElastiCache](#)
- [Migración de nodos de generación anterior](#)

Entre las opciones importantes en relación con los nodos, se encuentran las siguientes:

- [Adición de nodos a un clúster](#)
- [Eliminación de nodos de un clúster](#)
- [Escalado ElastiCache \(Redis OSS\)](#)
- [Búsqueda de puntos de conexión](#)

Visualización ElastiCache del estado de los nodos

Con la [ElastiCache consola](#), puede acceder rápidamente al estado de su ElastiCache nodo. El estado de un ElastiCache nodo indica el estado del nodo. Puedes usar los siguientes procedimientos para ver el estado del ElastiCache nodo en la ElastiCache consola de Amazon, el AWS CLI comando o la operación de la API.

Los posibles valores de estado de ElastiCache los nodos se muestran en la siguiente tabla. En esta tabla también se muestra si se le facturará por el ElastiCache nodo.

Tipo	Facturado	Descripción
available	Facturado	El ElastiCache nodo está en buen estado y disponible.
creating	No facturado	Se está creando el ElastiCache nodo. No se puede obtener acceso al nodo mientras se está creando.
deleting	No facturado	Se está eliminando el ElastiCache nodo.
modifying	Facturado	El ElastiCache nodo se está modificando debido a una solicitud de un cliente para modificarlo.
updating	Facturado	Un estado de actualización indica que se cumple una o más de las siguientes

Tipo	Facturado	Descripción
		<p>condiciones en el ElastiCache nodo de Amazon:</p> <ul style="list-style-type: none">• El ElastiCache nodo se está parcheando como parte de la actualización del servicio. Para obtener más información sobre las actualizaciones del servicio, consulta la página de ayuda de Amazon ElastiCache Managed Maintenance and Service Updates.• Los grupos de seguridad de VPC se están actualizando para el ElastiCache clúster.• El ElastiCache clúster se está ampliando o reduciendo.• Se están modificando las configuraciones de entrega de registros para el ElastiCache clúster.• Está pendiente una operación de eliminación del ElastiCache nodo.• La contraseña ElastiCache (Redis OSS) se actualiza o rota mediante AWS Secrets Manager

Tipo	Facturado	Descripción
<code>rebooting cache cluster nodes</code>	Facturado	El ElastiCache nodo se está reiniciando debido a una solicitud de un cliente o a un ElastiCache proceso de Amazon que requiere el reinicio del nodo.
<code>incompatible_parameters</code>	No facturado	Amazon no ElastiCache puede iniciar el nodo porque los parámetros especificados en el grupo de parámetros del nodo no son compatibles con el nodo. Revierta los cambios de los parámetros o hágalos compatibles con el nodo para recuperar el acceso a este. Para obtener más información sobre los parámetros incompatibles, consulte la lista de eventos del ElastiCache nodo.

Tipo	Facturado	Descripción
<code>incompatible_network</code>	No facturado	<p>Un estado de red incompatible indica que se dan una o más de las siguientes condiciones en el nodo de Amazon ElastiCache</p> <ul style="list-style-type: none">• No hay direcciones IP disponibles en la subred en la que se lanzó el ElastiCache nodo.• La subred mencionada en el grupo de ElastiCache subredes ya no existe en Amazon Virtual Private Cloud (Amazon VPC).

Tipo	Facturado	Descripción
restore_failed	No facturado	<p>Un estado de restauración fallida indica que se cumple una de las siguientes condiciones en el nodo de Amazon: ElastiCache</p> <ul style="list-style-type: none">• Se produjo un error al reemplazar el nodo debido a un estado repetido de capacidad insuficiente de la instancia. Esto suele ocurrir cuando se ejecutan nodos de generaciones anteriores que sí lo son. end-of-life Sin embargo, también puede ocurrir al reemplazar los nodos de la generación actual cuando AWS no hay suficiente capacidad bajo demanda para cumplir con la solicitud en la zona de disponibilidad especificada. Para obtener más información sobre cómo arreglar o eliminar estos nodos, consulte Migración de nodos de generación anterior.• No se pudo restaurar la instantánea de RDB especificada.• Se suspendió la AWS cuenta del ElastiCache clúster.

Tipo	Facturado	Descripción
		<ul style="list-style-type: none"> Se produjo un error en el nodo y no se pudo recuperar.
snapshotting	Facturado	ElastiCache está creando una instantánea del nodo ElastiCache (Redis OSS).

Visualización ElastiCache del estado del nodo con la consola

Para ver el estado de un ElastiCache nodo con la consola:

1. Inicia sesión en la ElastiCache consola de Amazon AWS Management Console y ábrela en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación, selecciona Redis OSS Clusters o Memcached Clusters. Aparece la página de cachés con la lista de nodos. ElastiCache Para cada nodo, se muestra el valor del estado.
3. A continuación, puede ir a la pestaña Actualizaciones del servicio de la memoria caché para ver la lista de actualizaciones de servicio aplicables a la memoria caché.

Visualización ElastiCache del estado de los nodos con AWS CLI

Para ver ElastiCache el nodo y su información de estado mediante el AWS CLI, utilice el `describe-cache-cluster` comando. Por ejemplo, el siguiente AWS CLI comando muestra cada ElastiCache nodo.

```
aws elasticache describe-cache-clusters
```

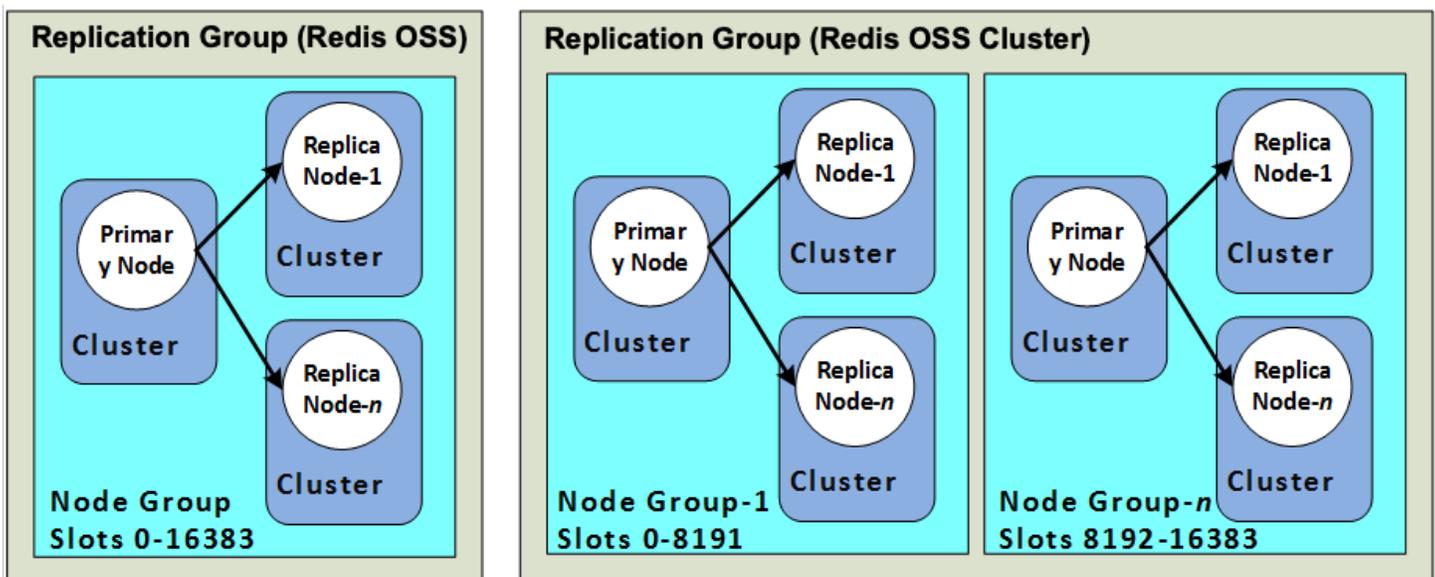
Visualización ElastiCache del estado de los nodos a través de la API

Para ver el estado del ElastiCache nodo mediante la ElastiCache API de Amazon, llama a `DescribeCacheClusteroperation with the ShowCacheNodeInfo flag` para recuperar información sobre los nodos de caché individuales.

Nodos y fragmentos de Redis OSS

Una partición (en la API y la CLI, un grupo de nodos) es un grupo de nodos clasificados de forma jerárquica en un clúster. Las particiones son compatibles con la reproducción. En una partición, un nodo funciona como nodo principal de lectura/escritura. Todos los demás nodos de la partición funcionan como réplicas de solo lectura del nodo principal. La versión 3.2 y posteriores de Redis OSS admiten varios fragmentos dentro de un clúster (en la API y la CLI, un grupo de replicación). Este soporte permite particionar los datos en un clúster de Redis OSS (habilitado para el modo de clúster).

El siguiente diagrama ilustra las diferencias entre un clúster OSS de Redis (modo de clúster desactivado) y un clúster de Redis OSS (modo de clúster habilitado).



Los clústeres OSS de Redis (modo de clúster activado) admiten la replicación mediante fragmentos. La operación API [DescribeReplicationGroups](#) (CLI: [describe-replication-groups](#)) enumera los grupos de nodos con los nodos miembros, la función del nodo dentro del grupo de nodos y también otra información.

Al crear un clúster de OSS de Redis, debe especificar si desea crear un clúster con la agrupación en clústeres habilitada. Los clústeres OSS de Redis (modo de clúster desactivado) nunca tienen más de un fragmento, que se puede escalar horizontalmente añadiendo (hasta un total de cinco) o eliminando nodos de réplica de lectura. Para obtener más información, consulte [Alta disponibilidad a través de grupos de reproducción](#), [Agregar una réplica de lectura, para grupos de replicación de Redis OSS \(modo de clúster deshabilitado\)](#) o [Eliminar una réplica de lectura, para grupos de replicación de Redis OSS \(modo de clúster deshabilitado\)](#). Los clústeres OSS de Redis (modo

de clúster desactivado) también se pueden escalar verticalmente cambiando los tipos de nodos. Para obtener más información, consulte [Escalado de clústeres de Redis OSS \(modo de clúster desactivado\) con nodos de réplica](#).

El límite de nodos o fragmentos se puede aumentar hasta un máximo de 500 por clúster si la versión del motor OSS de Redis es 5.0.6 o superior. Por ejemplo, puede elegir configurar un clúster de 500 nodos que oscila entre 83 particiones (uno primario y 5 réplicas por partición) y 500 particiones (único primario y sin réplicas). Asegúrese de que hay suficientes direcciones IP disponibles para acomodar el aumento. Algunos problemas comunes incluyen que las subredes del grupo de subredes tienen un rango CIDR demasiado pequeño o que otros clústeres comparten y utilizan considerablemente las subredes. Para obtener más información, consulte [Creación de un grupo de subredes](#).

Para las versiones inferiores a 5.0.6, el límite es de 250 por clúster.

Para solicitar un aumento del límite, consulte [AWS Service Limits](#) y elija el tipo de límite Nodes per cluster per instance type (Nodos por clúster por tipo de instancias).

Una vez creado un clúster de Redis OSS (modo de clúster activado), se puede modificar (ampliarlo o reducirlo). Para obtener más información, consulte [Escalado ElastiCache \(Redis OSS\)](#) y [Sustitución de nodos](#).

Cuando cree un clúster nuevo, puede inicializarlo con datos del clúster anterior para que no comience vacío. El enfoque anterior solo funciona si el grupo de clústeres tiene el mismo número de particiones que el clúster anterior. Esto puede resultar útil si necesita cambiar el tipo de nodo o la versión del motor. Para obtener más información, consulte [Copias de seguridad manuales](#) y [Restauración desde una copia de seguridad a una nueva caché](#).

Conexión a los nodos

Antes de intentar conectarse a los nodos del OSS clúster de Redis, debe disponer de los puntos finales de los nodos. Para encontrar los puntos de enlace, consulte lo siguiente:

- [Búsqueda de los puntos finales de un clúster de Redis OSS \(modo de clúster desactivado\) \(consola\)](#)
- [Búsqueda de puntos finales para un clúster de Redis OSS \(modo de clúster activado\) \(consola\)](#)
- [Búsqueda de puntos de conexión \(AWS CLI\)](#)
- [Búsqueda de puntos finales \(\) ElastiCache API](#)

En el siguiente ejemplo, utiliza la utilidad `redis-cli` para conectarse a un clúster que ejecuta Redis OSS

Note

Para obtener más información sobre Redis OSS y los OSS comandos de Redis disponibles, consulte la página web <http://redis.io/commands>.

Para conectarse a un OSS clúster de Redis mediante `redis-cli`

1. Conéctate a tu EC2 instancia de Amazon mediante la utilidad de conexión que prefieras.

Note

Para obtener instrucciones sobre cómo conectarse a una EC2 instancia de Amazon, consulta la [Guía de EC2 introducción de Amazon](#).

2. Para compilar `redis-cli`, descargar e instalar la colección de GNU compiladores (`gcc`). En la línea de comandos de la EC2 instancia, ingresa el siguiente comando y escríbelo y en la línea de confirmación.

```
sudo yum install gcc
```

Aparecerá una salida similar a la siguiente.

```
Loaded plugins: priorities, security, update-motd, upgrade-helper
```

```

Setting up Install Process
Resolving Dependencies
--> Running transaction check

...(output omitted)...

Total download size: 27 M
Installed size: 53 M
Is this ok [y/N]: y
Downloading Packages:
(1/11): binutils-2.22.52.0.1-10.36.amzn1.x86_64.rpm      | 5.2 MB    00:00
(2/11): cpp46-4.6.3-2.67.amzn1.x86_64.rpm             | 4.8 MB    00:00
(3/11): gcc-4.6.3-3.10.amzn1.noarch.rpm                | 2.8 kB    00:00

...(output omitted)...

Complete!

```

3. Descargue y compile la utilidad redis-cli. Esta utilidad se incluye en la distribución del OSS software Redis. En la línea de comandos de la EC2 instancia, escriba los siguientes comandos:

Note

En sistemas Ubuntu, antes de ejecutar make, ejecute `make distclean`.

```

wget https://download.redis.io/releases/redis-7.2.5.tar.gz
tar xvzf redis-7.2.5.tar.gz
cd redis-7.2.5
make distclean      # ubuntu systems only
make

```

4. En la línea de comandos de tu EC2 instancia, escribe el siguiente comando.

```
src/redis-cli -c -h mycachecuster.eaogs8.0001.usw2.cache.amazonaws.com -p 6379
```

Aparece una línea de OSS comandos de Redis similar a la siguiente.

```
redis mycachecuster.eaogs8.0001.usw2.cache.amazonaws.com 6379>
```

5. Pruebe la conexión ejecutando los comandos de RedisOSS.

Ahora está conectado al clúster y puede ejecutar los comandos de RedisOSS. Los siguientes son algunos ejemplos de comandos con sus respuestas de RedisOSS.

```
set a "hello"           // Set key "a" with a string value and no expiration
OK
get a                   // Get value for key "a"
"hello"
get b                   // Get value for key "b" results in miss
(nil)
set b "Good-bye" EX 5   // Set key "b" with a string value and a 5 second expiration
get b
"Good-bye"

                        // wait 5 seconds

get b
(nil)                   // key has expired, nothing returned
quit                    // Exit from redis-cli
```

Para conectarse a nodos o clústeres que tienen el cifrado Secure Sockets Layer (SSL) (activado en tránsito), consulte. [ElastiCache cifrado en tránsito \(\) TLS](#)

Tipos de nodos compatibles

ElastiCache admite los siguientes tipos de nodos. En general, los tipos de la generación actual ofrecen más capacidad de memoria y potencia de computación a menor costo comparados con sus equivalentes de generaciones previas.

Para obtener más información sobre los detalles del rendimiento de cada tipo de nodo, consulte [Tipos de instancia de Amazon EC2](#).

Para obtener información sobre el tamaño de nodo que debe utilizarse, consulte [Elección del tamaño del nodo](#).

Generación actual

Para obtener más información sobre la generación anterior, consulte [Nodos de la generación anterior](#).

Note

Los tipos de instancias con rendimiento de red ampliable utilizan un mecanismo de créditos de E/S de red para superar el ancho de banda de base en función del esfuerzo.

General

Tipo de instancia	Versión mínima admitida de Redis OSS	E/S mejorada (Redis OSS 5.0.6+)	Descarga de TLS (Redis OSS 6.2.5+)	Multiple Banda Banda acción de de E/S base con mejorad (Gbps)(Gbps) (Redis OSS 7.0.4+)		
cache.m7g.large	6.2	N	N	N	0.937	12,5
cache.m7g.xlarge	6.2	Y	Y	Y	1.876	12,5

Tipo de instancia	Versión mínima admitida de Redis OSS	E/S mejorada (Redis OSS 5.0.6+)	Descarga de TLS (Redis OSS 6.2.5+)	Múltiple E/S mejorada (Redis OSS 7.0.4+)	Banda ancha de base (Gbps)	Banda ancha con ráfagas (Gbps)
cache.m7g .2xlarge	6.2	Y	Y	Y	3.75	15
cache.m7g .4xlarge	6.2	Y	Y	Y	7.5	15
cache.m7g .8xlarge	6.2	Y	Y	Y	15	N/A
cache.m7g .12xlarge	6.2	Y	Y	Y	22,5	N/A
cache.m7g .16xlarge	6.2	Y	Y	Y	30	N/A
cache.m6g.large	5.0.6	N	N	N	0.75	10.0
cache.m6g .xlarge	5.0.6	Y	Y	Y	1,25	10.0
cache.m6g .2xlarge	5.0.6	Y	Y	Y	2,5	10.0
cache.m6g .4xlarge	5.0.6	Y	Y	Y	5.0	10.0
cache.m6g .8xlarge	5.0.6	Y	Y	Y	12	N/A

Tipo de instancia	Versión mínima admitida de Redis OSS	E/S mejorada (Redis OSS 5.0.6+)	Descarga de TLS (Redis OSS 6.2.5+)	Múltiple E/S mejorada (Redis OSS 7.0.4+)	Banda ancha de base (Gbps)	Banda ancha con ráfagas (Gbps)
cache.m6g .12xlarge	5.0.6	Y	Y	Y	20	N/A
cache.m6g .16xlarge	5.0.6	Y	Y	Y	25	N/A
cache.m5.large	3.2.4	N	N	N	0.75	10.0
cache.m5.xlarge	3.2.4	Y	N	N	1,25	10.0
cache.m5. 2xlarge	3.2.4	Y	Y	Y	2,5	10.0
cache.m5. 4xlarge	3.2.4	Y	Y	Y	5.0	10.0
cache.m5. 12xlarge	3.2.4	Y	Y	Y	12	N/A
cache.m5. 24xlarge	3.2.4	Y	Y	Y	25	N/A
cache.m4.large	3.2.4	N	N	N	0,45	1.2
cache.m4.xlarge	3.2.4	Y	N	N	0.75	2.8
cache.m4. 2xlarge	3.2.4	Y	Y	Y	1.0	10.0

Tipo de instancia	Versión mínima admitida de Redis OSS	E/S mejorada (Redis OSS 5.0.6+)	Descarga de TLS (Redis OSS 6.2.5+)	Múltiple E/S mejorada (Redis OSS 7.0.4+)	Banda ancha de base (Gbps)	Banda ancha con ráfagas (Gbps)
cache.m4.4xlarge	3.2.4	Y	Y	Y	2.0	10.0
cache.m4.10xlarge	3.2.4	Y	Y	Y	5.0	10.0
cache.t4g.micro	3.2.4	N	N	N	0,064	5.0
cache.t4g.small	5.0.6	N	N	N	0,128	5.0
cache.t4g.medium	5.0.6	N	N	N	0,256	5.0
cache.t3.micro	3.2.4	N	N	N	0,064	5.0
cache.t3.small	3.2.4	N	N	N	0,128	5.0
cache.t3.medium	3.2.4	N	N	N	0,256	5.0
cache.t2.micro	3.2.4	N	N	N	0,064	1.024
cache.t2.small	3.2.4	N	N	N	0,128	1.024
cache.t2.medium	3.2.4	N	N	N	0,256	1.024

Optimizada para memoria

Tipo de instancia	Versión mínima admitida de Redis OSS	E/S mejorada (Redis OSS 5.0.6+)	Descarga de TLS (Redis OSS 6.2.5+)	Múltiple Banda de E/S mejorada (Redis OSS 7.0.4+)	Banda ancha de base (Gbps)	Banda ancha con ráfagas (Gbps)
cache.r7g.large	6.2	N	N	N	0.937	12,5
cache.r7g.xlarge	6.2	Y	Y	Y	1.876	12,5
cache.r7g.2xlarge	6.2	Y	Y	Y	3.75	15
cache.r7g.4xlarge	6.2	Y	Y	Y	7.5	15
cache.r7g.8xlarge	6.2	Y	Y	Y	15	N/A
cache.r7g.12xlarge	6.2	Y	Y	Y	22,5	N/A
cache.r7g.16xlarge	6.2	Y	Y	Y	30	N/A
cache.r6g.large	5.0.6	N	N	N	0.75	10.0
cache.r6g.xlarge	5.0.6	Y	Y	Y	1,25	10.0
cache.r6g.2xlarge	5.0.6	Y	Y	Y	2,5	10.0
cache.r6g.4xlarge	5.0.6	Y	Y	Y	5.0	10.0

Tipo de instancia	Versión mínima admitida de Redis OSS	E/S mejorada (Redis OSS 5.0.6+)	Descarga de TLS (Redis OSS 6.2.5+)	Múltiple E/S mejorada (Redis OSS 7.0.4+)	Banda ancha de base (Gbps)	Banda ancha con ráfagas (Gbps)
cache.r6g.8xlarge	5.0.6	Y	Y	Y	12	N/A
cache.r6g.12xlarge	5.0.6	Y	Y	Y	20	N/A
cache.r6g.16xlarge	5.0.6	Y	Y	Y	25	N/A
cache.r5.large	3.2.4	N	N	N	0.75	10.0
cache.r5.xlarge	3.2.4	Y	N	N	1,25	10.0
cache.r5.2xlarge	3.2.4	Y	Y	Y	2,5	10.0
cache.r5.4xlarge	3.2.4	Y	Y	Y	5.0	10.0
cache.r5.12xlarge	3.2.4	Y	Y	Y	12	N/A
cache.r5.24xlarge	3.2.4	Y	Y	Y	25	N/A
cache.r4.large	3.2.4	N	N	N	0.75	10.0
cache.r4.xlarge	3.2.4	Y	N	N	1,25	10.0
cache.r4.2xlarge	3.2.4	Y	Y	Y	2,5	10.0
cache.r4.4xlarge	3.2.4	Y	Y	Y	5.0	10.0

Tipo de instancia	Versión mínima admitida de Redis OSS	E/S mejorada (Redis OSS 5.0.6+)	Descarga de TLS (Redis OSS 6.2.5+)	Múltiple acción de E/S mejorada (Redis OSS 7.0.4+)	Banda ancha de base (Gbps)	Banda ancha con ráfagas (Gbps)
cache.r4.8xlarge	3.2.4	Y	Y	Y	12	N/A
cache.r4.16xlarge	3.2.4	Y	Y	Y	25	N/A

Memoria optimizada con la organización de datos en niveles

Tipo de instancia	Versión mínima admitida de Redis OSS	E/S mejorada (Redis OSS 5.0.6+)	Descarga de TLS (Redis OSS 6.2.5+)	Múltiple acción de E/S mejorada (Redis OSS 7.0.4+)	Banda ancha de base (Gbps)	Banda ancha con ráfagas (Gbps)
cache.r6gd.xlarge	6.2.0	Y	N	N	1,25	10
cache.r6gd.2xlarge	6.2.0	Y	Y	Y	2,5	10
cache.r6gd.4xlarge	6.2.0	Y	Y	Y	5.0	10
cache.r6gd.8xlarge	6.2.0	Y	Y	Y	12	N/A

Tipo de instancia	Versión mínima admitida de Redis OSS	E/S mejorada (Redis OSS 5.0.6+)	Descarga de TLS (Redis OSS 6.2.5+)	Múltiple acción de E/S mejorada (Redis OSS 7.0.4+)	Banda ancha de base (Gbps)	Banda ancha con ráfagas (Gbps)
cache.r6gd.12xlarge	6.2.0	Y	Y	Y	20	N/A
cache.r6gd.16xlarge	6.2.0	Y	Y	Y	25	N/A

Optimizado para red

Tipo de instancia	Versión mínima admitida de Redis OSS	E/S mejorada (Redis OSS 5.0.6+)	Descarga de TLS (Redis OSS 6.2.5+)	Múltiple acción de E/S mejorada (Redis OSS 7.0.4+)	Banda ancha de base (Gbps)	Banda ancha con ráfagas (Gbps)
cache.c7gn.large	6.2	N	N	N	6.25	30
cache.c7gn.xlarge	6.2	Y	Y	Y	12,5	40
cache.c7gn.2xlarge	6.2	Y	Y	Y	25	50
cache.c7gn.4xlarge	6.2	Y	Y	Y	50	N/A

Tipo de instancia	Versión mínima admitida de Redis OSS	E/S mejorada (Redis OSS 5.0.6+)	Descarga de TLS (Redis OSS 6.2.5+)	Múltiple E/S mejorada (Redis OSS 7.0.4+)	Banda ancha de base (Gbps)	Banda ancha con ráfagas (Gbps)
cache.c7g n.8xlarge	6.2	Y	Y	Y	100	N/A
cache.c7g n.12xlarge	6.2	Y	Y	Y	150	N/A
cache.c7g n.16xlarge	6.2	Y	Y	Y	200	N/A

Tipos de nodos admitidos por región de AWS

Los tipos de nodos compatibles pueden variar de una AWS región a otra. Para obtener más información, consulta los [ElastiCacheprecios de Amazon](#).

Instancias de rendimiento ampliable

Puede lanzar nodos de caché T4g, T3-Standard y T2-Standard en ráfagas de uso general en Amazon. ElastiCache Estos nodos proporcionan un nivel básico de rendimiento de la CPU con la capacidad de explotar el uso de la CPU en cualquier momento hasta que se agoten los créditos acumulados. Un crédito de CPU proporciona el rendimiento de un núcleo de CPU completo durante un minuto.

Los nodos T4g, T3 y T2 de Amazon ElastiCache están configurados de forma estándar y son adecuados para cargas de trabajo con un uso medio de la CPU que está constantemente por debajo del rendimiento básico de la instancia. Para ampliar por encima de la referencia, la instancia gasta créditos que ha acumulado en su saldo de créditos de CPU. Si el nodo se queda sin créditos acumulados, el rendimiento se reduce gradualmente hasta el nivel de rendimiento de referencia. Esta disminución gradual garantiza que el nodo no experimente una caída brusca del rendimiento cuando se agote el saldo de créditos de CPU acumulado. Para obtener más información, consulte

[Créditos de CPU y rendimiento de referencia para las instancias de rendimiento ampliable](#) en la Guía del usuario de Amazon EC2.

En la siguiente tabla, se enumeran los tipos de nodos de rendimiento ampliables y la velocidad a la que se obtienen los créditos de CPU por hora. También se muestra el número máximo de créditos de CPU obtenidos que un nodo puede acumular y el número de vCPU por nodo. Además, se proporciona el nivel de rendimiento de referencia como porcentaje de un rendimiento básico completo (mediante una única vCPU).

Créditos de CPU obtenidos por hora	Créditos máximos ganados que se pueden acumular*	vCPU	Rendimiento de referencia por vCPU	Memoria (GiB)	Rendimiento de la red
12	288	2	10%	0,5	Hasta 5 gigabits
24	576	2	20%	1,37	Hasta 5 gigabits
24	576	2	20%	3,09	Hasta 5 gigabits
12	288	2	10%	0,5	Hasta 5 gigabits
24	576	2	20%	1,37	Hasta 5 gigabits
24	576	2	20%	3,09	Hasta 5 gigabits
6	144	1	10%	0,5	Baja a moderada
12	288	1	20%	1.55	Baja a moderada

Créditos de CPU obtenidos por hora	Créditos máximos ganados que se pueden acumular*	vCPU	Rendimiento de referencia por vCPU	Memoria (GiB)	Rendimiento de la red
24	576	2	20%	3.22	Baja a moderada

* El número de créditos que se pueden acumular es equivalente a la cantidad de créditos que se pueden obtener en un periodo de 24 horas.

** El rendimiento de referencia de la tabla es por vCPU. Algunos tamaños de nodos que tienen más de una vCPU. Para estos, calcule la utilización de CPU de referencia correspondiente al nodo con la multiplicación del porcentaje de vCPU por el número de vCPU.

Las siguientes métricas de crédito de CPU se encuentran disponibles para las instancias de rendimiento ampliable T3 y T4g:

 Note

Estas métricas no se encuentran disponibles para las instancias de rendimiento ampliable T2.

- CPUCreditUsage
- CPUCreditBalance

Para obtener más información sobre estas métricas, consulte [Métricas de crédito de CPU](#).

Además, tenga en cuenta estos detalles:

- Todos los tipos de nodos de generación actuales se crean en una Virtual Private Cloud (VPC) basada en Amazon VPC de forma predeterminada.

- Los archivos de solo datos adjuntos (AOF) de Redis OSS no son compatibles con las instancias T2. Son variables de configuración de Redis OSS `appendonly` y `appendfsync` no son compatibles con la versión 2.8.22 y posteriores de Redis OSS.

Información relacionada

- [Características y detalles de los ElastiCache productos de Amazon](#)
- [Parámetros específicos de Redis OSS](#)
- [Cifrado en tránsito \(TLS\)](#)

Reinicio de nodos (solo modo de clúster deshabilitado)

Algunos cambios requieren que los nodos del clúster se reinicien para que se apliquen los cambios. Por ejemplo, en el caso de algunos parámetros, el cambio del valor del parámetro en un grupo de parámetros solo se aplica después de reiniciar.

Para los clústeres de Redis OSS (modo de clúster desactivado), esos parámetros son:

- `activeresharding`
- bases de datos

Puede reiniciar un nodo utilizando únicamente la ElastiCache consola. Solo puede reiniciar un solo nodo a la vez. Si desea reiniciar varios nodos, debe repetir el proceso para cada uno de ellos.

Cambios en los parámetros del OSS (modo de clúster activado) de Redis

Si realiza cambios en los siguientes parámetros en un clúster de Redis OSS (modo de clúster activado), siga los pasos siguientes.

- `activeresharding`
- bases de datos

1. Cree una copia de seguridad manual de su clúster. Consulte [Copias de seguridad manuales](#).
2. Elimine el clúster de Redis OSS (modo de clúster activado). Consulte [Eliminación de un clúster](#).

3. Restaure el clúster con la copia de seguridad y el grupo de parámetros modificados para propagar los datos en el nuevo clúster. Consulte [Restauración desde una copia de seguridad a una nueva caché](#).

Los cambios en otros parámetros no requieren esta acción.

Uso del AWS Management Console

Puede reiniciar un nodo mediante la ElastiCache consola.

Para reiniciar un nodo (consola)

1. Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en <https://console.aws.amazon.com/elasticache/>.
2. En la lista de la esquina superior derecha, selecciona la AWS región que corresponda.
3. En el panel de navegación izquierdo, elija Redis OSS.

Aparece una lista de clústeres que ejecutan Redis OSS.

4. Elija el clúster en Cluster Name (Nombre de clúster).
5. En Node name (Nombre de nodo), elija el botón de opción situado junto al nodo que desea reiniciar.
6. Seleccione Actions (Acciones) y, a continuación, elija Reboot node (Reiniciar nodo).

Para reiniciar varios nodos, repita los pasos del 2 al 5 por cada nodo que desee reiniciar. No es necesario esperar a que un nodo termine de reiniciarse para reiniciar otro.

Sustitución de nodos

Amazon ElastiCache (Redis OSS) actualiza su flota con frecuencia con parches y actualizaciones que se aplican a las instancias sin problemas. Sin embargo, de vez en cuando necesitamos volver a lanzar sus nodos ElastiCache (Redis OSS) para aplicar las actualizaciones obligatorias del sistema operativo al host subyacente. Estas sustituciones son necesarias para aplicar actualizaciones que refuerzan la seguridad, la fiabilidad y el rendimiento operativo.

Puede optar por administrar personalmente estas sustituciones en cualquier momento antes del periodo programado para la sustitución de nodos. Cuando administre personalmente una sustitución, la instancia recibirá la actualización del sistema operativo cuando vuelva a lanzar el nodo y se cancelará la sustitución de nodos programada. Es posible que reciba alertas que indiquen que va a tener lugar la sustitución de nodos. Si ya ha mitigado manualmente la necesidad de mantenimiento, puede hacer caso omiso de estas alertas.

Note

Los nodos de caché de reemplazo generados automáticamente por Amazon ElastiCache pueden tener direcciones IP diferentes. Usted es responsable de revisar la configuración de la aplicación para asegurarse de que los nodos de caché estén asociados con las direcciones IP apropiadas.

La siguiente lista identifica las acciones que puede realizar al programar ElastiCache la sustitución de uno de sus nodos OSS de Redis. Para agilizar la búsqueda de la información que necesita para su situación, elija una opción del menú siguiente.

- [Do nothing](#)— Deje que Amazon ElastiCache reemplace el nodo según lo programado.
- [Change your maintenance window](#): cambie el periodo de mantenimiento a un mejor momento.
- Configuraciones de Redis OSS (modo de clúster habilitado)
 - [Replace the only node in any Redis OSS cluster](#)— Un procedimiento para reemplazar un nodo de un clúster de Redis OSS mediante copias de seguridad y restauración.
 - [Replace a replica node in any Redis OSS cluster](#)— Un procedimiento para reemplazar una réplica de lectura en cualquier clúster de OSS de Redis aumentando y disminuyendo el número de réplicas sin tiempo de inactividad del clúster.

- [Replace any node in a Redis OSS \(cluster mode enabled\) shard](#)— Un procedimiento dinámico sin tiempo de inactividad del clúster para reemplazar un nodo de un clúster de Redis OSS (habilitado en modo de clúster) mediante la ampliación horizontal y horizontal.
- Configuraciones de Redis OSS (modo de clúster desactivado)
 - [Replace the only node in any Redis OSS cluster](#)— Procedimiento para reemplazar cualquier nodo de un clúster de Redis OSS mediante copias de seguridad y restauración.
 - [Replace a replica node in any Redis OSS cluster](#)— Procedimiento para reemplazar una réplica de lectura en cualquier clúster de OSS de Redis aumentando y disminuyendo el número de réplicas sin tiempo de inactividad del clúster.
 - [Replace a node in a Redis OSS \(cluster mode disabled\) cluster](#)— Procedimiento para reemplazar un nodo de un clúster de Redis OSS (modo de clúster desactivado) mediante la replicación.
 - [Replace a Redis OSS \(cluster mode disabled\) read-replica](#)— Procedimiento para reemplazar manualmente una réplica de lectura en un grupo de replicación de Redis OSS (modo de clúster desactivado).
 - [Replace a Redis OSS \(cluster mode disabled\) primary node](#)— Un procedimiento para reemplazar manualmente el nodo principal de un grupo de replicación de Redis OSS (modo de clúster desactivado).

Opciones de reemplazo de nodos de Redis OSS

- No hacer nada: si no hace nada, ElastiCache reemplaza el nodo según lo programado.

En el caso de las configuraciones que no son de clúster y que tienen habilitada la conmutación automática por error, los clústeres de Redis OSS 5.0.6 y versiones posteriores se sustituyen por completo mientras el clúster permanece en línea y atiende las solicitudes de escritura entrantes. En el caso de los clústeres habilitados para la conmutación automática por error en Redis OSS 4.0.10 o versiones anteriores, es posible que observe una breve interrupción de escritura de hasta unos segundos asociada a las actualizaciones de DNS.

Si el nodo es miembro de un clúster con conmutación por error automática, ElastiCache (Redis OSS) proporciona una disponibilidad mejorada durante la aplicación de parches, las actualizaciones y otros reemplazos de nodos relacionados con el mantenimiento.

En el caso de las configuraciones de clúster ElastiCache (Redis OSS) que están configuradas para usar clientes de clúster ElastiCache (Redis OSS), la sustitución ahora se completa mientras el clúster atiende las solicitudes de escritura entrantes.

En el caso de las configuraciones que no son de clúster y que tienen habilitada la conmutación automática por error, los clústeres de Redis OSS 5.0.6 y versiones posteriores se sustituyen por completo mientras el clúster permanece en línea y atiende las solicitudes de escritura entrantes. En el caso de los clústeres habilitados para la conmutación automática por error en Redis OSS 4.0.10 o versiones anteriores, es posible que observe una breve interrupción de escritura de hasta unos segundos asociada a las actualizaciones de DNS.

Si el nodo es independiente, Amazon lanza ElastiCache primero un nodo de reemplazo y, a continuación, se sincroniza desde el nodo existente. El nodo existente no estará disponible para las solicitudes de servicio durante este periodo. Una vez finalizada la sincronización, el nodo existente finaliza y el nuevo nodo ocupa su lugar. ElastiCache hace todo lo posible por conservar los datos durante esta operación.

- Cambie el período de mantenimiento: en el caso de los eventos de mantenimiento programados, recibirá un correo electrónico o una notificación de un evento de su parte ElastiCache. En estos casos, si cambia el periodo de mantenimiento antes de la hora de sustitución programada, su nodo se sustituirá en ese momento a la nueva hora. Para más información, consulte los siguientes temas:
 - [Modificación de un ElastiCache clúster](#)
 - [Modificación de un grupo de reproducción](#)

 Note

La posibilidad de cambiar la ventana de reemplazo moviendo la ventana de mantenimiento solo está disponible cuando la ElastiCache notificación incluye una ventana de

mantenimiento. Si la notificación no incluye un periodo de mantenimiento, no se puede cambiar el periodo de sustitución.

Por ejemplo, supongamos que es jueves 9 de noviembre a las 15:00 h y el próximo periodo de mantenimiento es el viernes 10 de noviembre a las 17:00 h. A continuación, se exponen tres escenarios con sus resultados:

- Cambia el periodo de mantenimiento a los viernes a las 16:00, después de la fecha y hora actuales y antes del siguiente periodo de mantenimiento programado. El nodo se sustituye el viernes 10 de noviembre a las 16:00.
- Cambia el periodo de mantenimiento al sábado a las 16:00, después de la fecha y hora actuales y después del siguiente periodo de mantenimiento programado. El nodo se sustituye el sábado 11 de noviembre a las 16:00.
- Cambia el periodo de mantenimiento al miércoles a las 16:00 (un día anterior de la misma semana que la fecha y hora actuales). El nodo se sustituye el próximo miércoles 15 de noviembre a las 16:00.

Para ver instrucciones, consulte [Administración del mantenimiento](#).

- Sustituya el único nodo de cualquier clúster de OSS de Redis: si el clúster no tiene réplicas de lectura, puede utilizar el siguiente procedimiento para reemplazar el nodo.

Para sustituir el único nodo mediante operaciones de copia de seguridad y restauración

1. Cree una instantánea del clúster del nodo. Para ver instrucciones, consulte [Copias de seguridad manuales](#).
2. Cree un nuevo clúster mediante la propagación de los datos a partir de la instantánea. Para ver instrucciones, consulte [Restauración desde una copia de seguridad a una nueva caché](#).
3. Elimine el clúster con el nodo programado para su sustitución. Para ver instrucciones, consulte [Eliminación de un clúster](#).
4. En su aplicación, sustituya el punto de enlace del nodo anterior por el punto de enlace del nuevo nodo.

- Reemplace un nodo de réplica en cualquier clúster de OSS de Redis: para reemplazar un clúster de réplicas, aumente el número de réplicas. Para hacerlo, agregue una réplica y, a continuación, reduzca el número de réplicas eliminando la réplica que desee reemplazar. Este proceso es dinámico y no tiene asociado ningún tiempo de inactividad del clúster.

 Note

Si la partición o el grupo de reproducción ya tiene cinco réplicas, invierta los pasos 1 y 2.

Para reemplazar una réplica en cualquier clúster de OSS de Redis

1. Aumente el número de réplicas agregando una réplica al fragmento o grupo de reproducción. Para obtener más información, consulte [Aumento del número de réplicas de una partición](#).
 2. Elimine la réplica que desea sustituir. Para obtener más información, consulte [Reducción del número de réplicas de una partición](#).
 3. Actualice los puntos de enlace de la aplicación.
- Reemplace cualquier nodo de un fragmento OSS (modo de clúster habilitado) de Redis: para reemplazar el nodo de un clúster sin tiempo de inactividad, utilice la refragmentación en línea. Primero agregue un fragmento mediante el escalado ascendente y, a continuación, elimine el fragmento con el nodo que se reemplazará mediante el escalado descendente.

Para reemplazar cualquier nodo de un clúster de Redis OSS (modo de clúster habilitado)

1. Escalado ascendente: agregue un fragmento adicional con la misma configuración que el existente con el nodo que se reemplazará. Para obtener más información, consulte [Adición de particiones con los cambios de particiones en línea](#).
2. Reducción horizontal: elimine la partición con el nodo que se sustituirá. Para obtener más información, consulte [Eliminación de particiones con los cambios de particiones en línea](#).
3. Actualice los puntos de enlace de la aplicación.

- Reemplazar un nodo en un clúster de Redis OSS (modo de clúster desactivado): si el clúster es un clúster de Redis OSS (modo de clúster desactivado) sin réplicas de lectura, utilice el siguiente procedimiento para reemplazar el nodo.

Para sustituir el nodo mediante la reproducción (solo modo de clúster desactivado), realice el siguiente procedimiento:

1. Agregue la reproducción al clúster con el nodo programado para reemplazo como nodo primario. No habilite Multi-AZ en este clúster. Para ver instrucciones, consulte [Para agregar la replicación a un clúster de Redis OSS sin fragmentos](#).
 2. Agregue una réplica de lectura al clúster. Para ver instrucciones, consulte [Para agregar nodos a un clúster \(consola\)](#).
 3. Promocione la réplica de lectura recién creada a nodo principal. Para ver instrucciones, consulte [Convertir una réplica de lectura en principal, para grupos de replicación de Redis OSS \(modo de clúster desactivado\)](#).
 4. Elimine el nodo programado para su sustitución. Para ver instrucciones, consulte [Eliminación de nodos de un clúster](#).
 5. En su aplicación, sustituya el punto de enlace del nodo anterior por el punto de enlace del nuevo nodo.
- Sustituya una réplica de lectura de Redis OSS (modo de clúster desactivado): si el nodo es una réplica de lectura, sustituya el nodo.

Si su clúster tiene solo un nodo de réplica y están habilitadas Multi-AZ, debe deshabilitarlas para poder eliminar la réplica. Para ver instrucciones, consulte [Modificación de un grupo de reproducción](#).

Para reemplazar un OSS de Redis (modo de clúster desactivado), lea una réplica

1. Elimine la réplica programada para la sustitución. Para obtener instrucciones, consulte lo siguiente:
 - [Reducción del número de réplicas de una partición](#)
 - [Eliminación de nodos de un clúster](#)

2. Agregue una nueva réplica para reemplazar la programada para el reemplazo. Si utiliza el mismo nombre que el de la réplica que acaba de eliminar, puede pasar al paso 3. Para obtener instrucciones, consulte lo siguiente:
 - [Aumento del número de réplicas de una partición](#)
 - [Agregar una réplica de lectura, para grupos de replicación de Redis OSS \(modo de clúster deshabilitado\)](#)
 3. En su aplicación, sustituya el punto de enlace de la réplica anterior por el punto de enlace de la nueva réplica.
 4. Si deshabilitó las Multi-AZ al inicio, vuelva a habilitarlas ahora. Para ver instrucciones, consulte [Habilitación de Multi-AZ](#).
- Reemplace un nodo principal de Redis OSS (modo de clúster desactivado): si el nodo es el nodo principal, primero convierta una réplica de lectura en principal. A continuación, elimine la réplica que era el nodo primario.

Si su clúster tiene solo una réplica y están habilitadas Multi-AZ, debe deshabilitarlas para poder eliminar la réplica en el paso 2. Para ver instrucciones, consulte [Modificación de un grupo de reproducción](#).

Para reemplazar un nodo principal de Redis OSS (modo de clúster desactivado)

1. Promocione una réplica de lectura a primaria. Para ver instrucciones, consulte [Convertir una réplica de lectura en principal, para grupos de replicación de Redis OSS \(modo de clúster desactivado\)](#).
2. Elimine el nodo cuya sustitución está programada (antiguo nodo primario). Para ver instrucciones, consulte [Eliminación de nodos de un clúster](#).
3. Agregue una nueva réplica para reemplazar la programada para el reemplazo. Si utiliza el mismo nombre que el del nodo que acaba de eliminar, puede omitir el cambio de puntos de enlace en la aplicación.

Para ver instrucciones, consulte [Agregar una réplica de lectura, para grupos de replicación de Redis OSS \(modo de clúster deshabilitado\)](#).

4. En su aplicación, sustituya el punto de enlace del nodo anterior por el punto de enlace del nuevo nodo.

5. Si deshabilitó las Multi-AZ al inicio, vuelva a habilitarlas ahora. Para obtener instrucciones, consulte [Habilitación de Multi-AZ](#) .

Nodos reservados de ElastiCache

Reservar uno o varios nodos puede ser una forma de reducir costos. Los nodos reservados se cobran por adelantado en función de una tarifa que depende del tipo de nodo y la duración de la reserva: uno o tres años.

Para ver si los nodos reservados suponen un ahorro de costos para sus casos de uso, determine primero el tamaño del nodo y la cantidad de nodos que necesita. A continuación, calcule la utilización del nodo y compare el costo total que supondría utilizar nodos en diferido en relación con los nodos reservados. Puede mezclar y emparejar nodos reservados con nodos en diferido en sus clústeres. Para obtener información sobre los precios, consulte [Precios de Amazon ElastiCache](#).

Note

Los nodos reservados no son flexibles, solo se aplican al tipo de instancia exacto que se reserva.

Administración de costos con los nodos reservados

Reservar uno o varios nodos puede ser una forma de reducir costos. Los nodos reservados se cobran por adelantado en función de una tarifa que depende del tipo de nodo y la duración de la reserva: uno o tres años. Dicho cargo es mucho menor que el cargo por uso por hora que se aplica con los nodos bajo demanda.

Para ver si los nodos reservados suponen un ahorro de costos para sus casos de uso, determine primero el tamaño del nodo y la cantidad de nodos que necesita. A continuación, calcule el uso del nodo y compare el costo total que supondría utilizar nodos bajo demanda en relación con los nodos reservados. Puede mezclar y emparejar nodos reservados con nodos en diferido en sus clústeres. Para obtener información sobre los precios, consulte [Precios de Amazon ElastiCache](#).

La región de AWS, el tipo de nodo y la longitud del término se deben elegir en el momento de la compra y no se pueden cambiar más adelante.

Puede utilizar la AWS Management Console, la AWS CLI o la API de ElastiCache para mostrar y adquirir ofertas de nodos reservados disponibles.

Para obtener más información sobre los nodos reservados, consulte [Nodos reservados de Amazon ElastiCache](#).

Temas

- [Ofertas de nodos reservados estándar](#)
- [Ofertas de nodos reservados heredados](#)
- [Obtención de información sobre las ofertas de nodos reservados](#)
- [Compra de un nodo reservado](#)
- [Obtención de información sobre los nodos reservados](#)

Ofertas de nodos reservados estándar

Cuando compra una instancia de nodo reservada (RI) estándar en Amazon ElastiCache, adquiere un compromiso para obtener una tarifa con descuento en un tipo de instancias de nodo específico y una región de AWS por la duración de la instancia de nodo reservada. Para utilizar una instancia reservada de nodo de Amazon ElastiCache, debe crear una instancia de nodo de ElastiCache nueva, al igual que lo haría para una instancia en diferido.

La instancia de nodo nueva que cree deberá tener exactamente las mismas especificaciones que la instancia de nodo reservada. Si las especificaciones de la instancia de nodo nueva coinciden con una instancia reservada de nodo existente de su cuenta, se facturará con la tarifa con descuento ofrecida para la instancia reservada de nodo. De lo contrario, la instancia de nodo se factura con una tarifa en diferido. Estas RI estándar están disponibles a partir de las familias de instancias R5 y M5 en adelante.

Note

Los tres tipos de ofertas que se analizan a continuación están disponibles en plazos de un año y de tres años.

Tipos de ofertas

Una RI sin pago inicial proporciona acceso a una instancia reservada de ElastiCache sin que haya que hacer un pago inicial. La instancia reservada de ElastiCache sin pago inicial se factura con una tarifa por hora con descuento por cada hora dentro del término, independientemente de si se utiliza o no.

Una RI con pago inicial parcial requiere que parte de la instancia reservada de ElastiCache se pague por adelantado. Las horas restantes del término se cobran a una tarifa por hora con descuento,

independientemente de la utilización que haga. Esta opción sustituye a la opción heredada utilización intensa, que se explica en la sección siguiente.

La RI de Pago total anticipado requiere que el pago completo se realice al comenzar el término de RI. No se aplicarán otros costos durante el resto del término, independientemente del número de horas utilizadas.

Ofertas de nodos reservados heredados

Existen tres niveles de reservas de nodo heredado: uso intenso, uso medio y uso bajo. Los nodos pueden reservarse en cualquier nivel de uso durante uno o tres años. El tipo de nodo, el nivel de uso y el término de la reserva afectarán a los costos totales. Verifique el ahorro que los nodos reservados pueden proporcionar a la empresa al comparar varios modelos antes de adquirir los nodos reservados.

Los nodos que se compran con un término o nivel de utilización determinado no pueden cambiarse a un término o nivel de utilización distinto.

Niveles de utilización

Los nodos reservados de utilización intensa permiten cargas de trabajo que tienen una base constante de capacidad, así como la ejecución de cargas de trabajo estables. Los nodos reservados de utilización intensa requieren un elevado nivel de compromiso por anticipado; sin embargo, si planea mantener el nodo en ejecución más del 79 por ciento del término de la reserva del nodo, podrá conseguir un mayor ahorro (hasta un 70 por ciento en comparación con el precio de los nodos en diferido). Con los nodos reservados de uso intenso, deberá realizar un pago por única vez. A continuación, seguirá una tarifa por hora más baja mientras dure el término, independientemente de si el nodo se encuentre en funcionamiento.

Los nodos reservados de uso medio son la mejor opción si tiene previsto aprovechar los nodos reservados durante una buena parte del tiempo, pero desea disponer de una tarifa de pago por única vez menor o de la flexibilidad para poder dejar de pagar por el nodo al apagarlo. Los nodos reservados de utilización media son una opción más rentable cuando tiene planeado mantener en ejecución los nodos reservados más del 40 por ciento del término que dura la reserva. Esta opción puede ahorrar hasta un 64 por ciento con respecto al precio en diferido. Con los nodos reservados de utilización media, el pago por única vez será ligeramente superior que con los nodos reservados de utilización baja y disfrutará de unas tarifas de uso por hora más reducidas al ejecutar un nodo.

Los nodos reservados de utilización baja son la opción perfecta para cargas de trabajo periódicas que solo se ejecutan durante un par de horas al día o durante unos días a la semana. Con los nodos

reservados de utilización baja, deberá realizar un pago por única vez seguido del pago de una tarifa de utilización por horas con descuento cuando el nodo se ejecute. Puede comenzar a ahorrar cuando el nodo se ejecute durante más del 17 % del término reservado del nodo. Puede ahorrar hasta un 56 % de las tarifas bajo demanda durante todo el término del nodo reservado.

Ofertas de nodos reservados heredados

Ofertas	Costo inicial	Tarifa de uso	Beneficio
Utilización intensa	Máxima	Tasa por hora más baja. Se aplica durante todo el término independientemente de si utiliza el nodo reservado o no.	Menor costo total si planea ejecutar los nodos reservados más del 79 por ciento del término de tres años.
Utilización media	Media	Tasa de utilización por hora aplicada por cada hora de ejecución del nodo. No se aplica ningún cargo por hora cuando el nodo no está en ejecución.	Adecuado para cargas de trabajo elásticas o para cuando se espera una utilización moderada, durante más del 40 por ciento del término de tres años.
Utilización baja	Mínima	Tasa de uso por hora aplicada por cada hora de ejecución del nodo. No se aplica ningún cargo por hora cuando el nodo no está en ejecución. Tasa por hora más alta de todos los tipos de	El costo total más elevado si planea ejecutar todo el tiempo. Sin embargo, ofrece el mínimo costo total si prevé que utilizará el nodo reservado con poca frecuencia, más del

Ofertas	Costo inicial	Tarifa de uso	Beneficio
		ofertas; sin embargo, las tasas solo se aplican cuando el nodo reservado está en ejecución.	15 % de un término de tres años.
Utilización en diferido (sin nodos reservados)	Ninguno	Tasa por hora más alta. Se aplica cuando el nodo está en ejecución.	Costo por hora más alto.

Para obtener más información, consulte [Precios de Amazon ElastiCache](#).

Obtención de información sobre las ofertas de nodos reservados

Antes de comprar nodos reservados, puede obtener información acerca de las ofertas de nodos reservados.

Los siguientes ejemplos muestran cómo obtener datos de precios e información sobre las ofertas disponibles de nodos reservados mediante la AWS Management Console, la AWS CLI y la API de ElastiCache.

Temas

- [Obtención de información sobre las ofertas de nodos reservados \(consola\)](#)
- [Obtención de información sobre las ofertas de nodos reservados \(AWS CLI\)](#)
- [Obtención de información sobre las ofertas de nodos reservados \(API de ElastiCache\)](#)

Obtención de información sobre las ofertas de nodos reservados (consola)

Para obtener datos de precios e información sobre las ofertas disponibles de clústeres reservados mediante la AWS Management Console, utilice el siguiente procedimiento.

Para obtener información acerca de las ofertas de nodos reservados disponibles

1. Inicie sesión en la AWS Management Console y abra la consola de ElastiCache en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación, seleccione Reserved Nodes (Nodos reservados).
3. Elija Purchase Reserved Node (Comprar nodo reservado).
4. En Engine (Motor), elija Redis.
5. Para determinar las ofertas disponibles, seleccione las siguientes opciones:
 - Node Type
 - Term (Término)
 - Offering Type(Tipo de oferta)

Después de realizar estas selecciones, el costo por nodo y el costo total de las selecciones se mostrarán en Reservation details (Detalles de reserva).

6. Elija Cancel (Cancelar) para evitar comprar estos nodos e incurrir en gastos.

Obtención de información sobre las ofertas de nodos reservados (AWS CLI)

Para obtener información acerca de precios y otra información disponible acerca de las ofertas de nodos reservados, escriba el comando siguiente en el símbolo del sistema:

```
aws elasticache describe-reserved-cache-nodes-offerings
```

Esta operación produce un resultado similar al siguiente (formato JSON):

```
{
  "ReservedCacheNodesOfferingId": "0xxxxxxxx-xxeb-44ex-xx3c-xxxxxxxx072",
  "CacheNodeType": "cache.xxx.large",
  "Duration": 94608000,
  "FixedPrice": XXXX.X,
  "UsagePrice": X.X,
  "ProductDescription": "redis",
  "OfferingType": "All Upfront",
  "RecurringCharges": [
    {
      "RecurringChargeAmount": X.X,
      "RecurringChargeFrequency": "Hourly"
    }
  ]
},
{
  "ReservedCacheNodesOfferingId": "0xxxxxxxx-xxeb-44ex-xx3c-xxxxxxxx072",
  "CacheNodeType": "cache.xxx.xlarge",
  "Duration": 94608000,
  "FixedPrice": XXXX.X,
  "UsagePrice": X.X,
  "ProductDescription": "redis",
  "OfferingType": "Partial Upfront",
  "RecurringCharges": [
    {
      "RecurringChargeAmount": X.XXX,
      "RecurringChargeFrequency": "Hourly"
    }
  ]
},
{
  "ReservedCacheNodesOfferingId": "0xxxxxxxx-xxeb-44ex-xx3c-xxxxxxxx072",
  "CacheNodeType": "cache.xxx.large",
  "Duration": 31536000,
```

```
    "FixedPrice": X.X,  
    "UsagePrice": X.X,  
    "ProductDescription": "redis",  
    "OfferingType": "No Upfront",  
    "RecurringCharges": [  
      {  
        "RecurringChargeAmount": X.XXX,  
        "RecurringChargeFrequency": "Hourly"  
      }  
    ]  
  }  
}
```

Para obtener más información, consulte [describe-reserved-cache-nodes-offerings](#) en la referencia de la AWS CLI.

Obtención de información sobre las ofertas de nodos reservados (API de ElastiCache)

Para obtener datos acerca de precios e información acerca de las ofertas de nodos reservados disponibles, realice una llamada a la acción `DescribeReservedCacheNodesOfferings`.

Example

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=DescribeReservedCacheNodesOfferings  
&Version=2014-12-01  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20141201T220302Z  
&X-Amz-Algorithm  
&X-Amz-SignedHeaders=Host  
&X-Amz-Expires=20141201T220302Z  
&X-Amz-Credential=<credential>  
&X-Amz-Signature=<signature>
```

Para obtener más información, consulte [DescribeReservedCacheNodesOfferings](#) en la referencia de la API de ElastiCache.

Compra de un nodo reservado

Los siguientes ejemplos muestran cómo comprar una oferta de nodo reservado mediante la AWS Management Console, la AWS CLI y la API de ElastiCache.

Important

Los ejemplos que se muestran a continuación en esta sección incurrirán en gastos en la cuenta de AWS que no podrá revertir.

Temas

- [Compra de un nodo reservado \(Consola\)](#)
- [Compra de un nodo reservado \(AWS CLI\)](#)
- [Compra de un nodo reservado \(API de ElastiCache\)](#)

Compra de un nodo reservado (Consola)

En este ejemplo, se muestra la compra de una oferta de nodo reservado específica, 649fd0c8-cf6d-47a0-bfa6-060f8e75e95f, con un ID de nodo reservado de myreservationID.

En el procedimiento siguiente, se utiliza la AWS Management Console para comprar la oferta de nodo reservado por identificador de oferta.

Para comprar nodos reservados

1. Inicie sesión en la AWS Management Console y abra la consola de ElastiCache en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación, elija el enlace Reserved Nodes (Nodos reservados).
3. Elija el botón Purchase reserved nodes (Comprar nodos reservados).
4. En Engine (Motor), elija Redis.
5. Para determinar las ofertas disponibles, seleccione las siguientes opciones:
 - Node Type
 - Term (Término)
 - Offering Type(Tipo de oferta)
 - Un Reserved node ID (ID de nodo reservado) opcional

Después de realizar estas selecciones, el costo por nodo y el costo total de las selecciones se mostrarán en Reservation details (Detalles de reserva).

6. Elija Purchase (Comprar).

Compra de un nodo reservado (AWS CLI)

En el siguiente ejemplo, se muestra la compra de una oferta de clúster reservado específica, 649fd0c8-cf6d-47a0-bfa6-060f8e75e95f, con un ID de nodo reservado de myreservationID.

Escriba el comando siguiente en el símbolo del sistema:

Para Linux, macOS o Unix:

```
aws elasticache purchase-reserved-cache-nodes-offering \
  --reserved-cache-nodes-offering-id 649fd0c8-cf6d-47a0-bfa6-060f8e75e95f \
  --reserved-cache-node-id myreservationID
```

Para Windows:

```
aws elasticache purchase-reserved-cache-nodes-offering ^
  --reserved-cache-nodes-offering-id 649fd0c8-cf6d-47a0-bfa6-060f8e75e95f ^
  --reserved-cache-node-id myreservationID
```

El comando devuelve un resultado similar al siguiente:

RESERVATION	ReservationId	Class	Start Time	Duration	
Fixed Price	Usage Price	Count	State	Description	Offering Type
RESERVATION	myreservationid	cache.xx.small	2013-12-19T00:30:23.247Z	1y	
XXX.XX USD	X.XXX USD	1	payment-pending	memcached	Medium Utilization

Para obtener más información, consulte [purchase-reserved-cache-nodes-offering](#) en la referencia de la AWS CLI.

Compra de un nodo reservado (API de ElastiCache)

En el siguiente ejemplo, se muestra la compra de una oferta de nodo reservado específica, 649fd0c8-cf6d-47a0-bfa6-060f8e75e95f, con un ID de clúster reservado de myreservationID.

Realice una llamada a la operación `PurchaseReservedCacheNodesOffering` con los parámetros siguientes:

- `ReservedCacheNodesOfferingId` = 649fd0c8-cf6d-47a0-bfa6-060f8e75e95f
- `ReservedCacheNodeID` = myreservationID
- `CacheNodeCount` = 1

Example

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=PurchaseReservedCacheNodesOffering  
  &ReservedCacheNodesOfferingId=649fd0c8-cf6d-47a0-bfa6-060f8e75e95f  
  &ReservedCacheNodeID=myreservationID  
  &CacheNodeCount=1  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Timestamp=20141201T220302Z  
  &X-Amz-Algorithm=&AWS;4-HMAC-SHA256  
  &X-Amz-Date=20141201T220302Z  
  &X-Amz-SignedHeaders=Host  
  &X-Amz-Expires=20141201T220302Z  
  &X-Amz-Credential=<credential>  
  &X-Amz-Signature=<signature>
```

Para obtener más información, consulte [PurchaseReservedCacheNodesOffering](#) en la referencia de la API de ElastiCache.

Obtención de información sobre los nodos reservados

Puede obtener información sobre los nodos reservados que ha comprado con la AWS Management Console, la AWS CLI y la API de ElastiCache.

Temas

- [Obtención de información sobre los nodos reservados \(consola\)](#)
- [Obtención de información sobre los nodos reservados \(AWS CLI\)](#)
- [Obtención de información sobre los nodos reservados \(API de ElastiCache\)](#)

Obtención de información sobre los nodos reservados (consola)

En el procedimiento siguiente, se describe cómo utilizar la AWS Management Console para obtener información acerca de los nodos reservados que ha comprado.

Para obtener información acerca de los nodos reservados comprados

1. Inicie sesión en la AWS Management Console y abra la consola de ElastiCache en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación, elija el enlace Reserved nodes (Nodos reservados).

Los nodos reservados para la cuenta aparecen en la lista de nodos reservados. Puede elegir cualquiera de los nodos reservados de la lista para ver información detallada acerca del nodo reservado en el panel de detalles situado en la parte inferior de la consola.

Obtención de información sobre los nodos reservados (AWS CLI)

Para obtener información sobre los nodos reservados de la cuenta de AWS, escriba el siguiente comando en el símbolo del sistema:

```
aws elasticache describe-reserved-cache-nodes
```

Esta operación produce un resultado similar al siguiente (formato JSON):

```
{
  "ReservedCacheNodeId": "myreservationid",
  "ReservedCacheNodesOfferingId": "649fd0c8-cf6d-47a0-bfa6-060f8e75e95f",
  "CacheNodeType": "cache.xx.small",
  "DataTiering": "disabled",
```

```
"Duration": "31536000",
"ProductDescription": "memcached",
"OfferingType": "Medium Utilization",
"MaxRecords": 0
}
```

Para obtener más información, consulte [describe--reserved-cache-nodes](#) en la referencia de la AWS CLI.

Obtención de información sobre los nodos reservados (API de ElastiCache)

Para obtener información sobre los nodos reservados de la cuenta de AWS, realice una llamada a la operación `DescribeReservedCacheNodes`.

Example

```
https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeReservedCacheNodes
&Version=2014-12-01
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20141201T220302Z
&X-Amz-Algorithm=&AWS;4-HMAC-SHA256
&X-Amz-Date=20141201T220302Z
&X-Amz-SignedHeaders=Host
&X-Amz-Expires=20141201T220302Z
&X-Amz-Credential=<credential>
&X-Amz-Signature=<signature>
```

Para obtener más información, consulte [DescribeReservedCacheNodes](#) en la referencia de la API de ElastiCache.

Migración de nodos de generación anterior

Los nodos de generación anterior son tipos de nodos que se eliminan gradualmente. Si no tiene clústeres existentes que utilicen un tipo de nodo de la generación anterior, ElastiCache no admite la creación de nuevos clústeres con ese tipo de nodo.

Debido a la cantidad limitada de tipos de nodos de generación anterior, no podemos garantizar un reemplazo exitoso cuando un nodo pasa a estar en mal estado en los clústeres. En tal situación, la disponibilidad del clúster puede verse afectada de forma negativa.

Recomendamos migrar los clústeres a un tipo de nodo nuevo para mejorar la disponibilidad y el rendimiento. Para obtener un tipo de nodo recomendado que permita migrar, consulte [Rutas de actualización](#). Para obtener una lista completa de los tipos de nodos compatibles y de los tipos de nodos de la generación anterior ElastiCache, consulte [Tipos de nodos compatibles](#).

Migración de nodos en un clúster de OSS de Redis

El siguiente procedimiento describe cómo migrar el tipo de nodo del clúster de Redis OSS mediante la consola. ElastiCache Durante este proceso, el clúster de Redis OSS seguirá atendiendo las solicitudes con un tiempo de inactividad mínimo. Según la configuración del clúster, puede ver los siguientes tiempos de inactividad. Las siguientes son estimaciones y pueden diferir en función de sus configuraciones específicas:

- El modo de clúster deshabilitado (nodo único) puede ver aproximadamente 60 segundos, principalmente debido a la propagación de DNS.
- El modo de clúster desactivado (con el nodo de réplica) puede tardar aproximadamente 1 segundo en los clústeres que ejecutan Redis OSS 5.0.6 o versiones posteriores. Todas las versiones inferiores pueden experimentar aproximadamente 10 segundos.
- El modo de clúster habilitado puede ver aproximadamente 1 segundo.

Para modificar el tipo de nodo de un clúster OSS de Redis mediante la consola:

1. Inicie sesión en la consola y ábrala ElastiCache en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación, elija los clústeres de Redis OSS.
3. En la lista de clústeres, seleccione el clúster que desea migrar.
4. Elija Actions (Acciones) y después Modify (Modificar).
5. Elija el nuevo tipo de nodo de la lista de tipos de nodos.
6. Si desea realizar el proceso de migración de forma inmediata, elija Apply immediately (Aplicar de inmediato). Si no selecciona Apply immediately (Aplicar de inmediato), el proceso de migración se realizará durante el siguiente periodo de mantenimiento del clúster.
7. Elija Modify (Modificar). Si eligió Apply immediately (Aplicar inmediatamente) en el paso anterior, el estado del clúster cambiará a modifying (en modificación). Cuando el estado cambie a available (disponible), la modificación se habrá completado y podrá empezar a utilizar el nuevo clúster.

Para modificar el tipo de nodo de un clúster OSS de Redis mediante: AWS CLI

Utilice la [modify-replication-group](#) API como se muestra a continuación:

Para Linux, macOS o Unix:

```
aws elasticache modify-replication-group /  
  --replication-group-id my-replication-group /  
  --cache-node-type new-node-type /  
  --apply-immediately
```

Para Windows:

```
aws elasticache modify-replication-group ^  
  --replication-group-id my-replication-group ^  
  --cache-node-type new-node-type ^  
  --apply-immediately
```

En este escenario, el valor de *new-node-type* es el tipo de nodo al que se está migrando. Al pasar el parámetro `--apply-immediately`, la actualización se aplicará de inmediato cuando el grupo de reproducción pase del estado de `modifying` (modificando) a `modification` (disponible). Si no selecciona `Apply immediately` (Aplicar de inmediato), el proceso de migración se realizará durante el siguiente periodo de mantenimiento del clúster.

Note

Si no puede modificar el clúster con un error `InvalidCacheClusterState`, primero debe eliminar un nodo con error de restauración.

Reparar o eliminar `restore-failed-node`

El siguiente procedimiento describe cómo reparar o eliminar los nodos con errores de restauración del clúster de Redis OSS. Para obtener más información sobre cómo los ElastiCache nodos entran en un estado de restauración fallida, consulte [Visualización ElastiCache del estado de los nodos](#). Recomendamos eliminar primero los nodos que se encuentren en un estado de restauración fallida, migrar los nodos restantes de la generación anterior del ElastiCache clúster a un tipo de nodo de nueva generación y, por último, volver a añadir el número de nodos necesario.

Eliminar el nodo con error de restauración (consola):

1. [Inicie sesión en la consola y ábrala en https://console.aws.amazon.com/elasticache/](https://console.aws.amazon.com/elasticache/).
[ElastiCache](#)
2. En el panel de navegación, elija los clústeres de Redis OSS.
3. En la lista de clústeres, elija el clúster del que desea eliminar un nodo.
4. En la lista de particiones, elija la partición de la que desea eliminar un nodo. Omita este paso si el modo clúster se encuentra deshabilitado para el clúster.
5. En la lista de nodos, elija el nodo con un estado `restore-failed`.
6. Elija Actions (Acciones) y, a continuación, elija Delete node (Eliminar nodo).

Una vez que haya eliminado del ElastiCache clúster los nodos con errores de restauración, podrá migrar a un tipo de nueva generación. Para obtener más información, consulte lo descrito anteriormente sobre [Migración de nodos en un clúster de OSS de Redis](#).

Para volver a añadir nodos al ElastiCache clúster, consulte. [Adición de nodos a un clúster](#)

Administración de clústeres

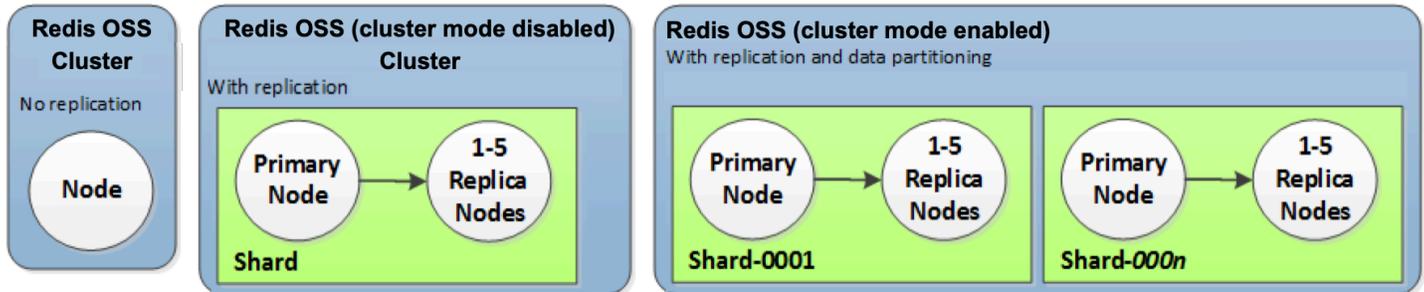
Un clúster es una colección de uno o varios nodos de caché que, en conjunto, ejecutan una instancia de software del motor de caché de Redis. Al crear un clúster, debe especificar el motor y la versión que se va a utilizar en todos los nodos.

El siguiente diagrama ilustra un clúster OSS de Redis típico. Los clústeres de Redis OSS pueden contener un solo nodo o hasta seis nodos dentro de un fragmento (API/CLI: grupo de nodos), un clúster de Redis OSS (modo de clúster deshabilitado) de un solo nodo no tiene ningún fragmento y un clúster de Redis OSS (modo de clúster deshabilitado) de varios nodos tiene un solo fragmento. Los clústeres de Redis OSS (modo de clúster activado) pueden tener hasta 500 fragmentos, y los datos se dividen entre ellos. El límite de nodos o fragmentos se puede aumentar hasta un máximo de 500 por clúster si la versión del motor OSS de Redis es 5.0.6 o superior. Por ejemplo, puede elegir configurar un clúster de 500 nodos que oscila entre 83 particiones (uno primario y 5 réplicas por partición) y 500 particiones (único primario y sin réplicas). Asegúrese de que hay suficientes direcciones IP disponibles para acomodar el aumento. Algunos problemas comunes incluyen que las subredes del grupo de subredes tienen un rango CIDR demasiado pequeño o que otros clústeres comparten y utilizan considerablemente las subredes. Para obtener más información, consulte [Creación de un grupo de subredes](#). Para las versiones inferiores a 5.0.6, el límite es de 250 por clúster.

Para solicitar un aumento del límite, consulte [AWS Service Limits](#) y elija el tipo de límite Nodes per cluster per instance type (Nodos por clúster por tipo de instancias).

Si tiene varios nodos en una partición, uno de los nodos será un nodo primario de lectura/escritura. Todos los demás nodos de la partición son réplicas de solo lectura.

Los clústeres OSS de Redis típicos tienen el siguiente aspecto.



La mayoría ElastiCache de las operaciones se realizan a nivel de clúster. Puede configurar un clúster con un número específico de nodos y un grupo de parámetros que controle las propiedades de cada nodo. Todos los nodos de un clúster están diseñados para ser del mismo tipo y tener los mismos valores de configuración de parámetros y grupo de seguridad.

Cada clúster debe tener un identificador de clúster. El identificador del clúster es un nombre suministrado por el cliente para el clúster. Este identificador especifica un clúster concreto al interactuar con la ElastiCache API y AWS CLI los comandos. El identificador del clúster debe ser único para ese cliente en una AWS región.

ElastiCache admite varias versiones de motor. A menos que tenga motivos específicos, recomendamos que utilice la versión más reciente.

ElastiCache Los clústeres están diseñados para que se pueda acceder a ellos mediante una instancia de Amazon EC2. Si el clúster se lanza en una nube virtual privada (VPC) en función del servicio de Amazon VPC, se puede acceder a él desde fuera de AWS. Para obtener más información, consulte [Acceder a ElastiCache los recursos desde el exterior AWS](#).

Para obtener una lista de las versiones de Redis OSS compatibles, consulte. [Versiones compatibles ElastiCache \(Redis OSS\)](#)

Elección de un tipo de red

ElastiCache es compatible con las versiones 4 y 6 del Protocolo de Internet (IPv4 e IPv6), lo que le permite configurar el clúster para que acepte:

- solo conexiones IPv4,
- solo conexiones IPv6,
- conexiones IPv4 e IPv6 (doble pila).

[Se admite IPv6 para las cargas de trabajo que utilizan el motor OSS de Redis a partir de la versión 6.2 en adelante en todas las instancias integradas en el sistema Nitro.](#) El acceso a través de IPv6 no conlleva cargos adicionales. ElastiCache

Note

No se admite la migración de clústeres creados antes de la disponibilidad de IPV6/doble pila. Tampoco se admite el cambio entre tipos de red en los clústeres recién creados.

Configuración de subredes para el tipo de red

Si crea un clúster en una Amazon VPC, debe especificar un grupo de subredes. ElastiCache usa ese grupo de subredes para elegir una subred y direcciones IP dentro de esa subred para asociarlas a sus nodos. ElastiCache Los clústeres requieren una subred de doble pila con direcciones IPv4 e IPv6 asignadas para funcionar en modo de doble pila y una subred solo de IPv6 para que funcione solo de IPv6.

Usar doble pila

Cuando se utiliza ElastiCache (Redis OSS) con el modo de clúster activado, desde la perspectiva de una aplicación, conectarse a todos los nodos del clúster a través del punto final de configuración no es diferente a conectarse directamente a un nodo de caché individual. Para lograr esto, un cliente que tiene en cuenta los clústeres debe interactuar en un proceso de detección de clústeres y solicitar la información de configuración de todos los nodos. El protocolo de detección de Redis solo admite una IP por nodo.

Para mantener la compatibilidad con versiones anteriores de todos los clientes existentes, se usa la detección de IP, que permite seleccionar el tipo de IP (es decir, IPv4 o IPv6) que se anunciará en el

protocolo de detección. Aunque esto limita la detección automática a un solo tipo de IP, la doble pila sigue siendo beneficiosa para las cargas de trabajo habilitadas en modo de clúster, ya que permite migraciones (o retrocesos) de un tipo de IP de detección IPv4 a uno IPv6 sin tiempo de inactividad.

Clústeres de doble pila compatibles con TLS ElastiCache

Cuando el TLS está habilitado para ElastiCache los clústeres, las funciones de detección de clústeres (`cluster slotscluster shards`, `ycluster nodes`) devuelven nombres de host en lugar de direcciones IP. A continuación, se utilizan los nombres de host en lugar de las IP para conectarse al ElastiCache clúster y realizar un protocolo de enlace TLS. Esto significa que los clientes no se verán afectados por el parámetro de detección de IP. En el caso de los clústeres habilitados para TLS, el parámetro de detección de IP no tiene ningún efecto en el protocolo IP preferido. En cambio, el protocolo IP utilizado se determinará según el protocolo IP que prefiera el cliente al resolver los nombres de host de DNS.

Para ver ejemplos sobre cómo configurar una preferencia de protocolo IP al resolver nombres de host DNS, consulte [Clústeres de doble pila compatibles con TLS ElastiCache](#).

Usando el AWS Management Console

Al crear un clúster mediante el AWS Management Console, en Conectividad, elija un tipo de red, ya sea IPv4, IPv6 o doble pila. Si va a crear un clúster OSS de Redis (modo de clúster habilitado) y elige doble pila, debe seleccionar un tipo de IP de descubrimiento, IPv6 o IPv4.

Para obtener más información, consulte [Creación de un clúster de Redis OSS \(modo de clúster activado\) \(consola\)](#) o [Creación de un OSS de Redis \(modo de clúster desactivado\) \(consola\)](#).

Al crear un grupo de replicación mediante AWS Management Console, elija un tipo de red, ya sea IPv4, IPv6 o doble pila. Si elige doble pila, debe seleccionar un Discovery IP type (Tipo de IP de detección) IPv6 o IPv4.

Para obtener más información, consulte [Creación de un grupo de replicación de Redis OSS \(modo de clúster desactivado\) desde cero](#) o [Crear un grupo de replicación en Redis OSS \(modo de clúster habilitado\) desde cero](#).

Uso de la CLI

Al crear un clúster de caché mediante la CLI, se utiliza el `create-cache-cluster` comando y se especifican los `IPDiscovery` parámetros `NetworkType` y:

Para Linux, macOS o Unix:

```
aws elasticache create-cache-cluster \  
  --cache-cluster-id "cluster-test" \  
  --engine redis \  
  --cache-node-type cache.m5.large \  
  --num-cache-nodes 1 \  
  --network-type dual_stack \  
  --ip-discovery ipv4
```

Para Windows:

```
aws elasticache create-cache-cluster ^  
  --cache-cluster-id "cluster-test" ^  
  --engine redis ^  
  --cache-node-type cache.m5.large ^  
  --num-cache-nodes 1 ^  
  --network-type dual_stack ^  
  --ip-discovery ipv4
```

Al crear un grupo de replicación con el modo de clúster desactivado mediante la CLI, utilice el [create-replication-group](#) comando y especifique los IPDiscovery parámetros NetworkType y:

Para Linux, macOS o Unix:

```
aws elasticache create-replication-group \  
  --replication-group-id sample-repl-group \  
  --replication-group-description "demo cluster with replicas" \  
  --num-cache-clusters 3 \  
  --primary-cluster-id redis01 \  
  --network-type dual_stack \  
  --ip-discovery ipv4
```

Para Windows:

```
aws elasticache create-replication-group ^  
  --replication-group-id sample-repl-group ^  
  --replication-group-description "demo cluster with replicas" ^  
  --num-cache-clusters 3 ^  
  --primary-cluster-id redis01 ^
```

```
--network-type dual_stack ^
--ip-discovery ipv4
```

Al crear un grupo de replicación con el modo de clúster habilitado y utilizar IPv4 para la detección de IP mediante la CLI, utilice el [create-replication-group](#) comando y especifique los IPDiscovery parámetros NetworkType y:

Para Linux, macOS o Unix:

```
aws elasticache create-replication-group \
  --replication-group-id demo-cluster \
  --replication-group-description "demo cluster" \
  --cache-node-type cache.m5.large \
  --num-node-groups 2 \
  --engine redis \
  --cache-subnet-group-name xyz \
  --network-type dual_stack \
  --ip-discovery ipv4 \
  --region us-east-1
```

Para Windows:

```
aws elasticache create-replication-group ^
  --replication-group-id demo-cluster ^
  --replication-group-description "demo cluster" ^
  --cache-node-type cache.m5.large ^
  --num-node-groups 2 ^
  --engine redis ^
  --cache-subnet-group-name xyz ^
  --network-type dual_stack ^
  --ip-discovery ipv4 ^
  --region us-east-1
```

Al crear un grupo de replicación con el modo de clúster habilitado y utilizar IPv6 para la detección de IP mediante la CLI, utilice el [create-replication-group](#) comando y especifique los IPDiscovery parámetros NetworkType y:

Para Linux, macOS o Unix:

```
aws elasticache create-replication-group \
  --replication-group-id demo-cluster \
```

```
--replication-group-description "demo cluster" \  
--cache-node-type cache.m5.large \  
--num-node-groups 2 \  
--engine redis \  
--cache-subnet-group-name xyz \  
--network-type dual_stack \  
--ip-discovery ipv6 \  
--region us-east-1
```

Para Windows:

```
aws elasticache create-replication-group ^  
--replication-group-id demo-cluster ^  
--replication-group-description "demo cluster" ^  
--cache-node-type cache.m5.large ^  
--num-node-groups 2 ^  
--engine redis ^  
--cache-subnet-group-name xyz ^  
--network-type dual_stack ^  
--ip-discovery ipv6 ^  
--region us-east-1
```

Organización de datos en niveles

Los clústeres que componen un grupo de replicación y utilizan un tipo de nodo de la familia r6gd tienen sus datos en niveles entre la memoria y el almacenamiento SSD local (unidades de estado sólido). La organización de datos en niveles ofrece una nueva opción de precio-rendimiento para las cargas de trabajo de Redis OSS, ya que utiliza unidades de estado sólido (SSD) de menor costo en cada nodo del clúster, además de almacenar los datos en la memoria. Es ideal para cargas de trabajo que acceden regularmente hasta un 20 por ciento de su conjunto de datos general y para aplicaciones que pueden tolerar latencia adicional al acceder a los datos en SSD.

En los clústeres con niveles de datos, ElastiCache supervisa la hora del último acceso de cada elemento que almacena. Cuando la memoria disponible (DRAM) se agota por completo, ElastiCache utiliza un algoritmo utilizado recientemente (LRU) para mover automáticamente los elementos a los que se accede con poca frecuencia de la memoria a la SSD. Cuando se accede posteriormente a los datos de la SSD, los devuelve a la memoria de ElastiCache forma automática y asíncrona antes de procesar la solicitud. Si tiene una carga de trabajo que solo accede a un subconjunto de sus datos regularmente, la organización de datos en niveles es una forma óptima de escalar su capacidad de forma rentable.

Tenga en cuenta que cuando se utiliza la organización de datos en niveles, las propias claves siempre permanecen en la memoria, mientras que la LRU rige la ubicación de los valores en la memoria frente al disco. En general, recomendamos que los tamaños de las claves sean más pequeños que los tamaños de los valores al usar la organización de datos en niveles.

La organización de datos en niveles está diseñada para tener un impacto mínimo en el rendimiento de las cargas de trabajo. Por ejemplo, suponiendo valores de cadena de 500 bytes, puede esperar 300 microsegundos adicionales de latencia en promedio para las solicitudes de datos almacenados en SSD en comparación con las solicitudes de datos de la memoria.

Con el mayor tamaño de nodo de organización de datos en niveles (cache.r6gd.16xlarge), puede almacenar hasta 1 petabyte en un solo clúster de 500 nodos (500 TB cuando se utiliza 1 réplica de lectura). La organización de datos en niveles es compatible con todos los comandos y estructuras de datos del OSS de Redis compatibles. ElastiCache No es necesario cambiar el lado del cliente para usar esta característica.

Temas

- [Prácticas recomendadas](#)
- [Limitaciones](#)
- [Precios](#)
- [Monitorización](#)
- [Uso de la organización de datos en niveles](#)
- [Restauración de datos desde copias de seguridad en clústeres con la organización de datos en niveles](#)

Prácticas recomendadas

Recomendamos que siga las siguientes prácticas recomendadas:

- La organización de datos en niveles es ideal para cargas de trabajo que acceden regularmente hasta un 20 % de su conjunto de datos general y para aplicaciones que pueden tolerar latencia adicional al acceder a los datos en SSD.
- Al utilizar la capacidad de SSD disponible en nodos con niveles de datos, recomendamos que el tamaño del valor sea mayor que el tamaño de la clave. Cuando se mueven elementos entre DRAM y SSD, las claves siempre permanecerán en la memoria y solo los valores se moverán al nivel de SSD.

Limitaciones

La organización de datos en niveles tiene las siguientes restricciones:

- Solo puede utilizar la organización de datos en niveles en clústeres que forman parte de un grupo de replicación.
- El tipo de nodo que utilice debe pertenecer a la familia r6gd, que está disponible en las siguientes regiones: us-east-2, us-east-1, us-west-2, us-west-1, eu-west-1, eu-central-1, eu-north-1, eu-west-3, ap-northeast-1, ap-southeast-1, ap-southeast-2, ap-south-1, ca-central-1 y sa-east-1.
- Debe utilizar el motor Redis OSS 6.2 o posterior.
- No se puede restaurar una copia de seguridad de un clúster r6gd en otro clúster a menos que utilice también r6gd.
- No se puede exportar una copia de seguridad a Amazon S3 para clústeres de organización de datos en niveles.
- La migración en línea no se admite para clústeres que se ejecutan en el tipo de nodo r6gd.
- No se admite el escalado desde un clúster de organización de datos en niveles (por ejemplo, un clúster que utiliza un tipo de nodo r6gd) a un clúster que no utiliza la organización de datos en niveles (por ejemplo, un clúster que utiliza un tipo de nodo r6g). Para obtener más información, consulte [Escalado ElastiCache \(Redis OSS\)](#).
- El escalado automático se admite en los clústeres que utilizan la organización de datos por niveles en la versión 7.0.7 y versiones posteriores de Redis OSS. Para obtener más información, consulte [Clústeres de Auto Scaling ElastiCache \(Redis OSS\)](#).
- La organización de datos en niveles solo admite las políticas maxmemory volatile-lru, allkeys-lru, volatile-lfu, allkeys-lfu y noeviction.
- La versión 7.0.7 y posteriores de Redis OSS admiten el almacenamiento sin formalización. Para obtener más información, consulte [Cómo se implementan la sincronización y la copia de seguridad](#).
- Los artículos de más de 128 MiB no se mueven a SSD.

Precios

Los nodos R6gd tienen 4,8 veces más capacidad total (memoria + SSD) y pueden ayudarle a ahorrar más del 60 por ciento cuando se ejecuta con la máxima utilización en comparación con los nodos R6g (solo memoria). [Para obtener más información, consulte los precios. ElastiCache](#)

Monitorización

ElastiCache (Redis OSS) ofrece métricas diseñadas específicamente para monitorear los clústeres de rendimiento que utilizan la organización de datos por niveles. Para controlar la proporción de elementos de la DRAM en comparación con la de la SSD, puede utilizar la `CurrentItems` métrica de [Metrics for Redis OSS](#). Puede calcular el porcentaje de la siguiente manera: (CurrentItems con dimensión: nivel = memoria * 100)/(CurrentItems sin filtro de dimensiones).

Si la política de desalojo configurada lo permite, ElastiCache (Redis OSS) empezará a desalojar los elementos cuando el porcentaje de elementos en la memoria disminuya por debajo del 5 por ciento. En los nodos configurados con la política de desalojo, las operaciones de escritura recibirán un error de memoria insuficiente.

Aun así, se recomienda considerar la posibilidad de escalar los clústeres habilitados en modo de clúster o ampliarlos para los clústeres deshabilitados en modo de clúster cuando el porcentaje de elementos en la memoria disminuya por debajo del 5 por ciento. Para obtener más información sobre el escalado, consulte [Escalar clústeres en Redis OSS \(modo de clúster habilitado\)](#). Para obtener más información sobre las métricas de los clústeres OSS de Redis que utilizan la organización de datos en niveles, consulte. [Métricas para Redis OSS](#)

Uso de la organización de datos en niveles

Uso de la organización de datos por niveles mediante el AWS Management Console

Al crear un clúster como parte de un grupo de replicación, se utiliza la organización de datos en niveles seleccionando un tipo de nodo de la familia `r6gd`, como `cache.r6gd.xlarge`. La selección de ese tipo de nodo habilita automáticamente la organización de datos en niveles.

Para obtener más información sobre la creación de clústeres, consulte [Creación de un clúster](#).

Habilitar la estratificación de datos mediante el AWS CLI

Al crear un grupo de replicación mediante el AWS CLI, se utiliza la estratificación de datos seleccionando un tipo de nodo de la familia `r6gd`, como `cache.r6gd.xlarge`, y configurando el parámetro. `--data-tiering-enabled`

No puede excluirse de la organización de datos en niveles al seleccionar un tipo de nodo de la familia `r6gd`. Si configura el parámetro `--no-data-tiering-enabled`, la operación no se llevará a cabo correctamente.

Para Linux, macOS o Unix:

```
aws elasticache create-replication-group \  
  --replication-group-id redis-dt-cluster \  
  --replication-group-description "Redis OSS cluster with data tiering" \  
  --num-node-groups 1 \  
  --replicas-per-node-group 1 \  
  --cache-node-type cache.r6gd.xlarge \  
  --engine redis \  
  --cache-subnet-group-name default \  
  --automatic-failover-enabled \  
  --data-tiering-enabled
```

Para Windows:

```
aws elasticache create-replication-group ^  
  --replication-group-id redis-dt-cluster ^  
  --replication-group-description "Redis OSS cluster with data tiering" ^  
  --num-node-groups 1 ^  
  --replicas-per-node-group 1 ^  
  --cache-node-type cache.r6gd.xlarge ^  
  --engine redis ^  
  --cache-subnet-group-name default ^  
  --automatic-failover-enabled ^  
  --data-tiering-enabled
```

Después de ejecutar esta operación, verá una respuesta parecida a la siguiente:

```
{  
  "ReplicationGroup": {  
    "ReplicationGroupId": "redis-dt-cluster",  
    "Description": "Redis OSS cluster with data tiering",  
    "Status": "creating",  
    "PendingModifiedValues": {},  
    "MemberClusters": [  
      "redis-dt-cluster"  
    ],  
    "AutomaticFailover": "enabled",  
    "DataTiering": "enabled",  
    "SnapshotRetentionLimit": 0,  
    "SnapshotWindow": "06:00-07:00",  
    "ClusterEnabled": false,  
    "CacheNodeType": "cache.r6gd.xlarge",  
    "TransitEncryptionEnabled": false,
```

```
    "AtRestEncryptionEnabled": false
  }
}
```

Restauración de datos desde copias de seguridad en clústeres con la organización de datos en niveles

Puede restaurar una copia de seguridad en un clúster nuevo con la organización de datos en niveles habilitada mediante la (consola), () o (API).AWS CLI ElastiCache Cuando crea un clúster mediante tipos de nodos de la familia r6gd, se habilita la organización de datos en niveles.

Restauración de datos desde la copia de seguridad en clústeres con la organización de datos en niveles activada (consola)

Para restaurar una copia de seguridad a un nuevo clúster con la organización de datos en niveles habilitada (consola)

1. Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación, seleccione Backups (Copias de seguridad).
3. En la lista de copias de seguridad, active la casilla situada a la izquierda del nombre de la copia de seguridad que desea restaurar.
4. Elija Restore (Restaurar).
5. Complete el cuadro de diálogo Restore Cluster. Asegúrese de completar todos los campos Required (Obligatorios) y los demás campos cuyos valores predeterminados desee modificar.
 1. ID del clúster: obligatorio. Se trata del nombre del nuevo clúster.
 2. Modo de clúster activado (escalado horizontal): seleccione esta opción para un clúster de Redis OSS (modo de clúster activado).
 3. Tipo de nodo: especificar cache.r6gd.xlarge o cualquier otro tipo de nodo de la familia r6gd.
 4. Number of Shards: elija el número de fragmentos que desea en el nuevo clúster (API/CLI: grupos de nodos).
 5. Replicas per Shard: elija el número de nodos de réplica de lectura que desea en cada fragmento.
 6. Ranuras y espacios de claves: elija cómo desea que se distribuyan las claves entre las particiones. Si decide especificar las distribuciones de claves, complete la tabla especificando los rangos de claves para cada fragmento.

7. Availability zone(s): especifique cómo desea que se seleccionen las zonas de disponibilidad del clúster.
 8. Port: modifique este valor solo si desea que el nuevo clúster use un puerto distinto.
 9. Choose a VPC: elija la VPC en la que va a crear este clúster.
 - 10.Grupo de parámetros: elija un grupo de parámetros que reserve suficiente memoria para la sobrecarga de Redis OSS para el tipo de nodo que haya seleccionado.
6. Cuando esté conforme con los ajustes, elija Crear.

Para obtener más información sobre la creación de clústeres, consulte [Creación de un clúster](#).

Restauración de datos desde copias de seguridad en clústeres con la organización de datos en niveles habilitada (AWS CLI)

Al crear un grupo de replicación mediante el AWS CLI, la organización de datos por niveles se utiliza de forma predeterminada. Para ello, se selecciona un tipo de nodo de la familia r6gd, como cache.r6gd.xlarge, y se establece el parámetro. `--data-tiering-enabled`

No puede excluirse de la organización de datos en niveles al seleccionar un tipo de nodo de la familia r6gd. Si configura el parámetro `--no-data-tiering-enabled`, la operación no se llevará a cabo correctamente.

Para Linux, macOS o Unix:

```
aws elasticache create-replication-group \  
  --replication-group-id redis-dt-cluster \  
  --replication-group-description "Redis OSS cluster with data tiering" \  
  --num-node-groups 1 \  
  --replicas-per-node-group 1 \  
  --cache-node-type cache.r6gd.xlarge \  
  --engine redis \  
  --cache-subnet-group-name default \  
  --automatic-failover-enabled \  
  --data-tiering-enabled \  
  --snapshot-name my-snapshot
```

Para Linux, macOS o Unix:

```
aws elasticache create-replication-group ^ \  
  --replication-group-id redis-dt-cluster ^ \  
  --replication-group-description "Redis OSS cluster with data tiering" ^
```

```
--num-node-groups 1 ^
--replicas-per-node-group 1 ^
--cache-node-type cache.r6gd.xlarge ^
--engine redis ^
--cache-subnet-group-name default ^
--automatic-failover-enabled ^
--data-tiering-enabled ^
--snapshot-name my-snapshot
```

Después de ejecutar esta operación, verá una respuesta parecida a la siguiente:

```
{
  "ReplicationGroup": {
    "ReplicationGroupId": "redis-dt-cluster",
    "Description": "Redis OSS cluster with data tiering",
    "Status": "creating",
    "PendingModifiedValues": {},
    "MemberClusters": [
      "redis-dt-cluster"
    ],
    "AutomaticFailover": "enabled",
    "DataTiering": "enabled",
    "SnapshotRetentionLimit": 0,
    "SnapshotWindow": "06:00-07:00",
    "ClusterEnabled": false,
    "CacheNodeType": "cache.r6gd.xlarge",
    "TransitEncryptionEnabled": false,
    "AtRestEncryptionEnabled": false
  }
}
```

Preparación de un clúster

A continuación, puede encontrar instrucciones sobre cómo crear un clúster a través de la consola de ElastiCache, la AWS CLI o la API de ElastiCache.

También puede crear un clúster de ElastiCache con [AWS CloudFormation](#). Para obtener más información, consulte [AWS::ElastiCache::CacheCluster](#) en la Guía del usuario de AWS CloudFormation, que contiene directrices acerca de cómo implementar este enfoque.

Siempre que cree un clúster o un grupo de reproducción, es conveniente realizar algunos preparativos para que no sea necesario actualizar o efectuar cambios de inmediato.

Temas

- [Determinación de los requisitos](#)
- [Elección del tamaño del nodo](#)

Determinación de los requisitos

Preparación

Conocer las respuestas a las siguientes preguntas ayuda a que la creación del clúster sea más fluida:

- ¿Qué tipo de instancia de nodo necesita?

Para obtener instrucciones sobre cómo elegir un tipo de nodo de instancia, consulte [Elección del tamaño del nodo](#).

- ¿Lanzará su clúster en una nube virtual privada (VPC) en función de Amazon VPC?

Important

Si va a lanzar su clúster en una VPC, debe asegurarse de crear un grupo de subredes en la misma VPC antes de comenzar a crear un clúster. Para obtener más información, consulte [Subredes y grupos de subredes](#).

ElastiCache está diseñado para que se pueda acceder a él desde dentro de AWS mediante Amazon EC2. Sin embargo, si se lanza en una VPC basada en Amazon VPC y el clúster se encuentra en una VPC, se puede proporcionar acceso desde fuera de AWS. Para obtener más información, consulte [Acceder a ElastiCache los recursos desde el exterior AWS](#).

- ¿Necesita personalizar los valores de algún parámetro?

Si lo hace, cree un grupo de parámetros personalizado. Para obtener más información, consulte [Creación de un grupo de parámetros](#).

Si está ejecutando Redis OSS, considere la posibilidad de configurar `reserved-memory` o `reserved-memory-percent`. Para obtener más información, consulte [Administración de la memoria reservada](#).

- ¿Necesita crear su propio grupo de seguridad de VPC?

Para obtener más información, consulte [Seguridad en la VPC](#).

- ¿Cómo pretende implementar la tolerancia a errores?

Para obtener más información, consulte [Mitigación de errores](#).

Temas

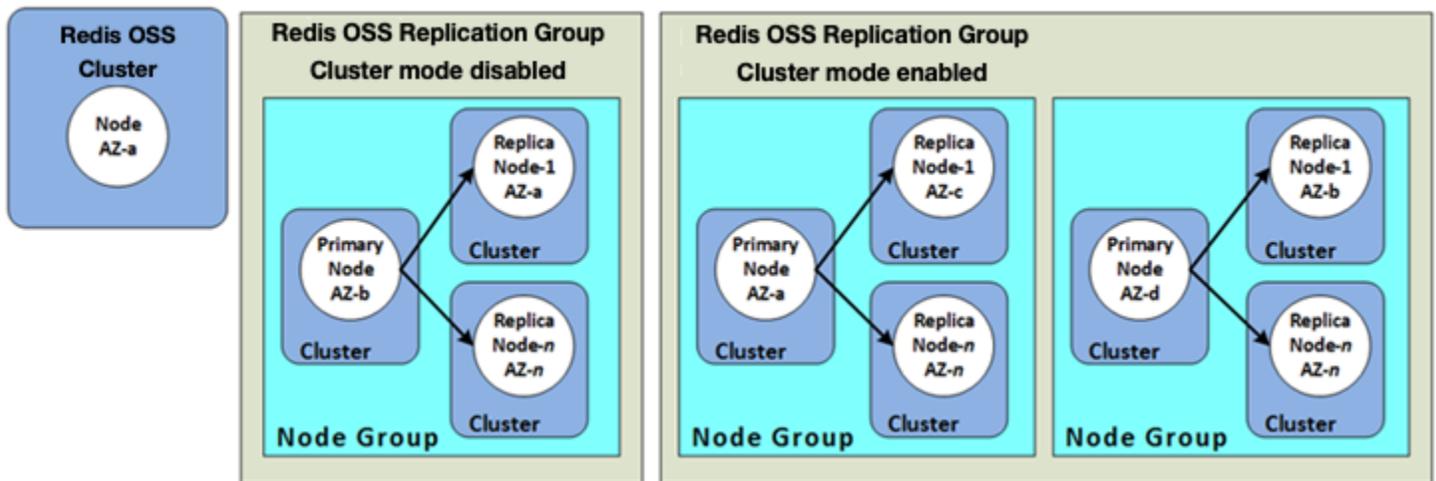
- [Requisitos de procesador y memoria](#)
- [Configuración de clústeres de Redis OSS](#)
- [Requisitos de escalado](#)
- [Requisitos de acceso](#)
- [Requisitos de región, zona de disponibilidad y zona local](#)

Requisitos de procesador y memoria

El componente básico de Amazon ElastiCache es el nodo. Los nodos se configuran individualmente o en agrupaciones para formar clústeres. A la hora de determinar el tipo de nodo que desea utilizar para el clúster, tenga en cuenta la configuración del nodo del clúster y la cantidad de datos que tiene que almacenar.

Configuración de clústeres de Redis OSS

ElastiCache (Redis OSS) los clústeres se componen de 0 a 500 fragmentos (también denominados grupos de nodos). Los datos de un clúster de OSS de Redis se dividen entre los fragmentos del clúster. La aplicación se conecta a un clúster de OSS de Redis mediante una dirección de red denominada punto final. Los nodos de un fragmento OSS de Redis cumplen una de dos funciones: una principal de lectura/escritura y todos los demás nodos secundarios de solo lectura (también denominadas réplicas de lectura). Además de los puntos finales de los nodos, el propio clúster de OSS de Redis tiene un punto final denominado punto final de configuración. La aplicación puede usar este punto final para leer o escribir en el clúster, dejando en manos del usuario la determinación del nodo desde el que leer o escribir ElastiCache (Redis OSS).



Para obtener más información, consulte [Administración de clústeres](#).

Requisitos de escalado

Todos los clústeres pueden ampliarse mediante la creación de un nuevo clúster con el nuevo tipo de nodo, que es más grande. Al escalar un clúster OSS de Redis, puede iniciarlo a partir de una copia de seguridad y evitar que el nuevo clúster comience vacío.

Para obtener más información, consulte la sección [Escalado ElastiCache \(Redis OSS\)](#) de esta guía.

Requisitos de acceso

Por diseño, se accede a ElastiCache los clústeres de Amazon desde las instancias de Amazon EC2. El acceso de red a un ElastiCache clúster está limitado a la cuenta que creó el clúster. Por lo tanto, antes de poder acceder a un clúster desde una instancia de Amazon EC2, debe autorizar a dicha instancia para el acceso al clúster. Los pasos para ello varían en función de si realiza el lanzamiento en EC2-VPC o en EC2-Classic.

Si ha lanzado el clúster en EC2-VPC, necesita conceder acceso de red al clúster. Si lanzó su clúster en EC2-Classic, debe conceder al grupo de seguridad de Amazon Elastic Compute Cloud asociado a la instancia el acceso a su grupo de ElastiCache seguridad. Para obtener instrucciones detalladas, consulte [Paso 3: autorizar acceso al clúster](#) en esta guía.

Requisitos de región, zona de disponibilidad y zona local

Amazon ElastiCache es compatible con todas AWS las regiones. Al ubicar sus ElastiCache clústeres en una AWS región cercana a su aplicación, puede reducir la latencia. Si el clúster tiene varios nodos, ubicar los nodos en distintas zonas de disponibilidad o en Local Zones puede reducir el impacto de los errores en el clúster.

Para más información, consulte los siguientes temas:

- [Elección de regiones y zonas de disponibilidad](#)
- [Uso de zonas locales con ElastiCache](#)
- [Mitigación de errores](#)

Elección del tamaño del nodo

El tamaño del nodo que seleccione para el clúster afectará a los costos, al rendimiento y a la tolerancia a errores.

Elección del tamaño del nodo

Para obtener información sobre las ventajas de los procesadores Graviton, consulte [Procesador AWS Graviton](#).

Responder a las siguientes preguntas puede ayudarle a determinar el tipo de nodo mínimo que necesita para la implementación de Redis OSS:

- ¿Espera cargas de trabajo limitadas por el rendimiento con múltiples conexiones de cliente?

Si este es el caso y utiliza la versión 5.0.6 o superior de Redis OSS, puede mejorar el rendimiento y la latencia con nuestra función de E/S mejorada, en la que se utilizan las CPU disponibles para descargar las conexiones de los clientes, en nombre del motor OSS de Redis. Si utiliza la versión 7.0.4 o superior de Redis OSS, además de la E/S mejorada, obtendrá una aceleración adicional con la multiplexación de E/S mejorada, en la que cada subproceso de E/S de red dedicado canaliza los comandos de varios clientes al motor de Redis OSS, aprovechando la capacidad de Redis OSS de procesar eficientemente los comandos en lotes. En la versión 7.1 y ElastiCache posteriores (Redis OSS), ampliamos la funcionalidad mejorada de subprocesos de E/S para gestionar también la lógica de la capa de presentación. Por capa de presentación, lo que queremos decir es que los subprocesos de E/S mejorados ahora no solo leen las entradas del cliente, sino que también las analizan en el formato de comandos binarios OSS de Redis, que luego se reenvían al subproceso principal para su ejecución, lo que mejora el rendimiento. Consulte la [entrada de blog](#) y la página de [versiones compatibles](#) para obtener más información.

- ¿Tiene cargas de trabajo que acceden regularmente a un pequeño porcentaje de sus datos?

Si este es el caso y utiliza la versión 6.2 o posterior del motor OSS de Redis, puede aprovechar la organización de los datos por niveles eligiendo el tipo de nodo r6gd. Con la organización de datos en niveles, los datos menos usados recientemente se almacenan en SSD. Cuando se recupera,

hay un pequeño costo de latencia, que se equilibra con el ahorro de costos. Para obtener más información, consulte [Organización de datos en niveles](#).

Para obtener más información, consulte [Tipos de nodos compatibles](#).

- ¿Cuánta memoria total necesita para sus datos?

Para obtener una estimación general, tome el tamaño de los elementos que desea almacenar en la caché. Multiplique este tamaño por el número de elementos que desea conservar en la caché al mismo tiempo. Para obtener una estimación razonable del tamaño de los elementos, serialice los elementos de la caché y cuente los caracteres. A continuación, divida esto entre el número de particiones de su clúster.

Para obtener más información, consulte [Tipos de nodos compatibles](#).

- ¿Qué versión de Redis OSS utiliza?

Las versiones de Redis OSS anteriores a la 2.8.22 requieren reservar más memoria para la conmutación por error, las instantáneas, la sincronización y la conversión de una réplica a las operaciones principales. Este requisito se debe a que debe disponer de suficiente memoria disponible para todas las operaciones de escritura que se producen durante el proceso.

La versión 2.8.22 y posteriores de Redis OSS utilizan un proceso de guardado sin bifurcación que requiere menos memoria disponible que el proceso anterior.

Para más información, consulte los siguientes temas:

- [Cómo se implementan la sincronización y la copia de seguridad](#)
- [Asegurarse de tener suficiente memoria para crear una instantánea de Redis OSS](#)
- ¿Qué volumen de operaciones de escritura realiza su aplicación?

Las aplicaciones que realizan un elevado volumen de operaciones de escritura pueden requerir un mayor volumen de memoria disponible (memoria no usada por los datos) a la hora de tomar instantáneas o en casos de conmutación por error. Cuando se aplica el proceso BGSAVE, debe tener suficiente memoria que no sea utilizada por los datos para alojar todas las escrituras que se producen durante el proceso BGSAVE. Algunos ejemplos son: tomar una instantánea, sincronizar un clúster principal con una réplica en un clúster y habilitar la característica de archivo de solo anexo (AOF). Otro ejemplo es cuando se promueve una réplica a principal (si tiene habilitado Multi-AZ). El peor de los casos es cuando todos los datos se reescriben durante el proceso. En este caso, necesita un tamaño de instancia de nodo con el doble de la memoria necesaria solo para los datos.

Para obtener información más detallada, consulte [Asegurarse de tener suficiente memoria para crear una instantánea de Redis OSS](#).

- ¿Su implementación será un clúster de Redis OSS (modo de clúster desactivado) independiente o un clúster de Redis OSS (modo de clúster habilitado) con varios fragmentos?

Clúster Redis OSS (modo de clúster desactivado)

Si está implementando un clúster OSS (modo de clúster deshabilitado) de Redis, su tipo de nodo debe poder acomodar todos sus datos más la sobrecarga necesaria, tal como se describe en el bullet anterior.

Por ejemplo, supongamos que estima que el tamaño total de todos los elementos es de 12 GB. En este caso, puede utilizar un nodo de cache .m3.xlarge con 13.3 GB de memoria o un nodo de cache .r3.large con 13.5 GB de memoria. Sin embargo, es posible que necesite más memoria para las operaciones BGSAVE. Si la aplicación tiene un gran volumen de operaciones de escritura, duplique los requisitos de memoria a 24 GB como mínimo. Por lo tanto, utilice un cache .m3.2xlarge con 27.9 GB de memoria o un cache .r3.xlarge con 30.5 GB de memoria.

Redis OSS (modo de clúster activado) con varios fragmentos

Si está implementando un clúster de Redis OSS (modo de clúster habilitado) con varios fragmentos, el tipo de nodo debe poder alojar `bytes-for-data-and-overhead / number-of-shards` bytes de datos.

Por ejemplo, suponga que estima que el tamaño total de todos los elementos es de 12 GB y tiene dos particiones. En este caso, puede utilizar un nodo de cache .m3.large con 6.05 GB de memoria (12 GB/2). Sin embargo, es posible que necesite más memoria para las operaciones BGSAVE. Si la aplicación tiene un gran volumen de operaciones de escritura, duplique los requisitos de memoria a 12 GB por partición como mínimo. Por lo tanto, utilice un cache .m3.xlarge con 13.3 GB de memoria o un cache .r3.large con 13.5 GB de memoria.

- ¿Utiliza Local Zones?

[Las Zonas Locales](#) le permiten colocar recursos, como un ElastiCache clúster, en varias ubicaciones cercanas a sus usuarios. Sin embargo, cuando elija el tamaño del nodo, tenga en cuenta que, independientemente de los requisitos de capacidad, los tamaños de nodo disponibles tienen los siguientes límites en este momento:

- Generación actual:

Tipos de nodos M5: `cache.m5.large`, `cache.m5.xlarge`, `cache.m5.2xlarge`, `cache.m5.4xlarge`, `cache.m5.12xlarge`, `cache.m5.24xlarge`

Tipos de nodos R5: `cache.r5.large`, `cache.r5.xlarge`, `cache.r5.2xlarge`, `cache.r5.4xlarge`, `cache.r5.12xlarge`, `cache.r5.24xlarge`

Tipos de nodos T3: `cache.t3.micro`, `cache.t3.small`, `cache.t3.medium`

Mientras el clúster está en ejecución, puede supervisar las métricas de uso de memoria, uso del procesador, aciertos y errores de caché publicadas en ellas. CloudWatch Es posible que observe que el clúster no tiene la tasa de aciertos que desea o que las claves se expulsan con demasiada frecuencia. En estos casos, puede elegir un tamaño de nodo diferente con mayores especificaciones de memoria y CPU.

Cuando supervise el uso de la CPU, recuerde que el OSS de Redis es de un solo subproceso. Por lo tanto, multiplique el uso de la CPU por el número de núcleos de CPU para obtener el uso real. Por ejemplo, una CPU de cuatro núcleos que presenta una tasa de uso del 20 por ciento es en realidad la única CPU de Redis OSS que funciona con un 80 por ciento de utilización.

Creación de un clúster

Los siguientes ejemplos muestran cómo crear un clúster OSS de Redis mediante la AWS Management Console ElastiCache API AWS CLI y.

Creación de un OSS de Redis (modo de clúster desactivado) (consola)

ElastiCache admite la replicación cuando se utiliza el motor OSS de Redis. Para supervisar la latencia entre el momento en que los datos se escriben en un clúster principal de lectura/escritura de Redis OSS y el momento en que se propagan a un clúster secundario de solo lectura, ElastiCache agrega al clúster una clave especial, `ElastiCacheMasterReplicationTimestamp`. Esta clave es la hora de tiempo universal (UCT). Como un clúster de OSS de Redis se puede añadir a un grupo de replicación más adelante, esta clave se incluye en todos los clústeres de OSS de Redis, incluso si inicialmente no son miembros de un grupo de replicación. Para obtener más información acerca de los grupos de reproducción, consulte [Alta disponibilidad a través de grupos de reproducción](#).

Para crear un clúster de Redis OSS (modo de clúster desactivado), siga los pasos que se indican en [Creación de un clúster de Redis OSS \(modo de clúster desactivado\) \(consola\)](#)

En cuanto el estado de su clúster sea `available` (disponible), podrá conceder a Amazon EC2 acceso a este, así como conectarse a él y comenzar a utilizarlo. Para obtener más información, consulte [Paso 3: autorizar acceso al clúster](#) y [Paso 4: conectarse al nodo de un clúster](#).

Important

Cuando su clúster esté disponible, se cobrará por cada hora u hora parcial que el clúster esté activo, incluso si no lo está utilizando de forma activa. Para dejar de incurrir en cargos por este clúster, debe eliminarlo. Consulte [Eliminación de un clúster](#).

Creación de un clúster de Redis OSS (modo de clúster activado) (consola)

Si ejecuta Redis OSS 3.2.4 o una versión posterior, puede crear un clúster de Redis OSS (modo de clúster habilitado). Los clústeres OSS de Redis (modo de clúster habilitado) admiten la partición de los datos en 1 a 500 fragmentos (API/CLI: grupos de nodos), pero con algunas limitaciones. Para ver una comparación entre Redis OSS (modo de clúster desactivado) y Redis OSS (modo de clúster activado), consulte [Versiones compatibles ElastiCache \(Redis OSS\)](#)

Para crear un clúster de Redis OSS (modo de clúster activado) mediante la consola ElastiCache

1. Inicia sesión en la ElastiCache consola de Amazon AWS Management Console y ábrela en <https://console.aws.amazon.com/elasticache/>.
2. En la lista de la esquina superior derecha, selecciona la AWS región en la que quieres lanzar este clúster.
3. En el panel de navegación, seleccione Get started (Introducción).
4. Elija Create VPC (Crear una VPC) y siga los pasos que se detallan en [Creating a Virtual Private Cloud \(VPC\)](#) (Creación de una nube privada virtual).
5. En la página del ElastiCache panel, elija Crear clúster y, a continuación, elija Crear clúster de Redis OSS.
6. En Cluster settings (Configuración de clúster), realice la siguiente operación:
 - a. Elija Configure and create a new cluster (Configurar y crear un nuevo clúster).
 - b. Para Cluster mode (Modo clúster), elija Enabled (Habilitado).
 - c. Para Cluster info (Información del clúster) introduzca un valor para Name (Nombre).
 - d. (Opcional) Introduzca un valor para Description (Descripción).
7. En Location (Ubicación):

AWS Cloud

1. Para Nube de AWS , le recomendamos que acepte la configuración predeterminada de Multi-AZ y Conmutación por error automática. Para obtener más información, consulte [Minimizar el tiempo de inactividad en ElastiCache \(Redis OSS\) con Multi-AZ](#).
2. En Cluster settings (Configuración del clúster)
 - a. Para Engine version (Versión del motor), elija una versión disponible.
 - b. En Port (Puerto), utilice el puerto predeterminado, 6379. Si tiene un motivo para utilizar un puerto diferente, escriba el número de puerto.
 - c. En Grupo de parámetros, elija un grupo de parámetros o cree uno nuevo. Los grupos de parámetros controlan los parámetros de tiempo de ejecución de su clúster. Para obtener más información acerca de los grupos de parámetros, consulte [Parámetros específicos de Redis OSS](#) y [Creación de un grupo de parámetros](#).

Note

Cuando se selecciona un grupo de parámetros para definir los valores de configuración del motor, dicho grupo de parámetros se aplica a todos los clústeres del almacén de datos global. En la página Parameter Groups (Grupos de parámetros), el atributo Global sí/no, indica si un grupo de parámetros forma parte de un almacén de datos global.

- d. Para Node type, elija la flecha hacia abajo (▼).

En el cuadro de diálogo Cambiar tipo de nodo, elija un valor para Familia de instancias para el tipo de nodo que desea. A continuación, elija el tipo de nodo que desea utilizar para este clúster y, a continuación, elija Guardar.

Para obtener más información, consulte [Elección del tamaño del nodo](#).

Si elige un tipo de nodo r6gd, la organización de datos en niveles se activará automáticamente. Para obtener más información, consulte [Organización de datos en niveles](#).

- e. En Número de fragmentos, elija el número de fragmentos (particiones/grupos de nodos) que desee para este clúster de Redis OSS (modo de clúster habilitado).

En algunas versiones de Redis OSS (modo de clúster activado), puede cambiar el número de fragmentos del clúster de forma dinámica:

- Redis OSS 3.2.10 y versiones posteriores: si su clúster ejecuta Redis OSS 3.2.10 o versiones posteriores, puede cambiar la cantidad de fragmentos del clúster de forma dinámica. Para obtener más información, consulte [Escalar clústeres en Redis OSS \(modo de clúster habilitado\)](#).
- Otras versiones de Redis OSS: si su clúster ejecuta una versión de Redis OSS anterior a la versión 3.2.10, existe otro enfoque. Para cambiar el número de particiones del clúster en este caso, cree un nuevo clúster con el nuevo número de particiones. Para obtener más información, consulte [Restauración desde una copia de seguridad a una nueva caché](#).

- f. En Réplicas por partición, elija el número de nodos de réplica de lectura que desea en cada partición.

Existen las siguientes restricciones para Redis OSS (modo de clúster activado).

- Si tiene habilitado Multi-AZ, asegúrese de tener al menos una réplica por partición.
- El número de réplicas es el mismo para cada fragmento al crear el clúster utilizando la consola.
- El número de réplicas de lectura por fragmento es fijo y no puede modificarse. Si se da cuenta de que necesita más o menos réplicas por fragmento (API/CLI: grupo de nodos), debe crear un nuevo clúster con el nuevo número de réplicas. Para obtener más información, consulte [Inicialización de datos en un nuevo clúster de autodiseño con una copia de seguridad creada externamente](#).

3. En Connectivity (Conectividad)

- a. Para el Network type (Tipo de red), elija las versiones IP que admitirá este clúster.
- b. Para los grupos de subredes, elija la subred que desee aplicar a este clúster. ElastiCache usa ese grupo de subredes para elegir una subred y las direcciones IP dentro de esa subred para asociarlas a los nodos. ElastiCache Los clústeres requieren una subred de doble pila con direcciones IPv4 e IPv6 asignadas para funcionar en modo de doble pila y una subred solo de IPv6 para que funcione solo de IPv6.

Al crear un nuevo grupo de subredes, escriba el VPC ID (ID de VPC) al que pertenece.

Seleccione un Discovery IP type (Tipo de IP de detección). Solo se devuelven las direcciones IP del protocolo elegido.

Para obtener más información, consulte:

- [Elección de un tipo de red](#).
- [Cree una subred en la VPC](#).

Si es [Uso de zonas locales con ElastiCache](#), debe crear o elegir una subred que se encuentre en la zona local.

Para obtener más información, consulte [Subredes y grupos de subredes](#).

4. En Availability zone placements Ubicaciones de las zonas de disponibilidad), tiene dos opciones:
 - ElastiCache Sin preferencia: elige la zona de disponibilidad.
 - Specify availability zones (Especificar zonas de disponibilidad): especifique la zona de disponibilidad para cada clúster.

Si decide especificar las zonas de disponibilidad, para cada clúster en cada fragmento, elija la zona de disponibilidad de la lista.

Para obtener más información, consulte [Elección de regiones y zonas de disponibilidad](#).

5. Elija Next (Siguiente).
6. En la configuración avanzada de Redis OSS
 - Para Security (Seguridad):
 - i. Para cifrar sus datos, tiene las siguientes opciones:
 - Encryption at rest (Cifrado en reposo): permite el cifrado de los datos almacenados en el disco. Para obtener más información, consulte [Cifrado en reposo](#).

 Note

Tiene la opción de proporcionar una clave de cifrado diferente; para ello, seleccione la clave AWS KMS gestionada por el cliente y elija la clave. Para obtener más información, consulte [Uso de claves de AWS KMS administradas por el cliente](#).

- Encryption in-transit (Cifrado en tránsito): permite el cifrado de datos del cable. Para obtener más información, consulte [Cifrado en tránsito](#). Para la versión 6.0 y superior del motor OSS de Redis, si habilita el cifrado en tránsito, se le pedirá que especifique una de las siguientes opciones de control de acceso:
 - No Access Control (Sin control de acceso): este es el valor predeterminado. Esto indica que no hay restricciones en el acceso de los usuarios al clúster.

- User Group Access Control List (Lista de control de acceso del grupo de usuarios): seleccione un grupo de usuarios con un conjunto definido de usuarios que pueden acceder al clúster. Para obtener más información, consulte [Administración de grupos de usuarios con la consola y la CLI](#).
- Usuario predeterminado de AUTH de Redis OSS: mecanismo de autenticación para el servidor OSS de Redis. Para obtener más información, consulte [Redis OSS AUTH](#).
- Redis OSS AUTH: un mecanismo de autenticación para el servidor Redis OSS. Para obtener más información, consulte [Redis OSS AUTH](#).

 Note

Para las versiones de Redis OSS desde la 3.2.6 en adelante, excepto la versión 3.2.10, la única opción es Redis OSS AUTH.

- ii. En Grupos de seguridad, elija los grupos de seguridad que desea para este clúster. Un grupo de seguridad actúa como un firewall para controlar el acceso de red al clúster. Puede utilizar el grupo de seguridad predeterminado para la VPC o crear uno nuevo.

Para obtener más información sobre los grupos de seguridad, consulte [Grupos de seguridad de su VPC](#) en la Guía del usuario de Amazon VPC.
7. En las copias de seguridad automáticas programadas regularmente, seleccione Enable automatic backups (Habilitar copias de seguridad automáticas) y, a continuación, escriba el número de días que desea conservar cada copia de seguridad automática antes de que se elimine automáticamente. Si no quiere tener copias de seguridad automáticas programadas regularmente, desactive la casilla de verificación Enable automatic backups (Habilitar copias de seguridad automáticas). En cualquier caso, siempre tendrá la opción de crear copias de seguridad manuales.

Para obtener más información sobre la copia de seguridad y la restauración de Redis OSS, consulte [Instantánea y restauración](#)

8. (Opcional) Especifique un periodo de mantenimiento. El período de mantenimiento es el tiempo, que suele durar una hora cada semana, en el que se ElastiCache programa el mantenimiento del sistema del clúster. Puede ElastiCache elegir el día y la hora de su período de mantenimiento (sin preferencia) o puede elegir el día, la hora y la duración

usted mismo (especifique el período de mantenimiento). Si elige Specify maintenance window, elija Start day, Start time y Duration (en horas) de las listas para el periodo de mantenimiento. Todas las horas se indican en UCT.

Para obtener más información, consulte [Administración del mantenimiento](#).

9. (Opcional) Para Logs (Registros):

- En Log format (Formato de registro), elija Text o JSON.
- En Tipo de destino, elija CloudWatch Logs o Kinesis Firehose.
- En Destino del registro, elija Crear nuevo e introduzca el nombre del grupo de CloudWatch registros de Logs o el nombre de la transmisión de Firehose, o bien elija Seleccionar existente y, a continuación, elija el nombre del grupo de CloudWatch registros de Logs o el nombre de la transmisión de Firehose,

10. En el caso de las etiquetas, para ayudarte a gestionar tus clústeres y otros ElastiCache recursos, puedes asignar tus propios metadatos a cada recurso en forma de etiquetas. Para obtener más información, consulte [Etiquetado de los recursos de ElastiCache](#).

11. Elija Siguiente.

12. Revise todas las entradas y opciones y, a continuación, realice todos los cambios necesarios. Cuando haya terminado, elija Create (Crear).

On premises

1. Para On premises (En las instalaciones), le recomendamos que deje Auto-failover (Conmutación por error automática) habilitada. Para obtener más información, consulte [Minimizar el tiempo de inactividad en ElastiCache \(Redis OSS\) con Multi-AZ](#)
2. Siga los pasos de [Uso de Outposts](#).

Para crear el equivalente mediante la ElastiCache API o AWS CLI en lugar de la ElastiCache consola, consulte lo siguiente:

- API: [CreateReplicationGroup](#)
- CLI: [create-replication-group](#)

En cuanto el estado de su clúster sea available, podrá conceder a EC2 acceso a este, así como conectarse a él y comenzar a utilizarlo. Para obtener más información, consulte [Paso 3: autorizar acceso al clúster](#) y [Paso 4: conectarse al nodo de un clúster](#).

 Important

Cuando su clúster esté disponible, se cobrará por cada hora u hora parcial que el clúster esté activo, incluso si no lo está utilizando de forma activa. Para dejar de incurrir en cargos por este clúster, debe eliminarlo. Consulte [Eliminación de un clúster](#).

Creación de un clúster (AWS CLI)

Para crear un clúster mediante AWS CLI, utilice el `create-cache-cluster` comando.

Important

Cuando su clúster esté disponible, se cobrará por cada hora u hora parcial que el clúster esté activo, incluso si no lo está utilizando de forma activa. Para dejar de incurrir en cargos por este clúster, debe eliminarlo. Consulte [Eliminación de un clúster](#).

Creación de un clúster (CLI) de Redis OSS (modo de clúster desactivado)

Example — Un clúster de Redis OSS (modo de clúster desactivado) sin réplicas de lectura

El siguiente código CLI crea un clúster de caché de Redis OSS (modo de clúster desactivado) sin réplicas.

Note

Al crear clúster utilizando un tipo de nodo de la familia `r6gd`, debe pasar el parámetro `data-tiering-enabled`.

Para Linux, macOS o Unix:

```
aws elasticache create-cache-cluster \  
--cache-cluster-id my-cluster \  
--cache-node-type cache.r4.large \  
--engine redis \  
--num-cache-nodes 1 \  
--cache-parameter-group default.redis6.x \  
--snapshot-arns arn:aws:s3:::my_bucket/snapshot.rdb
```

Para Windows:

```
aws elasticache create-cache-cluster ^  
--cache-cluster-id my-cluster ^  
--cache-node-type cache.r4.large ^  
--engine redis ^
```

```
--num-cache-nodes 1 ^  
--cache-parameter-group default.redis6.x ^  
--snapshot-arns arn:aws:s3:::my_bucket/snapshot.rdb
```

Crear un clúster de Redis OSS (modo de clúster habilitado) (AWS CLI)

Los clústeres de Redis OSS (modo de clúster habilitado) (API/CLI: grupos de replicación) no se pueden crear mediante esta operación. `create-cache-cluster` Para crear un clúster de Redis OSS (modo de clúster habilitado) (API/CLI: grupo de replicación), consulte [Crear un grupo de replicación de Redis OSS \(modo de clúster activado\) desde cero \(AWS CLI\)](#)

Para obtener más información, consulte el tema de referencia AWS CLI . ElastiCache [create-replication-group](#)

Crear un clúster (ElastiCache API)

Para crear un clúster mediante la ElastiCache API, usa la `CreateCacheCluster` acción.

Important

Cuando su clúster esté disponible, se le cobrará por cada hora u hora parcial que el clúster esté activo, incluso si no lo está utilizando. Para dejar de incurrir en cargos por este clúster, debe eliminarlo. Consulte [Eliminación de un clúster](#).

Temas

- [Crear un clúster de caché \(ElastiCache API\) de Redis OSS \(modo de clúster desactivado\)](#)
- [Crear un clúster de caché en Redis OSS \(modo de clúster activado\) \(ElastiCache API\)](#)

Crear un clúster de caché (ElastiCache API) de Redis OSS (modo de clúster desactivado)

El siguiente código crea un clúster de caché (ElastiCache API) de Redis OSS (modo de clúster desactivado).

Se agregan saltos de línea para facilitar la lectura.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=CreateCacheCluster  
&CacheClusterId=my-cluster
```

```
&CacheNodeType=cache.r4.large
&CacheParameterGroup=default.redis3.2
&Engine=redis
&EngineVersion=3.2.4
&NumCacheNodes=1
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&SnapshotArns.member.1=arn%3Aaws%3As3%3A%3A%3AmyS3Bucket%2Fdump.rdb
&Timestamp=20150508T220302Z
&Version=2015-02-02
&X-Amz-Algorithm=&AWS;4-HMAC-SHA256
&X-Amz-Credential=<credential>
&X-Amz-Date=20150508T220302Z
&X-Amz-Expires=20150508T220302Z
&X-Amz-SignedHeaders=Host
&X-Amz-Signature=<signature>
```

Crear un clúster de caché en Redis OSS (modo de clúster activado) (ElastiCache API)

Los clústeres de Redis OSS (modo de clúster habilitado) (API/CLI: grupos de replicación) no se pueden crear mediante esta operación. `CreateCacheCluster` Para crear un clúster de Redis OSS (modo de clúster habilitado) (API/CLI: grupo de replicación), consulte [Crear un grupo de replicación en Redis OSS \(modo de clúster activado\) desde cero \(ElastiCache API\)](#)

Para obtener más información, consulte el tema de referencia de la ElastiCache API. [CreateReplicationGroup](#)

Visualización de los detalles de un clúster

Puede ver información detallada sobre uno o más clústeres mediante la ElastiCache consola o la ElastiCache API. AWS CLI

Visualización de los detalles de un clúster de Redis OSS (modo de clúster desactivado) (consola)

Puede ver los detalles de un clúster de Redis OSS (modo de clúster desactivado) mediante la ElastiCache consola, el AWS CLI formulario ElastiCache o la ElastiCache API.

El siguiente procedimiento detalla cómo ver los detalles de un clúster de Redis OSS (modo de clúster desactivado) mediante la ElastiCache consola.

Para ver los detalles de un clúster de Redis OSS (modo de clúster desactivado)

1. Inicia sesión en la ElastiCache consola de Amazon AWS Management Console y ábrela en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de la ElastiCache consola, selecciona Redis OSS para ver una lista de todos los clústeres que ejecutan cualquier versión de Redis OSS.
3. Para ver los detalles de un clúster, seleccione la casilla de verificación situada a la izquierda del nombre del clúster. Asegúrese de seleccionar un clúster que ejecute el motor Redis OSS, no un Redis OSS agrupado. Al hacerlo se muestran los detalles sobre el clúster, incluido su punto de conexión principal.
4. Para ver la información de los nodos:
 - a. Elija el nombre del clúster.
 - b. Elija la pestaña Shards and nodes (Particiones y nodos). Se mostrarán los detalles de cada nodo, incluido el punto de conexión del nodo que tiene que usar para leer del clúster.
5. Para ver las métricas, elija la pestaña Metrics (Métricas), que muestra las métricas pertinentes de todos los nodos del clúster. Para obtener más información, consulte [Monitoreo del uso con métricas de CloudWatch](#).
6. Para ver los registros, elija la pestaña Logs (Registros), que indica si el clúster utiliza registros lentos o registros del motor y proporciona los detalles pertinentes. Para obtener más información, consulte [Entrega de registro](#).
7. Elija la pestaña Network and security (Red y seguridad) para ver los detalles sobre la conectividad de red del clúster y la configuración del grupo de subredes. Para obtener más información, consulte [Subredes y grupos de subredes](#).

8. Elija la pestaña Maintenance (Mantenimiento) para ver los detalles en la configuración de mantenimiento del clúster. Para obtener más información, consulte [Administración del mantenimiento](#).
9. Elija la pestaña Service updates (Actualizaciones de servicio) para ver los detalles de las actualizaciones de servicio disponibles junto con la fecha de caducidad recomendada. Para obtener más información, consulte [Actualizaciones de servicios en ElastiCache](#).
10. Elija la pestaña Tags (Etiquetas) para ver los detalles de las etiquetas aplicadas a los recursos del clúster. Para obtener más información, consulte [Etiquetado de los recursos de ElastiCache](#).

Visualización de los detalles de un clúster de Redis OSS (modo de clúster activado) (consola)

Puede ver los detalles de un clúster OSS (modo de clúster habilitado) de Redis mediante la ElastiCache consola, el formulario AWS CLI o la ElastiCache API. ElastiCache

El siguiente procedimiento detalla cómo ver los detalles de un clúster de Redis OSS (modo de clúster habilitado) mediante la ElastiCache consola.

Para ver los detalles de un clúster de Redis OSS (modo de clúster activado)

1. Inicia sesión en la ElastiCache consola de Amazon AWS Management Console y ábrela en <https://console.aws.amazon.com/elasticache/>.
2. En la lista de la esquina superior derecha, selecciona la AWS región que te interese.
3. En el panel de la ElastiCache consola, elija Redis OSS para ver una lista de todos los clústeres que ejecutan cualquier versión de Redis OSS.
4. Para ver los detalles de un clúster de Redis OSS (modo de clúster activado), seleccione la casilla situada a la izquierda del nombre del clúster. Asegúrese de elegir un clúster que ejecute el motor de OSS de Redis agrupado, no solo de Redis OSS.

La pantalla se ampliará para mostrar el clúster y los detalles sobre el clúster, incluido el punto de conexión de configuración del clúster.

5. Para ver una lista de las particiones del clúster y el número de nodos en cada partición, elija la pestaña Shards and nodes (Particiones y nodos).
6. Para ver información específica sobre un nodo:
 - Elija el ID del fragmento.

Al hacerlo, se muestra información sobre cada nodo, incluido el punto de conexión de cada nodo que debe utilizar para leer datos del clúster.

7. Para ver las métricas, elija la pestaña Metrics (Métricas), que muestra las métricas pertinentes de todos los nodos del clúster. Para obtener más información, consulte [Monitoreo del uso con métricas de CloudWatch](#).
8. Para ver los registros, elija la pestaña Logs (Registros), que indica si el clúster utiliza registros lentos o registros del motor y proporciona los detalles pertinentes. Para obtener más información, consulte [Entrega de registro](#).
9. Elija la pestaña Network and security (Red y seguridad) para ver los detalles sobre la configuración de la conectividad de red y el grupo de subredes del clúster, el grupo de seguridad de la VPC y qué, si lo hay, método de cifrado está habilitado en el clúster. Para obtener más información, consulte [Subredes y grupos de subredes](#) y [Seguridad de datos en Amazon ElastiCache](#).
10. Elija la pestaña Maintenance (Mantenimiento) para ver los detalles en la configuración de mantenimiento del clúster. Para obtener más información, consulte [Administración del mantenimiento](#).
11. Elija la pestaña Service updates (Actualizaciones de servicio) para ver los detalles de las actualizaciones de servicio disponibles junto con la fecha de caducidad recomendada. Para obtener más información, consulte [Actualizaciones de servicios en ElastiCache](#).
12. Elija la pestaña Tags (Etiquetas) para ver los detalles de las etiquetas aplicadas a los recursos del clúster. Para obtener más información, consulte [Etiquetado de los recursos de ElastiCache](#).

Visualización de los detalles de un clúster (AWS CLI)

El siguiente código muestra los detalles para *my-cluster*:

```
aws elasticache describe-cache-clusters --cache-cluster-id my-cluster
```

Sustituya *my-cluster* por el nombre del clúster en caso de que el clúster se cree con 1 nodo de caché y 0 particiones mediante el comando `create-cache-cluster`.

```
{
  "CacheClusters": [
    {
      "CacheClusterStatus": "available",
      "SecurityGroups": [
```

```

        {
            "Status": "active",
            "SecurityGroupId": "sg-dbe93fa2"
        }
    ],
    "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
    "Engine": "redis",
    "PreferredMaintenanceWindow": "wed:12:00-wed:13:00",
    "CacheSubnetGroupName": "default",
    "SnapshotWindow": "08:30-09:30",
    "TransitEncryptionEnabled": false,
    "AtRestEncryptionEnabled": false,
    "CacheClusterId": "my-cluster1",
    "CacheClusterCreateTime": "2018-02-26T21:06:43.420Z",
    "PreferredAvailabilityZone": "us-west-2c",
    "AuthTokenEnabled": false,
    "PendingModifiedValues": {},
    "CacheNodeType": "cache.r4.large",
    "DataTiering": "disabled",
    "CacheParameterGroup": {
        "CacheNodeIdsToReboot": [],
        "ParameterApplyStatus": "in-sync",
        "CacheParameterGroupName": "default.redis3.2"
    },
    "SnapshotRetentionLimit": 0,
    "AutoMinorVersionUpgrade": true,
    "EngineVersion": "3.2.10",
    "CacheSecurityGroups": [],
    "NumCacheNodes": 1
}

```

```

{
    "CacheClusters": [
        {
            "SecurityGroups": [
                {
                    "Status": "active",
                    "SecurityGroupId": "sg-dbe93fa2"
                }
            ],
            "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",

```

```

    "AuthTokenEnabled": false,
    "CacheSubnetGroupName": "default",
    "SnapshotWindow": "12:30-13:30",
    "AutoMinorVersionUpgrade": true,
    "CacheClusterCreateTime": "2018-02-26T21:13:24.250Z",
    "CacheClusterStatus": "available",
    "AtRestEncryptionEnabled": false,
    "PreferredAvailabilityZone": "us-west-2a",
    "TransitEncryptionEnabled": false,
    "ReplicationGroupId": "my-cluster2",
    "Engine": "redis",
    "PreferredMaintenanceWindow": "sun:08:30-sun:09:30",
    "CacheClusterId": "my-cluster2-001",
    "PendingModifiedValues": {},
    "CacheNodeType": "cache.r4.large",
    "DataTiering": "disabled",
    "CacheParameterGroup": {
      "CacheNodeIdsToReboot": [],
      "ParameterApplyStatus": "in-sync",
      "CacheParameterGroupName": "default.redis6.x"
    },
    "SnapshotRetentionLimit": 0,
    "EngineVersion": "6.0",
    "CacheSecurityGroups": [],
    "NumCacheNodes": 1
  },
  {
    "SecurityGroups": [
      {
        "Status": "active",
        "SecurityGroupId": "sg-dbe93fa2"
      }
    ],
    "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/home#client-download:",
    "AuthTokenEnabled": false,
    "CacheSubnetGroupName": "default",
    "SnapshotWindow": "12:30-13:30",
    "AutoMinorVersionUpgrade": true,
    "CacheClusterCreateTime": "2018-02-26T21:13:24.250Z",
    "CacheClusterStatus": "available",
    "AtRestEncryptionEnabled": false,
    "PreferredAvailabilityZone": "us-west-2b",
    "TransitEncryptionEnabled": false,

```

```

    "ReplicationGroupId": "my-cluster2",
    "Engine": "redis",
    "PreferredMaintenanceWindow": "sun:08:30-sun:09:30",
    "CacheClusterId": "my-cluster2-002",
    "PendingModifiedValues": {},
    "CacheNodeType": "cache.r4.large",
    "DataTiering": "disabled",
    "CacheParameterGroup": {
      "CacheNodeIdsToReboot": [],
      "ParameterApplyStatus": "in-sync",
      "CacheParameterGroupName": "default.redis6.x"
    },
    "SnapshotRetentionLimit": 0,
    "EngineVersion": "6.0",
    "CacheSecurityGroups": [],
    "NumCacheNodes": 1
  },
  {
    "SecurityGroups": [
      {
        "Status": "active",
        "SecurityGroupId": "sg-dbe93fa2"
      }
    ],
    "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/home#client-download:",
    "AuthTokenEnabled": false,
    "CacheSubnetGroupName": "default",
    "SnapshotWindow": "12:30-13:30",
    "AutoMinorVersionUpgrade": true,
    "CacheClusterCreateTime": "2018-02-26T21:13:24.250Z",
    "CacheClusterStatus": "available",
    "AtRestEncryptionEnabled": false,
    "PreferredAvailabilityZone": "us-west-2c",
    "TransitEncryptionEnabled": false,
    "ReplicationGroupId": "my-cluster2",
    "Engine": "redis",
    "PreferredMaintenanceWindow": "sun:08:30-sun:09:30",
    "CacheClusterId": "my-cluster2-003",
    "PendingModifiedValues": {},
    "CacheNodeType": "cache.r4.large",
    "DataTiering": "disabled",
    "CacheParameterGroup": {
      "CacheNodeIdsToReboot": [],

```

```

        "ParameterApplyStatus": "in-sync",
        "CacheParameterGroupName": "default.redis3.2"
    },
    "SnapshotRetentionLimit": 0,
    "EngineVersion": "3.2.10",
    "CacheSecurityGroups": [],
    "NumCacheNodes": 1
}

```

```

{
  "CacheClusters": [
    {
      "SecurityGroups": [
        {
          "Status": "active",
          "SecurityGroupId": "sg-dbe93fa2"
        }
      ],
      "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/home#client-download:",
      "AuthTokenEnabled": true,
      "CacheSubnetGroupName": "default",
      "SnapshotWindow": "12:30-13:30",
      "AutoMinorVersionUpgrade": true,
      "CacheClusterCreateTime": "2018-02-26T21:17:01.439Z",
      "CacheClusterStatus": "available",
      "AtRestEncryptionEnabled": true,
      "PreferredAvailabilityZone": "us-west-2a",
      "TransitEncryptionEnabled": true,
      "ReplicationGroupId": "my-cluster3",
      "Engine": "redis",
      "PreferredMaintenanceWindow": "thu:11:00-thu:12:00",
      "CacheClusterId": "my-cluster3-0001-001",
      "PendingModifiedValues": {},
      "CacheNodeType": "cache.r4.large",
      "DataTiering": "disabled",
      "CacheParameterGroup": {
        "CacheNodeIdsToReboot": [],
        "ParameterApplyStatus": "in-sync",
        "CacheParameterGroupName": "default.redis6.x.cluster.on"
      },
      "SnapshotRetentionLimit": 0,
      "EngineVersion": "6.0",
    }
  ]
}

```

```

    "CacheSecurityGroups": [],
    "NumCacheNodes": 1
  },
  {
    "SecurityGroups": [
      {
        "Status": "active",
        "SecurityGroupId": "sg-dbe93fa2"
      }
    ],
    "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
    "AuthTokenEnabled": true,
    "CacheSubnetGroupName": "default",
    "SnapshotWindow": "12:30-13:30",
    "AutoMinorVersionUpgrade": true,
    "CacheClusterCreateTime": "2018-02-26T21:17:01.439Z",
    "CacheClusterStatus": "available",
    "AtRestEncryptionEnabled": true,
    "PreferredAvailabilityZone": "us-west-2b",
    "TransitEncryptionEnabled": true,
    "ReplicationGroupId": "my-cluster3",
    "Engine": "redis",
    "PreferredMaintenanceWindow": "thu:11:00-thu:12:00",
    "CacheClusterId": "my-cluster3-0001-002",
    "PendingModifiedValues": {},
    "CacheNodeType": "cache.r4.large",
    "DataTiering": "disabled",
    "CacheParameterGroup": {
      "CacheNodeIdsToReboot": [],
      "ParameterApplyStatus": "in-sync",
      "CacheParameterGroupName": "default.redis3.2.cluster.on"
    },
    "SnapshotRetentionLimit": 0,
    "EngineVersion": "3.2.6",
    "CacheSecurityGroups": [],
    "NumCacheNodes": 1
  },
  {
    "SecurityGroups": [
      {
        "Status": "active",
        "SecurityGroupId": "sg-dbe93fa2"
      }
    ]
  }

```

```

    ],
    "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
    "AuthTokenEnabled": true,
    "CacheSubnetGroupName": "default",
    "SnapshotWindow": "12:30-13:30",
    "AutoMinorVersionUpgrade": true,
    "CacheClusterCreateTime": "2018-02-26T21:17:01.439Z",
    "CacheClusterStatus": "available",
    "AtRestEncryptionEnabled": true,
    "PreferredAvailabilityZone": "us-west-2c",
    "TransitEncryptionEnabled": true,
    "ReplicationGroupId": "my-cluster3",
    "Engine": "redis",
    "PreferredMaintenanceWindow": "thu:11:00-thu:12:00",
    "CacheClusterId": "my-cluster3-0001-003",
    "PendingModifiedValues": {},
    "CacheNodeType": "cache.r4.large",
    "DataTiering": "disabled",
    "CacheParameterGroup": {
      "CacheNodeIdsToReboot": [],
      "ParameterApplyStatus": "in-sync",
      "CacheParameterGroupName": "default.redis6.x.cluster.on"
    },
    "SnapshotRetentionLimit": 0,
    "EngineVersion": "6.0",
    "CacheSecurityGroups": [],
    "NumCacheNodes": 1
  },
  {
    "SecurityGroups": [
      {
        "Status": "active",
        "SecurityGroupId": "sg-dbe93fa2"
      }
    ],
    "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
    "AuthTokenEnabled": true,
    "CacheSubnetGroupName": "default",
    "SnapshotWindow": "12:30-13:30",
    "AutoMinorVersionUpgrade": true,
    "CacheClusterCreateTime": "2018-02-26T21:17:01.439Z",
    "CacheClusterStatus": "available",

```

```

    "AtRestEncryptionEnabled": true,
    "PreferredAvailabilityZone": "us-west-2b",
    "TransitEncryptionEnabled": true,
    "ReplicationGroupId": "my-cluster3",
    "Engine": "redis",
    "PreferredMaintenanceWindow": "thu:11:00-thu:12:00",
    "CacheClusterId": "my-cluster3-0002-001",
    "PendingModifiedValues": {},
    "CacheNodeType": "cache.r4.large",
    "DataTiering": "disabled",
    "CacheParameterGroup": {
      "CacheNodeIdsToReboot": [],
      "ParameterApplyStatus": "in-sync",
      "CacheParameterGroupName": "default.redis6.x.cluster.on"
    },
    "SnapshotRetentionLimit": 0,
    "EngineVersion": "6.0",
    "CacheSecurityGroups": [],
    "NumCacheNodes": 1
  },
  {
    "SecurityGroups": [
      {
        "Status": "active",
        "SecurityGroupId": "sg-dbe93fa2"
      }
    ],
    "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/home#client-download:",
    "AuthTokenEnabled": true,
    "CacheSubnetGroupName": "default",
    "SnapshotWindow": "12:30-13:30",
    "AutoMinorVersionUpgrade": true,
    "CacheClusterCreateTime": "2018-02-26T21:17:01.439Z",
    "CacheClusterStatus": "available",
    "AtRestEncryptionEnabled": true,
    "PreferredAvailabilityZone": "us-west-2c",
    "TransitEncryptionEnabled": true,
    "ReplicationGroupId": "my-cluster3",
    "Engine": "redis",
    "PreferredMaintenanceWindow": "thu:11:00-thu:12:00",
    "CacheClusterId": "my-cluster3-0002-002",
    "PendingModifiedValues": {},
    "CacheNodeType": "cache.r4.large",

```

```

    "DataTiering": "disabled",
    "CacheParameterGroup": {
      "CacheNodeIdsToReboot": [],
      "ParameterApplyStatus": "in-sync",
      "CacheParameterGroupName": "default.redis3.2.cluster.on"
    },
    "SnapshotRetentionLimit": 0,
    "EngineVersion": "3.2.6",
    "CacheSecurityGroups": [],
    "NumCacheNodes": 1
  },
  {
    "SecurityGroups": [
      {
        "Status": "active",
        "SecurityGroupId": "sg-dbe93fa2"
      }
    ],
    "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/home#client-download:",
    "AuthTokenEnabled": true,
    "CacheSubnetGroupName": "default",
    "SnapshotWindow": "12:30-13:30",
    "AutoMinorVersionUpgrade": true,
    "CacheClusterCreateTime": "2018-02-26T21:17:01.439Z",
    "CacheClusterStatus": "available",
    "AtRestEncryptionEnabled": true,
    "PreferredAvailabilityZone": "us-west-2a",
    "TransitEncryptionEnabled": true,
    "ReplicationGroupId": "my-cluster3",
    "Engine": "redis",
    "PreferredMaintenanceWindow": "thu:11:00-thu:12:00",
    "CacheClusterId": "my-cluster3-0002-003",
    "PendingModifiedValues": {},
    "CacheNodeType": "cache.r4.large",
    "DataTiering": "disabled",
    "CacheParameterGroup": {
      "CacheNodeIdsToReboot": [],
      "ParameterApplyStatus": "in-sync",
      "CacheParameterGroupName": "default.redis6.x.cluster.on"
    },
    "SnapshotRetentionLimit": 0,
    "EngineVersion": "6.0",
    "CacheSecurityGroups": [],

```

```
        "NumCacheNodes": 1
    }
]
}
```

En el caso de que el clúster se cree mediante AWS Management Console (el nodo del clúster está habilitado o deshabilitado con 1 o más fragmentos), utilice el siguiente comando para describir los detalles del clúster (sustituya *my-cluster* por el nombre del grupo de replicación (nombre de su clúster)):

```
aws elasticache describe-replication-groups --replication-group-id my-cluster
```

Para obtener más información, consulte el tema AWS CLI para ElastiCache . [describe-cache-clusters](#)

Visualización de los detalles de un clúster (ElastiCache API)

Puedes ver los detalles de un clúster mediante la DescribeCacheClusters acción de la ElastiCache API. Si se incluye el parámetro CacheClusterId, se devolverán detalles del clúster especificado. Si el parámetro CacheClusterId se omite, se devolverán los detalles de hasta MaxRecords clústeres (el valor predeterminado es 100). El valor de MaxRecords no puede ser inferior a 20 ni superior a 100.

El siguiente código enumera los detalles de my-cluster.

```
https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeCacheClusters
&CacheClusterId=my-cluster
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>
```

El siguiente código enumera los detalles de hasta 25 clústeres.

```
https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeCacheClusters
&MaxRecords=25
&Version=2015-02-02
```

```
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20150202T192317Z  
&X-Amz-Credential=<credential>
```

Para obtener más información, consulta el tema de referencia de la ElastiCache API

[DescribeCacheClusters](#).

Modificación de un ElastiCache clúster

Además de agregar o quitar nodos de un clúster, puede que haya veces en las que necesita realizar otros cambios en un clúster existente, como, por ejemplo, al agregar un grupo de seguridad o al cambiar el periodo de mantenimiento o un grupo de parámetros.

Recomendamos que el periodo de mantenimiento corresponda al momento de mínimo uso. Esto puede requerir alguna modificación de vez en cuando.

Cuando se modifican los parámetros de un clúster, el cambio se aplica al clúster inmediatamente o después de reiniciarlo. Esto es cierto tanto si se modifica el propio grupo de parámetros del clúster como si se modifica el valor de un parámetro del grupo. Para determinar cuándo se aplica un cambio de parámetros determinado, consulte la sección Aplicación de los cambios de la columna Detalles en las tablas para [Parámetros específicos de Redis OSS](#).

Uso del AWS Management Console

Pasos para modificar un clúster

1. Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en <https://console.aws.amazon.com/elasticache/>.
2. En la lista de la esquina superior derecha, elija la AWS región en la que se encuentra el clúster que desea modificar.
3. En el panel de navegación, elija el motor que se ejecuta en el clúster que desea modificar.

Se mostrará una lista de los clústeres del motor elegido.

4. En la lista de clústeres, para el clúster que desea modificar, elija su nombre.
5. Elija Actions (Acciones) y después Modify (Modificar).

Aparecerá la ventana Modify Cluster.

6. En la ventana Modificar clúster, haga las modificaciones que desee. Las opciones son:

- Descripción
- Modo de clúster: Para modificar el modo de clúster de desactivado a habilitado, primero debe establecer el modo de clúster en compatible.

El modo compatible permite que sus clientes OSS de Redis se conecten utilizando el modo de clúster activado y el modo de clúster desactivado. Después de migrar todos los clientes OSS

de Redis para que utilicen el modo de clúster activado, podrá completar la configuración del modo de clúster y establecerlo en Activado.

- Compatibilidad de la versión del motor

Important

Puede actualizar a versiones de motor más recientes. Si actualiza las versiones principales del motor, por ejemplo de 5.0.6 a 6.0, debe seleccionar una familia de grupos de parámetros que sea compatible con la versión del motor nueva. Para obtener más información sobre cómo hacerlo, consulte [Versiones del motor y actualizaciones](#). Sin embargo, no se puede volver a versiones del motor anteriores, salvo si se elimina el clúster existente y se vuelve a crear.

- Grupos de seguridad de VPC.
- Parameter Group
- Node Type

Note

Si el clúster utiliza un tipo de nodo de la familia r6gd, solo puede elegir un tamaño de nodo diferente dentro de esa familia. Si elige un tipo de nodo de la familia r6gd, la organización de datos en niveles se activará automáticamente. Para obtener más información, consulte [Organización de datos en niveles](#).

- Multi-AZ
- Conmutación por error automática (solo modo de clúster deshabilitado)
- Habilitar copias de seguridad automáticas
- ID del nodo de copia de seguridad
- Período de retención de backup
- Periodo de copia de seguridad
- Tema para la notificación de SNS

El cuadro Apply Immediately (Aplicar inmediatamente) solo se aplica a las modificaciones en la versión del motor. Para aplicar los cambios inmediatamente, active la casilla Apply Immediately (Aplicar inmediatamente). Si no se elige esta casilla, las modificaciones en el tipo

de nodo y la versión del motor se aplicarán durante el próximo periodo de mantenimiento. Otras modificaciones, como el cambio del periodo de mantenimiento, se aplican inmediatamente.

7. Elija Modificar.

Para habilitar/deshabilitar la entrega de registros

1. En la lista de clústeres, elija el clúster que desea modificar. Elija el Cluster name (Nombre del clúster) y no la casilla de verificación al lado.
2. En la página Cluster Details (Detalles del clúster), seleccione la pestaña Logs (Registros).
3. Para habilitar/deshabilitar los registros lentos, elija Enable (Habilitar) o Disable (Deshabilitar).

Si elige habilitar:

- a. En Log format (Formato de registro), elija Text (Texto) o JSON.
- b. En Tipo de destino del registro, elija CloudWatch Logs o Kinesis Firehose.
- c. En Destino del registro, elija Crear nuevo e introduzca el nombre del grupo de CloudWatchLogs registros o el nombre de la transmisión de Kinesis Data Firehose. O bien, elija Seleccionar existente y, a continuación, elija el nombre del grupo de CloudWatchLogs registros o el nombre de la transmisión de Kinesis Data Firehose.
- d. Seleccione Habilitar.

Para cambiar su configuración:

1. Elija Modificar.
2. En Log format (Formato de registro), elija Text (Texto) o JSON.
3. En Tipo de destino, elija CloudWatch Logs o Kinesis Firehose.
4. En Destino del registro, elija Crear nuevo e introduzca el nombre del grupo de CloudWatchLogs registros o el nombre de la transmisión de Kinesis Data Firehose. O bien, elija Seleccionar existente y, a continuación, elija el nombre del grupo de CloudWatchLogs registros o el nombre de la transmisión de Kinesis Data Firehose.

Usando el AWS CLI

Puede modificar un clúster existente mediante la AWS CLI `modify-cache-cluster` operación. Para modificar un valor de configuración de un clúster, especifique el ID del clúster, el parámetro

que desea cambiar y el nuevo valor del parámetro. El siguiente ejemplo cambia el periodo de mantenimiento de un clúster denominado `my-cluster` y aplica el cambio inmediatamente.

Important

Puede actualizar a versiones de motor más recientes. Si actualiza las versiones principales del motor, por ejemplo de 5.0.6 a 6.0, debe seleccionar una familia de grupos de parámetros que sea compatible con la versión del motor nueva. Para obtener más información sobre cómo hacerlo, consulte [Versiones del motor y actualizaciones](#). Sin embargo, no se puede volver a versiones del motor anteriores, salvo si se elimina el clúster existente y se vuelve a crear.

Para Linux, macOS o Unix:

```
aws elasticache modify-cache-cluster \  
  --cache-cluster-id my-cluster \  
  --preferred-maintenance-window sun:23:00-mon:02:00
```

Para Windows:

```
aws elasticache modify-cache-cluster ^  
  --cache-cluster-id my-cluster ^  
  --preferred-maintenance-window sun:23:00-mon:02:00
```

El parámetro `--apply-immediately` solo se aplica a modificaciones en el tipo de nodo, la versión del motor y el número de nodos de un clúster. Si desea aplicar cualquiera de estos cambios de forma inmediata, utilice el parámetro `--apply-immediately`. Si prefiere aplazar estos cambios al siguiente periodo de mantenimiento, utilice el parámetro `--no-apply-immediately`. Otras modificaciones, como el cambio del periodo de mantenimiento, se aplican inmediatamente.

Para obtener más información, consulte el ElastiCache tema AWS CLI para [modify-cache-cluster](#).

Uso de la ElastiCache API

Puede modificar un clúster existente mediante la `ModifyCacheCluster` operación de la ElastiCache API. Para modificar un valor de configuración de un clúster, especifique el ID del clúster, el parámetro que desea cambiar y el nuevo valor del parámetro. El siguiente ejemplo

cambia el periodo de mantenimiento de un clúster denominado `my-cluster` y aplica el cambio inmediatamente.

⚠ Important

Puede actualizar a versiones de motor más recientes. Si actualiza las versiones principales del motor, por ejemplo de 5.0.6 a 6.0, debe seleccionar una familia de grupos de parámetros que sea compatible con la versión del motor nueva. Para obtener más información sobre cómo hacerlo, consulte [Versiones del motor y actualizaciones](#). Sin embargo, no se puede volver a versiones del motor anteriores, salvo si se elimina el clúster existente y se vuelve a crear.

Se agregan saltos de línea para facilitar la lectura.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=ModifyCacheCluster  
&CacheClusterId=my-cluster  
&PreferredMaintenanceWindow=sun:23:00-mon:02:00  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20150901T220302Z  
&X-Amz-Algorithm=&AWS;4-HMAC-SHA256  
&X-Amz-Date=20150202T220302Z  
&X-Amz-SignedHeaders=Host  
&X-Amz-Expires=20150901T220302Z  
&X-Amz-Credential=<credential>  
&X-Amz-Signature=<signature>
```

El parámetro `ApplyImmediately` solo se aplica a modificaciones en el tipo de nodo, la versión del motor y el número de nodos de un clúster. Si desea aplicar cualquiera de estos cambios de forma inmediata, establezca el parámetro `ApplyImmediately` en `true`. Si prefiere aplazar estos cambios al siguiente periodo de mantenimiento, establezca el parámetro `ApplyImmediately` en `false`. Otras modificaciones, como el cambio del periodo de mantenimiento, se aplican inmediatamente.

Para obtener más información, consulta el tema de referencia de la ElastiCache API [ModifyCacheCluster](#).

Adición de nodos a un clúster

Para volver a configurar el clúster de Redis OSS (modo de clúster activado), consulte [Escalar clústeres en Redis OSS \(modo de clúster habilitado\)](#)

Puede usar la consola ElastiCache de administración, la ElastiCache API AWS CLI o la API para agregar nodos al clúster.

Uso de AWS Management Console

Si desea agregar un nodo a un clúster Redis OSS (modo de clúster desactivado) de un solo nodo (uno sin la replicación habilitada), se trata de un proceso de dos pasos: primero agregue la replicación y, a continuación, agregue un nodo de réplica.

Temas

- [Para agregar la replicación a un clúster de Redis OSS sin fragmentos](#)
- [Para agregar nodos a un clúster \(consola\)](#)

El siguiente procedimiento agrega la replicación a un Redis OSS de un solo nodo que no tiene habilitada la replicación. Al agregar la reproducción, el nodo existente se convierte en el nodo principal en el clúster con la reproducción habilitada. Tras agregar la reproducción, puede agregar hasta 5 nodos de réplica al clúster.

Para agregar la replicación a un clúster de Redis OSS sin fragmentos

1. [Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en https://console.aws.amazon.com/elasticache/.](https://console.aws.amazon.com/elasticache/)
2. En el panel de navegación, elija los clústeres de Redis OSS.

Aparece una lista de clústeres que ejecutan el motor OSS de Redis.

3. Elija el nombre de un clúster al que desea agregar nodos, pero no la casilla situada a la izquierda del nombre del clúster.

Lo siguiente es cierto en el caso de un clúster OSS de Redis que no tiene habilitada la replicación:

- Ejecuta Redis OSS, no Redis OSS agrupado.
- No tiene ningún fragmento.

Si el clúster tiene algún fragmento, la reproducción ya está habilitada y podrá continuar con [Para agregar nodos a un clúster \(consola\)](#).

4. Seleccione Add replication (Agregar reproducción).
5. En Agregar reproducción, escriba una descripción para este clúster con la reproducción habilitada.
6. Elija Añadir.

Cuando el estado del clúster vuelva a ser available, podrá continuar con el siguiente procedimiento y agregar réplicas al clúster.

Para agregar nodos a un clúster (consola)

El siguiente procedimiento se puede utilizar para agregar nodos a un clúster.

1. [Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en https://console.aws.amazon.com/elasticache/](https://console.aws.amazon.com/elasticache/).
2. En el panel de navegación, elija el motor que se ejecuta en el clúster en el que desea agregar nodos.

Se mostrará una lista de los clústeres que ejecutan el motor elegido.

3. En la lista de clústeres, para el clúster al que desea agregar un nodo, elija su nombre.

Si su clúster es un clúster OSS de Redis (modo de clúster activado), consulte [Escalar clústeres en Redis OSS \(modo de clúster habilitado\)](#).

Si su clúster es un clúster de Redis OSS (modo de clúster desactivado) sin fragmentos, complete primero los pasos que se indican en [Para agregar la replicación a un clúster de Redis OSS sin fragmentos](#)

4. Elija Add node (Agregar nodo).
5. Rellene la información que se solicita en el cuadro de diálogo Add Node (Agregar nodo).
6. Seleccione el botón Apply Immediately - Yes (Aplicar inmediatamente: sí) para aplicar este cambio de inmediato o seleccione No para aplazar el cambio hasta el próximo periodo de mantenimiento del clúster.

Impacto de las nuevas solicitudes de adición y eliminación en las solicitudes pendientes

Escenarios	Operación pendiente	Solicitud nueva	Resultados
Escenario 1	Delete	Delete	<p>La nueva solicitud de eliminación, pendiente o inmediata, reemplaza a la solicitud de eliminación pendiente.</p> <p>Por ejemplo, si los nodos 0001, 0003 y 0007 se encuentran pendientes de eliminación y se emite una solicitud nueva para eliminar los nodos 0002 y 0004, solo se eliminarán los nodos 0002 y 0004. Los nodos 0001, 0003 y 0007 no se eliminarán.</p>
Escenario 2	Delete	Creación	<p>La nueva solicitud de creación, pendiente o inmediata, reemplaza a la solicitud de eliminación pendiente.</p> <p>Por ejemplo, si los nodos 0001, 0003 y 0007 se encuentran pendientes de eliminación y se emite una solicitud nueva para crear un nodo, se creará un nodo nuevo y no se eliminarán los nodos 0001, 0003 y 0007.</p>
Escenario 3	Creación	Delete	<p>La nueva solicitud de eliminación, pendiente o inmediata, reemplaza a la solicitud de creación pendiente.</p> <p>Por ejemplo, si hay una solicitud pendiente para crear dos nodos y se emite una solicitud nueva para eliminar el nodo 0003, no se crearán nodos nuevos y se eliminará el nodo 0003.</p>

Escenarios	Operación pendiente	Solicitud nueva	Resultados
Escenario 4	Creación	Creación	<p>La nueva solicitud de creación se agrega a la solicitud de creación pendiente.</p> <p>Por ejemplo, si hay una solicitud pendiente para crear dos nodos y se emite una solicitud nueva para crear tres nodos, las solicitudes nuevas se agregarán a la solicitud pendiente y se crearán cinco nodos.</p> <div style="border: 1px solid #f08080; border-radius: 10px; padding: 10px; margin-top: 10px;"> <p>⚠ Important</p> <p>Si la nueva solicitud de creación se establece en Apply Immediately - Yes (Aplicar de inmediato: sí), todas las solicitudes de creación se ejecutan de inmediato. Si la nueva solicitud de creación se establece en Apply Immediately - No (Aplicar de inmediato: no), todas las solicitudes de creación quedarán pendientes.</p> </div>

Para determinar qué operaciones se encuentran pendientes, elija la pestaña de Description (Descripción) y verifique cuántas creaciones o eliminaciones pendientes se muestran. No puede tener creaciones y eliminaciones pendientes al mismo tiempo.

7. Seleccione el botón Add (Agregar).

Después de unos minutos, los nuevos nodos deberían aparecer en la lista de nodos con el estado creating. Si no aparecen, actualice página del navegador. Cuando el estado cambie a nodo disponible el nuevo nodo puede usarse.

Uso del AWS CLI

Si desea agregar nodos a un clúster OSS (modo de clúster desactivado) de Redis existente que no tiene habilitada la replicación, primero debe crear el grupo de replicación especificando el clúster

existente como principal. Para obtener más información, consulte [Crear un grupo de replicación mediante un clúster de caché de Redis OSS disponible \(AWS CLI\)](#). Una vez que el estado del grupo de replicación sea available, puede continuar con el siguiente proceso.

Para añadir nodos a un clúster mediante el AWS CLI, utilice la AWS CLI operación `increase-replica-count` con los siguientes parámetros:

- `--replication-group-id`: el ID del grupo de reproducción al que desea agregar nodos.
- `--new-replica-count` especifica el número de nodos que desea en este grupo de reproducción una vez que se aplica la modificación. Para añadir nodos a este clúster, el valor de `--new-replica-count` debe ser mayor que el número actual de nodos en este clúster.
- `--apply-immediately` o `--no-apply-immediately`, que especifica si desea añadir estos nodos de forma inmediata o durante el próximo periodo de mantenimiento.

Para Linux, macOS o Unix:

```
aws elasticache increase-replica-count \  
  --replication-group-id my-replication-group \  
  --new-replica-count 4 \  
  --apply-immediately
```

Para Windows:

```
aws elasticache increase-replica-count ^  
  --replication-group-id my-replication-group ^  
  --new-replica-count 4 ^  
  --apply-immediately
```

Esta operación produce un resultado similar al siguiente (formato JSON):

```
{  
  "ReplicationGroup": {  
    "ReplicationGroupId": "node-test",  
    "Description": "node-test",  
    "Status": "modifying",  
    "PendingModifiedValues": {},  
    "MemberClusters": [  
      "node-test-001",  
      "node-test-002",  
      "node-test-003",
```

```

        "node-test-004",
        "node-test-005"
    ],
    "NodeGroups": [
        {
            "NodeGroupId": "0001",
            "Status": "modifying",
            "PrimaryEndpoint": {
                "Address": "node-test.zzzzzz.ng.0001.usw2.cache.amazonaws.com",
                "Port": 6379
            },
            "ReaderEndpoint": {
                "Address": "node-test.zzzzzz.ng.0001.usw2.cache.amazonaws.com",
                "Port": 6379
            },
            "NodeGroupMembers": [
                {
                    "CacheClusterId": "node-test-001",
                    "CacheNodeId": "0001",
                    "ReadEndpoint": {
                        "Address": "node-
test-001.zzzzzz.0001.usw2.cache.amazonaws.com",
                        "Port": 6379
                    },
                    "PreferredAvailabilityZone": "us-west-2a",
                    "CurrentRole": "primary"
                },
                {
                    "CacheClusterId": "node-test-002",
                    "CacheNodeId": "0001",
                    "ReadEndpoint": {
                        "Address": "node-
test-002.zzzzzz.0001.usw2.cache.amazonaws.com",
                        "Port": 6379
                    },
                    "PreferredAvailabilityZone": "us-west-2c",
                    "CurrentRole": "replica"
                },
                {
                    "CacheClusterId": "node-test-003",
                    "CacheNodeId": "0001",
                    "ReadEndpoint": {
                        "Address": "node-
test-003.zzzzzz.0001.usw2.cache.amazonaws.com",

```

```

        "Port": 6379
      },
      "PreferredAvailabilityZone": "us-west-2b",
      "CurrentRole": "replica"
    }
  ]
}
],
"SnapshottingClusterId": "node-test-002",
"AutomaticFailover": "enabled",
"MultiAZ": "enabled",
"SnapshotRetentionLimit": 1,
"SnapshotWindow": "07:30-08:30",
"ClusterEnabled": false,
"CacheNodeType": "cache.r5.large",
  "DataTiering": "disabled",
"TransitEncryptionEnabled": false,
"AtRestEncryptionEnabled": false,
"ARN": "arn:aws:elasticache:us-west-2:123456789012:replicationgroup:node-test"
}
}

```

Para obtener más información, consulte el AWS CLI tema [increase-replica-count](#).

Uso de la ElastiCache API

Si desea añadir nodos a un clúster de Redis OSS (modo de clúster desactivado) existente que no tiene habilitada la replicación, primero debe crear el grupo de replicación especificando el clúster existente como principal. Para obtener más información, consulte [Añadir réplicas a un clúster \(API\) de Redis OSS \(modo de clúster desactivado\) independiente ElastiCache](#). Una vez que el estado del grupo de replicación sea available, puede continuar con el siguiente proceso.

Para añadir nodos a un clúster (ElastiCache API)

- Realice una llamada a la operación API `IncreaseReplicaCount` con los parámetros siguientes:
 - `ReplicationGroupId` El ID del clúster al que desea agregar nodos.
 - `NewReplicaCount` El parámetro `NewReplicaCount` especifica el número de nodos que desea en este clúster una vez aplicada la modificación. Para añadir nodos a este clúster, el valor de `NewReplicaCount` debe ser mayor que el número actual de nodos

en este clúster. Si este valor es menor que el número actual de nodos, utilice la API `DecreaseReplicaCount` con el número de nodos que se debe quitar del clúster.

- `ApplyImmediately` especifica si desea añadir estos nodos de forma inmediata o durante el próximo periodo de mantenimiento.
- `RegionEspecifica` la AWS región del clúster a la que desea agregar nodos.

En el siguiente ejemplo se muestra una llamada para añadir nodos a un clúster.

Example

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=IncreaseReplicaCount  
  &ApplyImmediately=true  
  &NumCacheNodes=4  
  &ReplicationGroupId=my-replication-group  
  &Region=us-east-2  
  &Version=2014-12-01  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Timestamp=20141201T220302Z  
  &X-Amz-Algorithm=&AWS;4-HMAC-SHA256  
  &X-Amz-Date=20141201T220302Z  
  &X-Amz-SignedHeaders=Host  
  &X-Amz-Expires=20141201T220302Z  
  &X-Amz-Credential=<credential>  
  &X-Amz-Signature=<signature>
```

Para obtener más información, consulte el tema ElastiCache API [IncreaseReplicaCount](#).

Eliminación de nodos de un clúster

Para eliminar un nodo del clúster, puede utilizar la AWS Management Console, la AWS CLI o la API de ElastiCache.

Utilización de la AWS Management Console

Para eliminar nodos de un clúster (consola)

1. Inicie sesión en la AWS Management Console y abra la consola de ElastiCache en <https://console.aws.amazon.com/elasticache/>.
2. En la lista de la esquina superior derecha, elija la región de AWS del clúster del que desea quitar nodos.
3. En el panel de navegación, elija el motor que se ejecuta en el clúster del que desea quitar un nodo.

Se mostrará una lista de los clústeres que ejecutan el motor elegido.

4. En la lista de clústeres, elija el nombre del clúster del que desea quitar un nodo.

Aparecerá una lista de nodos del clúster.

5. Elija la casilla situada a la izquierda del ID del nodo que desee quitar. Con la consola de ElastiCache, solo puede eliminar un nodo a la vez, por lo que elegir varios nodos significa que no puede utilizar el botón Delete node (Eliminar nodo).

Aparecerá la página Eliminar nodo.

6. Para eliminar el nodo, complete la página Eliminar nodo y elija Eliminar nodo. Para mantener el nodo, elija Cancelar.

Important

Si al eliminar el nodo, el clúster ya no es compatible con Multi-AZ, asegúrese de desactivar primero la casilla de verificación Multi-AZ y, a continuación, eliminar el nodo. Si desactiva la casilla de verificación Multi-AZ puede elegir habilitar la Conmutación por error automática.

Impacto de las nuevas solicitudes de adición y eliminación en las solicitudes pendientes

Escenarios	Operación pendiente	Solicitud nueva	Resultados
Escenario 1	Eliminar	Eliminar	<p>La nueva solicitud de eliminación, pendiente o inmediata, reemplaza a la solicitud de eliminación pendiente.</p> <p>Por ejemplo, si los nodos 0001, 0003 y 0007 se encuentran pendientes de eliminación y se emite una solicitud nueva para eliminar los nodos 0002 y 0004, solo se eliminarán los nodos 0002 y 0004. Los nodos 0001, 0003 y 0007 no se eliminarán.</p>
Escenario 2	Eliminar	Create	<p>La nueva solicitud de creación, pendiente o inmediata, reemplaza a la solicitud de eliminación pendiente.</p> <p>Por ejemplo, si los nodos 0001, 0003 y 0007 se encuentran pendientes de eliminación y se emite una solicitud nueva para crear un nodo, se creará un nodo nuevo y no se eliminarán los nodos 0001, 0003 y 0007.</p>
Escenario 3	Create	Eliminar	<p>La nueva solicitud de eliminación, pendiente o inmediata, reemplaza a la solicitud de creación pendiente.</p> <p>Por ejemplo, si hay una solicitud pendiente para crear dos nodos y se emite una solicitud nueva para eliminar el nodo 0003, no se crearán nodos nuevos y se eliminará el nodo 0003.</p>
Escenario 4	Create	Create	<p>La nueva solicitud de creación se agrega a la solicitud de creación pendiente.</p> <p>Por ejemplo, si hay una solicitud pendiente para crear dos nodos y se emite una solicitud nueva para crear</p>

Escenarios	Operación pendiente	Solicitud nueva	Resultados
			<p>tres nodos, las solicitudes nuevas se agregarán a la solicitud pendiente y se crearán cinco nodos.</p> <div data-bbox="732 386 1507 888" style="border: 1px solid #f08080; border-radius: 10px; padding: 10px; background-color: #fff9e6;"> <p>⚠ Important</p> <p>Si la nueva solicitud de creación se establece en Apply Immediately - Yes (Aplicar de inmediato: sí), todas las solicitudes de creación se ejecutan de inmediato. Si la nueva solicitud de creación se establece en Apply Immediately - No (Aplicar de inmediato: no), todas las solicitudes de creación quedarán pendientes.</p> </div>

Para determinar qué operaciones se encuentran pendientes, elija la pestaña de Description (Descripción) y verifique cuántas creaciones o eliminaciones pendientes se muestran. No puede tener creaciones y eliminaciones pendientes al mismo tiempo.

Utilización de la AWS CLI

1. Especifique los ID de los nodos que desee quitar. Para obtener más información, consulte [Visualización de los detalles de un clúster](#).
2. Utilice la operación `decrease-replica-count` de la CLI con una lista de los nodos que desea quitar, como en el siguiente ejemplo.

Para quitar nodos de un clúster a través de la interfaz de línea de comandos, utilice el comando `decrease-replica-count` con los siguientes parámetros:

- `--replication-group-id`: el ID del grupo de reproducción del que desea eliminar nodos.
- `--new-replica-count` El parámetro `--new-replica-count` especifica el número de nodos que desea en este clúster una vez aplicada la modificación.
- `--replicas-to-remove` Una lista de los ID de nodos que desea quitar de este clúster.

- `--apply-immediately` o `--no-apply-immediately` especifican si desea quitar estos nodos de forma inmediata o durante el próximo periodo de mantenimiento.
- `--region` especifica la región de AWS del clúster del que desea quitar nodos.

 Note

Solo puede pasar uno de los parámetros `--replicas-to-remove` o `--new-replica-count` al llamar a esta operación.

Para Linux, macOS o Unix:

```
aws elasticache decrease-replica-count \  
  --replication-group-id my-replication-group \  
  --new-replica-count 2 \  
  --region us-east-2 \  
  --apply-immediately
```

Para Windows:

```
aws elasticache decrease-replica-count ^  
  --replication-group-id my-replication-group ^  
  --new-replica-count 3 ^  
  --region us-east-2 ^  
  --apply-immediately
```

Esta operación produce un resultado similar al siguiente (formato JSON):

```
{  
  "ReplicationGroup": {  
    "ReplicationGroupId": "node-test",  
    "Description": "node-test"  
  },  
  "Status": "modifying",  
  "PendingModifiedValues": {},  
  "MemberClusters": [  
    "node-test-001",  
    "node-test-002",  
    "node-test-003",
```

```

        "node-test-004",
        "node-test-005",
        "node-test-006"
    ],
    "NodeGroups": [
        {
            "NodeId": "0001",
            "Status": "modifying",
            "PrimaryEndpoint": {
                "Address": "node-test.zzzzzz.ng.0001.usw2.cache.amazonaws.com",
                "Port": 6379
            },
            "ReaderEndpoint": {
                "Address": "node-test-
ro.zzzzzz.ng.0001.usw2.cache.amazonaws.com",
                "Port": 6379
            },
            "NodeGroupMembers": [
                {
                    "CacheClusterId": "node-test-001",
                    "CacheNodeId": "0001",
                    "ReadEndpoint": {
                        "Address": "node-
test-001.zzzzzz.0001.usw2.cache.amazonaws.com",
                        "Port": 6379
                    },
                    "PreferredAvailabilityZone": "us-west-2a",
                    "CurrentRole": "primary"
                },
                {
                    "CacheClusterId": "node-test-002",
                    "CacheNodeId": "0001",
                    "ReadEndpoint": {
                        "Address": "node-
test-002.zzzzzz.0001.usw2.cache.amazonaws.com",
                        "Port": 6379
                    },
                    "PreferredAvailabilityZone": "us-west-2c",
                    "CurrentRole": "replica"
                },
                {
                    "CacheClusterId": "node-test-003",
                    "CacheNodeId": "0001",
                    "ReadEndpoint": {

```

```

        "Address": "node-
test-003.zzzzzz.0001.usw2.cache.amazonaws.com",
        "Port": 6379
    },
    "PreferredAvailabilityZone": "us-west-2b",
    "CurrentRole": "replica"
},
{
    "CacheClusterId": "node-test-004",
    "CacheNodeId": "0001",
    "ReadEndpoint": {
        "Address": "node-
test-004.zzzzzz.0001.usw2.cache.amazonaws.com",
        "Port": 6379
    },
    "PreferredAvailabilityZone": "us-west-2c",
    "CurrentRole": "replica"
},
{
    "CacheClusterId": "node-test-005",
    "CacheNodeId": "0001",
    "ReadEndpoint": {
        "Address": "node-
test-005.zzzzzz.0001.usw2.cache.amazonaws.com",
        "Port": 6379
    },
    "PreferredAvailabilityZone": "us-west-2b",
    "CurrentRole": "replica"
},
{
    "CacheClusterId": "node-test-006",
    "CacheNodeId": "0001",
    "ReadEndpoint": {
        "Address": "node-
test-006.zzzzzz.0001.usw2.cache.amazonaws.com",
        "Port": 6379
    },
    "PreferredAvailabilityZone": "us-west-2b",
    "CurrentRole": "replica"
}
    ]
}
],
"SnapshottingClusterId": "node-test-002",

```

```

    "AutomaticFailover": "enabled",
    "MultiAZ": "enabled",
    "SnapshotRetentionLimit": 1,
    "SnapshotWindow": "07:30-08:30",
    "ClusterEnabled": false,
    "CacheNodeType": "cache.r5.large",
    "DataTiering": "disabled",
    "TransitEncryptionEnabled": false,
    "AtRestEncryptionEnabled": false,
    "ARN": "arn:aws:elasticache:us-west-2:123456789012:replicationgroup:node-
test"
  }
}

```

También puede llamar a `decrease-replica-count` y en lugar de pasar el parámetro `--new-replica-count`, podría pasar el parámetro `--replicas-to-remove`, como se muestra a continuación:

Para Linux, macOS o Unix:

```

aws elasticache decrease-replica-count \
  --replication-group-id my-replication-group \
  --replicas-to-remove node-test-003 \
  --region us-east-2 \
  --apply-immediately

```

Para Windows:

```

aws elasticache decrease-replica-count ^
  --replication-group-id my-replication-group ^
  --replicas-to-remove node-test-003 ^
  --region us-east-2 ^
  --apply-immediately

```

Para obtener más información, consulte los temas de la AWS CLI [decrease-replica-count](#).

Uso de la API de ElastiCache

Para eliminar nodos mediante la API de ElastiCache, llame a la operación de API `DecreaseReplicaCount` con el ID de grupo de reproducción y una lista de nodos a fin de eliminar, como se muestra a continuación:

- `ReplicationGroupId`: el ID del grupo de reproducción del que desea eliminar nodos.
- `ReplicasToRemove` El parámetro `ReplicasToRemove` especifica el número de nodos que desea en este clúster una vez aplicada la modificación.
- `ApplyImmediately` especifica si desea quitar estos nodos de forma inmediata o durante el próximo periodo de mantenimiento.
- `Region` especifica la región de AWS del clúster del que desea quitar un nodo.

En el ejemplo siguiente, se quitan de inmediato los nodos 0004 y 0005 del clúster my-cluster.

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=DecreaseReplicaCount  
  &ReplicationGroupId=my-replication-group  
  &ApplyImmediately=true  
  &ReplicasToRemove=node-test-003  
  &Region us-east-2  
  &Version=2014-12-01  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Timestamp=20141201T220302Z  
  &X-Amz-Algorithm=&AWS;4-HMAC-SHA256  
  &X-Amz-Date=20141201T220302Z  
  &X-Amz-SignedHeaders=Host  
  &X-Amz-Expires=20141201T220302Z  
  &X-Amz-Credential=<credential>  
  &X-Amz-Signature=<signature>
```

Para obtener más información, consulte el tema de la API de ElastiCache [DecreaseReplicaCount](#).

Cancelación de operaciones pendientes de adición o eliminación de nodos

Si ha optado por no aplicar un cambio de forma inmediata, la operación tiene el estado pending hasta que se realice en el próximo periodo de mantenimiento. Puede cancelar cualquier operación pendiente.

Para cancelar una operación pendiente

1. Inicie sesión en la AWS Management Console y abra la consola de ElastiCache en <https://console.aws.amazon.com/elasticache/>.
2. En la lista de la esquina superior derecha, elija la región de AWS en la que desea cancelar una operación pendiente de adición o eliminación de nodo.
3. En el panel de navegación, elija el motor que se ejecuta en el clúster que tiene operaciones pendientes que desea cancelar. Se mostrará una lista de los clústeres que ejecutan el motor elegido.
4. En la lista de clústeres, elija el nombre del clúster, no el cuadro situado a la izquierda del nombre del clúster, que tiene operaciones pendientes que desea cancelar.
5. Para determinar qué operaciones se encuentran pendientes, elija la pestaña de Description (Descripción) y verifique cuántas creaciones o eliminaciones pendientes se muestran. No puede tener creaciones y eliminaciones pendientes al mismo tiempo.
6. Elija la pestaña de Nodes (Nodos).
7. Para cancelar todas las operaciones pendientes, haga clic en Cancel Pending (Cancelar Pendiente). Aparecerá el cuadro de diálogo Cancel Pending.
8. Confirme que desea cancelar todas las operaciones pendientes seleccionando el botón Cancel Pending, o para conservar las operaciones, elija Cancel.

Eliminación de un clúster

Siempre que un clúster tenga el estado available, se cobrará por él, independientemente de si lo esté usando de forma activa o no. Para que dejen de devengarse cargos, elimine el clúster.

Warning

Al eliminar un clúster ElastiCache (RedisOSS), se conservan las instantáneas manuales. También puede crear una instantánea final antes de eliminar el clúster. Por el contrario, las instantáneas de caché automáticas no se conservan.

Usando el AWS Management Console

El siguiente procedimiento elimina un único clúster de su implementación. Para eliminar varios clústeres, repita el procedimiento por cada clúster que desee eliminar. No es necesario esperar a un clúster para terminar de eliminarlo antes de empezar el procedimiento para eliminar otro clúster.

Para eliminar un clúster

1. Inicia sesión en la ElastiCache consola de Amazon AWS Management Console y ábrela en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de control de la ElastiCache consola, selecciona el motor que está ejecutando el clúster que deseas eliminar.

Se mostrará una lista de todos los clústeres que ejecutan ese motor.

3. Para elegir el clúster que se va a eliminar, elija el nombre del clúster de la lista de clústeres.

Important

Solo puede eliminar un clúster a la vez desde la ElastiCache consola. Al elegir varios clústeres, se deshabilita la operación de eliminación.

4. En Actions (Acciones), seleccione Delete (Eliminar).
5. En la pantalla de confirmación Delete Cluster (Eliminar clúster), elija Delete (Eliminar) para eliminar el clúster o elija Cancel (Cancelar) para conservarlo.

Si elige Delete, el estado del clúster cambia a deleting.

En cuanto el clúster desaparezca de la lista de clústeres, dejará de incurrir en gastos.

Usando el AWS CLI

El código siguiente elimina el clúster de caché `my-cluster`.

```
aws elasticache delete-cache-cluster --cache-cluster-id my-cluster
```

La `delete-cache-cluster` CLI acción solo elimina un clúster de caché. Para eliminar varios clústeres de caché, llame a `delete-cache-cluster` por cada clúster de caché que desee eliminar. No es necesario esperar a un clúster de caché para terminar de eliminarlo antes de eliminar otro.

Para Linux, macOS o Unix:

```
aws elasticache delete-cache-cluster \  
  --cache-cluster-id my-cluster \  
  --region us-east-2
```

Para Windows:

```
aws elasticache delete-cache-cluster ^  
  --cache-cluster-id my-cluster ^  
  --region us-east-2
```

Para obtener más información, consulte el ElastiCache tema AWS CLI [delete-cache-cluster](#) para.

Uso del ElastiCache API

El código siguiente elimina el clúster `my-cluster`.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=DeleteCacheCluster  
&CacheClusterId=my-cluster  
&Region us-east-2  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20150202T220302Z  
&X-Amz-Algorithm=&AWS;4-HMAC-SHA256  
&X-Amz-Date=20150202T220302Z
```

```
&X-Amz-SignedHeaders=Host
&X-Amz-Expires=20150202T220302Z
&X-Amz-Credential=<credential>
&X-Amz-Signature=<signature>
```

La `DeleteCacheCluster` API operación solo elimina un clúster de caché. Para eliminar varios clústeres de caché, llame a `DeleteCacheCluster` por cada clúster de caché que desee eliminar. No es necesario esperar a un clúster de caché para terminar de eliminarlo antes de eliminar otro.

Para obtener más información, consulte el tema [DeleteCacheCluster](#) de ElastiCache API referencia.

Acceso al clúster o al grupo de reproducción

Tus ElastiCache instancias de Amazon están diseñadas para que puedas acceder a ellas a través de una EC2 instancia de Amazon.

Si lanzaste tu ElastiCache instancia en una Amazon Virtual Private Cloud (AmazonVPC), puedes acceder a ella ElastiCache desde una EC2 instancia de Amazon en el mismo AmazonVPC. O bien, mediante el VPC peering, puedes acceder a tu ElastiCache instancia desde un Amazon situado EC2 en otro AmazonVPC.

Si lanzaste tu ElastiCache instancia en EC2 Classic, permites que la EC2 instancia acceda a tu clúster al conceder al grupo de EC2 seguridad de Amazon asociado a la instancia acceso a tu grupo de seguridad de caché. De forma predeterminada, el acceso a un clúster está restringido a la cuenta que lanzó el clúster.

Temas

- [Conceder acceso al clúster o al grupo de reproducción](#)

Conceder acceso al clúster o al grupo de reproducción

Lanzó su clúster en EC2 - VPC

Si lanzaste tu clúster en una Amazon Virtual Private Cloud (AmazonVPC), solo podrás conectarte a tu ElastiCache clúster desde una EC2 instancia de Amazon que se ejecute en el mismo AmazonVPC. En este caso, necesitará conceder acceso de red al clúster.

Note

Si utiliza Local Zones, asegúrese de haberlo habilitado. Para obtener más información, consulte [Habilitar Local Zones](#). De este modo, VPC se extenderá a esa zona local y VPC tratará la subred como cualquier subred de cualquier otra zona de disponibilidad y ajustará automáticamente las puertas de enlace, las tablas de enrutamiento y otros grupos de seguridad pertinentes.

Para conceder la entrada de red desde un grupo de VPC seguridad de Amazon a un clúster

1. Inicia sesión en la EC2 consola de Amazon AWS Management Console y ábrela en <https://console.aws.amazon.com/ec2/>.

2. En el panel de navegación, en Network & Security (Redes y seguridad), seleccione Security Groups (Grupos de seguridad).
3. En la lista de grupos de seguridad, elige el grupo de seguridad para tu AmazonVPC. A menos que haya creado un grupo de seguridad para ElastiCache su uso, este grupo de seguridad se denominará predeterminado.
4. Elija la pestaña Inbound y haga lo siguiente:
 - a. Elija Editar.
 - b. Seleccione Agregar regla.
 - c. En la columna Tipo, elija TCPRegla personalizada.
 - d. En el cuadro Port range, escriba el número de puerto para su nodo de clúster. Este número debe ser el mismo que especificó cuando lanzó el clúster. El puerto predeterminado de OSS Redis es. **6379**
 - e. En el cuadro Fuente, selecciona Cualquier lugar que tenga el rango de puertos (0.0.0.0/0) para que cualquier EC2 instancia de Amazon que lances en tu Amazon VPC pueda conectarse a tus nodos. ElastiCache

 Important

Abrir el ElastiCache clúster a 0.0.0.0/0 no lo expone a Internet porque no tiene una dirección IP pública y, por lo tanto, no se puede acceder a él desde fuera. VPC Sin embargo, el grupo de seguridad predeterminado se puede aplicar a otras EC2 instancias de Amazon en la cuenta del cliente y esas instancias pueden tener una dirección IP pública. Si se está ejecutando algo en el puerto predeterminado, ese servicio podría exponerse de forma involuntaria. Por lo tanto, recomendamos crear un grupo de VPC seguridad que sea utilizado exclusivamente por ElastiCache. Para obtener más información, consulte [Grupos de seguridad personalizados](#).

- f. Seleccione Guardar.

Cuando lances una EC2 instancia de Amazon en tu AmazonVPC, esa instancia podrá conectarse a tu ElastiCache clúster.

Acceder a ElastiCache los recursos desde el exterior AWS

Amazon ElastiCache es un AWS servicio que proporciona un almacén de valores clave en memoria basado en la nube. El servicio está diseñado para que se pueda acceder exclusivamente desde dentro. Sin embargo, si el ElastiCache clúster está alojado en un VPC, puedes usar una instancia de traducción de direcciones de red (NAT) para proporcionar acceso externo.

Requisitos

Debes cumplir los siguientes requisitos para poder acceder a tus ElastiCache recursos desde el exterior AWS:

- El clúster debe residir dentro de una instancia de traducción de direcciones de red (NAT) VPC y acceder a él a través de ella. Este requisito no tiene excepciones.
- La NAT instancia debe lanzarse en el mismo lugar VPC que el clúster.
- La NAT instancia debe lanzarse en una subred pública independiente del clúster.
- Se debe asociar una dirección IP elástica (EIP) a la NAT instancia. La función de reenvío de puertos de iptables se utiliza para reenviar un puerto de la NAT instancia al puerto del nodo de caché dentro del VPC

Consideraciones

Debe tener en cuenta las siguientes consideraciones al acceder a sus ElastiCache recursos desde el exterior. ElastiCache

- Los clientes se conectan al puerto EIP y a la memoria caché de la NAT instancia. El reenvío de puertos de la NAT instancia reenvía el tráfico al nodo del clúster de caché correspondiente.
- Si un nodo del clúster se añade o se reemplaza, las reglas de tablas IP deben actualizarse para reflejar este cambio.

Limitaciones

Este enfoque solo debe usarse con fines de prueba y desarrollo. No se recomienda su uso para la producción debido a las limitaciones siguientes:

- La NAT instancia actúa como un proxy entre los clientes y varios clústeres. La adición de un proxy afecta al desempeño del clúster de caché. El impacto aumenta con la cantidad de clústeres de caché a los que se accede a través de la NAT instancia.

- El tráfico de los clientes a la NAT instancia no está cifrado. Por lo tanto, debe evitar enviar datos confidenciales a través de la NAT instancia.
- La NAT instancia añade la sobrecarga que supone el mantenimiento de otra instancia.
- La NAT instancia sirve como punto único de error. Para obtener información sobre cómo configurar la alta disponibilidad NAT en VPC, consulte [Alta disponibilidad para VPC NAT instancias de Amazon: un ejemplo](#).

¿Cómo acceder a ElastiCache los recursos desde fuera AWS

El siguiente procedimiento muestra cómo conectarse a ElastiCache los recursos mediante una NAT instancia.

En los pasos que se describen a continuación se da por sentado lo siguiente:

- `iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 6380 -j DNAT --to 10.0.1.231:6379`
- `iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 6381 -j DNAT --to 10.0.1.232:6379`

A continuación, necesita NAT ir en la dirección opuesta:

```
iptables -t nat -A POSTROUTING -o eth0 -j SNAT --to-source 10.0.0.55
```

También debe activar el reenvío de IP, que está desactivado de forma predeterminada:

```
sudo sed -i 's/net.ipv4.ip_forward=0/net.ipv4.ip_forward=1/g' /etc/sysctl.conf  
sudo sysctl --system
```

- Está accediendo a un OSS clúster de Redis con:
 - Dirección IP: 10.0.1.230
 - OSS Puerto de Redis predeterminado: 6379
 - Grupo de seguridad: sg-bd56b7da
 - AWS dirección IP de la instancia: sg-bd56b7da
- El cliente tiene de confianza tiene la dirección IP 198.51.100.27.
- La NAT instancia tiene la dirección IP elástica 203.0.113.73.
- La NAT instancia tiene el grupo de seguridad sg-ce56b7a9.

Para conectarse a sus recursos mediante una instancia ElastiCache NAT

1. Crea una NAT instancia en el mismo lugar VPC que tu clúster de caché, pero en una subred pública.

De forma predeterminada, el VPC asistente lanzará un nodo del tipo cache.m1.small. Seleccione un tamaño de nodo en función de sus necesidades. Debe usarlo EC2 NAT AMI para poder acceder desde el exterior. ElastiCache AWS

Para obtener información sobre la creación de una NAT instancia, consulte [NATInstancias](#) en la Guía del AWS VPC usuario.

2. Cree reglas de grupos de seguridad para el clúster y la NAT instancia de caché.

El grupo de seguridad de la NAT instancia y la instancia del clúster deben tener las siguientes reglas:

- Dos reglas de entrada
 - Uno para permitir TCP las conexiones de los clientes de confianza a cada puerto de caché reenviado desde la NAT instancia (6379 - 6381).
 - Un segundo para permitir el SSH acceso a clientes de confianza.

NATgrupo de seguridad de instancias: reglas de entrada

Tipo	Protocolo	Intervalo de puertos	Origen
Regla personalizada TCP	TCP	6379-6380	198.51.100.27/32
SSH	TCP	22	203.0.113.73/32

- Una regla de salida que permite TCP las conexiones al puerto de caché (6379).

NATgrupo de seguridad de instancias: regla de salida

Tipo	Protocolo	Rango de puerto	Destino
Regla personalizada TCP	TCP	6379	sg-ce56b7a9 (grupo de seguridad de la instancia de clúster)

- Una regla de entrada para el grupo de seguridad del clúster que permite TCP las conexiones desde la NAT instancia al puerto de caché (6379).

Grupo de seguridad de la instancia del clúster: regla entrante

Tipo	Protocolo	Intervalo de puertos	Origen
Regla personalizada TCP	TCP	6379	sg-bd56b7da (grupo de seguridad del clúster)

3. Valide las reglas.

- Confirma que el cliente de confianza puede acceder SSH a la NAT instancia.
- Confirma que el cliente de confianza puede conectarse al clúster desde la NAT instancia.

4. Agrega una regla de iptables a la NAT instancia.

Se debe agregar una regla de iptables a la NAT tabla para cada nodo del clúster para reenviar el puerto de caché de la NAT instancia al nodo del clúster. Un ejemplo de esto sería el siguiente:

```
iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 6379 -j DNAT --to 10.0.1.230:6379
```

El número de puerto debe ser único para cada nodo del clúster. Por ejemplo, si trabajas con un OSS clúster de Redis de tres nodos con los puertos 6379 a 6381, las reglas serían las siguientes:

```
iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 6379 -j DNAT --to 10.0.1.230:6379
iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 6380 -j DNAT --to 10.0.1.231:6379
iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 6381 -j DNAT --to 10.0.1.232:6379
```

5. Confirme que el cliente de confianza es capaz de conectarse al clúster.

El cliente de confianza debe conectarse al puerto EIP asociado a la NAT instancia y al puerto del clúster correspondiente al nodo del clúster correspondiente. Por ejemplo, la cadena de conexión de PHP podría tener el siguiente aspecto:

```
redis->connect( '203.0.113.73', 6379 );  
redis->connect( '203.0.113.73', 6380 );  
redis->connect( '203.0.113.73', 6381 );
```

También puede usar un cliente de Telnet para comprobar la conexión. Por ejemplo:

```
telnet 203.0.113.73 6379  
telnet 203.0.113.73 6380  
telnet 203.0.113.73 6381
```

6. Guarde la configuración de tablas IP.

Guarde las reglas tras realizar las pruebas y verificarlas. Si usa una distribución de Linux basada en Redhat (como Amazon Linux) ejecute el comando siguiente:

```
service iptables save
```

Temas relacionados de

Los temas siguientes pueden ser de su interés.

- [Patrones de acceso para acceder a una ElastiCache caché en una Amazon VPC](#)
- [Acceso a una ElastiCache memoria caché desde una aplicación que se ejecuta en el centro de datos de un cliente](#)
- [NATInstancias](#)
- [Configuración de ElastiCache clientes](#)
- [Alta disponibilidad para Amazon VPC NAT Instances: un ejemplo](#)

Búsqueda de puntos de conexión

Su aplicación se conecta a su clúster mediante puntos de conexión. Un punto de conexión es un nodo o la dirección única del clúster.

Si opta por no usar la función de detección automática, deberá configurar su cliente para usar los puntos de conexión individuales para las operaciones de lectura y escritura. También deberá realizar un seguimiento conforme agregue y elimine nodos.

Qué puntos de conexión usar

- Nodo OSS independiente de Redis, utilice el punto final del nodo para las operaciones de lectura y escritura.
- Los clústeres de Redis OSS (modo de clúster desactivado) utilizan el punto final principal para todas las operaciones de escritura. Use el punto de conexión de lector para dividir de manera uniforme las conexiones entrantes al punto de enlace entre todas las réplicas de lectura. Utilice los puntos finales de los nodos individuales para las operaciones de lectura (en API/, se CLI denominan puntos finales de lectura).
- En los clústeres de Redis OSS (modo de clúster activado), se utiliza el punto de conexión de configuración del clúster para todas las operaciones que admiten comandos activados en modo de clúster. Debe utilizar un cliente compatible con Redis OSS Cluster (Redis OSS 3.2). Aún puede leer desde los puntos finales de los nodos individuales (en la API/, se CLI denominan puntos finales de lectura).

Las secciones siguientes le guiarán en el proceso de detección de los puntos de conexión que necesitará para el motor que esté usando.

Búsqueda de los puntos finales de un clúster de Redis OSS (modo de clúster desactivado) (consola)

Si un clúster de Redis OSS (modo de clúster desactivado) tiene un solo nodo, el punto final del nodo se utiliza tanto para las lecturas como para las escrituras. Si un clúster de Redis OSS (modo de clúster desactivado) tiene varios nodos, hay tres tipos de puntos finales: el punto final principal, el punto final del lector y los puntos finales del nodo.

El punto final principal es un DNS nombre que siempre corresponde al nodo principal del clúster. El punto de conexión principal es inmune a los cambios en el clúster, como la promoción de una réplica de lectura a la función principal. Para una actividad de escritura, recomendamos que sus aplicaciones se conecten al punto de conexión principal.

Un punto final del lector dividirá equitativamente las conexiones entrantes al punto final entre todas las réplicas de lectura de un clúster ElastiCache (RedisOSS). Factores adicionales como cuando la aplicación crea las conexiones o cómo la aplicación (re)utiliza las conexiones determinará la distribución del tráfico. Los puntos de enlace de lector se mantienen actualizados con los cambios del clúster en tiempo real cuando se añaden o eliminan réplicas. Puede colocar las múltiples réplicas de lectura de su clúster ElastiCache (RedisOSS) en diferentes zonas de disponibilidad (AZ) para garantizar una alta AWS disponibilidad de los puntos finales de los lectores.

Note

Un punto de conexión del lector no es un equilibrador de carga. Se trata de un DNS registro que se convierte en una dirección IP de uno de los nodos de la réplica de forma rotativa.

Para la actividad de lectura, las aplicaciones pueden conectarse también a cualquier nodo del clúster. A diferencia del punto de conexión principal, los puntos de conexión de nodo se resuelven en puntos de enlace específicos. Si realiza un cambio en su clúster, como añadir o eliminar una réplica, debe actualizar los puntos de enlace del nodo en su aplicación.

Para encontrar los puntos finales de un clúster de Redis OSS (modo de clúster desactivado)

1. Inicie sesión en AWS Management Console y abra la ElastiCache consola en. <https://console.aws.amazon.com/elasticache/>
2. En el panel de navegación, elija OSSclústeres de Redis.

Aparecerá la pantalla de clústeres con una lista de clústeres de Redis OSS (modo de clúster desactivado) y Redis OSS (modo de clúster activado).

- Para buscar los puntos de conexión principal o de lector, elija el nombre del clúster (no el botón a su izquierda).

▼ Cluster details			
Cluster name	Description	Node type cache.r6g.large	Status Available
Engine Redis OSS	Engine version 6.0.5	Global datastore -	Global datastore role -
Update status Update available	Cluster mode Off	Shards 1	Number of nodes 3
Data tiering Disabled	Multi-AZ Enabled	Auto-failover Enabled	Encryption in transit Disabled
Encryption at rest Disabled	Parameter group default.redis6.x	Outpost ARN -	Configuration endpoint -
Primary endpoint [icon] [redacted]-encrypted.llru6f.ng.0001.use1.cache.amazonaws.com:6379	Reader endpoint [icon] [redacted]-encrypted-ro.llru6f.ng.0001.use1.cache.amazonaws.com:6379	ARN [redacted]	

Puntos finales principal y de lector de un clúster de Redis OSS (modo de clúster desactivado)

Si el clúster tiene solo un nodo, quiere decir que no hay punto de conexión principal y, por lo tanto, puede continuar con el paso siguiente.

- Si el clúster de Redis OSS (modo de clúster desactivado) tiene nodos de réplica, puede encontrar los puntos finales de los nodos de réplica del clúster eligiendo el nombre del clúster y, a continuación, seleccionando la pestaña Nodos.

Aparecerá la pantalla de nodos con los nodos del clúster, principal y réplicas; cada uno con su punto de conexión.

<input type="checkbox"/>	Node Name	Status	Current Role	Port	Endpoint
<input type="checkbox"/>	test-no-001	available	primary	6379	[redacted].amazonaws.com
<input type="checkbox"/>	test-no-002	available	replica	6379	[redacted].amazonaws.com
<input type="checkbox"/>	test-no-003	available	replica	6379	[redacted].amazonaws.com

Puntos finales de nodo de un clúster de Redis OSS (modo de clúster desactivado)

- Para copiar un punto de conexión al portapapeles:
 - Un punto de conexión cada vez, encuentre el punto de conexión que desea copiar.

- b. Elija el icono de copia directamente delante del punto de conexión.

El punto de conexión se ha copiado ahora en el portapapeles. Para obtener información sobre el uso del punto de conexión a fin de conectarse a un nodo, consulte [Conexión a los nodos](#).

El punto final principal de Redis OSS (modo de clúster desactivado) tiene un aspecto similar al siguiente. Existe una diferencia dependiendo de si está habilitado el cifrado en tránsito o no lo está.

Cifrado en tránsito no habilitado

```
clusterName.xxxxxx.nodeId.regionAndAz.cache.amazonaws.com:port
```

```
redis-01.7abc2d.0001.usw2.cache.amazonaws.com:6379
```

Cifrado en tránsito habilitado

```
master.clusterName.xxxxxx.regionAndAz.cache.amazonaws.com:port
```

```
master.ncit.ameaqx.use1.cache.amazonaws.com:6379
```

Búsqueda de puntos finales para un clúster de Redis OSS (modo de clúster activado) (consola)

Un clúster de Redis OSS (habilitado para el modo de clúster) tiene un único punto final de configuración. Al conectarse al punto de conexión de configuración, la aplicación puede detectar los puntos de conexión principal y de lectura de cada partición del clúster.

Para encontrar el punto final de un clúster de Redis OSS (habilitado para el modo de clúster)

1. Inicie sesión en AWS Management Console y abra la ElastiCache consola en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación, elija OSSclústeres de Redis.

Aparecerá la pantalla de clústeres con una lista de clústeres de Redis OSS (modo de clúster desactivado) y Redis OSS (modo de clúster activado). Elija el clúster de Redis OSS (modo de clúster activado) al que desee conectarse.

3. Para buscar el punto de conexión de configuración del clúster, elija el nombre del clúster (no el botón de opción).

4. El Configuration endpoint (Punto de conexión de configuración) se muestra en Cluster details (Detalles de clúster). Para copiarlo, elija el ícono copiar a la izquierda del punto de conexión.

Búsqueda de puntos de conexión (AWS CLI)

Puede utilizar AWS CLI para Amazon ElastiCache para descubrir los puntos finales de los nodos, los clústeres y los grupos de replicación.

Temas

- [Búsqueda de puntos de conexión para nodos y clústeres \(AWS CLI\)](#)
- [Búsqueda de los puntos de enlace de grupos de reproducción \(AWS CLI\)](#)

Búsqueda de puntos de conexión para nodos y clústeres (AWS CLI)

Puede utilizar el AWS CLI para descubrir los puntos finales de un clúster y sus nodos con el `describe-cache-clusters` comando. En el caso de OSS los clústeres de Redis, el comando devuelve el punto final del clúster. Si incluye el parámetro opcional `--show-cache-node-info`, el comando también devolverá los puntos de conexión de los nodos individuales del clúster.

Example

El siguiente comando recupera la información del clúster `mycluster` de Redis OSS (modo de clúster desactivado) de un solo nodo.

Important

El parámetro `--cache-cluster-id` puede usar con un identificador de clúster de Redis de un solo nodo OSS (modo de clúster desactivado) o con identificadores de nodos específicos en los grupos de replicación de Redis. OSS El `--cache-cluster-id` de un grupo de OSS replicación de Redis es un valor de 4 dígitos, como. `0001` Si `--cache-cluster-id` es el identificador de un clúster (nodo) de un grupo de OSS replicación de Redis, `replication-group-id` se incluye en el resultado.

Para Linux, macOS o Unix:

```
aws elasticache describe-cache-clusters \  
  --cache-cluster-id redis-cluster \  
  --show-cache-node-info
```

Para Windows:

```
aws elasticache describe-cache-clusters ^
--cache-cluster-id redis-cluster ^
--show-cache-node-info
```

El resultado de la operación anterior debería tener un aspecto similar al siguiente (JSONformato).

```
{
  "CacheClusters": [
    {
      "CacheClusterStatus": "available",
      "SecurityGroups": [
        {
          "SecurityGroupId": "sg-77186e0d",
          "Status": "active"
        }
      ],
      "CacheNodes": [
        {
          "CustomerAvailabilityZone": "us-east-1b",
          "CacheNodeCreateTime": "2018-04-25T18:19:28.241Z",
          "CacheNodeStatus": "available",
          "CacheNodeId": "0001",
          "Endpoint": {
            "Address": "redis-cluster.amazonaws.com",
            "Port": 6379
          },
          "ParameterGroupStatus": "in-sync"
        }
      ],
      "AtRestEncryptionEnabled": false,
      "CacheClusterId": "redis-cluster",
      "TransitEncryptionEnabled": false,
      "CacheParameterGroup": {
        "ParameterApplyStatus": "in-sync",
        "CacheNodeIdsToReboot": [],
        "CacheParameterGroupName": "default.redis3.2"
      },
      "NumCacheNodes": 1,
      "PreferredAvailabilityZone": "us-east-1b",
      "AutoMinorVersionUpgrade": true,
      "Engine": "redis",
      "AuthTokenEnabled": false,
      "PendingModifiedValues": {},
    }
  ]
}
```

```

        "PreferredMaintenanceWindow": "tue:08:30-tue:09:30",
        "CacheSecurityGroups": [],
        "CacheSubnetGroupName": "default",
        "CacheNodeType": "cache.t2.small",
        "DataTiering": "disabled"
        "EngineVersion": "3.2.10",
        "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
        "CacheClusterCreateTime": "2018-04-25T18:19:28.241Z"
    }
]
}

```

Para obtener más información, consulte el tema [describe-cache-clusters](#).

Búsqueda de los puntos de enlace de grupos de reproducción (AWS CLI)

Puede utilizar el AWS CLI para descubrir los puntos finales de un grupo de replicación y sus clústeres con el `describe-replication-groups` comando. El comando devuelve el punto de conexión principal del grupo de reproducción y una lista de todos los clústeres (nodos) del grupo de reproducción con sus puntos de conexión, junto con el punto de conexión de lector.

La operación siguiente recupera el punto de conexión principal y el punto de conexión de lector para el grupo de reproducción `myreplgroup`. Use el punto de conexión principal para todas las operaciones de escritura.

```
aws elasticache describe-replication-groups \
  --replication-group-id myreplgroup
```

Para Windows:

```
aws elasticache describe-replication-groups ^
  --replication-group-id myreplgroup
```

El resultado de esta operación debería tener un aspecto similar al siguiente (JSONformato).

```

{
  "ReplicationGroups": [
    {
      "Status": "available",
      "Description": "test",
      "NodeGroups": [

```

```
{
  "Status": "available",
  "NodeGroupMembers": [
    {
      "CurrentRole": "primary",
      "PreferredAvailabilityZone": "us-west-2a",
      "CacheNodeId": "0001",
      "ReadEndpoint": {
        "Port": 6379,
        "Address": "myreplgroup-001.amazonaws.com"
      },
      "CacheClusterId": "myreplgroup-001"
    },
    {
      "CurrentRole": "replica",
      "PreferredAvailabilityZone": "us-west-2b",
      "CacheNodeId": "0001",
      "ReadEndpoint": {
        "Port": 6379,
        "Address": "myreplgroup-002.amazonaws.com"
      },
      "CacheClusterId": "myreplgroup-002"
    },
    {
      "CurrentRole": "replica",
      "PreferredAvailabilityZone": "us-west-2c",
      "CacheNodeId": "0001",
      "ReadEndpoint": {
        "Port": 6379,
        "Address": "myreplgroup-003.amazonaws.com"
      },
      "CacheClusterId": "myreplgroup-003"
    }
  ],
  "NodeGroupId": "0001",
  "PrimaryEndpoint": {
    "Port": 6379,
    "Address": "myreplgroup.amazonaws.com"
  },
  "ReaderEndpoint": {
    "Port": 6379,
    "Address": "myreplgroup-ro.amazonaws.com"
  }
}
```

```
    ],  
    "ReplicationGroupId": "myreplgroup",  
    "AutomaticFailover": "enabled",  
    "SnapshottingClusterId": "myreplgroup-002",  
    "MemberClusters": [  
        "myreplgroup-001",  
        "myreplgroup-002",  
        "myreplgroup-003"  
    ],  
    "PendingModifiedValues": {}  
  }  
]  
}
```

Para obtener más información, consulte [describe-replication-groups](#) la Referencia de AWS CLI comandos.

Búsqueda de puntos finales () ElastiCache API

Puede usar Amazon ElastiCache API para descubrir los puntos finales de los nodos, los clústeres y los grupos de replicación.

Temas

- [Búsqueda de puntos finales para nodos y clústeres \(\) ElastiCache API](#)
- [Búsqueda de puntos finales para grupos de replicación \(\) ElastiCache API](#)

Búsqueda de puntos finales para nodos y clústeres () ElastiCache API

Puede utilizarla ElastiCache API para descubrir los puntos finales de un clúster y sus nodos con la `DescribeCacheClusters` acción. En el caso de OSS los clústeres de Redis, el comando devuelve el punto final del clúster. Si incluye el parámetro opcional `ShowCacheNodeInfo`, la acción también devuelve los puntos de conexión de los nodos individuales del clúster.

Example

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=DescribeCacheClusters  
&CacheClusterId=mycluster  
&ShowCacheNodeInfo=true  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20150202T192317Z  
&Version=2015-02-02  
&X-Amz-Credential=<credential>
```

Búsqueda de puntos finales para grupos de replicación () ElastiCache API

Puede utilizarla ElastiCache API para descubrir los puntos finales de un grupo de replicación y sus clústeres con la `DescribeReplicationGroups` acción. La acción devuelve el punto de conexión principal del grupo de reproducción y una lista de todos los clústeres del grupo de reproducción con sus puntos de enlace, junto con el punto de conexión de lector.

La siguiente operación recupera el punto final principal (`PrimaryEndpoint`), el punto final del lector (`ReaderEndpoint`) y los puntos finales del nodo individual (`ReadEndpoint`) del grupo de replicación. `myrep1group` Use el punto de conexión principal para todas las operaciones de escritura.

```
https://elasticache.us-west-2.amazonaws.com/
```

```
?Action=DescribeReplicationGroups
&ReplicationGroupId=myreplgroup
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&Version=2015-02-02
&X-Amz-Credential=<credential>
```

Para obtener más información, consulte [DescribeReplicationGroups](#)

Trabajar con particiones

Un fragmento (API/CLI: grupo de nodos) es un conjunto de uno a seis nodos ElastiCache (Redis OSS). Un clúster de Redis OSS (modo de clúster desactivado) nunca tendrá más de un fragmento. Con los fragmentos, puede separar las bases de datos grandes en partes más pequeñas, rápidas y fáciles de gestionar denominadas fragmentos de datos. Esto puede aumentar la eficiencia de la base de datos al distribuir las operaciones en varias secciones independientes. El uso de fragmentos puede ofrecer muchos beneficios, como la mejora del rendimiento, la escalabilidad y la rentabilidad.

Puede crear un clúster con un mayor número de particiones y un menor número de réplicas con un total de hasta 90 nodos por clúster. Esta configuración de clúster puede variar desde 90 particiones y 0 réplicas hasta 15 particiones y 5 réplicas, que es el número máximo de réplicas permitido. Los datos del clúster están particionados en las distintas particiones del clúster. Si hay más de un nodo en una partición, esta implementa la reproducción con un nodo, siendo el nodo principal de lectura/escritura y los demás, nodos de réplica de solo lectura.

El límite de nodos o fragmentos se puede aumentar hasta un máximo de 500 por clúster si la versión del motor OSS de Redis es 5.0.6 o superior. Por ejemplo, puede elegir configurar un clúster de 500 nodos que oscila entre 83 particiones (uno primario y 5 réplicas por partición) y 500 particiones (único primario y sin réplicas). Asegúrese de que hay suficientes direcciones IP disponibles para acomodar el aumento. Algunos problemas comunes incluyen que las subredes del grupo de subredes tienen un rango CIDR demasiado pequeño o que otros clústeres comparten y utilizan considerablemente las subredes. Para obtener más información, consulte [Creación de un grupo de subredes](#).

Para las versiones inferiores a 5.0.6, el límite es de 250 por clúster.

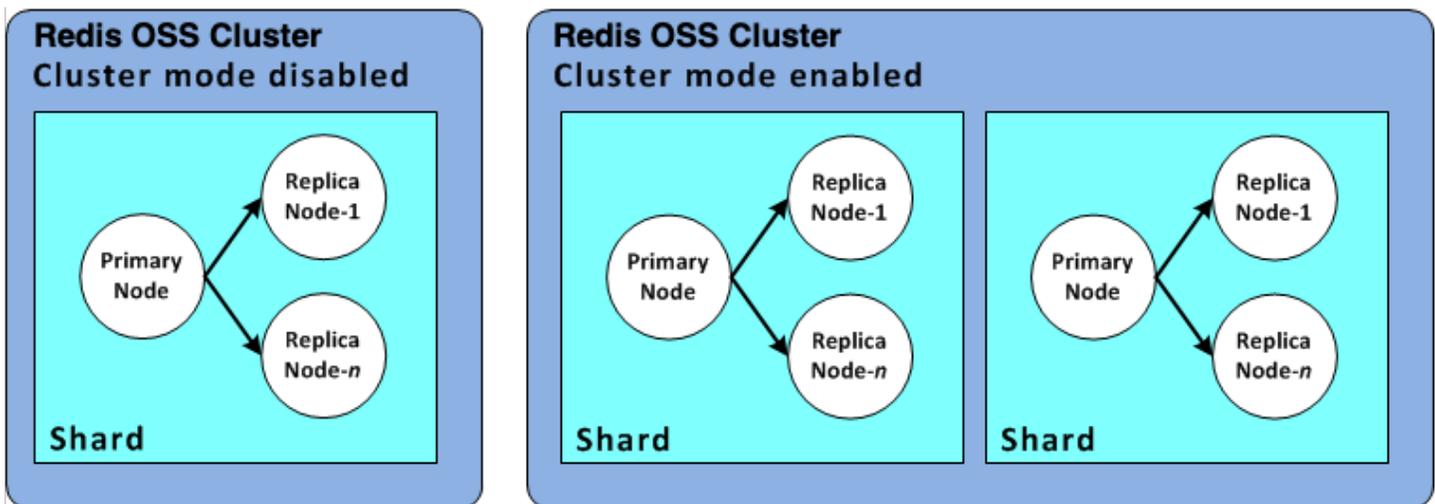
Para solicitar un aumento del límite, consulte [AWS Service Limits](#) y elija el tipo de límite Nodes per cluster per instance type (Nodos por clúster por tipo de instancias).

Al crear un clúster de Redis OSS (habilitado para el modo de clúster) mediante la ElastiCache consola, se especifica el número de fragmentos del clúster y el número de nodos de los fragmentos.

Para obtener más información, consulte [Creación de un clúster de Redis OSS \(modo de clúster activado\) \(consola\)](#). Si utiliza la ElastiCache API o AWS CLI crea un clúster (denominado grupo de replicación en la API/CLI), puede configurar el número de nodos de un fragmento (API/CLI: grupo de nodos) de forma independiente. Para más información, consulte los siguientes temas:

- API: [CreateReplicationGroup](#)
- CLI: [create-replication-group](#)

Los nodos de las particiones tienen las mismas especificaciones de memoria, almacenamiento y computación. La ElastiCache API le permite controlar los atributos de todo el fragmento, como la cantidad de nodos, la configuración de seguridad y los períodos de mantenimiento del sistema.



Configuraciones compartidas de Redis OSS

Para obtener más información, consulte [Refragmentación y reequilibrio de particiones sin conexión para Redis OSS \(modo de clúster habilitado\)](#) y [Refragmentación y reequilibrio de particiones en línea para Redis OSS \(modo de clúster habilitado\)](#).

Búsqueda del ID de una partición

Puede encontrar el ID de un fragmento mediante la AWS Management Console, la AWS CLI o la API. ElastiCache

Usando la AWS Management Console

Temas

- [Para Redis OSS \(modo de clúster desactivado\)](#)

- [Para Redis OSS \(modo de clúster activado\)](#)

Para Redis OSS (modo de clúster desactivado)

Los ID de los fragmentos de los grupos de replicación de Redis OSS (modo de clúster desactivado) siempre lo son. 0001

Para Redis OSS (modo de clúster activado)

El siguiente procedimiento utiliza el AWS Management Console para buscar el ID de fragmento de un grupo de replicación de Redis OSS (modo de clúster habilitado).

Para encontrar el ID del fragmento en un grupo de replicación de Redis OSS (modo de clúster activado)

1. [Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en https://console.aws.amazon.com/elasticache/.](https://console.aws.amazon.com/elasticache/)
2. En el panel de navegación, elija Redis OSS y, a continuación, elija el nombre del grupo de replicación de Redis OSS (modo de clúster habilitado) del que desea buscar los ID de los fragmentos.
3. En la columna de Shard Name (Nombre de la partición), el ID de partición lo forman los cuatro últimos dígitos del nombre de la partición.

Uso del AWS CLI

Para encontrar los identificadores de fragmentos (grupo de nodos) para los grupos de replicación de Redis OSS (modo de clúster deshabilitado) o de Redis OSS (modo de clúster habilitado), utilice la AWS CLI operación `describe-replication-groups` con el siguiente parámetro opcional.

- **--replication-group-id**: un parámetro opcional que, cuando se utiliza, limita los resultados de los detalles del grupo de reproducción especificado. Si se omite este parámetro, se devuelven los detalles de hasta 100 grupos de reproducción.

Example

Este comando devuelve los detalles de `sample-repl-group`.

Para Linux, macOS o Unix:

```
aws elasticache describe-replication-groups \  
  --replication-group-id sample-repl-group
```

Para Windows:

```
aws elasticache describe-replication-groups ^  
  --replication-group-id sample-repl-group
```

La salida de este comando es similar a la siguiente. Aquí están *resaltados* los ID de fragmento (grupo de nodos) para que resulte más fácil encontrarlos.

```
{  
  "ReplicationGroups": [  
    {  
      "Status": "available",  
      "Description": "2 shards, 2 nodes (1 + 1 replica)",  
      "NodeGroups": [  
        {  
          "Status": "available",  
          "Slots": "0-8191",  
          "NodeGroupId": "0001",  
          "NodeGroupMembers": [  
            {  
              "PreferredAvailabilityZone": "us-west-2c",  
              "CacheNodeId": "0001",  
              "CacheClusterId": "sample-repl-group-0001-001"  
            },  
            {  
              "PreferredAvailabilityZone": "us-west-2a",  
              "CacheNodeId": "0001",  
              "CacheClusterId": "sample-repl-group-0001-002"  
            }  
          ]  
        },  
        {  
          "Status": "available",  
          "Slots": "8192-16383",  
          "NodeGroupId": "0002",  
          "NodeGroupMembers": [  
            {  
              "PreferredAvailabilityZone": "us-west-2b",  
              "CacheNodeId": "0001",  
              "CacheClusterId": "sample-repl-group-0002-001"  
            }  
          ]  
        }  
      ]  
    }  
  ]  
}
```

```

        "CacheClusterId": "sample-repl-group-0002-001"
    },
    {
        "PreferredAvailabilityZone": "us-west-2a",
        "CacheNodeId": "0001",
        "CacheClusterId": "sample-repl-group-0002-002"
    }
]
}
],
"ConfigurationEndpoint": {
    "Port": 6379,
    "Address": "sample-repl-
group.9dcv5r.clustercfg.usw2.cache.amazonaws.com"
},
"ClusterEnabled": true,
"ReplicationGroupId": "sample-repl-group",
"SnapshotRetentionLimit": 1,
"AutomaticFailover": "enabled",
"SnapshotWindow": "13:00-14:00",
"MemberClusters": [
    "sample-repl-group-0001-001",
    "sample-repl-group-0001-002",
    "sample-repl-group-0002-001",
    "sample-repl-group-0002-002"
],
"CacheNodeType": "cache.m3.medium",
"DataTiering": "disabled",
"PendingModifiedValues": {}
}
]
}

```

Uso de la API ElastiCache

Para encontrar los identificadores de fragmentos (grupo de nodos) para los grupos de replicación de Redis OSS (modo de clúster deshabilitado) o de Redis OSS (modo de clúster habilitado), utilice la AWS CLI operación `describe-replication-groups` con el siguiente parámetro opcional.

- **ReplicationGroupId:** un parámetro opcional que, cuando se utiliza, limita los resultados de los detalles del grupo de reproducción especificado. Si se omite este parámetro, se devuelven los detalles de hasta **xxx** grupos de reproducción.

Example

Este comando devuelve los detalles de `sample-repl-group`.

Para Linux, macOS o Unix:

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=DescribeReplicationGroup  
&ReplicationGroupId=sample-repl-group  
&Version=2015-02-02  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20150202T192317Z  
&X-Amz-Credential=<credential>
```

Comparación de las cachés autodiseñadas de Memcached y Redis OSS

Amazon ElastiCache es compatible con los motores de caché OSS de Memcached y Redis. Cada motor proporciona algunas ventajas. Use la información de este tema para elegir el motor y la versión que mejor se adapte a sus necesidades.

Important

Después de crear una caché, un clúster de diseño propio o un grupo de replicación, puede actualizar a una versión del motor más reciente, pero no puede cambiar a una versión anterior del motor. Si desea utilizar una versión anterior del motor, debe eliminar la memoria caché, el clúster de diseño propio o el grupo de replicación existentes y volver a crearlos con la versión anterior del motor.

En líneas generales, los motores son similares. Cada uno de ellos funciona como almacén de claves-valores en memoria. Sin embargo, en la práctica, existen diferencias importantes.

Elija Memcached si se dan las condiciones siguientes:

- Necesita el modelo más sencillo que sea posible.
- Necesita ejecutar nodos de gran tamaño con varios núcleos o subprocesos.

- Necesita tener la capacidad de escalar de forma ascendente y descendente, y agregar y eliminar nodos en función de la demanda del sistema.
- Necesita almacenar en la caché objetos.

Elija Redis OSS con una versión de ElastiCache (Redis OSS) si se ajusta a lo siguiente:

- ElastiCache (Redis OSS) versión 7.0 (mejorada)

[Desea utilizar las mejoras de Redis OSS Functions, Sharded Pub/Sub o Redis OSS ACL.](#) Para obtener más información, consulte [Redis OSS](#), versión 7.0 (mejorada).

- ElastiCache (Redis OSS) versión 6.2 (mejorada)

Desea tener la posibilidad de establecer niveles de datos entre memoria y SSD utilizando el tipo de nodo r6gd. Para obtener más información, consulte [Organización de datos en niveles](#).

- ElastiCache (Redis OSS) versión 6.0 (mejorada)

Desea autenticar usuarios con el control de acceso basado en roles.

Para obtener más información, consulte [Redis OSS versión 6.0 \(mejorada\)](#).

- ElastiCache (Redis OSS) versión 5.0.0 (mejorada)

Desea utilizar las [transmisiones OSS de Redis](#), una estructura de datos de registro que permite a los productores añadir nuevos elementos en tiempo real y también permite a los consumidores consumir los mensajes de forma bloqueada o no bloqueante.

Para obtener más información, consulte [Redis OSS, versión 5.0.0](#) (mejorada).

- ElastiCache (Redis OSS) versión 4.0.10 (mejorada)

Admite el cifrado y la adición o eliminación dinámica de fragmentos de su clúster de Redis OSS (habilitado para el modo de clúster).

Para obtener más información, consulte la [versión 4.0.10 \(mejorada\) de Redis OSS](#).

Las siguientes versiones están en desuso, han llegado o están a punto de llegar al final de su vida útil.

- ElastiCache (Redis OSS) versión 3.2.10 (mejorada)

Admite la capacidad de añadir o eliminar fragmentos de forma dinámica de su clúster de Redis OSS (habilitado para el modo de clúster).

 Important

Actualmente ElastiCache (Redis OSS) 3.2.10 no admite el cifrado.

Para más información, consulte los siguientes temas:

- [Redis OSS versión 3.2.10 \(mejorada\)](#)
- Prácticas recomendadas de refragmentación en línea para Redis OSS. Para obtener más información, consulte lo siguiente:
 - [Prácticas recomendadas: cambios de los fragmentos online](#)
 - [Refragmentación y reequilibrio de fragmentos en línea para Redis OSS \(modo de clúster activado\)](#)
- [Para obtener más información sobre cómo escalar los clústeres de Redis OSS, consulte Escalado.](#)
- ElastiCache (Redis OSS) versión 3.2.6 (mejorada)

Si necesita la funcionalidad de las versiones anteriores de Redis OSS además de las siguientes funciones, elija ElastiCache (Redis OSS) 3.2.6:

- Cifrado en tránsito. Para obtener más información, consulte [Amazon ElastiCache \(Redis OSS\) In-Transit Encryption](#).
- Cifrado en reposo. Para obtener más información, consulte [Amazon ElastiCache \(Redis OSS\) At-Rest Encryption](#).
- ElastiCache (Redis OSS) (modo de clúster activado) versión 3.2.4

Si necesita la funcionalidad de Redis OSS 2.8.x además de las siguientes funciones, elija Redis OSS 3.2.4 (modo agrupado):

- Es necesario que los datos se repartan entre dos a 500 grupos de nodos (solo en modo clúster).
- Necesita indexación geoespacial (modo agrupado en clúster y no agrupado en clúster).
- No necesita la compatibilidad con varias bases de datos.
- ElastiCache (Redis OSS) (modo no agrupado) 2.8.x y 3.2.4 (mejorado)

Si lo siguiente es su caso, elija Redis OSS 2.8.x o Redis OSS 3.2.4 (modo no agrupado):

- Necesita tipos de datos complejos como, por ejemplo, cadenas, hash, listas, conjuntos, conjuntos clasificados mapas de bits.
- Necesita ordenar o clasificar conjuntos de datos en memoria.
- Necesita persistencia del almacén de claves.
- Necesita replicar sus datos desde el nodo principal a una o varias réplicas de lectura para aplicaciones de lectura intensivas.
- Necesita conmutación por error automática en caso de error en el nodo principal.
- Necesita publicar y suscribirse a capacidades para informar a los clientes acerca de eventos en el servidor.
- Necesita capacidades de copia de seguridad y restauración para los clústeres de diseño propio, así como para las cachés sin servidor.
- Necesita compatibilidad con varias bases de datos.

Resumen comparativo de Memcached, Redis OSS (modo de clúster desactivado) y Redis OSS (modo de clúster activado)

	Memcached	Redis OSS (modo de clúster desactivado)	Redis OSS (modo de clúster activado)
Versiones del motor +	1.4.5 y versiones posteriores	4.0.10 y versiones posteriores	4.0.10 y versiones posteriores
Tipos de datos	Sencillez	2.8.x - Complejo * Complejo	3.2.x y posteriores - Complejo
Particiones de datos	Sí	No	Sí
Clúster modificable	Sí	Sí	3.2.10 y posteriores - Limitado
Cambios de los fragmentos online	No	No	3.2.10 y versiones posteriores
Cifrado	in-transit 1.6.12 y versiones posteriores	4.0.10 y versiones posteriores	4.0.10 y versiones posteriores
Organización de datos en niveles	No	6.2 y versiones posteriores	6.2 y versiones posteriores
Certificaciones de conformidad			
Certificación de conformidad			
FedRAMP	Sí - 1.6.12 y posteriores	4.0.10 y versiones posteriores	4.0.10 y versiones posteriores
HIPAA	Sí - 1.6.12 y posteriores	4.0.10 y versiones posteriores	4.0.10 y versiones posteriores
PCI DSS	Sí	4.0.10 y versiones posteriores	4.0.10 y versiones posteriores
Múltiples subprocesos	Sí	No	No

	Memcached	Redis OSS (modo de clúster desactivado)	Redis OSS (modo de clúster activado)
Actualización de tipo de nodo	No	Sí	Sí
Actualización del motor	Sí	Sí	Sí
Alta disponibilidad (replicación)	No	Sí	Sí
Conmutación por error automática	No	Opcional	Obligatoria
Capacidades de publicación/suscripción	No	Sí	Sí
Conjuntos ordenados	No	Sí	Sí
Copia de seguridad y restauración	Solo para Memcached sin servidor, no para clústeres de Memcached de diseño propio	Sí	Sí
Indexación geoespacial	No	4.0.10 y versiones posteriores	Sí

Notas:

cadena, objetos (como bases de datos)

* cadena, conjuntos, conjuntos ordenados, listas, hash, mapas de bits, hyperloglog

cadena, conjuntos, conjuntos ordenados, listas, hash, mapas de bits, hyperloglog, índices geoespaciales

Memcached

Redis OSS (modo de
clúster desactivado)Redis OSS (modo de
clúster activado)

+ Se excluyen las versiones que están en desuso, que han llegado al final de su vida útil o que están a punto de llegar al final de su vida útil.

Tras elegir el motor para su clúster, recomendamos que use la versión más reciente de dicho motor. Para obtener más información, consulte Versiones [compatibles ElastiCache \(Memcached\)](#) o [Versiones compatibles ElastiCache \(Redis OSS\)](#).

Migración en línea a ElastiCache

Al utilizar Online Migration, puede migrar sus datos de Redis OSS de código abierto y autohospedado en Amazon EC2 a Amazon. ElastiCache

Note

No se admite la migración en línea a las cachés ni a los clústeres ElastiCache sin servidor que se ejecutan en el tipo de nodo r6gd.

Información general

Para migrar sus datos de Redis OSS de código abierto que se ejecuta en Amazon EC2 a ElastiCache Amazon, se necesita una implementación de Amazon existente o recién creada ElastiCache . La implementación debe tener una configuración que esté preparada para el proceso de migración. También debe ajustarse a la configuración que desee, incluidos atributos como el tipo de instancia, el número de particiones y el número de réplicas.

La migración en línea está diseñada para la migración de datos desde Redis OSS de código abierto autohospedado en Amazon EC2 a ElastiCache (Redis OSS), y no para mover datos entre ElastiCache clústeres (Redis OSS).

Important

Recomendamos que lea las siguientes secciones en su totalidad antes de iniciar el proceso de migración online.

La migración comienza cuando llama a la operación de la API `StartMigration` o al comando `AWS CLI`. Al migrar clústeres deshabilitados en modo clúster de Redis OSS, el proceso de migración convierte el nodo principal del clúster ElastiCache (Redis OSS) en una réplica del Redis OSS principal de Redis de origen. Al migrar clústeres habilitados para el modo clúster de Redis OSS, el proceso de migración convierte el nodo principal de cada fragmento en una réplica del ElastiCache fragmento correspondiente del clúster de origen que posee las mismas ranuras.

Después de llevar a cabo los cambios en la parte del cliente, llame a la operación de la API `CompleteMigration`. Esta operación de API convierte su implementación en su ElastiCache implementación principal de Redis OSS con nodos principales y de réplica (según corresponda). Ahora puede redirigir la aplicación cliente para empezar a escribir datos en ella. ElastiCache Durante la migración, puede comprobar el estado de la replicación ejecutando el comando [REDIS-cli INFO](#) en los nodos OSS de Redis y en los nodos principales. ElastiCache

Pasos para realizar la migración

Los siguientes temas describen el proceso de la migración de datos:

- [Preparación de los nodos OSS de Redis de origen y destino para la migración](#)
- [Prueba de la migración de datos](#)
- [Inicio de la migración](#)
- [Verificación del progreso de migración de datos](#)
- [Finalización de la migración de datos](#)

Preparación de los nodos OSS de Redis de origen y destino para la migración

Debe asegurarse de que se cumplen los cuatro requisitos previos que se mencionan a continuación antes de iniciar la migración desde la ElastiCache consola, la API o la `AWS CLI`.

Para preparar los nodos OSS de Redis de origen y destino para la migración

1. Identifique la ElastiCache implementación de destino y asegúrese de poder migrar los datos a ella.

Una ElastiCache implementación existente o recién creada debe cumplir los siguientes requisitos de migración:

- Utiliza la versión 5.0.6 o superior del motor OSS de Redis.
 - No cuenta con cifrado en tránsito o en reposo habilitados.
 - Tiene habilitado Multi-AZ.
 - Dispone de memoria suficiente para almacenar los datos del clúster de Redis OSS. Para configurar los ajustes adecuados de memoria reservada, consulte [Administración de la memoria reservada](#).
 - Si el modo clúster está desactivado, puede migrar directamente desde las versiones 2.8.21 de Redis OSS a las versiones 5.0.6 de Redis OSS en adelante si utiliza la CLI o las versiones 5.0.6 de Redis OSS en adelante mediante la CLI o la consola. Si el modo de clúster está activado, puede migrar directamente desde cualquier versión de Redis OSS habilitada para el modo de clúster a Redis OSS versión 5.0.6 en adelante si utiliza la CLI o las versiones 5.0.6 de Redis OSS en adelante mediante la CLI o la consola.
 - Número de particiones en origen y destino coincidentes.
 - No forma parte de un almacén de datos global.
 - Tiene desactivada la agrupación de datos por niveles.
2. Asegúrese de que las configuraciones de su Redis OSS de código abierto y de la implementación (Redis OSS) sean compatibles. ElastiCache

Como mínimo, todo lo siguiente de la ElastiCache implementación de destino debe ser compatible con la configuración de Redis OSS para la replicación de Redis OSS:

- Su clúster de Redis OSS no debe tener habilitada la autenticación de Redis OSS.
- La configuración de Redis OSS `protected-mode` debe estar establecida en `no`.
- Si tiene una `bind` configuración en su configuración de Redis OSS, debe actualizarse para permitir las solicitudes de ElastiCache los nodos.
- El número de bases de datos lógicas debe ser el mismo en el ElastiCache nodo y en el clúster de Redis OSS. Este valor se establece mediante la configuración `databases` de Redis OSS.
- No se debe cambiar el nombre de los comandos de Redis OSS que modifican datos para permitir que la replicación de los datos se realice correctamente. Por ejemplo `sync,,,psyncinfo, config y. command cluster`.
- Para replicar los datos de su clúster de OSS de Redis ElastiCache, asegúrese de que haya suficiente CPU y memoria para gestionar esta carga adicional. Esta carga proviene del archivo RDB creado por su clúster de OSS de Redis y transferido a través de la red al nodo. ElastiCache

- Todas las instancias de Redis OSS del clúster de origen deben ejecutarse en el mismo puerto.
3. Asegúrese de que sus instancias se puedan conectar ElastiCache haciendo lo siguiente:
 - Asegúrese de que la dirección IP de cada instancia sea privada.
 - Asigne o cree la ElastiCache implementación en la misma nube privada virtual (VPC) que su OSS de Redis en su instancia (recomendado).
 - Si las VPC son distintas, configure la interconexión de VPC para permitir el acceso entre los nodos. Para obtener más información acerca de la interconexión con VPC, consulte [Patrones de acceso para acceder a una ElastiCache caché en una Amazon VPC](#).
 - El grupo de seguridad adjunto a las instancias de Redis OSS debe permitir el tráfico entrante desde los nodos. ElastiCache
 4. Asegúrese de que su aplicación pueda dirigir el tráfico a ElastiCache los nodos una vez finalizada la migración de los datos. Para obtener más información, consulte [Patrones de acceso para acceder a una ElastiCache caché en una Amazon VPC](#).

Prueba de la migración de datos

Después de completar todos los requisitos previos, puede validar la configuración de migración con la AWS Management Console, la API de ElastiCache o la AWS CLI. El siguiente ejemplo muestra el uso de la CLI.

Pruebe la migración con el comando `test-migration` con los siguientes parámetros:

- `--replication-group-id`: el ID del grupo de replicación al que se van a migrar los datos.
- `--customer-node-endpoint-list`: lista de puntos de conexión desde los que deben migrarse los datos. La lista debe tener solo un elemento.

A continuación se muestra un ejemplo del uso de la CLI.

```
aws elasticache test-migration --replication-group-id test-cluster --customer-node-endpoint-list "Address='10.0.0.241',Port=6379"
```

ElastiCache validará la configuración de migración sin ninguna migración de datos real.

Inicio de la migración

Una vez completados todos los requisitos previos, puede comenzar la migración de datos mediante la AWS Management Console ElastiCache API o AWS CLI. Para el modo clúster activado, si la migración de ranura difiere, se realizará un cambio de particiones antes de la migración en activo. El siguiente ejemplo muestra el uso de la CLI.

Note

Recomendamos utilizar la API `TestMigration` para validar la configuración de la migración. Pero esto es estrictamente opcional.

Inicie la migración llamando al comando `start-migration` con los siguientes parámetros:

- `--replication-group-id`— Identificador del grupo de ElastiCache replicación de destino
- `--customer-node-endpoint-list`— Una lista de puntos finales con direcciones DNS o IP y el puerto en el que se ejecuta el clúster OSS de Redis de origen. La lista solo puede tener un elemento tanto para el modo clúster desactivado como para el modo clúster activado. Si ha habilitado la replicación en cadena, el punto final puede apuntar a una réplica en lugar del nodo principal del clúster de Redis OSS.

A continuación se muestra un ejemplo del uso de la CLI.

```
aws elasticache start-migration --replication-group-id test-cluster --customer-node-endpoint-list "Address='10.0.0.241',Port=6379"
```

Al ejecutar este comando, el nodo ElastiCache principal (en cada fragmento) se configura para convertirse en una réplica de su instancia de Redis OSS (en el fragmento correspondiente que posee las mismas ranuras en los redis habilitados para clústeres). El estado del ElastiCache clúster cambia a migración y los datos comienzan a migrar de la instancia de Redis OSS al nodo principal. ElastiCache Según el tamaño de los datos y la carga de la instancia de Redis OSS, la migración puede tardar un tiempo en completarse. Puede comprobar el progreso de la migración ejecutando el comando [redis-cli INFO en la instancia de Redis](#) OSS y en el nodo principal. ElastiCache

Tras una replicación correcta, todas las escrituras en las instancias de Redis OSS se propagan al clúster. ElastiCache Puede usar ElastiCache nodos para las lecturas. Sin embargo, no puede escribir en el ElastiCache clúster. Si un nodo ElastiCache principal tiene otros nodos de réplica conectados a

él, estos nodos de réplica seguirán replicándose desde el nodo ElastiCache principal. De esta forma, todos los datos del clúster de OSS de Redis se replican en ElastiCache todos los nodos del clúster.

Si un nodo ElastiCache principal no puede convertirse en una réplica de su instancia de Redis OSS, lo volverá a intentar varias veces antes de volver a pasar a ser principal. A continuación, el estado del ElastiCache clúster cambia a disponible y se envía un evento del grupo de replicación sobre el error al iniciar la migración. Para solucionar problemas como un error, compruebe lo siguiente:

- Busque el evento del grupo de reproducción. Utilice cualquier información específica del evento para arreglar el error de migración.
- Si el evento no ofrece ninguna información específica, asegúrese de que ha seguido las directrices que se indican en [Preparación de los nodos OSS de Redis de origen y destino para la migración](#).
- Asegúrese de que la configuración de enrutamiento de la VPC y las subredes permita el tráfico entre ElastiCache los nodos y las instancias de Redis OSS.
- Asegúrese de que el grupo de seguridad adjunto a sus instancias de Redis OSS permita el tráfico enlazado de entrada desde los nodos. ElastiCache
- Consulte los registros de Redis OSS de sus instancias de Redis OSS para obtener más información sobre los errores específicos de la replicación.

Verificación del progreso de migración de datos

Después de iniciar la migración de datos, puede hacer lo siguiente para realizar un seguimiento del progreso:

- Compruebe que Redis OSS `master_link_status` esté up en el INFO comando de los nodos ElastiCache principales. También puede encontrar esta información en la ElastiCache consola. Seleccione el clúster y, en las CloudWatch métricas, observe el estado de salud de Primary Link. Una vez que el valor alcanza 1, los datos se encuentran sincronizados.
- Puede comprobar que la ElastiCache réplica está en línea ejecutando el INFO comando en sus instancias de Redis OSS. Esta acción también ofrece información sobre el retardo de reproducción.
- Compruebe que el búfer de salida del cliente es bajo mediante el comando [CLIENT LIST](#) Redis OSS en sus instancias de Redis OSS.

Una vez completada la migración de datos, los datos se sincronizan con cualquier escritura nueva que llegue a los nodos principales del clúster de Redis OSS.

Finalización de la migración de datos

Cuando esté listo para pasar al ElastiCache clúster, utilice el comando `complete-migration` CLI con los siguientes parámetros:

- `--replication-group-id`: el identificador del grupo de reproducción.
- `--force`: un valor que obliga a la migración a detenerse sin asegurarse de que los datos se encuentren sincronizados.

A continuación, se muestra un ejemplo.

```
aws elasticache complete-migration --replication-group-id test-cluster
```

Al ejecutar este comando, el nodo ElastiCache principal (en cada fragmento) deja de replicarse desde la instancia de Redis OSS y pasa a ser principal. Esta promoción se completa normalmente en cuestión de minutos. Para confirmar la promoción a principal, compruebe el evento `Complete Migration successful for test-cluster`. En este punto, puede dirigir la aplicación para que ElastiCache escriba y lea. ElastiCache el estado del clúster debería cambiar de migratorio a disponible.

Si se produce un error en la promoción a la instancia principal, el nodo ElastiCache principal continúa replicándose desde la instancia de OSS de Redis. El ElastiCache clúster sigue en estado de migración y se envía un mensaje de evento del grupo de replicación sobre el error. Para solucionar este error, consulte lo siguiente:

- Compruebe el evento del grupo de reproducción. Utilice información específica del evento para arreglar el error.
- Puede que reciba un mensaje del evento sobre los datos no sincronizados. Si es así, asegúrese de que la instancia ElastiCache principal se pueda replicar desde su instancia de OSS de Redis y de que ambas estén sincronizadas. Si todavía quiere detener la migración, puede ejecutar el comando anterior con la opción `-force`.
- Es posible que reciba un mensaje de evento si se está reemplazando uno de ElastiCache los nodos. Puede volver a intentar el paso de migración completo después de que la sustitución finalice.

Realización de la migración de datos en línea con la consola

Puede usarlo AWS Management Console para migrar los datos del clúster al clúster de Redis OSS.

Para realizar la migración de datos online utilizando la consola

1. Inicie sesión en la consola y ábrala ElastiCache en <https://console.aws.amazon.com/elasticache/>.
2. Cree un nuevo clúster de Redis OSS o elija uno existente. Asegúrese de que el clúster cumple los siguientes requisitos:
 - La versión del motor OSS de Redis debe ser 5.0.6 o superior.
 - Su clúster de Redis OSS no debe tener habilitada la autenticación de Redis OSS.
 - La configuración de Redis OSS `protected-mode` debe estar establecida en `no`.
 - Si tiene una `bind` configuración en su configuración de Redis OSS, debe actualizarse para permitir las solicitudes de ElastiCache los nodos.
 - El número de bases de datos debe ser el mismo entre el ElastiCache nodo y el clúster de Redis OSS. Este valor se establece mediante la configuración `databases` de Redis OSS.
 - No se debe cambiar el nombre de los comandos de Redis OSS que modifican datos para permitir que la replicación de los datos se realice correctamente.
 - Para replicar los datos de su clúster de OSS de Redis ElastiCache, asegúrese de que haya suficiente CPU y memoria para gestionar esta carga adicional. Esta carga proviene del archivo RDB creado por su clúster de OSS de Redis y transferido a través de la red al nodo. ElastiCache
 - El clúster tiene el estado `available` (disponible).
3. Con el clúster seleccionado, elija `Migrate Data from Endpoint` (Migrar datos desde el punto de enlace) para `Actions` (Acciones).
4. En el cuadro de diálogo `Migrar datos desde un punto final`, introduzca la dirección IP y el puerto en los que está disponible el clúster de Redis OSS.

Important

La dirección IP debe ser exacta. Si introduce una dirección que no es correcta, la migración da error.

5. Elija `Start Migration` (Iniciar migración).

Cuando el clúster comienza la migración, cambia al estado Modifying (Modificando) y, a continuación, a Migrating (Migrando).

6. Monitoree el proceso de migración al elegir Events (Eventos) en el panel de navegación.

Puede detener la migración en cualquier punto durante el proceso. Para hacerlo, elija el clúster y Stop Data Migration (Detener migración de datos) para Actions (Acciones). Después, el clúster pasa al estado Available (Disponible).

Si la migración se realiza correctamente, el clúster pasa al estado Available (Disponible) y el registro del evento muestra lo siguiente:

```
Migration operation succeeded for replication group ElastiCacheClusterName.
```

Si la migración falla, el clúster pasa al estado Available (Disponible) y el registro del evento muestra lo siguiente:

```
Migration operation failed for replication group ElastiCacheClusterName.
```

Elección de regiones y zonas de disponibilidad

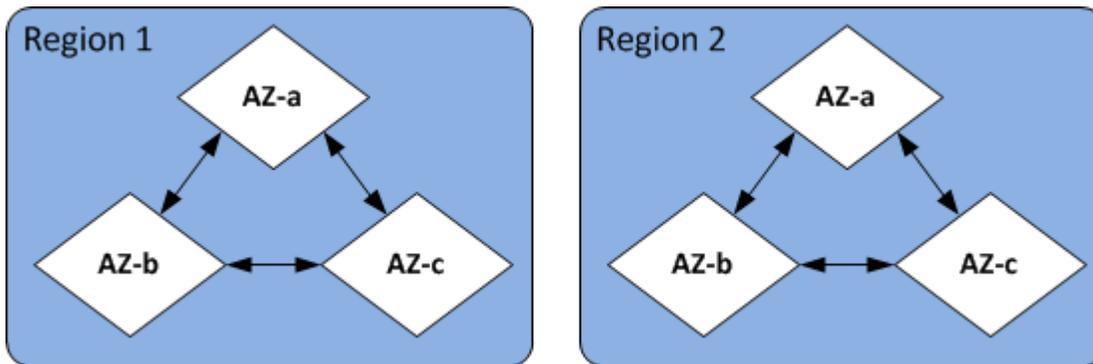
AWS Los recursos de computación en la nube se alojan en instalaciones de centros de datos de alta disponibilidad. Para proporcionar escalabilidad y fiabilidad adicionales, estas instalaciones de centros de datos se encuentran en ubicaciones físicas diferentes. Dichas ubicaciones están categorizadas por regiones y zonas de disponibilidad.

AWS Las regiones son extensas y están muy dispersas en distintas ubicaciones geográficas. Las zonas de disponibilidad son ubicaciones distintas dentro de una AWS región que están diseñadas para aislarlas de los errores en otras zonas de disponibilidad. Proporcionan una conectividad de red económica y de baja latencia con otras zonas de disponibilidad de la misma AWS región.

Important

Cada región es totalmente independiente. Cualquier ElastiCache actividad que inicie (por ejemplo, la creación de clústeres) se ejecuta únicamente en la región predeterminada actual.

Para crear o trabajar con un clúster de una región específica, use el punto de conexión de servicio regional correspondiente. Para obtener información acerca de los puntos de conexión del servicio, consulte [Regiones y puntos de enlace admitidos](#).



Regiones y zonas de disponibilidad

Temas

- [Ubicación de los nodos](#)
- [Regiones y puntos de enlace admitidos](#)
- [Uso de zonas locales con ElastiCache](#)
- [Uso de Outposts](#)

Ubicación de los nodos

Amazon ElastiCache admite la ubicación de todos los nodos de un clúster en una o varias zonas de disponibilidad (AZ). Además, si opta por ubicar sus nodos en varias zonas de disponibilidad (recomendado), ElastiCache le permite elegir la zona de disponibilidad para cada nodo o bien ElastiCache elegir la zona de disponibilidad por usted.

Al ubicar los nodos en distintas zonas de disponibilidad, se elimina la posibilidad de que un fallo, como un corte del suministro eléctrico, en una zona de disponibilidad provoque errores en todo el sistema. Las pruebas han demostrado que no hay ninguna diferencia de latencia importante entre ubicar todos los nodos en una zona de disponibilidad o repartirlos en varias zonas de disponibilidad.

Puede especificar una zona de disponibilidad para cada nodo cuando cree un clúster o agregando nodos cuando modifique un clúster existente. Para más información, consulte los siguientes temas:

- [Creación de un clúster](#)
- [Modificación de un ElastiCache clúster](#)
- [Adición de nodos a un clúster](#)

Regiones y puntos de enlace admitidos

Amazon ElastiCache está disponible en varias AWS regiones. Esto significa que puede lanzar ElastiCache clústeres en ubicaciones que cumplan sus requisitos. Por ejemplo, puede lanzarlos en la AWS región más cercana a sus clientes o en una AWS región concreta para cumplir determinados requisitos legales.

Cada región de se ha diseñado para que se encuentre totalmente aislada de las demás regiones de . Dentro de cada región hay varias zonas de disponibilidad (AZ). ElastiCache Las cachés sin servidor replican automáticamente los datos en varias zonas de disponibilidad (excepto us-west-1 cuando los datos se replican en dos zonas de disponibilidad) para lograr una alta disponibilidad. Al diseñar su propio ElastiCache clúster, puede optar por lanzar los nodos en diferentes zonas de disponibilidad para lograr la tolerancia a los errores. Para obtener más información acerca de las regiones y zonas de disponibilidad, consulte [Elección de regiones y zonas de disponibilidad](#) al comienzo de este tema.

Regiones en las ElastiCache que es compatible

Nombre de la región/ Región	Punto de conexión	Protocolo	
Región del este de EE. UU. (Ohio) us-east-2	elasticache.us-east-2.amazonaws.com	HTTPS	
Región del este de EE. UU. (Norte de Virginia) us-east-1	elasticache.us-east-1.amazonaws.com	HTTPS	
Región del oeste de EE. UU. (Norte de California) us-west-1	elasticache.us-west-1.amazonaws.com	HTTPS	
Región del oeste de EE. UU. (Oregón) us-west-2	elasticache.us-west-2.amazonaws.com	HTTPS	
Región de Canadá (centro) ca-central-1	elasticache.ca-central-1.amazonaws.com	HTTPS	
Región Canadá (Oeste) ca-west-1	elasticache.ca-west-1.amazonaws.com	HTTPS	
Asia-Pacífico (Yakarta) ap-southeast-3	elasticache.ap-southeast-3.amazonaws.com	HTTPS	

Nombre de la región/ Región	Punto de conexión	Protocolo	
Región de Asia-Pacífico (Bombay) ap-south-1	elasticache.ap-south-1.amazonaws.com	HTTPS	
Región de Asia Pacífico (Hyderabad) ap-south-2	elasticache.ap-south-2.amazonaws.com	HTTPS	
Asia Pacífico (Tokio) ap-northeast-1	elasticache.ap-northeast-1.amazonaws.com	HTTPS	
Región de Asia-Pacífico (Seúl) ap-northeast-2	elasticache.ap-northeast-2.amazonaws.com	HTTPS	
Región Asia-Pacífico (Osaka) ap-northeast-3	elasticache.ap-northeast-3.amazonaws.com	HTTPS	
Región de Asia-Pacífico (Singapur) ap-southeast-1	elasticache.ap-southeast-1.amazonaws.com	HTTPS	
Región de Asia-Pacífico (Sídney) ap-southeast-2	elasticache.ap-southeast-2.amazonaws.com	HTTPS	

Nombre de la región/ Región	Punto de conexión	Protocolo	
Región de Europa (Fráncfort) eu-central-1	elasticache.eu- central-1.am azonaws.com	HTTPS	
Región Europa (Zúrich) eu-central-2	elasticache.eu- central-2.am azonaws.com	HTTPS	
Región Europa (Estocolmo) eu-north-1	elasticache.eu- north-1.amaz onaws.com	HTTPS	
Región Medio Oriente (Baréin) me-south-1	elasticache.me- south-1.amaz onaws.com	HTTPS	
Región de Oriente Medio (EAU) me-central-1	elasticache.me- central-1.am azonaws.com	HTTPS	
Región de Europa (Irlanda) eu-west-1	elasticache.eu- west-1.amazo naws.com	HTTPS	
Región de Europa (Londres) eu-west-2	elasticache.eu- west-2.amazo naws.com	HTTPS	

Nombre de la región/ Región	Punto de conexión	Protocolo	
Región EU (París) eu-west-3	elasticache.eu-west-3.amazonaws.com	HTTPS	
Región Europa (Milán) eu-south-1	elasticache.eu-south-1.amazonaws.com	HTTPS	
Región Europa (España) eu-south-2	elasticache.eu-south-2.amazonaws.com	HTTPS	
Región de América del Sur (São Paulo) sa-east-1	elasticache.sa-east-1.amazonaws.com	HTTPS	
Región China (Pekín) cn-north-1	elasticache.cn-north-1.amazonaws.com.cn	HTTPS	
Región China (Ningxia) cn-northwest-1	elasticache.cn-northwest-1.amazonaws.com.cn	HTTPS	
Región de Asia-Pacífico (Hong Kong) ap-east-1	elasticache.ap-east-1.amazonaws.com	HTTPS	
Región África (Ciudad del Cabo) af-south-1	elasticache.af-south-1.amazonaws.com	HTTPS	

Nombre de la región/ Región	Punto de conexión	Protocolo
Región de Israel (Tel Aviv) il-central-1	elasticache.il-central-1.amazonaws.com	HTTPS
AWS GovCloud (EE. UU.-Oeste) us-gov-west-1	elasticache.us-gov-west-1.amazonaws.com	HTTPS
AWS GovCloud (EE. UU.-Este) us-gov-east-1	elasticache.us-gov-east-1.amazonaws.com	HTTPS

Para obtener información sobre el uso de AWS GovCloud (EE. UU.) con ElastiCache, consulte [Servicios en la región AWS GovCloud \(EE. UU.\): ElastiCache](#).

Algunas regiones admiten un subconjunto de tipos de nodos. Para ver una tabla de los tipos de nodos compatibles por AWS región, consulte [Tipos de nodos admitidos por región de AWS](#).

Para ver una tabla de AWS productos y servicios por región, consulta [Productos y servicios por región](#).

Uso de zonas locales con ElastiCache

Una zona local es una extensión de una AWS región que está geográficamente cerca de sus usuarios. Puede extender cualquier nube privada virtual (VPC) de una AWS región principal a una Zona Local creando una nueva subred y asignándola a la Zona Local. Cuando crea una subred en una zona local, la VPC también se amplía a dicha zona local. La subred de la zona local funciona igual que otras subredes de su VPC.

Al utilizar las Zonas Locales, puede colocar recursos, como un ElastiCache clúster, en varias ubicaciones cercanas a sus usuarios.

Al crear un ElastiCache clúster, puede elegir una subred en una zona local. Las zonas locales tienen sus propias conexiones a internet y admiten AWS Direct Connect. Por lo tanto, los recursos creados en una zona local pueden prestar servicio a los usuarios locales con comunicaciones de muy baja latencia. Para obtener más información, consulte [AWS Local Zones](#).

Una zona local se representa mediante un código de AWS región seguido de un identificador que indica la ubicación, por ejemplo `us-west-2-lax-1a`.

En este momento, las Local Zones disponibles son `us-west-2-lax-1a` y `us-west-2-lax-1b`.

Las siguientes limitaciones se aplican a ElastiCache las Zonas Locales:

- No se admiten almacenes de datos globales.
- No se admite la migración en línea.
- Las Local Zones admiten los siguientes tipos de nodos:
 - Generación actual:

Tipos de nodos M5: `cache.m5.large`, `cache.m5.xlarge`, `cache.m5.2xlarge`,
`cache.m5.4xlarge`, `cache.m5.12xlarge`, `cache.m5.24xlarge`

Tipos de nodos R5: `cache.r5.large`, `cache.r5.xlarge`, `cache.r5.2xlarge`,
`cache.r5.4xlarge`, `cache.r5.12xlarge`, `cache.r5.24xlarge`

Tipos de nodos T3: `cache.t3.micro`, `cache.t3.small`, `cache.t3.medium`

Habilitación de una zona local

1. Habilite la zona local en la consola de Amazon EC2.

Para obtener más información, consulte [Habilitación de Local Zones](#) en la Guía del usuario de Amazon EC2.

2. Cree una subred en la zona local.

Para obtener más información, consulte [Creación de una subred en la VPC](#) en la Guía del usuario de Amazon VPC.

3. Cree un grupo de ElastiCache subredes en la zona local.

Al crear un grupo de ElastiCache subredes, elija el grupo de zonas de disponibilidad para la zona local.

Para obtener más información, consulte [Creación de un grupo de subredes](#) en la Guía del ElastiCache usuario.

4. Cree un clúster ElastiCache (Redis OSS) que utilice la ElastiCache subred de la zona local. Para obtener más información, consulte uno de los siguientes temas:

- [Creación de un clúster de Redis OSS \(modo de clúster desactivado\) \(consola\)](#)
- [Creación de un clúster de Redis OSS \(modo de clúster activado\) \(consola\)](#)

Uso de Outposts

AWS Outposts es un servicio totalmente gestionado que extiende la AWS infraestructura, los servicios, las API y las herramientas a las instalaciones del cliente. Al proporcionar acceso local a la infraestructura AWS gestionada, AWS Outposts permite a los clientes crear y ejecutar aplicaciones in situ utilizando las mismas interfaces de programación que en AWS Regions, a la vez que utilizan recursos informáticos y de almacenamiento locales para reducir la latencia y las necesidades de procesamiento de datos locales. Un Outpost es un conjunto de capacidades AWS informáticas y de almacenamiento desplegadas en las instalaciones de un cliente. AWS opera, supervisa y administra esta capacidad como parte de una AWS región. Puedes crear subredes en tu Outpost y especificarlas al crear AWS recursos, como ElastiCache clústeres.

Note

En esta versión, se aplican las siguientes limitaciones:

- ElastiCache for Outposts solo admite las familias de nodos M5 y R5.
- Multi-AZ (no se admite la reproducción cruzada de Outpost).
- La migración en directo no es compatible.
- No se admiten instantáneas locales.
- Los registros del motor y los registros lentos no se pueden habilitar.
- ElastiCache on Outposts no admite CoIP.
- ElastiCache for Outposts no está disponible en las siguientes regiones: cn-north-1, cn-northwest-1 y ap-northeast-3.

Uso de Outposts con la consola OSS de Redis

1. [Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en https://console.aws.amazon.com/elasticache/.](https://console.aws.amazon.com/elasticache/)
2. En el panel de navegación, seleccione Cachés OSS de Redis.
3. Seleccione Crear caché OSS de Redis.
4. En Configuración del clúster, seleccione Diseñe su propia caché y Caché de clúster. Deje el modo de clúster desactivado. A continuación, cree un nombre y una descripción opcional para la caché.
5. Para la ubicación, elija En las instalaciones.
6. En la sección Local, verá el campo Outpost ID. Introduce el ID en el que se ejecutará el clúster.

Todas las demás configuraciones de la configuración del clúster pueden permanecer como predeterminadas.

7. En Conectividad, selecciona Crear un nuevo grupo de subredes e introduce el ID de VPC. Deje el resto como predeterminado y seleccione Siguiente.

Configurar opciones en las instalaciones

Puede seleccionar un Outpost disponible para agregar el clúster de caché o, si no hay Outposts disponibles, crear uno nuevo mediante los siguientes pasos:

En On-Premises options (Opciones en las instalaciones):

1. En la configuración de Redis OSS:
 - a. Nombre: introduzca un nombre para el clúster de Redis OSS
 - b. Descripción: introduzca una descripción para el clúster de OSS de Redis.
 - c. Compatibilidad de la versión del motor: la versión del motor se basa en la región de Outpost AWS
 - d. Puerto: acepte el puerto predeterminado, 6379. Si tiene un motivo para utilizar un puerto diferente, escriba el número de puerto.
 - e. Grupo de parámetros: utilice el menú desplegable para seleccionar un grupo de parámetros predeterminado o personalizado.
 - f. Tipo de nodo: las instancias disponibles se basan en la disponibilidad de Outposts. El asistente de portación de .NET para Outposts solo admite familias de nodos M5 y R5.

En la lista desplegable, seleccione Outposts y, a continuación, seleccione el tipo de nodo disponible que desee utilizar para este clúster. A continuación, seleccione Guardar.

- g. Número de réplicas: ingrese el número de réplicas de lectura que desea crear para este grupo de reproducción. Debe tener al menos una y no más de cinco réplicas de lectura. El valor predeterminado es 2.

Los nombres generados de forma automática de las réplicas de lectura siguen el mismo patrón que el nombre del clúster principal, con un guion y un número secuencial de tres dígitos agregados al final, que comienza por -002. Por ejemplo, si el grupo de reproducción se denomina MyGroup, entonces los nombres de los secundarios serían MyGroup-002, MyGroup-003, MyGroup-004, MyGroup-005, MyGroup-006.

2. En Conectividad:

- a. Grupo de subredes: en la lista, seleccione Create new (Crear nuevo).
 - Name: ingrese un nombre para el grupo de la subred.
 - Descripción: ingrese una descripción para el grupo de la subred.
 - ID de VPC: el ID de VPC debe coincidir con la VPC de Outpost. Si selecciona una VPC que no tiene ID de subred en los Outposts, la lista se devolverá vacía.
 - Zona de disponibilidad o Outpost: seleccione el Outpost que utiliza.
 - ID de subred: seleccione un ID de subred que se encuentre disponible para el Outpost. Si no hay ID de subred disponibles, debe crearlos. Para obtener más información, consulte [Creación de una subred](#).
- b. Seleccione Crear.

Visualización de detalles del clúster de Outpost

En la página de la lista de OSS de Redis, seleccione un clúster que pertenezca a un AWS Outpost y tenga en cuenta lo siguiente al ver los detalles del clúster:

- Zona de disponibilidad: representará el puesto de avanzada, utilizando un ARN (nombre de recurso de Amazon) y AWS el número de recurso.
- Nombre del puesto de avanzada: el nombre del puesto de avanzada. AWS

Uso de Outposts con la CLI AWS

Puede usar el AWS Command Line Interface (AWS CLI) para controlar varios AWS servicios desde la línea de comandos y automatizarlos mediante scripts. Puede usar la AWS CLI para operaciones ad hoc (únicas).

Descargar y configurar el AWS CLI

Se AWS CLI ejecuta en Windows, macOS o Linux. Utilice el siguiente procedimiento para descargarlo y configurarlo.

Para descargar, instalar y configurar la CLI

1. Descargue la AWS CLI en la página web de la [interfaz de línea de AWS comandos](#).
2. Siga las instrucciones para [instalar la AWS CLI](#) y [configurar la AWS CLI](#) de la Guía del AWS Command Line Interface usuario.

Uso de la AWS CLI con Outposts

Utilice la siguiente operación de CLI para crear un clúster de caché que utilice Outposts:

- [create-cache-cluster](#)— Con esta operación, el `outpost-mode` parámetro acepta un valor que especifica si los nodos del clúster de caché se crean en un solo Outpost o en varios Outposts.

Note

En este momento, solo el modo `single-outpost` es compatible.

```
aws elasticache create-cache-cluster \  
  --cache-cluster-id cache cluster id \  
  --outpost-mode single-outpost \  
  \
```

Trabajando con ElastiCache

En esta sección encontrará detalles sobre cómo gestionar los distintos componentes de la ElastiCache implementación.

Temas

- [Instantánea y restauración](#)
- [Versiones del motor y actualizaciones](#)
- [ElastiCache mejores prácticas y estrategias de almacenamiento en caché](#)
- [Administración de un clúster de autodiseño](#)
- [Escalado ElastiCache \(Redis OSS\)](#)
- [Introducción a JSON en ElastiCache \(Redis OSS\)](#)
- [Etiquetado de los recursos de ElastiCache](#)
- [Uso del enfoque Well-Architected de Amazon ElastiCache](#)
- [Pasos comunes para la solución de problemas y prácticas recomendadas](#)
- [Pasos adicionales de solución de problemas](#)

Instantánea y restauración

ElastiCache Las memorias caché de Amazon que ejecutan Redis OSS pueden hacer copias de seguridad de sus datos mediante la creación de una instantánea. Puede utilizar la copia de seguridad para restaurar una caché o para propagar los datos iniciales de una nueva caché. La copia de seguridad se compone de los metadatos de la caché, junto con todos los datos de la caché. Todas las copias de seguridad se escriben en Amazon Simple Storage Service (Amazon S3), lo que proporciona un almacenamiento duradero. En cualquier momento, puede restaurar sus datos creando una nueva caché de Redis OSS y rellenándola con los datos de una copia de seguridad. Con ElastiCache, puede gestionar las copias de seguridad mediante la AWS Management Console, AWS Command Line Interface (AWS CLI) y la API. ElastiCache

Si está pensando en eliminar la caché y es importante conservar los datos, puede tomar una medida de precaución adicional. Para ello, cree primero una copia de seguridad manual, compruebe que su estado sea disponible y elimine la caché. Con este procedimiento, se asegura de que seguirá disponiendo de los datos de la caché aunque haya un error en la copia de seguridad. Puede

volver a intentar realizar una copia de seguridad siguiendo las prácticas recomendadas descritas anteriormente.

Temas

- [Limitaciones del proceso de copia de seguridad](#)
- [El impacto en el rendimiento de las copias de seguridad de los clústeres de autodiseño](#)
- [Programación de copias de seguridad automáticas](#)
- [Copias de seguridad manuales](#)
- [Creación de una copia de seguridad final](#)
- [Descripción de copias de seguridad](#)
- [Copiar copias de seguridad](#)
- [Exportación de una copia de seguridad](#)
- [Restauración desde una copia de seguridad a una nueva caché](#)
- [Eliminación de una copia de seguridad](#)
- [Etiquetado de copias de seguridad](#)
- [Inicialización de datos en un nuevo clúster de autodiseño con una copia de seguridad creada externamente](#)

Limitaciones del proceso de copia de seguridad

Debe tener en cuenta las limitaciones siguientes a la hora de planear o realizar copias de seguridad:

- El backup y la restauración solo son compatibles con las cachés que se ejecutan en Redis OSS o Serverless Memcached.
- En el caso de los clústeres de Redis OSS (modo de clúster desactivado), las copias de seguridad y la restauración no se admiten en los nodos. `cache.t1.micro` Todos los demás tipos de nodos de caché son compatibles con la característica.
- En el caso de los clústeres OSS de Redis (modo de clúster activado), se admiten la copia de seguridad y la restauración en todos los tipos de nodos.
- Durante cualquier período contiguo de 24 horas, no puede crear más de 24 copias de seguridad manuales por caché sin servidor. En el caso de los clústeres de diseño propio de Redis OSS, no puede crear más de 20 copias de seguridad manuales por nodo del clúster.
- Redis OSS (modo de clúster habilitado) solo admite la realización de copias de seguridad a nivel de clúster (para la API o CLI, el nivel de grupo de replicación). Redis OSS (modo de clúster

habilitado) no admite la realización de copias de seguridad a nivel de fragmentos (para la API o la CLI, a nivel de grupo de nodos).

- Durante el proceso de copia de seguridad, no puede ejecutar ninguna otra operación de API o CLI en la memoria caché sin servidor. Puede ejecutar operaciones de API o CLI en un clúster de diseño propio durante la copia de seguridad.
- Si utiliza cachés con niveles de datos, no puede exportar una copia de seguridad a Amazon S3.
- Puede restaurar una copia de seguridad de un clúster utilizando el tipo de nodo r6gd solo en clústeres que utilicen el tipo de nodo r6gd.

El impacto en el rendimiento de las copias de seguridad de los clústeres de autodiseño

Las copias de seguridad en las cachés sin servidor son transparentes para la aplicación y no afectan al rendimiento. Sin embargo, al crear copias de seguridad para clústeres de autodiseño, el rendimiento puede verse afectado en cierta medida en función de la memoria reservada disponible. Los clústeres de diseño propio no están disponibles con ElastiCache Memcached, pero sí con Redis OSS. ElastiCache

A continuación, se ofrecen algunas pautas para mejorar el rendimiento de la realización de copias de seguridad para clústeres de autodiseño.

- Defina el `reserved-memory-percent` parámetro: para mitigar el exceso de paginación, le recomendamos que configure el parámetro `reserved-memory-percent`. Este parámetro evita que Redis OSS consuma toda la memoria disponible del nodo y puede ayudar a reducir la cantidad de paginación. También puede ver mejoras de rendimiento simplemente usando un nodo de mayor tamaño. Para obtener más información sobre la memoria reservada y los parámetros, consulte [reserved-memory-percent Administración de la memoria reservada](#)
- Cree copias de seguridad a partir de una réplica de lectura: si ejecuta Redis OSS en un grupo de nodos con más de un nodo, puede realizar una copia de seguridad del nodo principal o de una de las réplicas de lectura. Dado los orígenes del sistema requeridos durante BGSAVE, recomendamos que cree copias de seguridad de una de las réplicas de lectura. Aunque la copia de seguridad se está creando a partir de la réplica, el nodo principal no se ve afectado por los requisitos de recursos del comando BGSAVE. El nodo principal puede continuar atendiendo a las solicitudes sin ralentizarse.

Para hacer esto, consulte [Creación de una copia de seguridad manual \(consola\)](#) y en el campo Cluster Name (Nombre del clúster) en la ventana Create Backup (Crear copia de seguridad), elija una réplica en lugar del nodo primario predeterminado.

Si elimina un grupo de replicación y solicita una copia de seguridad final, ElastiCache siempre toma la copia de seguridad del nodo principal. Esto garantiza la captura de los datos más recientes de Redis OSS antes de eliminar el grupo de replicación.

Programación de copias de seguridad automáticas

Puede habilitar las copias de seguridad automáticas para cualquier caché sin servidor o clúster de diseño propio de Redis OSS. Cuando las copias de seguridad automáticas están habilitadas, ElastiCache crea una copia de seguridad de la memoria caché a diario. Esto no afecta a la caché y el cambio es inmediato. Las copias de seguridad automáticas pueden ayudarle a protegerse frente a la pérdida de datos. En caso de error, puede crear una nueva caché y restaurar los datos de la copia de seguridad más reciente. El resultado es una caché iniciada en activo; es decir, ya está cargada con sus datos y lista para el uso. Para obtener más información, consulte [Restauración desde una copia de seguridad a una nueva caché](#).

Al programar copias de seguridad automáticas, debe planificar los ajustes siguientes:

- Hora de inicio de la copia de seguridad: hora del día en la que se ElastiCache comienza a crear una copia de seguridad. Puede configurar la ventana de copia de seguridad en el momento que le resulte más conveniente. Si no especifica una ventana de copia de seguridad, ElastiCache asigna una automáticamente.
- Límite de retención de copia de seguridad: número de días que se retiene la copia de seguridad en Amazon S3. Por ejemplo, si establece el límite de retención en 5, una copia de seguridad que se realice hoy se conservaría durante 5 días. Al finalizar el límite de retención, la copia de seguridad se eliminará automáticamente.

El límite máximo de retención de copia de seguridad es de 35 días. Si el límite de retención de copia de seguridad se establece en 0, las copias de seguridad se deshabilitarán en la caché.

Puede activar o desactivar las copias de seguridad automáticas al crear una nueva caché o al actualizar una caché OSS de Redis existente mediante la ElastiCache consola AWS CLI, la o la ElastiCache API. Esto se hace marcando la casilla Habilitar copias de seguridad automáticas en la sección Configuración avanzada de Redis OSS.

Copias de seguridad manuales

Además de las copias de seguridad automáticas, puede crear una copia de seguridad manual en cualquier momento. A diferencia de las copias de seguridad automáticas, que se eliminan automáticamente después de un periodo de retención determinado, las copias de seguridad manuales no tienen periodo de retención que determine su eliminación automática. Aunque elimine la memoria caché, se conservarán todas las copias de seguridad manuales de esa memoria caché. Si ya no desea conservar una copia de seguridad manual, deberá eliminarla de forma explícita.

Además de crear una copia de seguridad manual, puede crear copias de seguridad manuales de las maneras siguientes:

- [Copiar copias de seguridad](#). No importa si la copia de seguridad de origen se creó automáticamente o manualmente.
- [Creación de una copia de seguridad final](#). Cree una copia de seguridad inmediatamente antes de eliminar un clúster o nodo.

Puede crear una copia de seguridad manual de una memoria caché mediante la AWS Management Console, la CLI o la ElastiCache API.

Creación de una copia de seguridad manual (consola)

Para crear una copia de seguridad de una caché (consola)

1. [Inicie sesión en la consola Amazon EC2 AWS Management Console y ábrala en https://console.aws.amazon.com/ec2/](https://console.aws.amazon.com/ec2/).
2. En el panel de navegación, elija las cachés OSS de Redis.
3. Seleccione la casilla situada a la izquierda del nombre de la memoria caché de la que quiere hacer una copia de seguridad.
4. Elija Backup (Copia de seguridad).
5. En el cuadro de diálogo Create Backup, escriba un nombre para la copia de seguridad en el cuadro Backup Name. Recomendamos que el nombre indique que el clúster proviene de una copia de seguridad y especifique la fecha y la hora en que se creó la copia de seguridad.

Las restricciones para la asignación de nombres de clúster son las siguientes:

- Deben contener entre 1 y 40 caracteres alfanuméricos o guiones.

- Deben comenzar por una letra.
- No pueden contener dos guiones consecutivos.
- No pueden terminar con un guion.

6. Elija Create Backup.

El estado del clúster cambia a snapshotting.

Creación de una copia de seguridad manual (AWS CLI)

Copia de seguridad manual de una memoria caché sin servidor con AWS CLI

Para crear una copia de seguridad manual de una caché mediante el AWS CLI, utilice la `create-serverless-snapshot` AWS CLI operación con los siguientes parámetros:

- `--serverless-cache-name`: el nombre de la memoria caché sin servidor cuya copia de seguridad está realizando.
- `--serverless-cache-snapshot-name`: nombre de la instantánea que se creará.

Para Linux, macOS o Unix:

```
aws elasticache create-serverless-snapshot \  
    --serverless-cache-name CacheName \  
    --serverless-cache-snapshot-name bkup-20231127
```

Para Windows:

```
aws elasticache create-serverless-snapshot ^  
    --serverless-cache-name CacheName ^  
    --serverless-cache-snapshot-name bkup-20231127
```

Copia de seguridad manual de un clúster de diseño propio con AWS CLI

Para crear una copia de seguridad manual de un clúster de diseño propio mediante el AWS CLI, utilice la `create-snapshot` AWS CLI operación con los siguientes parámetros:

- `--cache-cluster-id`

- Si el clúster que está copiando no tiene nodos de réplica, `--cache-cluster-id` será el nombre del clúster cuya copia de seguridad está creando; por ejemplo, *mycluster*.
- Si el clúster que está copiando tiene uno o varios nodos de réplica, `--cache-cluster-id` será el nombre del nodo del clúster que desea usar para la copia de seguridad. Por ejemplo, el nombre puede ser *mycluster-002*.

Utilice este parámetro únicamente cuando haga una copia de seguridad de un clúster de Redis OSS (modo de clúster desactivado).

- `--replication-group-id`— Nombre del clúster OSS (modo de clúster habilitado) de Redis (CLI/API: un grupo de replicación) que se utilizará como fuente de la copia de seguridad. Utilice este parámetro al hacer una copia de seguridad de un clúster de Redis OSS (modo de clúster activado).
- `--snapshot-name`: nombre de la instantánea que se creará.

Las restricciones para la asignación de nombres de clúster son las siguientes:

- Deben contener entre 1 y 40 caracteres alfanuméricos o guiones.
- Deben comenzar por una letra.
- No pueden contener dos guiones consecutivos.
- No pueden terminar con un guion.

Ejemplo 1: hacer una copia de seguridad de un clúster de Redis OSS (modo de clúster desactivado) que no tiene nodos de réplica

La siguiente AWS CLI operación crea la copia `bkup-20150515` de seguridad desde el clúster OSS de Redis (modo de clúster desactivado) `myNonClusteredRedis` que no tiene réplicas de lectura.

Para Linux, macOS o Unix:

```
aws elasticache create-snapshot \  
  --cache-cluster-id myNonClusteredRedis \  
  --snapshot-name bkup-20150515
```

Para Windows:

```
aws elasticache create-snapshot ^
  --cache-cluster-id myNonClusteredRedis ^
  --snapshot-name bkup-20150515
```

Ejemplo 2: hacer una copia de seguridad de un clúster de Redis OSS (modo de clúster desactivado) con nodos de réplica

La siguiente AWS CLI operación crea la copia *bkup-20150515* de seguridad desde el clúster de Redis OSS (modo de clúster desactivado). *myNonClusteredRedis* Esta copia de seguridad dispone de una o más réplicas de lectura.

Para Linux, macOS o Unix:

```
aws elasticache create-snapshot \
  --cache-cluster-id myNonClusteredRedis-001 \
  --snapshot-name bkup-20150515
```

Para Windows:

```
aws elasticache create-snapshot ^
  --cache-cluster-id myNonClusteredRedis-001 ^
  --snapshot-name bkup-20150515
```

Ejemplo de resultado: hacer una copia de seguridad de un clúster de Redis OSS (modo de clúster desactivado) con nodos de réplica

La salida de la operación será similar a lo que se indica a continuación.

```
{
  "Snapshot": {
    "Engine": "redis",
    "CacheParameterGroupName": "default.redis6.x",
    "VpcId": "vpc-91280df6",
    "CacheClusterId": "myNonClusteredRedis-001",
    "SnapshotRetentionLimit": 0,
    "NumCacheNodes": 1,
    "SnapshotName": "bkup-20150515",
    "CacheClusterCreateTime": "2017-01-12T18:59:48.048Z",
    "AutoMinorVersionUpgrade": true,
    "PreferredAvailabilityZone": "us-east-1c",
    "SnapshotStatus": "creating",
```

```

    "SnapshotSource": "manual",
    "SnapshotWindow": "08:30-09:30",
    "EngineVersion": "6.0",
    "NodeSnapshots": [
      {
        "CacheSize": "",
        "CacheNodeId": "0001",
        "CacheNodeCreateTime": "2017-01-12T18:59:48.048Z"
      }
    ],
    "CacheSubnetGroupName": "default",
    "Port": 6379,
    "PreferredMaintenanceWindow": "wed:07:30-wed:08:30",
    "CacheNodeType": "cache.m3.2xlarge",
    "DataTiering": "disabled"
  }
}

```

Ejemplo 3: Hacer una copia de seguridad de un clúster para Redis OSS (modo de clúster activado)

La siguiente AWS CLI operación crea la copia `bkup-20150515` de seguridad desde el clúster OSS de Redis (modo de clúster activado). `myClusteredRedis` Tenga en cuenta que debe usar el `--replication-group-id` en lugar del `--cache-cluster-id` para identificar el origen.

Para Linux, macOS o Unix:

```

aws elasticache create-snapshot \
  --replication-group-id myClusteredRedis \
  --snapshot-name bkup-20150515

```

Para Windows:

```

aws elasticache create-snapshot ^
  --replication-group-id myClusteredRedis ^
  --snapshot-name bkup-20150515

```

Ejemplo de resultado: hacer una copia de seguridad de un clúster de Redis OSS (modo de clúster activado)

La salida de esta operación será similar a lo que se indica a continuación.

```
{
```

```
"Snapshot": {
  "Engine": "redis",
  "CacheParameterGroupName": "default.redis6.x.cluster.on",
  "VpcId": "vpc-91280df6",
  "NodeSnapshots": [
    {
      "CacheSize": "",
      "NodeGroupId": "0001"
    },
    {
      "CacheSize": "",
      "NodeGroupId": "0002"
    }
  ],
  "NumNodeGroups": 2,
  "SnapshotName": "bkup-20150515",
  "ReplicationGroupId": "myClusteredRedis",
  "AutoMinorVersionUpgrade": true,
  "SnapshotRetentionLimit": 1,
  "AutomaticFailover": "enabled",
  "SnapshotStatus": "creating",
  "SnapshotSource": "manual",
  "SnapshotWindow": "10:00-11:00",
  "EngineVersion": "6.0",
  "CacheSubnetGroupName": "default",
  "ReplicationGroupDescription": "2 shards 2 nodes each",
  "Port": 6379,
  "PreferredMaintenanceWindow": "sat:03:30-sat:04:30",
  "CacheNodeType": "cache.r3.large",
  "DataTiering": "disabled"
}
}
```

Temas relacionados de

Para obtener más información, consulte [create-snapshot](#) en la Referencia de comandos de la AWS CLI .

Creación de una copia de seguridad final

Puede crear una copia de seguridad final mediante la ElastiCache consola AWS CLI, la o la ElastiCache API.

Creación de una copia de seguridad final (consola)

Puede crear una copia de seguridad final al eliminar una caché sin servidor de Redis OSS o un clúster de diseño propio mediante la consola. ElastiCache

Para crear una copia de seguridad final al eliminar una caché, en el cuadro de diálogo de eliminación, seleccione Sí en Crear copia de seguridad y asigne un nombre a la copia de seguridad.

Temas relacionados de

- [Usando el AWS Management Console](#)
- [Eliminación de un grupo de reproducción \(consola\)](#)

Creación de una copia de seguridad final (AWS CLI)

Puede crear una copia de seguridad final al eliminar una caché mediante AWS CLI.

Temas

- [Al eliminar una caché sin servidor](#)
- [Al eliminar un clúster de diseño propio de Redis OSS sin réplicas de lectura](#)
- [Al eliminar un clúster de OSS de Redis con réplicas de lectura](#)

Al eliminar una caché sin servidor

Para crear una copia de seguridad final, utilice la `delete-serverless-cache` AWS CLI operación con los siguientes parámetros.

- `--serverless-cache-name`: nombre de la caché que va a eliminar.
- `--final-snapshot-name`: nombre de la copia de seguridad.

El siguiente código crea la copia de seguridad final `bkup-20231127-final` al eliminar la caché `myserverlesscache`.

Para Linux, macOS o Unix:

```
aws elasticache delete-serverless-cache \  
    --serverless-cache-name myserverlesscache \  
    --final-snapshot-name bkup-20231127-final
```

Para Windows:

```
aws elasticache delete-serverless-cache ^  
    --serverless-cache-name myserverlesscache ^  
    --final-snapshot-name bkup-20231127-final
```

Para obtener más información, consulte [delete-serverless-cache](#) la Referencia de AWS CLI comandos.

Al eliminar un clúster de diseño propio de Redis OSS sin réplicas de lectura

Para crear una copia de seguridad final para un clúster de diseño propio sin réplicas de lectura, utilice la `delete-cache-cluster` AWS CLI operación con los siguientes parámetros.

- `--cache-cluster-id`: nombre del clúster que va a eliminar.
- `--final-snapshot-identifier`: nombre de la copia de seguridad.

El siguiente código crea la copia de seguridad final `bkup-20150515-final` al eliminar el clúster `myRedisCluster`.

Para Linux, macOS o Unix:

```
aws elasticache delete-cache-cluster \  
    --cache-cluster-id myRedisCluster \  
    --final-snapshot-identifier bkup-20150515-final
```

Para Windows:

```
aws elasticache delete-cache-cluster ^  
    --cache-cluster-id myRedisCluster ^  
    --final-snapshot-identifier bkup-20150515-final
```

Para obtener más información, consulte la [delete-cache-cluster](#) Referencia de AWS CLI comandos.

Al eliminar un clúster de OSS de Redis con réplicas de lectura

Para crear una copia de seguridad final al eliminar un grupo de replicación, utilice la `delete-replication-group` AWS CLI operación con los siguientes parámetros:

- `--replication-group-id`: nombre del grupo de replicación que se va a eliminar.
- `--final-snapshot-identifier`: nombre de la copia de seguridad final.

El siguiente código realiza la copia de seguridad final `bkup-20150515-final` al eliminar el grupo de reproducción `myReplGroup`.

Para Linux, macOS o Unix:

```
aws elasticache delete-replication-group \  
    --replication-group-id myReplGroup \  
    --final-snapshot-identifier bkup-20150515-final
```

Para Windows:

```
aws elasticache delete-replication-group ^  
    --replication-group-id myReplGroup ^  
    --final-snapshot-identifier bkup-20150515-final
```

Para obtener más información, consulte [delete-replication-group](#) la Referencia de AWS CLI comandos.

Descripción de copias de seguridad

Los procedimientos siguientes muestran cómo mostrar una lista de sus copias de seguridad. Si lo desea, también puede ver los detalles de una copia de seguridad determinada.

Descripción de copias de seguridad (consola)

Para mostrar las copias de seguridad mediante el AWS Management Console

1. Inicie sesión AWS Management Console y abra la ElastiCache consola en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación, seleccione Backups (Copias de seguridad).
3. Para ver los detalles de una copia de seguridad determinada, elija la casilla situada a la izquierda del nombre de la copia de seguridad.

Descripción de las copias de seguridad sin servidor (AWS CLI)

Para mostrar una lista de las copias de seguridad sin servidor y, de forma opcional, los detalles de una copia de seguridad específica, use la operación `describe-serverless-cache-snapshots` de la CLI.

Ejemplos

La operación siguiente usa el parámetro `--max-records` para mostrar hasta 20 copias de seguridad asociadas a su cuenta. La omisión del parámetro `--max-records` permite mostrar hasta 50 copias de seguridad.

```
aws elasticache describe-serverless-cache-snapshots --max-records 20
```

La operación siguiente usa el parámetro `--serverless-cache-name` para mostrar solo las copias de seguridad asociadas a la caché `my-cache`.

```
aws elasticache describe-serverless-cache-snapshots --serverless-cache-name my-cache
```

La operación siguiente usa el parámetro `--serverless-cache-snapshot-name` para mostrar los detalles de la copia de seguridad `my-backup`.

```
aws elasticache describe-serverless-cache-snapshots --serverless-cache-snapshot-name my-backup
```

Para obtener más información, consulte [describe-serverless-cache-snapshots en la Referencia de comandos](#). AWS CLI

Descripción de las copias de seguridad de clústeres de autodiseño (AWS CLI)

Para mostrar una lista de las copias de seguridad de clústeres de autodiseño y, de forma opcional, los detalles de una copia de seguridad específica, use la operación `describe-snapshots` de la CLI.

Ejemplos

La operación siguiente usa el parámetro `--max-records` para mostrar hasta 20 copias de seguridad asociadas a su cuenta. La omisión del parámetro `--max-records` permite mostrar hasta 50 copias de seguridad.

```
aws elasticache describe-snapshots --max-records 20
```

La operación siguiente usa el parámetro `--cache-cluster-id` para mostrar solo las copias de seguridad asociadas al clúster `my-cluster`.

```
aws elasticache describe-snapshots --cache-cluster-id my-cluster
```

La operación siguiente usa el parámetro `--snapshot-name` para mostrar los detalles de la copia de seguridad `my-backup`.

```
aws elasticache describe-snapshots --snapshot-name my-backup
```

Para obtener más información, consulte [describe-snapshots en la Referencia de comandos](#). AWS CLI

Copiar copias de seguridad

Puede realizar una copia de cualquier copia de seguridad, independientemente de si se creó de forma automática o manual. También puede exportar su copia de seguridad para poder acceder a ella desde fuera ElastiCache. Para obtener información sobre cómo exportar la copia de seguridad, consulte [Exportación de una copia de seguridad](#).

Los procedimientos siguientes muestran cómo copiar una copia de seguridad.

Copia de copias de seguridad (consola)

Para copiar una copia de seguridad (consola)

1. Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en <https://console.aws.amazon.com/elasticache/>.
2. Para ver una lista de las copias de seguridad, en el panel de navegación izquierdo, elija Backups (Copias de seguridad).
3. En la lista de copias de seguridad, active la casilla situada a la izquierda del nombre de la copia de seguridad que desea copiar.
4. Seleccione Acciones y Copiar.
5. En el cuadro New backup name (Nombre del nuevo backup), escriba un nombre para la nueva copia de seguridad.
6. Elija Copiar.

Copia de una copia de seguridad sin servidor (AWS CLI)

Para copiar una copia de seguridad de una caché sin servidor, utilice la operación `copy-serverless-cache-snapshot`.

Parámetros

- `--source-serverless-cache-snapshot-name`: nombre de la copia de seguridad que se copiará.
- `--target-serverless-cache-snapshot-name`: nombre de la copia de seguridad.

El ejemplo siguiente realiza una copia de una copia de seguridad automática.

Para Linux, macOS o Unix:

```
aws elasticache copy-serverless-cache-snapshot \  
  --source-serverless-cache-snapshot-name automatic.my-cache-2023-11-27-03-15 \  
  --target-serverless-cache-snapshot-name my-backup-copy
```

Para Windows:

```
aws elasticache copy-serverless-cache-snapshot ^  
  --source-serverless-cache-snapshot-name automatic.my-cache-2023-11-27-03-15 ^  
  --target-serverless-cache-snapshot-name my-backup-copy
```

Para obtener más información, consulte [copy-serverless-cache-snapshot](#) en la AWS CLI.

Copia de una copia de seguridad de un clúster de autodiseño (AWS CLI)

Para copiar una copia de seguridad de un clúster de autodiseño, utilice la operación `copy-snapshot`.

Parámetros

- `--source-snapshot-name`: nombre de la copia de seguridad que se copiará.
- `--target-snapshot-name`: nombre de la copia de seguridad.
- `--target-bucket`: reservado para la exportación de la copia de seguridad. No use este parámetro cuando realice una copia de una copia de seguridad. Disponible únicamente para las cachés sin servidor de Redis OSS y los clústeres de diseño propio de Redis OSS. Para obtener más información, consulte [Exportación de una copia de seguridad](#).

El ejemplo siguiente realiza una copia de una copia de seguridad automática.

Para Linux, macOS o Unix:

```
aws elasticache copy-snapshot \  
  --source-snapshot-name automatic.my-redis-primary-2014-03-27-03-15 \  
  --target-snapshot-name my-backup-copy
```

Para Windows:

```
aws elasticache copy-snapshot ^  
  --source-snapshot-name automatic.my-redis-primary-2014-03-27-03-15 ^  
  --target-snapshot-name my-backup-copy
```

Para obtener más información, consulte [copy-snapshot](#) en la AWS CLI.

Exportación de una copia de seguridad

Amazon ElastiCache admite la exportación de su copia de seguridad ElastiCache (Redis OSS) a un bucket de Amazon Simple Storage Service (Amazon S3), lo que le permite acceder a ella desde el exterior. ElastiCache Puede exportar una copia de seguridad mediante la ElastiCache consola AWS CLI, la o la ElastiCache API.

Exportar una copia de seguridad puede resultar útil si necesitas lanzar un clúster en otra AWS región. Puede exportar sus datos a una AWS región, copiar el archivo.rdb a la nueva AWS región y, a continuación, usar ese archivo.rdb para almacenar la nueva caché en lugar de esperar a que el nuevo clúster se llene debido al uso. Para obtener información acerca de la propagación de datos en un nuevo clúster, consulte [Inicialización de datos en un nuevo clúster de autodiseño con una copia de seguridad creada externamente](#). Otra razón por la que quizás quieras exportar los datos de la caché es usar el archivo.rdb para procesarlos sin conexión.

Important

- La ElastiCache copia de seguridad y el bucket de Amazon S3 en el que desea copiarla deben estar en la misma AWS región.

Aunque las copias de seguridad copiadas en un bucket de Amazon S3 se encuentran cifradas, recomendamos encarecidamente que no conceda acceso a otras personas al bucket de Amazon S3 en el que desea almacenar las copias de seguridad.

- La exportación de una copia de seguridad a Amazon S3 no se admite en clústeres que utilizan la organización de datos en niveles. Para obtener más información, consulte [Organización de datos en niveles](#).
- La exportación de una copia de seguridad está disponible para los clústeres de diseño propio de Redis OSS, Redis OSS sin servidor y Memcached sin servidor. La exportación de una copia de seguridad no está disponible para los clústeres de Memcached de diseño propio.

Para poder exportar una copia de seguridad a un bucket de Amazon S3, debe tener un bucket de Amazon S3 en la misma AWS región que la copia de seguridad. Conceda ElastiCache acceso al depósito. Los primeros dos pasos muestran cómo realizar esto último.

Paso 1: Crear un bucket de Amazon S3

En los siguientes pasos, utilice la consola de Amazon S3 para crear un bucket de Amazon S3 en el que exporte y almacene la ElastiCache copia de seguridad.

Creación de un bucket de Amazon S3

1. Inicie sesión en la consola de Amazon S3 AWS Management Console y ábrala en <https://console.aws.amazon.com/s3/>.
2. Seleccione la opción Crear bucket.
3. En Create a Bucket - Select a Bucket Name and Region, haga lo siguiente:
 - a. En Bucket Name (Nombre del bucket), escriba un nombre para el bucket de Amazon S3.

El nombre del bucket de Amazon S3 debe estar conforme con DNS. De lo contrario, no ElastiCache podrá acceder al archivo de respaldo. Las reglas para la conformidad con DNS son:

- Los nombres deben tener un mínimo de 3 y un máximo de 63 caracteres de largo.
 - Los nombres deben ser una serie de una o más etiquetas separadas por un punto (.) en el que cada etiqueta:
 - Comienza por una letra minúscula o un número.
 - Termina con una letra minúscula o un número.
 - Solo contiene letras minúsculas, números y guiones.
 - Los nombres no pueden tener el formato de una dirección IP (por ejemplo, 192.0.2.0).
- b. En la lista de regiones, selecciona una AWS región para tu bucket de Amazon S3. Esta AWS región debe ser la misma AWS región que la ElastiCache copia de seguridad que desea exportar.
 - c. Seleccione Crear.

Para obtener más información sobre la creación de un bucket de Amazon S3, consulte la sección de [Creación de un bucket](#) en la Guía del usuario de Amazon Simple Storage Service.

Paso 2: Conceda ElastiCache acceso a su bucket de Amazon S3

ElastiCache Para poder copiar una instantánea en un bucket de Amazon S3, debe actualizar su política de bucket para conceder ElastiCache acceso al bucket.

⚠ Warning

Aunque las copias de seguridad copiadas en un bucket de Amazon S3 se encuentran cifradas, cualquier persona que tenga acceso al bucket de Amazon S3 puede acceder a sus datos. Por tanto, recomendamos encarecidamente que establezca políticas de IAM para impedir el acceso no autorizado a este bucket de Amazon S3. A fin de obtener más información, consulte [Administración del acceso](#) en la Guía del usuario de Amazon S3.

Para crear los permisos adecuados en un bucket de Amazon S3, siga los pasos que se describen a continuación.

Para conceder ElastiCache acceso a un bucket de S3

1. Inicie sesión en la consola de Amazon S3 AWS Management Console y ábrala en <https://console.aws.amazon.com/s3/>.
2. Elija el nombre del bucket de Amazon S3 en el que desea copiar la copia de seguridad. Este debe ser el bucket de S3 que creó en [Paso 1: Crear un bucket de Amazon S3](#).
3. Elija la pestaña Permissions (Permisos) y debajo Permissions (Permisos), elija Access control list (ACL) (Lista de control de acceso (ACL)) y luego Edit (Editar).
4. Agregue la identificación canónica
540804c33a284a299d2547575ce1010f2312ef3da9b3a053c8bc45bf233e4353 con las siguientes opciones:
 - Objetos: List (Lista) o Write (Escribir)
 - ACL de bucket: Read (Lectura) o Write (Escritura)

i Note

- Para la GovCloud región PDT, el identificador canónico es.
40fa568277ad703bd160f66ae4f83fc9dfdfd06c2f1b5060ca22442ac3ef8be6
- Para la GovCloud región OSU, el identificador canónico es.
c54286759d2a83da9c480405349819c993557275cf37d820d514b42da6893f5c

5. Seleccione Guardar.

Paso 3: Exportar y hacer una copia de seguridad ElastiCache

Ahora ha creado su bucket de S3 y le ha concedido ElastiCache permisos para acceder a él. A continuación, puede usar la ElastiCache consola, la AWS CLI o la ElastiCache API para exportarle la instantánea. En los siguientes ejemplos se da por sentado que la identidad de IAM de la persona que llama dispone de los siguientes permisos adicionales de IAM específicos de S3.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
      "s3:GetBucketLocation",
      "s3:ListAllMyBuckets",
      "s3:PutObject",
      "s3:GetObject",
      "s3:DeleteObject",
      "s3:ListBucket"
    ],
    "Resource": "arn:aws:s3:::*"
  }]
}
```

A continuación, se muestra un ejemplo del aspecto que tendría la política actualizada para el bucket de S3 de las regiones a las que se suscribió. (En este ejemplo se utiliza la región Asia Pacífico (Hong Kong)).

```
{
  "Version": "2012-10-17",
  "Id": "Policy15397346",
  "Statement": [
    {
      "Sid": "Stmt15399483",
      "Effect": "Allow",
      "Principal": {
        "Service": "elasticache.amazonaws.com"
      },
      "Action": "s3:*",
      "Resource": [
        "arn:aws:s3:::hkg-elasticache-backup",
        "arn:aws:s3:::hkg-elasticache-backup/*"
      ]
    }
  ]
}
```

```
    },
    {
      "Sid": "Stmt15399484",
      "Effect": "Allow",

      "Principal": {
        "Service": "ap-east-1.elasticache-snapshot.amazonaws.com"
      },
      "Action": "s3:*",
      "Resource": [
        "arn:aws:s3:::hkg-elasticache-backup",
        "arn:aws:s3:::hkg-elasticache-backup/*"
      ]
    }
  ]
}
```

Exportación y ElastiCache copia de seguridad (consola)

En los siguientes pasos, se utiliza la ElastiCache consola para exportar una copia de seguridad a un bucket de Amazon S3 para que pueda acceder a ella desde fuera ElastiCache. El bucket de Amazon S3 debe estar en la misma AWS región que la ElastiCache copia de seguridad.

Para exportar una ElastiCache copia de seguridad a un bucket de Amazon S3

1. Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en <https://console.aws.amazon.com/elasticache/>.
2. Para ver una lista de las copias de seguridad, en el panel de navegación izquierdo, elija Backups (Copias de seguridad).
3. De la lista de copias de seguridad, active la casilla situada a la izquierda del nombre de la copia de seguridad que desea exportar.
4. Elija Copiar.
5. En Create a Copy of the Backup? (¿Desea crear una copia del backup?), haga lo siguiente:
 - a. En el cuadro New backup name (Nombre del nuevo backup), escriba un nombre para la nueva copia de seguridad.

El nombre debe tener entre 1 y 1 000 caracteres y debe admitir la codificación UTF-8.

ElastiCache añade un identificador de instancia y `.rdb` al valor que introduzcas aquí. Por ejemplo, si especifica `my-exported-backup`, ElastiCache creará `my-exported-backup-0001.rdb`.

- b. Desde la lista de Target S3 Location (Ubicación de S3 de destino), elija el nombre del bucket de Amazon S3 al que desea copiar la copia de seguridad (el bucket que creó en [Paso 1: Crear un bucket de Amazon S3](#)).

La ubicación S3 de destino debe ser un depósito de Amazon S3 en la AWS región de la copia de seguridad con los siguientes permisos para que el proceso de exportación se realice correctamente.

- Acceso al objeto: Read (Lectura) y Write (Escritura).
- Permisos de acceso: lectura.

Para obtener más información, consulte [Paso 2: Conceda ElastiCache acceso a su bucket de Amazon S3](#).

- c. Elija Copiar.

Note

Si su bucket de S3 no tiene los permisos necesarios ElastiCache para exportar una copia de seguridad a él, recibirá uno de los siguientes mensajes de error. Vuelva a [Paso 2: Conceda ElastiCache acceso a su bucket de Amazon S3](#) para añadir los permisos especificados y vuelva a intentar exportar la copia de seguridad.

- ElastiCache no se le han concedido los permisos de LECTURA %s en el bucket de S3.

Solución: añada los permisos Read en el bucket.

- ElastiCache no se le han concedido los permisos de ESCRITURA %s en el bucket de S3.

Solución: añada los permisos Write en el bucket.

- ElastiCache no se le han concedido los permisos READ_ACP %s en el bucket de S3.

Solución: añada Read como permiso de acceso en el bucket.

Si desea copiar la copia de seguridad en otra AWS región, utilice Amazon S3 para copiarla. Para obtener más información, consulte [Copia de objetos](#) en la Guía del usuario de Amazon Simple Storage Service.

Exportación de una copia de seguridad ElastiCache sin servidor ()AWS CLI

Exportación de una copia de seguridad de una caché sin servidor

Exporte la copia de seguridad a un bucket de Amazon S3 con la operación de la CLI `export-serverless-cache-snapshot` con los siguientes parámetros:

Parámetros

- `--serverless-cache-snapshot-name`: nombre de la copia de seguridad que se copiará.
- `--s3-bucket-name`: escriba el nombre del bucket de Amazon S3 donde desea exportar la copia de seguridad. Se realizará una copia de la copia de seguridad en el bucket especificado.
 - `--s3-bucket-name` Debe ser un bucket de Amazon S3 en la AWS región de la copia de seguridad con los siguientes permisos para que el proceso de exportación se realice correctamente.
 - Acceso al objeto: Read (Lectura) y Write (Escritura).
 - Permisos de acceso: lectura.

La operación siguiente permite copiar una copia de seguridad en `my-s3-bucket`.

Para Linux, macOS o Unix:

```
aws elasticache export-serverless-cache-snapshot \  
  --serverless-cache-snapshot-name automatic.my-redis-2023-11-27 \  
  --s3-bucket-name my-s3-bucket
```

Para Windows:

```
aws elasticache export-serverless-cache-snapshot ^  
  --serverless-cache-snapshot-name automatic.my-redis-2023-11-27 ^  
  --s3-bucket-name my-s3-bucket
```

Exportación de una copia de seguridad en ElastiCache clúster de diseño propio ()AWS CLI

Exportación de una copia de seguridad de un clúster de autodiseño

Exporte la copia de seguridad a un bucket de Amazon S3 con la operación de la CLI `copy-snapshot` con los siguientes parámetros:

Parámetros

- `--source-snapshot-name`: nombre de la copia de seguridad que se copiará.
- `--target-snapshot-name`: nombre de la copia de seguridad.

El nombre debe tener entre 1 y 1 000 caracteres y debe admitir la codificación UTF-8.

ElastiCache añade un identificador de instancia y `.rdb` al valor que introduzcas aquí. Por ejemplo, si especifica `my-exported-backup`, ElastiCache creará `my-exported-backup-0001.rdb`.

- `--target-bucket`: escriba el nombre del bucket de Amazon S3 donde desea exportar la copia de seguridad. Se realizará una copia de la copia de seguridad en el bucket especificado.
 - `--target-bucket` Debe ser un bucket de Amazon S3 en la AWS región de la copia de seguridad con los siguientes permisos para que el proceso de exportación se realice correctamente.
 - Acceso al objeto: Read (Lectura) y Write (Escritura).
 - Permisos de acceso: lectura.

Para obtener más información, consulte [Paso 2: Conceda ElastiCache acceso a su bucket de Amazon S3](#).

La operación siguiente permite copiar una copia de seguridad en `my-s3-bucket`.

Para Linux, macOS o Unix:

```
aws elasticache copy-snapshot \  
  --source-snapshot-name automatic.my-redis-primary-2016-06-27-03-15 \  
  --target-snapshot-name my-exported-backup \  
  --target-bucket my-s3-bucket
```

Para Windows:

```
aws elasticache copy-snapshot ^  
  --source-snapshot-name automatic.my-redis-primary-2016-06-27-03-15 ^  
  --target-snapshot-name my-exported-backup ^  
  --target-bucket my-s3-bucket
```

Restauración desde una copia de seguridad a una nueva caché

Puede restaurar una copia de seguridad existente en una nueva memoria caché sin servidor o en un clúster de diseño propio.

Restauración de una copia de seguridad en una caché sin servidor (consola)

Note

ElastiCache Serverless admite archivos RDB compatibles con las versiones de Redis OSS entre la 5.0 y la última versión disponible.

Para restaurar una copia de seguridad en una caché sin servidor (consola)

1. [Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en https://console.aws.amazon.com/elasticache/.](https://console.aws.amazon.com/elasticache/)
2. En el panel de navegación, seleccione Backups (Copias de seguridad).
3. En la lista de copias de seguridad, active la casilla situada a la izquierda del nombre de la copia de seguridad que desea restaurar.
4. Elija Acciones y, a continuación, Restaurar.
5. Escriba un nombre para la nueva caché sin servidor y, si lo desea, una descripción.
6. Haga clic en Crear para crear la nueva caché e importar los datos de la copia de seguridad.

Restauración de una copia de seguridad en un clúster de autodiseño (consola)

Para restaurar una copia de seguridad en un clúster de autodiseño (consola)

1. Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación, seleccione Backups (Copias de seguridad).
3. En la lista de copias de seguridad, active la casilla situada a la izquierda del nombre de la copia de seguridad que desea restaurar.
4. Elija Acciones y, a continuación, Restaurar.
5. Elija Diseñar su propia caché y personalice la configuración del clúster con cosas como el tipo de nodo, los tamaños, la cantidad de particiones, las réplicas, la ubicación en las zonas de disponibilidad y la configuración de seguridad.

6. Seleccione **Crear** para crear la nueva caché de autodiseño e importar los datos de la copia de seguridad.

Restauración de una copia de seguridad en una caché sin servidor (AWS CLI)

Note

ElastiCache Serverless admite archivos RDB compatibles con las versiones 5.0 y la última versión disponible de Redis OSS.

Para restaurar una copia de seguridad en una caché sin servidor (AWS CLI)

El siguiente AWS CLI ejemplo crea una nueva caché utilizando `create-serverless-cache` e importa datos de una copia de seguridad.

Para Linux, macOS o Unix:

```
aws elasticache create-serverless-cache \  
  --serverless-cache-name CacheName \  
  --engine redis \  
  --snapshot-arns-to-restore Snapshot-ARN
```

Para Windows:

```
aws elasticache create-serverless-cache ^ \  
  --serverless-cache-name CacheName ^ \  
  --engine redis ^ \  
  --snapshot-arns-to-restore Snapshot-ARN
```

Para Windows:

Restauración de una copia de seguridad en un clúster de autodiseño (AWS CLI)

Para restaurar una copia de seguridad en un clúster de autodiseño (AWS CLI)

Puede restaurar una copia de seguridad de la caché sin servidor de Redis OSS y también puede restaurar un clúster de diseño propio de Redis OSS.

Puede restaurar una copia de seguridad en caché sin servidor de Redis OSS de dos maneras.

- Puede restaurar un clúster de Redis OSS (modo de clúster desactivado) de un solo nodo mediante la operación. AWS CLI `create-cache-cluster`
- Puede restaurar en un clúster de OSS de Redis con réplicas de lectura (un grupo de replicación). Para ello, puede utilizar Redis OSS (modo de clúster desactivado) o Redis OSS (modo de clúster activado) con la operación. AWS CLI `create-replication-group` En este caso, se inicia la restauración con un archivo `.rdb` de Redis OSS. Para obtener más información sobre la propagación de información inicial en un nuevo clúster de autodiseño, consulte [Inicialización de datos en un nuevo clúster de autodiseño con una copia de seguridad creada externamente](#).

Puede restaurar una copia de seguridad de Redis OSS (modo de clúster desactivado) de dos maneras.

- Puede restaurar un clúster de Redis OSS (modo de clúster desactivado) de un solo nodo mediante la operación. AWS CLI `create-cache-cluster`
- Puede restaurar en un clúster de OSS de Redis con réplicas de lectura (un grupo de replicación). Para ello, puede utilizar Redis OSS (modo de clúster desactivado) o Redis OSS (modo de clúster activado) con la operación. AWS CLI `create-replication-group` En este caso, se inicia la restauración con un archivo `.rdb` de Redis OSS. Para obtener más información sobre la propagación de información inicial en un nuevo clúster de autodiseño, consulte [Inicialización de datos en un nuevo clúster de autodiseño con una copia de seguridad creada externamente](#).

Cuando use la operación `create-cache-cluster` o `create-replication-group`, asegúrese de incluir el parámetro `--snapshot-name` o `--snapshot-arn` para propagar los datos de la copia de seguridad en el nuevo clúster o grupo de réplica.

Eliminación de una copia de seguridad

Las copias de seguridad automáticas se eliminan automáticamente cuando finaliza el límite de retención. Si elimina un clúster, también se eliminarán todas sus copias de seguridad automáticas. Si elimina un grupo de reproducción, también se eliminarán todas las copias de seguridad automáticas de los clústeres de dicho grupo.

ElastiCache proporciona una operación de API de eliminación que le permite eliminar una copia de seguridad en cualquier momento, independientemente de si la copia de seguridad se creó automática o manualmente. Dado que las copias de seguridad manuales no tienen límite de retención, estas copias solo se pueden eliminar de forma manual.

Puede eliminar una copia de seguridad mediante la ElastiCache consola AWS CLI, la ElastiCache API o la consola.

Eliminación de una copia de seguridad (consola)

El siguiente procedimiento elimina una copia de seguridad mediante la ElastiCache consola.

Para eliminar una copia de seguridad

1. Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación, elija Backups.
Aparecerá la pantalla Backups (Copias de seguridad) con una lista de sus copias de seguridad.
3. Elija la casilla situada a la izquierda del nombre de la copia de seguridad que desee eliminar.
4. Elija Eliminar.
5. Si desea eliminar este backup, seleccione Delete en la pantalla de confirmación Delete Backup. El estado cambia a deleting.

Eliminación de una copia de seguridad sin servidor (AWS CLI)

Utilice la AWS CLI operación delete-snapshot con el siguiente parámetro para eliminar una copia de seguridad sin servidor.

- `--serverless-cache-snapshot-name`: nombre de la copia de seguridad que se va a eliminar.

El código siguiente elimina la copia de seguridad myBackup.

```
aws elasticache delete-serverless-cache-snapshot --serverless-cache-snapshot-name myBackup
```

Para obtener más información, consulte [delete-serverless-cache-snapshot](#) en la Referencia de comandos de la AWS CLI .

Eliminación de una copia de seguridad de un clúster de autodiseño (AWS CLI)

Utilice la AWS CLI operación de eliminación de instantáneas con el siguiente parámetro para eliminar una copia de seguridad de clúster de diseño propio.

- `--snapshot-name`: nombre de la copia de seguridad que se va a eliminar.

El código siguiente elimina la copia de seguridad `myBackup`.

```
aws elasticache delete-snapshot --snapshot-name myBackup
```

Para obtener más información, consulte [delete-snapshot](#) en la Referencia de comandos de la AWS CLI .

Etiquetado de copias de seguridad

Puede asignar sus propios metadatos a cada copia de seguridad en forma de etiquetas. Las etiquetas permiten clasificar las copias de seguridad de diversas maneras, por ejemplo, según su finalidad, propietario o entorno. Esto es útil cuando tiene muchos recursos del mismo tipo: puede identificar rápidamente un recurso específico en función de las etiquetas que le haya asignado. Para obtener más información, consulte [Recursos que se pueden etiquetar](#).

Las etiquetas de asignación de costes son una forma de realizar un seguimiento de los costes de varios AWS servicios al agrupar los gastos de las facturas por valores de etiqueta. Para obtener más información sobre las etiquetas de asignación de costos, consulte [Uso de etiquetas de asignación de costos](#).

Con la ElastiCache consola, la ElastiCache API o la API AWS CLI, puede añadir, enumerar, modificar, eliminar o copiar las etiquetas de asignación de costes en sus copias de seguridad. Para obtener más información, consulte [Monitoreo de costos con etiquetas de asignación de costos](#).

Inicialización de datos en un nuevo clúster de autodiseño con una copia de seguridad creada externamente

Al crear un nuevo clúster de diseño propio de Redis OSS, puede iniciarlo con datos de un archivo de copia de seguridad .rdb de Redis OSS. La configuración inicial del clúster resulta útil si actualmente gestiona una instancia de Redis OSS externa ElastiCache y desea rellenar su nuevo clúster ElastiCache (Redis OSS) de diseño propio con los datos de Redis OSS existentes.

Para iniciar un nuevo clúster de diseño propio de Redis OSS a partir de una copia de seguridad de Redis OSS creada en Amazon ElastiCache, consulte [Restauración desde una copia de seguridad a una nueva caché](#)

Si utiliza un archivo.rdb de Redis OSS para iniciar un nuevo clúster de diseño propio de Redis OSS, puede hacer lo siguiente:

- Actualice de un clúster no particionado a un clúster de diseño propio de Redis OSS (modo de clúster habilitado) que ejecute Redis OSS versión 3.2.4.
- Especificar un número de particiones (denominados grupos de nodos en la API y la CLI) en el nuevo clúster de autodiseño. Este número puede ser distinto del número de particiones en el clúster de autodiseño que se utilizó para crear el archivo de copia de seguridad.
- Especificar un tipo de nodo distinto para el nuevo clúster de autodiseño, más grande o más pequeño que el que se utilizó en el clúster que creó la copia de seguridad. Si escala a un tipo de nodo más pequeño, asegúrese de que el nuevo tipo de nodo tenga suficiente memoria para sus datos y la sobrecarga de Redis OSS. Para obtener más información, consulte [Asegurarse de tener suficiente memoria para crear una instantánea de Redis OSS](#).
- Distribuya las claves en las ranuras del nuevo clúster OSS de Redis (habilitado para el modo de clúster) de forma diferente a como se distribuyó en el clúster que se utilizó para crear el archivo de copia de seguridad.

Note

No se puede iniciar un clúster de Redis OSS (modo de clúster desactivado) a partir de un archivo.rdb creado a partir de un clúster de Redis OSS (modo de clúster activado).

⚠ Important

- Debe asegurarse de que los datos de copia de seguridad de Redis OSS no superen los recursos del nodo. Por ejemplo, no puede cargar un archivo.rdb con 5 GB de datos de Redis OSS a un nodo cache.m3.medium que tenga 2,9 GB de memoria.

Si la copia de seguridad es demasiado grande, el clúster resultante tendrá el estado `restore-failed`. Si esto ocurre, deberá eliminar el clúster y empezar de nuevo.

Para obtener una lista completa de los tipos y especificaciones de los nodos, consulta las [características Parámetros específicos del tipo de nodo de Redis OSS y detalles de los ElastiCache productos de Amazon](#).

- Puede cifrar un archivo.rdb de Redis OSS únicamente con el cifrado del lado del servidor de Amazon S3 (SSE-S3). Para obtener más información, consulte [Protección de los datos con el cifrado del lado del servidor](#).

A continuación, encontrará temas que le ayudarán a migrar su clúster de Redis OSS de servicios externos a (Redis OSS). ElastiCache

Migración a (Redis OSS ElastiCache)

- [Paso 1: Crear una copia de seguridad de Redis OSS](#)
- [Paso 2: crear un bucket y una carpeta de Amazon S3](#)
- [Paso 3: cargar la copia de seguridad a Amazon S3](#)
- [Paso 4: Otorgue acceso de ElastiCache lectura al archivo.rdb](#)

Paso 1: Crear una copia de seguridad de Redis OSS

Para crear la copia de seguridad de Redis OSS para iniciar su instancia ElastiCache (Redis OSS)

1. Conéctese a su instancia OSS de Redis existente.
2. Ejecute el OSS BGSAVE o la SAVE operación de Redis para crear una copia de seguridad. Tenga en cuenta la ubicación de su archivo .rdb.

BGSAVE es una operación asincrónica y no bloquea otros clientes durante el procesamiento.

Para obtener más información, consulte [BGSAVE](#) en el sitio web de Redis OSS.

SAVE es una operación sincrónica y bloquea otros procesos hasta que finalice. Para obtener más información, consulte [SAVE](#) en el sitio web de Redis OSS.

Para obtener información adicional sobre la creación de una copia de seguridad, consulte la [persistencia de Redis OSS](#) en el sitio web de Redis OSS.

Paso 2: crear un bucket y una carpeta de Amazon S3

Una vez que se crea el archivo de copia de seguridad, deberá cargarlo en una carpeta de un bucket de Amazon S3. Para ello, primero debe disponer de un bucket de Amazon S3 y de una carpeta en dicho bucket. Si ya dispone de un bucket de Amazon S3 y una carpeta con los permisos pertinentes, puede pasar a [Paso 3: cargar la copia de seguridad a Amazon S3](#).

Creación de un bucket de Amazon S3

1. Inicie sesión en la consola de Amazon S3 AWS Management Console y ábrala en <https://console.aws.amazon.com/s3/>.
2. Siga las instrucciones para crear un bucket de Amazon S3 en [Creación de un bucket](#) en la Guía del usuario de Amazon Simple Storage Service.

El nombre del bucket de Amazon S3 debe estar conforme con DNS. De lo contrario, no ElastiCache podrá acceder al archivo de respaldo. Las reglas para la conformidad con DNS son:

- Los nombres deben tener un mínimo de 3 y un máximo de 63 caracteres de largo.
- Los nombres deben ser una serie de una o más etiquetas separadas por un punto (.) en el que cada etiqueta:
 - Comienza por una letra minúscula o un número.
 - Termina con una letra minúscula o un número.
 - Solo contiene letras minúsculas, números y guiones.
- Los nombres no pueden tener el formato de una dirección IP (por ejemplo, 192.0.2.0).

Debe crear su bucket de Amazon S3 en la misma AWS región que su nuevo clúster ElastiCache (Redis OSS). Este enfoque garantiza la máxima velocidad de transferencia de datos al ElastiCache leer el archivo.rdb de Amazon S3.

 Note

Para conservar la máxima seguridad de los datos, asegúrese de que los permisos de su bucket de Amazon S3 sean lo más restrictivos posible. Al mismo tiempo, los permisos siguen siendo necesarios para permitir que el depósito y su contenido se utilicen para iniciar el nuevo clúster de Redis OSS.

Para agregar una carpeta a un bucket de Amazon S3

1. Inicie sesión en la consola de Amazon S3 AWS Management Console y ábrala en <https://console.aws.amazon.com/s3/>.
2. Elija el nombre del bucket en el que va a cargar el archivo .rdb.
3. Elija Crear carpeta.
4. Escriba un nombre para la nueva carpeta.
5. Seleccione Guardar.

Anote el nombre del bucket y el nombre de la carpeta.

Paso 3: cargar la copia de seguridad a Amazon S3

Ahora, cargue el archivo .rdb que creó en [Paso 1: Crear una copia de seguridad de Redis OSS](#). Cárguelo en el bucket de Amazon S3 y la carpeta que creó en [Paso 2: crear un bucket y una carpeta de Amazon S3](#). Para obtener más información sobre esta tarea, consulte [Agregar un objeto a un bucket](#). Entre los pasos 2 y 3, elija el nombre de la carpeta que creó.

Para cargar el archivo .rdb a una carpeta de Amazon S3

1. Inicie sesión en la consola de Amazon S3 AWS Management Console y ábrala en <https://console.aws.amazon.com/s3/>.
2. Elija el nombre del bucket de Amazon S3 que creó en el paso 2.
3. Elija el nombre de la carpeta que creó en el paso 2.
4. Seleccione Cargar.
5. Elija Add files.

6. Examine el archivo o los archivos que desea cargar y, a continuación, elija el archivo o los archivos. Para elegir varios archivos, mantenga pulsada la tecla Ctrl al mismo tiempo que selecciona un nombre de archivo.
7. Elija Open.
8. Asegúrese de que se muestran los archivos correctos en el cuadro de diálogo Upload y, a continuación, elija Upload.

Escriba la ruta del archivo .rdb. Por ejemplo, si el nombre del bucket es myBucket y la ruta es myFolder/redis.rdb, escriba myBucket/myFolder/redis.rdb. Necesitará esta ruta para propagar en el nuevo clúster los datos de la copia de seguridad.

Para obtener más información, consulte [Restricciones y limitaciones de los buckets](#) en la Guía del usuario de Amazon Simple Storage Service.

Paso 4: Otorgue acceso de ElastiCache lectura al archivo.rdb

Ahora, conceda acceso de ElastiCache lectura a su archivo de copia de seguridad .rdb. El ElastiCache acceso al archivo de copia de seguridad se concede de una forma diferente en función de si el depósito se encuentra en una AWS región predeterminada o en una región opcional AWS .

AWS Las regiones introducidas antes del 20 de marzo de 2019 están habilitadas de forma predeterminada. Puede empezar a trabajar en estas AWS regiones de forma inmediata. Las regiones que se presentaron después del 20 de marzo de 2019, como Asia-Pacífico (Hong Kong) y Medio Oriente (Baréin), se encuentran deshabilitadas de forma predeterminada. Debe habilitar o suscribirse a estas regiones antes de poder utilizarlas, tal y como se describe en [Administración de regiones de AWS](#) en Referencia general de AWS.

Elija su enfoque en función de su AWS región:

- Para una región predeterminada, utilice el procedimiento de [Conceda acceso de ElastiCache lectura al archivo.rdb en una región predeterminada](#).
- Para una región con suscripción, utilice el procedimiento de [Otorgue acceso de ElastiCache lectura al archivo.rdb en una región incluida](#).

Conceda acceso de ElastiCache lectura al archivo.rdb en una región predeterminada

AWS Las regiones introducidas antes del 20 de marzo de 2019 están habilitadas de forma predeterminada. Puede empezar a trabajar en estas AWS regiones de forma inmediata. Las regiones

que se presentaron después del 20 de marzo de 2019, como Asia-Pacífico (Hong Kong) y Medio Oriente (Baréin), se encuentran deshabilitadas de forma predeterminada. Debe habilitar o suscribirse a estas regiones antes de poder utilizarlas, tal y como se describe en [Administración de regiones de AWS](#) en Referencia general de AWS.

Conceder el acceso de ElastiCache lectura al archivo de respaldo en una AWS región, está activado de forma predeterminada

1. Inicie sesión en la consola de Amazon S3 AWS Management Console y ábrala en <https://console.aws.amazon.com/s3/>.
2. Elija el nombre del bucket de S3 que contiene su archivo .rdb.
3. Elija el nombre de la carpeta que contiene su archivo .rdb.
4. Elija el nombre de su archivo de copia de seguridad .rdb. El nombre del archivo seleccionado aparecerá encima de las pestañas, en la parte superior de la página.
5. Elija Permisos.
6. Si aws-scs-s3-readonly o uno de los ID canónicos de la siguiente lista no aparece como usuario, haga lo siguiente:
 - a. En Acceso para otras AWS cuentas, selecciona Añadir beneficiario.
 - b. En el cuadro, añada el ID canónico de la AWS región, tal y como se muestra a continuación:
 - AWS GovCloud Región (EE. UU.-Oeste):

```
40fa568277ad703bd160f66ae4f83fc9dfdfd06c2f1b5060ca22442ac3ef8be6
```

 Important

La copia de seguridad debe estar ubicada en un bucket de S3 AWS GovCloud (US) para poder descargarla en un clúster de Redis OSS. AWS GovCloud (US)

- AWS Regiones habilitadas de forma predeterminada:

```
540804c33a284a299d2547575ce1010f2312ef3da9b3a053c8bc45bf233e4353
```

- c. Establezca los permisos en el bucket eligiendo Yes (Sí) para los valores siguientes:
 - Objetos List (Lista) o Write (Escribir)

- Read/write object ACL permissions (Permisos de lectura y escritura de ACL)
- d. Seleccione Guardar.
7. Elija Overview (Información general) y, a continuación, elija Download (Descargar).

Otorgue acceso de ElastiCache lectura al archivo.rdb en una región incluida

AWS Las regiones introducidas antes del 20 de marzo de 2019 están habilitadas de forma predeterminada. Puede empezar a trabajar en estas AWS regiones de forma inmediata. Las regiones que se presentaron después del 20 de marzo de 2019, como Asia-Pacífico (Hong Kong) y Medio Oriente (Baréin), se encuentran deshabilitadas de forma predeterminada. Debe habilitar o suscribirse a estas regiones antes de poder utilizarlas, tal y como se describe en [Administración de regiones de AWS](#) en Referencia general de AWS.

Ahora, conceda acceso de ElastiCache lectura a su archivo de respaldo .rdb.

Para conceder acceso de ElastiCache lectura al archivo de copia de seguridad

1. Inicie sesión en la consola de Amazon S3 AWS Management Console y ábrala en <https://console.aws.amazon.com/s3/>.
2. Elija el nombre del bucket de S3 que contiene su archivo .rdb.
3. Elija el nombre de la carpeta que contiene su archivo .rdb.
4. Elija el nombre de su archivo de copia de seguridad .rdb. El nombre del archivo seleccionado aparecerá encima de las pestañas, en la parte superior de la página.
5. Elija la pestaña Permisos.
6. En Permissions (Permisos), elija Bucket policy (Política de bucket), y luego Edit (Editar).
7. Actualice la política para conceder los permisos ElastiCache necesarios para realizar operaciones:
 - Agregue ["Service" : "*region-full-name*.elasticache-snapshot.amazonaws.com"] a Principal.
 - Agregue los siguientes permisos necesarios para exportar una instantánea al bucket de Amazon S3:
 - "s3:GetObject"
 - "s3:ListBucket"
 - "s3:GetBucketAcl"

A continuación, se muestra un ejemplo del aspecto que tendría la política actualizada.

```
{
  "Version": "2012-10-17",
  "Id": "Policy15397346",
  "Statement": [
    {
      "Sid": "Stmt15399483",
      "Effect": "Allow",
      "Principal": {
        "Service": "ap-east-1.elasticache-snapshot.amazonaws.com"
      },
      "Action": [
        "s3:GetObject",
        "s3:ListBucket",
        "s3:GetBucketAcl"
      ],
      "Resource": [
        "arn:aws:s3:::example-bucket",
        "arn:aws:s3:::example-bucket/backup1.rdb",
        "arn:aws:s3:::example-bucket/backup2.rdb"
      ]
    }
  ]
}
```

8. Elija Guardar cambios.

Paso 5: siembre el ElastiCache clúster con los datos del archivo.rdb

Ahora está listo para crear un ElastiCache clúster e iniciarlo con los datos del archivo.rdb. Para crear el clúster, siga las instrucciones que se detallan en [Creación de un clúster](#) o [Creación de un grupo de replicación de Redis OSS desde cero](#). Asegúrese de elegir Redis OSS como motor de clúster.

El método que utilice para saber ElastiCache dónde encontrar la copia de seguridad de Redis OSS que cargó en Amazon S3 depende del método que utilice para crear el clúster:

Introduzca los datos del archivo.rdb en el clúster o grupo de replicación ElastiCache (Redis OSS)

- Uso de la consola ElastiCache

Al seleccionar la Cluster settings (Configuración del clúster), elija Restore from backups (Restaurar a partir de copias de seguridad) como método de creación del clúster y, a continuación, elija Other backups (Otras copias de seguridad) como Source (Origen) en la sección Backup source (Origen de copias de seguridad). En el cuadro Seed RDB file S3 location (Inicializar ubicación de S3 del archivo RDB), escriba la ruta de Amazon S3 de los archivos. Si tiene varios archivos.rdb, escriba la ruta para cada archivo en una lista separada por comas. La ruta de Amazon S3 tendrá un aspecto similar a *myBucket/myFolder/myBackupFilename*.rdb.

- Uso del AWS CLI

Si usa la operación `create-cache-cluster` o `create-replication-group`, use el parámetro `--snapshot-arns` para especificar un ARN completo para cada archivo .rdb. Por ejemplo, `arn:aws:s3:::myBucket/myFolder/myBackupFilename`.rdb. El ARN debe resolverse en los archivos de copia de seguridad que almacenó en Amazon S3.

- Uso de la ElastiCache API

Si utiliza la operación `CreateCacheCluster` o la `CreateReplicationGroup` ElastiCache API, utilice el parámetro `SnapshotArns` para especificar un ARN totalmente cualificado para cada archivo.rdb. Por ejemplo, `arn:aws:s3:::myBucket/myFolder/myBackupFilename`.rdb. El ARN debe resolverse en los archivos de copia de seguridad que almacenó en Amazon S3.

Important

Al sembrar un clúster de Redis OSS (modo de clúster habilitado), debe configurar cada grupo de nodos (fragmento) del nuevo clúster o grupo de replicación. Para ello, utilice el parámetro `--node-group-configuration` (API: `NodeGroupConfiguration`). Para más información, consulte los siguientes temas:

- CLI: [create-replication-group](#) en la AWS CLI referencia
- API: [CreateReplicationGroup](#) en la referencia de la ElastiCache API

Durante el proceso de creación del clúster, los datos de la copia de seguridad de Redis OSS se escriben en el clúster. Puede supervisar el progreso viendo los mensajes del ElastiCache evento. Para ello, ve a la ElastiCache consola y selecciona Caché los eventos. También puede utilizar la interfaz de línea de AWS ElastiCache comandos o la ElastiCache API para obtener los mensajes de eventos. Para obtener más información, consulte [Visualización de eventos de ElastiCache](#).

Versiones del motor y actualizaciones

En esta sección se describen las versiones compatibles del motor OSS de Redis y cómo actualizarlas.

Temas

- [Versiones del motor y actualizaciones](#)
- [Versiones compatibles ElastiCache \(Redis OSS\)](#)
- [Calendario de fin de vida de las versiones de Redis OSS](#)
- [Cómo actualizar las versiones del motor](#)
- [Resolver las actualizaciones bloqueadas del motor OSS de Redis](#)
- [Diferencias principales de compatibilidad y comportamiento de versiones](#)

Versiones del motor y actualizaciones

ElastiCache Las versiones (RedisOSS) se identifican con una versión semántica que consta de un componente y. MAJOR MINOR Por ejemplo, en Redis OSS 6.2, la versión principal es la 6 y la versión secundaria es 2. Cuando se utilizan clústeres de diseño propio, ElastiCache (RedisOSS) también expone el PATCH componente, por ejemplo, Redis OSS 6.2.1, y la versión del parche es la 1.

MAJOR las versiones son para cambios API incompatibles y MINOR las versiones son para nuevas funcionalidades añadidas de forma compatible con versiones anteriores. PATCH las versiones son para correcciones de errores compatibles con versiones anteriores y cambios no funcionales.

Note

Si un OSS clúster de Redis se replica en una o más regiones, la versión del motor se actualiza para las regiones secundarias y, después, para la región principal.

Administración de versiones para Serverless ElastiCache

ElastiCache Serverless aplica automáticamente la última versión MINOR y la versión de PATCH software a la memoria caché, sin que la aplicación se vea afectada ni se produzca un tiempo de inactividad. No tiene que hacer nada.

Cuando haya una nueva MAJOR versión disponible, ElastiCache Serverless le enviará una notificación en la consola y un evento en ella. EventBridge Puede optar por actualizar la memoria caché a la versión principal más reciente modificando la memoria caché mediante la consola o API seleccionando la última versión del motor. CLI

Administración de versiones para clústeres de diseño propio ElastiCache

Al trabajar con ElastiCache clústeres de diseño propio, puede controlar si el software que alimenta el clúster de caché se actualiza a las nuevas versiones compatibles con ellas. ElastiCache Puede controlar cuándo actualizar la memoria caché a las versiones y PATCH versiones más recientes disponibles MAJOR. MINOR Para iniciar la actualización de las versiones del motor en el clúster o el grupo de reproducción, modifíquelo y especifique una nueva versión del motor.

Puede controlar si el software compatible con el protocolo que alimenta su clúster de caché se actualiza a las nuevas versiones compatibles con ellas y cuándo. ElastiCache Este nivel de control

permite mantener la compatibilidad con versiones concretas, probar nuevas versiones con la aplicación antes de implementarlas en producción y realizar actualizaciones de versiones en los horarios y los plazos que más le convengan.

Como las actualizaciones de versión pueden conllevar algunos riesgos de compatibilidad, no se producen automáticamente. Debe iniciarlas.

Para iniciar la actualización de las versiones del motor en el clúster o el grupo de reproducción, modifíquelo y especifique una nueva versión del motor. Para más información, consulte los siguientes temas:

- [Modificación de clústeres](#)
- [Modificación de un grupo de reproducción](#)

Consideraciones sobre la actualización al trabajar con clústeres de autodiseño

Note

Las siguientes consideraciones solo son aplicables al actualizar clústeres de autodiseño. No se aplican a Serverless. ElastiCache

Cuando actualice un clúster de autodiseño, tenga en cuenta lo siguiente

- La administración de la versión del motor está diseñada para que pueda tener el mayor control posible sobre cómo se produce la aplicación de parches. Sin embargo, ElastiCache se reserva el derecho de aplicar parches al clúster en su nombre en el improbable caso de que se produzca una vulnerabilidad de seguridad crítica en el sistema o en el software de la memoria caché.
- A partir de Redis OSS 6.0, ElastiCache (RedisOSS) ofrecerá una única versión para cada versión OSS secundaria de Redis, en lugar de ofrecer varias versiones de parches.
- A partir de la versión 5.0.6 OSS del motor de Redis, puede actualizar la versión de clúster con un tiempo de inactividad mínimo. El clúster está disponible para operaciones de lectura durante toda la actualización y para operaciones de escritura durante la mayoría del proceso, excepto durante la operación de conmutación por error, que dura unos segundos.
- También puede actualizar sus ElastiCache clústeres con versiones anteriores a la 5.0.6. El proceso es el mismo, pero es posible que se prolongue el tiempo de conmutación por error durante la DNS propagación (30 segundos a 1 minuto).

- A partir de Redis OSS 7, ElastiCache (RedisOSS) permite cambiar entre Redis (modo de clúster desactivado) y Redis OSS (modo de clúster activado). OSS
- El proceso de actualización del motor Amazon ElastiCache (RedisOSS) está diseñado para hacer todo lo posible por conservar los datos existentes y requiere una replicación correcta de RedisOSS.
- Al actualizar el motor, ElastiCache (RedisOSS) finalizará las conexiones de clientes existentes. [Para minimizar el tiempo de inactividad durante las actualizaciones del motor, le recomendamos implementar las mejores prácticas para los OSS clientes de Redis que se repiten por error y se retrasan exponencialmente, así como las mejores prácticas para minimizar el tiempo de inactividad durante el mantenimiento.](#)
- No puede actualizar directamente de Redis OSS (modo de clúster desactivado) a Redis OSS (modo de clúster activado) al actualizar el motor. El siguiente procedimiento muestra cómo actualizar de Redis OSS (modo de clúster desactivado) a Redis OSS (modo de clúster activado).

Para actualizar de una versión de motor de Redis OSS (modo de clúster desactivado) a Redis OSS (modo de clúster activado)

1. Realice una copia de seguridad de su clúster o grupo de replicación de Redis OSS (modo de clúster desactivado). Para obtener más información, consulte [Copias de seguridad manuales](#).
 2. Utilice la copia de seguridad para crear e inicializar un clúster de Redis OSS (habilitado para el modo de clúster) con un fragmento (grupo de nodos). Especifique la nueva versión de motor y habilite el modo de clúster al crear el clúster o grupo de reproducción. Para obtener más información, consulte [Inicialización de datos en un nuevo clúster de autodiseño con una copia de seguridad creada externamente](#).
 3. Elimine el clúster o grupo de replicación de Redis OSS (modo de clúster desactivado) anterior. Para obtener más información, consulte [Eliminación de un clúster](#) o [Eliminación de un grupo de reproducción](#).
 4. Escale el nuevo clúster o grupo de replicación de Redis OSS (modo de clúster activado) en función del número de fragmentos (grupos de nodos) que necesite. Para obtener más información, consulte [Escalar clústeres en Redis OSS \(modo de clúster habilitado\)](#)
- Cuando actualiza las versiones principales del motor, por ejemplo de 5.0.6 a 6.0, debe seleccionar un grupo de parámetros nuevo que sea compatible con la versión del motor nueva.
 - Para los OSS clústeres de Redis únicos y los clústeres con Multi-AZ deshabilitado, recomendamos que Redis disponga de suficiente memoria, tal y como se describe en. OSS [Asegurarse de tener suficiente memoria para crear una instantánea de Redis OSS](#) En estos casos, el nodo principal no está disponible para las solicitudes de servicio durante el proceso de actualización.

- En el caso de OSS los clústeres de Redis con zonas de disponibilidad múltiples (Multi-AZ) habilitadas, también recomendamos programar las actualizaciones del motor durante los períodos de bajo tráfico de escritura entrante. Al actualizar a Redis OSS 5.0.6 o una versión posterior, el clúster principal seguirá estando disponible para atender las solicitudes de servicio durante el proceso de actualización.

Los clústeres y grupos de reproducción con varias particiones se procesan y se aplican parches de la siguiente manera:

- Todas las particiones se procesan en paralelo. Solo se realiza una operación de actualización en una partición a la vez.
- En cada partición, todas las réplicas se procesan antes que el principal. Si hay menos réplicas en una partición, el principal de esa partición podrá procesarse antes que las réplicas de otras particiones terminen de procesarse.
- En todas las particiones, los nodos principales se procesan en series. Solo se actualiza un nodo principal a la vez.
- Si el cifrado se encuentra habilitado en su grupo de reproducción o clúster actual, no puede actualizar a una versión del motor que no admita cifrado, como de la versión 3.2.6 a la 3.2.10.

Cómo actualizar las versiones del motor

Para iniciar las actualizaciones de versión de su clúster o grupo de replicación, debe modificarlo mediante la ElastiCache consola AWS CLI, el o el ElastiCache API y especificar una versión del motor más reciente. Para obtener más información, consulte los siguientes temas.

Cómo modificar clústeres y grupos de reproducción

Clústeres	Grupos de reproducción
Uso del AWS Management Console	Uso del AWS Management Console
Usando el AWS CLI	Uso del AWS CLI
Uso de la ElastiCache API	Uso de la API ElastiCache

Resolver las actualizaciones bloqueadas OSS del motor Redis

Como se muestra en la siguiente tabla, la operación de actualización OSS del motor de Redis está bloqueada si tiene una operación de ampliación pendiente.

Operaciones pendientes	Operaciones bloqueadas
Escalado ascendente	Actualización del motor inmediata
Actualización del motor	Ampliación inmediata
Ampliación y actualización del motor	Ampliación inmediata
	Actualización del motor inmediata

Para resolver un bloqueo en la actualización del motor Redis OSS

- Realice una de las siguientes acciones siguientes:
 - Programe la operación de actualización OSS del motor Redis para el siguiente período de mantenimiento desactivando la casilla de verificación Aplicar inmediatamente.
- Espere hasta el próximo período de mantenimiento (o después) para realizar la operación de actualización OSS del motor Redis.
- Añada la operación de OSS escalado de Redis a esta modificación de clúster con la casilla Aplicar inmediatamente seleccionada.

Con el CLI, usa `--no-apply-immediately`. Con el API, usa `ApplyImmediately=false`.

Con este enfoque, la actualización del motor se realizará inmediatamente, por lo que quedará cancelada para el siguiente periodo de mantenimiento.

Versiones compatibles ElastiCache (Redis OSS)

ElastiCache Las cachés sin servidor son compatibles con las siguientes versiones de Redis OSS:

- [ElastiCache \(Redis OSS\) versión 7.1 \(mejorada\)](#)

ElastiCache Los clústeres de diseño propio admiten las siguientes versiones de Redis OSS:

- [ElastiCache \(Redis OSS\) versión 7.1 \(mejorada\)](#)
- [ElastiCache \(Redis OSS\) versión 7.0 \(mejorada\)](#)
- [ElastiCache \(Redis OSS\) versión 6.2 \(mejorada\)](#)
- [ElastiCache \(Redis OSS\) versión 6.0 \(mejorada\)](#)
- [ElastiCache \(Redis OSS\) versión 5.0.6 \(mejorada\)](#)
- [ElastiCache \(Redis OSS\) versión 5.0.5 \(obsoleta, utilice la versión 5.0.6\)](#)
- [ElastiCache \(Redis OSS\) versión 5.0.4 \(obsoleta, utilice la versión 5.0.6\)](#)
- [ElastiCache \(Redis OSS\) versión 5.0.3 \(obsoleta, utilice la versión 5.0.6\)](#)
- [ElastiCache \(Redis OSS\) versión 5.0.0 \(obsoleta, utilice la versión 5.0.6\)](#)
- [ElastiCache \(Redis OSS\) versión 4.0.10 \(mejorada\)](#)
- [Versiones 3.x de fin de vida útil \(EOL\) anteriores](#)
- [Versiones 2.x de fin de vida útil \(EOL\) anteriores](#)

ElastiCache (Redis OSS) versión 7.1 (mejorada)

Esta versión contiene mejoras de rendimiento que permiten que las cargas de trabajo aumenten el rendimiento y reduzcan las latencias de operación. ElastiCache La 7.1 presenta [dos mejoras principales](#):

Hemos ampliado la funcionalidad mejorada de los subprocesos de E/S para gestionar también la lógica de la capa de presentación. Por capa de presentación, nos referimos a los subprocesos de E/S mejorados que ahora no solo leen la entrada del cliente, sino que también analizan la entrada en el formato de comandos binarios Redis OSS. Luego, se reenvía al subproceso principal para su ejecución, lo que proporciona un aumento del rendimiento. Patrón de acceso a la memoria OSS de Redis mejorado. Los pasos de ejecución de muchas operaciones de estructura de datos están intercalados para garantizar el acceso paralelo a la memoria y reducir la latencia de acceso a esta. Cuando se ejecuta ElastiCache en una versión basada en Graviton3 R7g .4xLarge o superior, los

clientes pueden recibir más de 1 millón de solicitudes por segundo por nodo. Con las mejoras de rendimiento de la versión 7.1 ElastiCache (Redis OSS), los clientes pueden lograr hasta un 100% más de rendimiento y un 50% menos de latencia de P99 en comparación con (Redis OSS) v7.0. ElastiCache Estas mejoras están disponibles en tamaños de nodo con al menos ocho núcleos físicos (2xlarge en Graviton y 4xlarge en x86), independientemente del tipo de CPU, y no es necesario hacer cambios en el cliente.

Note

ElastiCache La versión 7.1 es compatible con la versión 7.0 de Redis OSS.

ElastiCache (Redis OSS) versión 7.0 (mejorada)

ElastiCache (Redis OSS) 7.0 añade una serie de mejoras y compatibilidad con nuevas funciones:

- [Funciones OSS de Redis](#): ElastiCache (Redis OSS) 7 añade compatibilidad con las funciones OSS de Redis y proporciona una experiencia gestionada que permite a los desarrolladores ejecutar [scripts de LUA](#) con la lógica de la aplicación almacenada en el ElastiCache clúster, sin necesidad de que los clientes tengan que volver a enviar los scripts al servidor en cada conexión.
- [Mejoras en la ACL](#): ElastiCache (Redis OSS) 7 añade compatibilidad con la próxima versión de las listas de control de acceso (ACL) de Redis OSS. Con ElastiCache (Redis OSS) 7, los clientes ahora pueden especificar varios conjuntos de permisos en claves o espacios de teclas específicos en Redis OSS.
- [Sharded Pub/Sub](#): ElastiCache (Redis OSS) 7 añade soporte para ejecutar la funcionalidad Pub/Sub de Redis OSS de forma fragmentada cuando se ejecuta en modo de clúster activado (CME). ElastiCache Las funciones Pub/Sub de Redis OSS permiten a los editores enviar mensajes a cualquier número de suscriptores de un canal. Con Amazon ElastiCache (Redis OSS) 7, los canales están enlazados a un fragmento del ElastiCache clúster, lo que elimina la necesidad de propagar la información del canal entre los fragmentos, lo que mejora la escalabilidad.
- Multiplexación de E/S mejorada: la versión 7 ElastiCache (Redis OSS) presenta la multiplexación de E/S mejorada, que ofrece un mayor rendimiento y una latencia reducida para cargas de trabajo de alto rendimiento que tienen muchas conexiones de cliente simultáneas a un clúster. ElastiCache Por ejemplo, si utiliza un clúster de nodos r6g.xlarge y ejecuta 5200 clientes simultáneos, puede lograr un aumento de rendimiento de hasta un 72% (operaciones de lectura y escritura por segundo) y una disminución de la latencia de P99 de hasta un 71%, en comparación con la versión 6 ElastiCache (Redis OSS).

[Para obtener más información sobre la versión 7.0 de Redis OSS, consulte las notas de la versión de Redis OSS 7.0 en Redis OSS en adelante.](#) GitHub

ElastiCache (Redis OSS) versión 6.2 (mejorada)

ElastiCache (Redis OSS) 6.2 incluye mejoras de rendimiento para los clústeres habilitados para TLS que utilizan tipos de nodos x86 con 8 vCPU o más o tipos de nodos Graviton2 con 4 vCPU o más. Estas mejoras optimizan el rendimiento y reducen el tiempo de establecimiento de la conexión de clientes mediante la descarga del cifrado a otras vCPUs. Con Redis OSS 6.2, también puede administrar el acceso a los canales Pub/Sub con las reglas de la lista de control de acceso (ACL).

Con esta versión, también presentamos soporte para la organización de datos en niveles en nodos de clúster que contienen SSD NVMe adjunta localmente. Para obtener más información, consulte [Organización de datos en niveles](#).

La versión 6.2.6 del motor OSS de Redis también incluye la compatibilidad con el formato nativo de notación de JavaScript objetos (JSON), una forma sencilla y sin esquemas de codificar conjuntos de datos complejos dentro de los clústeres de Redis OSS. Gracias a la compatibilidad con JSON, puede aprovechar el rendimiento y las API OSS de Redis para las aplicaciones que funcionan a través de JSON. Para obtener más información, consulte [Introducción a JSON](#). También se incluyen métricas relacionadas con JSON `JsonBasedCmds` y `JsonBasedCmdsLatency`, que se incorporan CloudWatch para monitorear el uso de este tipo de datos. Para obtener más información, consulte [Métricas para Redis OSS](#).

Para especificar la versión del motor, utilice la versión 6.2. ElastiCache (Redis OSS) invocará automáticamente la versión de parche preferida de Redis OSS 6.2 que esté disponible. Por ejemplo, al crear o modificar un clúster de caché, establece el parámetro `--engine-version` en 6.2. El clúster se lanzará con la versión de parche preferida actualmente disponible de Redis OSS 6.2 en el momento de la creación o modificación. Si se especifica la versión 6.x del motor en la API, se obtendrá la última versión secundaria de Redis OSS 6.

Para los clústeres 6.0 existentes, puede optar por la siguiente actualización automática de la versión secundaria configurando el parámetro `AutoMinorVersionUpgrade` a `yes` en las API `CreateCacheCluster`, `ModifyCacheCluster`, `CreateReplicationGroup` o `ModifyReplicationGroup`. ElastiCache (Redis OSS) actualizará la versión secundaria de sus clústeres 6.0 existentes a la versión 6.2 mediante actualizaciones de autoservicio. Para obtener más información, consulta [Actualizaciones de autoservicio en Amazon ElastiCache](#).

Al llamar a la DescribeCacheEngineVersions API, el valor del EngineVersion parámetro se establecerá en 6.2 y en el CacheEngineVersionDescription campo se mostrará la versión real del motor con la versión del parche.

Para obtener más información sobre la versión 6.2 de Redis OSS, consulte las [notas de la versión 6.2 de Redis OSS](#) en Redis OSS en adelante. GitHub

ElastiCache (Redis OSS) versión 6.0 (mejorada)

Amazon ElastiCache (Redis OSS) presenta la próxima versión del motor Redis OSS, que incluye la [autenticación de los usuarios con un control de acceso basado en roles](#), almacenamiento en caché del lado del cliente e importantes mejoras operativas.

A partir de Redis OSS 6.0, ElastiCache (Redis OSS) ofrecerá una única versión para cada versión secundaria de Redis OSS, en lugar de ofrecer varias versiones de parches. ElastiCache (Redis OSS) gestionará automáticamente la versión parcheada de los clústeres de caché en ejecución, lo que garantizará un mejor rendimiento y una mayor seguridad.

También puede optar por la siguiente actualización automática de la versión secundaria configurando el AutoMinorVersionUpgrade parámetro en yes y ElastiCache (Redis OSS) gestionará la actualización de la versión secundaria mediante actualizaciones de autoservicio. Para obtener más información, consulte [Actualizaciones de servicios en ElastiCache](#).

Para especificar la versión del motor, utilice. 6.0 ElastiCache (Redis OSS) invocará automáticamente la versión de parche preferida de Redis OSS 6.0 que esté disponible. Por ejemplo, al crear o modificar un clúster de caché, establece la propiedad del parámetro --engine-version en 6.0. El clúster se lanzará con la versión de parche preferida actualmente disponible de Redis OSS 6.0 en el momento de la creación o modificación. Cualquier solicitud con un valor de versión de parche específico se rechazará, se lanzará una excepción y el proceso fallará.

Al llamar a la DescribeCacheEngineVersions API, el valor del EngineVersion parámetro se establecerá en 6.0 y la versión real del motor con la versión del parche aparecerá en el campo. CacheEngineVersionDescription

Para obtener más información sobre la versión 6.0 de Redis OSS, consulte las [notas de la versión 6.0 de Redis OSS](#) en Redis OSS en adelante. GitHub

ElastiCache (Redis OSS) versión 5.0.6 (mejorada)

Amazon ElastiCache (Redis OSS) presenta la próxima versión del motor Redis OSS, que incluye correcciones de errores y las siguientes actualizaciones acumulativas:

- Garantía de estabilidad del motor en condiciones especiales.
- Control de errores HyperLogLog mejorado.
- Comandos de protocolo de enlace mejorados para reproducción de confianza.
- Seguimiento de entrega de mensajes coherente a través de comando XCLAIM.
- Gestión de campo de LFU mejorada en objetos.
- Gestión de transacciones mejorada al utilizar ZPOP.
- Posibilidad de cambiar el nombre de los comandos: un parámetro llamado `rename-commands` así permite cambiar el nombre de comandos de Redis OSS potencialmente peligrosos o costosos que podrían provocar la pérdida accidental de datos, como `o. FLUSHALL FLUSHDB`. Es similar a la configuración de comandos de cambio de nombre de Redis OSS de código abierto. Sin embargo, ElastiCache ha mejorado la experiencia al proporcionar un flujo de trabajo totalmente gestionado. Los cambios al nombre de comandos se aplican de inmediato y se propagan automáticamente en todos los nodos del clúster que contienen la lista de comandos. No es necesaria la intervención del usuario, como reiniciar nodos.

Los siguientes ejemplos muestran cómo modificar los grupos de parámetros existentes. Incluyen el parámetro `rename-commands`, que es una lista de comandos separados por espacios cuyo nombre desea cambiar:

```
aws elasticache modify-cache-parameter-group --cache-parameter-group-name custom_param_group --parameter-name-values "ParameterName=rename-commands, ParameterValue='flushall restrictedflushall'" --region region
```

En este ejemplo, el parámetro `rename-commands` se utiliza para cambiar el nombre del comando `flushall` a `restrictedflushall`.

Para cambiar el nombre de varios comandos, utilice lo siguiente:

```
aws elasticache modify-cache-parameter-group --cache-parameter-group-name custom_param_group --parameter-name-values "ParameterName=rename-commands, ParameterValue='flushall restrictedflushall flushdb restrictedflushdb'" --region region
```

Para revertir un cambio, vuelva a ejecutar el comando y excluya los valores a los que se ha cambiado el nombre de la lista `ParameterValue` que desea conservar, tal y como se muestra a continuación:

```
aws elasticache modify-cache-parameter-group --cache-parameter-group-name custom_param_group --parameter-name-values "ParameterName=rename-commands, ParameterValue='flushall restrictedflushall'" --region region
```

En este caso, el nombre del comando `flushall` cambiará a `restrictedflushall` y los nombres de los demás comandos revertirán al nombre original.

Note

Al cambiar el nombre de los comandos, estará sometido a las siguientes limitaciones:

- Todos los nombres de comandos cambiados deben ser alfanuméricos.
- La longitud máxima de los nombres de comandos nuevos es de 20 caracteres alfanuméricos.
- Al cambiar el nombre de comandos, asegúrese de actualizar el grupo de parámetros asociados con el clúster.
- Para evitar por completo el uso de un comando, utilice la palabra clave `blocked`, tal y como se muestra a continuación:

```
aws elasticache modify-cache-parameter-group --cache-parameter-group-name custom_param_group --parameter-name-values "ParameterName=rename-commands, ParameterValue='flushall blocked'" --region region
```

Para obtener más información acerca de los cambios de los parámetros y obtener una lista de los comandos elegibles para el cambio de nombre, consulte [Cambios en los parámetros de Redis OSS 5.0.3](#).

- Redis OSS Streams: modela una estructura de datos de registro que permite a los productores añadir nuevos elementos en tiempo real. También permite a los clientes consumir mensajes mediante bloqueo o sin bloqueo. Las secuencias también permiten a los grupos de consumidores, que representan a un grupo de clientes, consumir cooperativamente distintas partes de la misma secuencia de mensajes, de forma similar a como se hace en [Apache Kafka](#). Para obtener más información, consulte [Introducción a Redis OSS Streams](#).

- Compatibilidad con una familia de comandos de secuencias, como XADD, XRANGE y XREAD. Para obtener más información, consulte Comandos de [Redis OSS Streams](#).
- Una serie de parámetros nuevos y con nombres nuevos. Para obtener más información, consulte [Cambios en los parámetros de Redis OSS 5.0.0](#).
- Una nueva métrica de Redis OSS, `StreamBasedCmds`.
- Tiempo de instantánea ligeramente más rápido para los nodos OSS de Redis.

Important

ElastiCache (Redis OSS) ha corregido dos errores críticos de la versión 5.0.1 de código abierto de [Redis OSS](#). Son las siguientes:

- Respuesta discrepante de RESTORE cuando algunas claves ya han vencido.
- El comando XCLAIM podría devolver una entrada incorrecta o desincronizar el protocolo.

Estas dos correcciones de errores se incluyen en la compatibilidad ElastiCache (Redis OSS) con la versión 5.0.0 del motor Redis OSS y se utilizan en futuras actualizaciones de versiones.

Para obtener más información, consulte las [notas de la versión 5.0.6 de Redis OSS](#) en Redis OSS en adelante. GitHub

ElastiCache (Redis OSS) versión 5.0.5 (obsoleta, utilice la versión 5.0.6)

Amazon ElastiCache (Redis OSS) presenta la próxima versión del motor Redis OSS;. Incluye cambios de configuración en línea para ElastiCache (Redis OSS) de los clústeres de conmutación por error automática durante todas las operaciones planificadas. Ahora puede escalar su clúster, actualizar la versión del motor OSS de Redis y aplicar parches y actualizaciones de mantenimiento mientras el clúster permanece en línea y sigue atendiendo las solicitudes entrantes. También incluye correcciones de errores.

Para obtener más información, consulte las [notas de la versión 5.0.5 de Redis OSS](#) en Redis OSS en adelante. GitHub

ElastiCache (Redis OSS) versión 5.0.4 (obsoleta, utilice la versión 5.0.6)

Amazon ElastiCache (Redis OSS) presenta la próxima versión del motor Redis OSS compatible con Amazon. ElastiCache contiene las mejoras siguientes:

- Garantía de estabilidad del motor en condiciones especiales.
- Control de errores HyperLogLog mejorado.
- Comandos de protocolo de enlace mejorados para reproducción de confianza.
- Seguimiento de entrega de mensajes coherente a través de comando XCLAIM.
- Gestión de campo de LFU mejorada en objetos.
- Gestión de transacciones mejorada al utilizar ZPOP.

Para obtener más información, consulte las [notas de la versión 5.0.4 de Redis OSS en Redis OSS en adelante](#). GitHub

ElastiCache (Redis OSS) versión 5.0.3 (obsoleta, utilice la versión 5.0.6)

Amazon ElastiCache (Redis OSS) presenta la próxima versión del motor Redis OSS compatible con Amazon, ElastiCache que incluye correcciones de errores.

ElastiCache (Redis OSS) versión 5.0.0 (obsoleta, utilice la versión 5.0.6)

Amazon ElastiCache (Redis OSS) presenta la próxima versión principal del motor Redis OSS compatible con Amazon. ElastiCache (Redis OSS) 5.0.0 admite las siguientes mejoras:

- Redis OSS Streams: modela una estructura de datos de registro que permite a los productores añadir nuevos elementos en tiempo real. También permite a los clientes consumir mensajes mediante bloqueo o sin bloqueo. Las secuencias también permiten a los grupos de consumidores, que representan a un grupo de clientes, consumir cooperativamente distintas partes de la misma secuencia de mensajes, de forma similar a como se hace en [Apache Kafka](#). Para obtener más información, consulte [Introducción a Redis OSS Streams](#).
- Compatibilidad con una familia de comandos de secuencias, como XADD, XRANGE y XREAD. Para obtener más información, consulte [Comandos de Redis OSS Streams](#).
- Una serie de parámetros nuevos y con nombres nuevos. Para obtener más información, consulte [Cambios en los parámetros de Redis OSS 5.0.0](#).
- Una nueva métrica de Redis OSS, `StreamBasedCmds`.
- Tiempo de instantánea ligeramente más rápido para los nodos OSS de Redis.

ElastiCache (Redis OSS) versión 4.0.10 (mejorada)

Amazon ElastiCache (Redis OSS) presenta la próxima versión principal del motor Redis OSS compatible con Amazon. ElastiCache ElastiCache (Redis OSS) 4.0.10 admite las siguientes mejoras:

- Tanto el redimensionamiento del clúster en línea como el cifrado en una sola versión ElastiCache (Redis OSS). Para más información, consulte los siguientes temas:
 - [Escalar clústeres en Redis OSS \(modo de clúster habilitado\)](#)
 - [Refragmentación y reequilibrio de particiones en línea para Redis OSS \(modo de clúster habilitado\)](#)
 - [Seguridad de datos en Amazon ElastiCache](#)
- Una serie de parámetros nuevos. Para obtener más información, consulte [Cambios en los parámetros de Redis OSS 4.0.10](#).
- Compatibilidad con la familia de comandos de memoria, como MEMORY. Para obtener más información, consulte [Comandos OSS de Redis](#) (busque en MEMO).
- Compatibilidad con la desfragmentación de memoria mientras está online, lo que permite un uso de memoria más eficiente y más disponibilidad de memoria para los datos.
- Support para descargas y eliminaciones asíncronas. ElastiCache (Redis OSS) admite comandos como UNLINK, por ejemplo, FLUSHDB que se ejecuten en un FLUSHALL hilo diferente al hilo principal. Hacer esto ayuda a mejorar el rendimiento y los tiempos de respuesta para sus aplicaciones mediante la liberación de memoria de forma asincrónica.
- Una nueva métrica de Redis OSS, ActiveDefragHits. Para obtener más información, consulte [Métricas para Redis OSS](#).

Los usuarios de Redis OSS (modo de clúster desactivado) que utilicen la versión 3.2.10 de Redis OSS pueden utilizar la consola para actualizar sus clústeres mediante una actualización en línea.

Compatibilidad con el cifrado y el cambio de tamaño de clústeres ElastiCache (Redis OSS)

Característica	3.2.6	3.2.10	4.0.10 y versiones posteriores
Cambio de tamaño de clústeres online *	No	Sí	Sí
Cifrado en tránsito **	Sí	No	Sí

Característica	3.2.6	3.2.10	4.0.10 y versiones posteriores
Cifrado en reposo**	Sí	No	Sí

* Adición, eliminación y reequilibrado de particiones.

** Necesario para las aplicaciones que se ajustan al estándar FedRAMP, HIPAA y PCI DSS. Para obtener más información, consulte [Validación de conformidad para Amazon ElastiCache](#).

Versiones 3.x de fin de vida útil (EOL) anteriores

ElastiCache (Redis OSS) versión 3.2.10 (mejorada)

Amazon ElastiCache (Redis OSS) presenta la próxima versión principal del motor Redis OSS compatible con Amazon. ElastiCache ElastiCache (Redis OSS) 3.2.10 introduce el cambio de tamaño del clúster en línea para añadir o eliminar fragmentos del clúster mientras se siguen atendiendo las solicitudes de E/S entrantes. ElastiCache (Redis OSS) Los usuarios de la versión 3.2.10 disponen de todas las funciones de las versiones anteriores de Redis OSS, excepto la capacidad de cifrar sus datos. Esta capacidad está disponible actualmente solo en la versión 3.2.6.

Comparación de las versiones 3.2.6 y ElastiCache 3.2.10 (Redis OSS)

Característica	3.2.6	3.2.10
Cambio de tamaño de clústeres online *	No	Sí
Cifrado en tránsito **	Sí	No
Cifrado en reposo**	Sí	No

* Adición, eliminación y reequilibrado de particiones.

** Necesario para las aplicaciones que se ajustan al estándar FedRAMP, HIPAA y PCI DSS. Para obtener más información, consulte [Validación de conformidad para Amazon ElastiCache](#).

Para más información, consulte los siguientes temas:

- [Refragmentación y reequilibrio de particiones en línea para Redis OSS \(modo de clúster habilitado\)](#)
- [Cambio de tamaño de clústeres online](#)

ElastiCache (Redis OSS) versión 3.2.6 (mejorada)

Amazon ElastiCache (Redis OSS) presenta la próxima versión principal del motor Redis OSS compatible con Amazon. ElastiCache ElastiCache (Redis OSS) 3.2.6 Los usuarios tienen todas las funciones de las versiones anteriores de Redis OSS, además de la opción de cifrar sus datos. Para más información, consulte los siguientes temas:

- [ElastiCache cifrado en tránsito \(\) TLS](#)
- [Cifrado en reposo en ElastiCache](#)
- [Validación de conformidad para Amazon ElastiCache](#)

ElastiCache (Redis OSS) versión 3.2.4 (mejorada)

La versión 3.2.4 de Amazon ElastiCache (Redis OSS) presenta la siguiente versión principal del motor OSS de Redis compatible con Amazon. ElastiCache ElastiCache (Redis OSS) 3.2.4 disponen de todas las funciones de las versiones anteriores de Redis OSS, además de la opción de ejecutar en modo clúster o sin clúster. La tabla siguiente ofrece un resumen.

Comparación entre el modo sin clúster y el modo clúster de Redis OSS 3.2.4

Característica	Modo sin clúster	Modo clúster
Particiones de datos	No	Sí
Indexación geoespacial	Sí	Sí
Cambio de tipo de nodo	Sí	Sí *
Escalado de réplica	Sí	Sí*
Escalado ascendente	No	Sí*
Compatibilidad con bases de datos	Múltiple	Única

Característica	Modo sin clúster	Modo clúster
Grupo de parámetros	<code>default.redis3.2</code> **	<code>default.redis3.2.cluster.on</code> **

* Consulte [Restauración desde una copia de seguridad a una nueva caché](#)

** O uno derivado.

Notas:

- **Particionamiento:** posibilidad de dividir los datos entre 2 a 500 grupos de nodos (particiones) con compatibilidad de reproducción para cada grupo de nodos.
- **Indexación geoespacial:** Redis OSS 3.2.4 incorpora la compatibilidad con la indexación geoespacial mediante seis comandos GEO. Para obtener más información, consulte la documentación sobre los comandos OSS GEO* de Redis OSS: [GEO en la página de comandos de Redis OSS \(filtrada por GEO\)](#).

[Para obtener información sobre las funciones adicionales de Redis OSS 3, consulte las notas de la versión 3.2 de Redis OSS y las notas de la versión 3.0 de Redis OSS.](#)

El OSS de Redis ElastiCache gestionado actualmente (modo de clúster activado) no es compatible con las siguientes funciones de Redis OSS 3.2:

- Migración de réplica
- Reequilibrado de clúster
- Depurador Lua

ElastiCache deshabilita los siguientes comandos de administración de Redis OSS 3.2:

- `cluster meet`
- `cluster replicate`
- `cluster flushslots`
- `cluster addslots`
- `cluster delslots`

- `cluster setslot`
- `cluster saveconfig`
- `cluster forget`
- `cluster failover`
- `cluster bumpepoch`
- `cluster set-config-epoch`
- `cluster reset`

Para obtener información sobre los parámetros de Redis OSS 3.2.4, consulte [Cambios en los parámetros de Redis OSS 3.2.4](#)

Versiones 2.x de fin de vida útil (EOL) anteriores

ElastiCache (Redis OSS) versión 2.8.24 (mejorada)

Las mejoras de Redis OSS añadidas desde la versión 2.8.23 incluyen la corrección de errores y el registro de direcciones de acceso a memoria incorrectas. Para obtener más información, consulte las notas de la versión 2.8 [de Redis OSS](#).

ElastiCache (Redis OSS) versión 2.8.23 (mejorada)

Las mejoras de Redis OSS añadidas desde la versión 2.8.22 incluyen correcciones de errores. Para obtener más información, consulte las notas de la versión 2.8 [de Redis OSS](#). Esta versión también incluye compatibilidad con el nuevo parámetro `close-on-slave-write` que, si está habilitado, desconecta a los clientes que intentan escribir en una réplica de solo lectura.

Para obtener más información sobre los parámetros de Redis OSS 2.8.23, consulte [Se agregaron parámetros a Redis OSS 2.8.23 \(mejorado\)](#) la Guía del usuario. ElastiCache

ElastiCache (Redis OSS) versión 2.8.22 (mejorada)

Entre las mejoras de Redis OSS añadidas desde la versión 2.8.21 se incluyen las siguientes:

- Compatibilidad con las copias de seguridad y las sincronizaciones sin ramificaciones que permiten asignar menos memoria para las copias de seguridad y más para su aplicación. Para obtener más información, consulte [Cómo se implementan la sincronización y la copia de seguridad](#). El proceso sin ramificaciones puede afectar a la latencia y al rendimiento. En caso de elevado rendimiento de escritura, al volver a sincronizar una réplica es posible que esta no esté disponible durante la sincronización.

- Si se produce una conmutación por error, los grupos de reproducción se recuperarán ahora más rápido porque las réplicas realizarán sincronizaciones parciales con el nodo principal en lugar de sincronizaciones completas siempre que sea posible. Además, tanto el nodo principal como las réplicas ya no usan el disco durante las sincronizaciones, lo que proporciona mayor velocidad.
- Support para dos CloudWatch métricas nuevas.
 - `ReplicationBytes`: número de bytes que el clúster principal de un grupo de reproducción envía a las réplicas de lectura.
 - `SaveInProgress`: valor binario que indica si hay un proceso de almacenamiento en ejecución en segundo plano.

Para obtener más información, consulte [Monitoreo del uso con métricas de CloudWatch](#).

- Varias correcciones de errores críticos en el comportamiento PSYNC de reproducción. Para obtener más información, consulte las [notas de la versión 2.8 de Redis OSS](#).
- Para mantener un rendimiento de replicación mejorado en los grupos de replicación en zonas de disponibilidad múltiples (Multi-AZ) y aumentar la estabilidad de los clústeres, ya no se admiten las ElastiCache réplicas.
- Para mejorar la coherencia de los datos entre el clúster principal y las réplicas de un grupo de reproducción, las réplicas ya no desalojan claves independientes del clúster principal.
- Las variables de configuración de Redis OSS no `appendfsync` son compatibles con la versión 2.8.22 `appendonly` y posteriores de Redis OSS.
- En situaciones de baja memoria, es posible que los clientes que tengan un búfer de salida de gran tamaño se desconecten del clúster de réplica. En caso de desconexión, el cliente debe volver a conectarse. Estas situaciones son más probables para clientes PUBSUB.

ElastiCache (Redis OSS) versión 2.8.21

Las mejoras de Redis OSS añadidas desde la versión 2.8.19 incluyen una serie de correcciones de errores. Para obtener más información, consulte las notas de la versión 2.8 [de Redis OSS](#).

ElastiCache (Redis OSS) versión 2.8.19

Entre las mejoras de Redis OSS añadidas desde la versión 2.8.6 se incluyen las siguientes:

- Support for HyperLogLog. Para obtener más información, consulte la [nueva estructura de datos de Redis OSS: HyperLogLog](#).

- El tipo de datos de conjuntos clasificados ahora es compatible con las consultas de gama lexicográfica gracias a los nuevos comandos ZRANGEBYLEX, ZLEXCOUNT y ZREMRANGEBYLEX.
- Para evitar que un nodo principal envíe obsoletos a los nodos de réplica, se producirá un error en la sincronización principal si se interrumpe un proceso secundario de almacenamiento en segundo plano (bgsave).
- Support para la HyperLogLogBasedCommands CloudWatch métrica. Para obtener más información, consulte [Métricas para Redis OSS](#).

ElastiCache (Redis OSS) versión 2.8.6

Entre las mejoras de Redis OSS añadidas desde la versión 2.6.13 se incluyen las siguientes:

- Mejora de la flexibilidad y la tolerancia a errores de las réplicas de lectura.
- Compatibilidad con la resincronización.
- Compatibilidad con un número mínimo definido por el usuario de réplicas de lectura que deben estar disponibles en todo momento.
- Compatibilidad completa con los clientes de notificación de publicaciones/suscripciones de eventos en el servidor.
- Detección automática de errores en un nodo principal y de conmutaciones por error del nodo principal al nodo secundario.

ElastiCache (Redis OSS) versión 2.6.13

La versión 2.6.13 de Redis OSS era la versión inicial de Redis OSS compatible con Amazon ElastiCache (Redis OSS). La función Multi-AZ no es compatible con Redis OSS 2.6.13.

Calendario de fin de vida de las versiones de Redis OSS

En esta sección se definen las fechas de fin de vida útil (EOL) de las versiones principales más antiguas a medida que se anuncian. Esto le permite tomar decisiones sobre versiones y actualizaciones para el futuro.

Note

ElastiCache Las versiones de los parches 5.0.0 a 5.0.5 (Redis OSS) están en desuso. Utilice las versiones 5.0.6 o superior.

La siguiente tabla resume cada versión y la fecha de fin de vida anunciada, así como la versión de destino de actualización recomendada.

EOL anterior

Versión de origen	Destino de actualización recomendado	Fecha de EOL
3.2.4, 3.2.6 y 3.2.10	Versión 6.2 o superior <div data-bbox="613 667 1040 1083" style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px;"> <p> Note</p> <p>Para las regiones US-ISO-EAST-1, US-ISO-WEST-1 y US-ISOB-EAST-1, recomendamos 5.0.6 o superior.</p> </div>	31 de julio de 2023
2.8.24, 2.8.23, 2.8.22, 2.8.21, 2.8.19, 2.8.12, 2.8.6, 2.6.13	Versión 6.2 o superior <div data-bbox="613 1194 1040 1610" style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px;"> <p> Note</p> <p>Para las regiones US-ISO-EAST-1, US-ISO-WEST-1 y US-ISOB-EAST-1, recomendamos 5.0.6 o superior.</p> </div>	13 de enero de 2023

Cómo actualizar las versiones del motor

Para iniciar las actualizaciones de versión de su clúster o grupo de replicación, debe modificarlo mediante la ElastiCache consola AWS CLI, la ElastiCache API o especificar una versión del motor más reciente. Para obtener más información, consulte los siguientes temas.

Cómo modificar clústeres y grupos de reproducción	
Clústeres	Grupos de reproducción
Uso del AWS Management Console	Uso del AWS Management Console
Usando el AWS CLI	Uso del AWS CLI
Uso de la ElastiCache API	Uso de la API ElastiCache

Resolver las actualizaciones bloqueadas del motor OSS de Redis

Como se muestra en la siguiente tabla, la operación de actualización del motor OSS de Redis está bloqueada si tiene una operación de ampliación pendiente.

Operaciones pendientes	Operaciones bloqueadas
Escalado ascendente	Actualización del motor inmediata
Actualización del motor	Ampliación inmediata
Ampliación y actualización del motor	Ampliación inmediata
	Actualización del motor inmediata

Para resolver un bloqueo en la actualización del motor OSS de Redis

- Realice una de las siguientes acciones siguientes:
 - Programe la operación de actualización del motor OSS de Redis para el siguiente período de mantenimiento desactivando la casilla Aplicar inmediatamente.

Con la CLI, utilice `--no-apply-immediately`. Con la API, utilice `ApplyImmediately=false`.

- Espere hasta el siguiente período de mantenimiento (o hasta después) para realizar la operación de actualización del motor Redis OSS.
- Añada la operación de escalado de Redis OSS a esta modificación de clúster con la casilla Aplicar inmediatamente seleccionada.

Con la CLI, utilice `--apply-immediately`. Con la API, utilice `ApplyImmediately=true`.

Con este enfoque, la actualización del motor se realizará inmediatamente, por lo que quedará cancelada para el siguiente periodo de mantenimiento.

Diferencias principales de compatibilidad y comportamiento de versiones

Important

La siguiente página está estructurada para indicar todas las diferencias de incompatibilidad entre las versiones e informarle de las consideraciones que debe tener en cuenta al actualizar a versiones más recientes. Esta lista incluye cualquier problema de incompatibilidad de versiones que pueda encontrar al actualizar.

Puede actualizar directamente desde su versión actual de Redis OSS a la última versión de Redis OSS disponible, sin necesidad de realizar actualizaciones secuenciales. Por ejemplo, puede actualizar directamente de la versión 3.0 de Redis OSS a la versión 7.0.

Las versiones de Redis OSS se identifican con una versión semántica que consta de un componente MAYOR, UNO MENOR y UN COMPONENTE PATCH. Por ejemplo, en Redis OSS 4.0.10, la versión principal es la 4, la versión secundaria es 0 y la versión del parche es 10. Por lo general, estos valores se incrementan en función de las siguientes convenciones:

- Las versiones PRINCIPALES son para cambios incompatibles con la API.
- Las versiones SECUNDARIAS son para nuevas funciones agregadas de manera compatible con versiones anteriores.
- Las versiones PARCHE son para correcciones de errores compatibles con versiones anteriores y cambios no funcionales.

Recomendamos mantener siempre la última versión del parche dentro de una versión PRINCIPAL.SECUNDARIO determinada para tener las últimas mejoras de rendimiento y estabilidad. A partir de Redis OSS 6.0, ElastiCache (Redis OSS) ofrecerá una única versión para cada versión secundaria de Redis OSS, en lugar de ofrecer varias versiones de parches. ElastiCache (Redis OSS) gestionará automáticamente la versión parcheada de los clústeres de caché en ejecución, lo que garantizará un mejor rendimiento y una mayor seguridad.

También recomendamos actualizar periódicamente a la última versión principal, ya que la mayoría de las mejoras importantes no se transfieren a versiones anteriores. A medida que ElastiCache amplía la disponibilidad a una nueva AWS región, ElastiCache (Redis OSS) es compatible con las dos versiones MAJOR.MINOR más recientes en ese momento para la nueva región. Por ejemplo, si se lanza una nueva AWS región y las últimas versiones de MAJOR.MINOR ElastiCache (Redis OSS) son 7.0 y 6.2, ElastiCache (Redis OSS) admitirá las versiones 7.0 y 6.2 en la nueva región. AWS A medida que se publiquen nuevas versiones MAJOR.MINOR de ElastiCache (Redis OSS), se ElastiCache seguirá añadiendo soporte para las versiones más recientes (Redis OSS). ElastiCache Para obtener más información sobre cómo elegir regiones ElastiCache, consulte [Elegir](#) regiones y zonas de disponibilidad.

Al realizar una actualización que abarque versiones principales o secundarias, tenga en cuenta la siguiente lista, que incluye los cambios de comportamiento e incompatibles con versiones anteriores publicados con Redis OSS a lo largo del tiempo.

El comportamiento de Redis OSS 7.0 y los cambios incompatibles con versiones anteriores

Para obtener una lista completa de cambios, consulte las notas de la [versión de Redis OSS 7.0](#).

- `SCRIPT LOAD` y `SCRIPT FLUSH` ya no se propagan a réplicas. Si necesita que los scripts tengan cierta durabilidad, le recomendamos que considere la posibilidad de utilizar las funciones de [Redis OSS](#).
- Los canales de pubsub ahora están bloqueados de forma predeterminada para los nuevos usuarios de ACL.
- El comando `STRALGO` se sustituyó por el comando `LCS`.
- El formato de ACL `GETUSER` ha cambiado para que todos los campos muestren el patrón de cadena de acceso estándar. Si utilizaba la automatización mediante ACL `GETUSER`, debe comprobar que maneje cualquiera de los formatos.

- Las categorías de ACL para SELECT, WAIT, ROLE, LASTSAVE, READONLY, READWRITE y ASKING han cambiado.
- El comando INFO ahora muestra las estadísticas de los comandos por subcomando en lugar de en los comandos del contenedor de nivel superior.
- Los valores devueltos de los comandos LPOP, RPOP, ZPOPMIN y ZPOPMAX han cambiado en determinados casos límite. Si utiliza estos comandos, debe consultar las notas de la versión y evaluar si se ve afectado.
- Los comandos SORT y SORT_RO ahora requieren acceso a todo el espacio de claves para poder utilizar los argumentos GET y BY.

El comportamiento de Redis OSS 6.2 y los cambios incompatibles con versiones anteriores

Para obtener una lista completa de cambios, consulte las notas de la [versión 6.2 de Redis OSS](#).

- Los indicadores de ACL de los comandos TIME, ECHO, ROLE y LASTSAVE se cambiaron. Esto puede provocar que se rechacen los comandos previamente permitidos y viceversa.

Note

Ninguno de estos comandos modifica ni da acceso a los datos.

- Al actualizar desde Redis OSS 6.0, se cambia el orden de los pares clave/valor devueltos por una respuesta de mapa a un script lua. Si sus scripts utilizan `redis.setresp()` o devuelven un mapa (nuevo en Redis OSS 6.0), tenga en cuenta las implicaciones que el script podría fallar en las actualizaciones.

El comportamiento de Redis OSS 6.0 y los cambios incompatibles con versiones anteriores

Para obtener una lista completa de cambios, consulte las notas de la [versión de Redis OSS 6.0](#).

- El número máximo de bases de datos permitidas se ha reducido de 1,2 millones a 10 mil. El valor predeterminado es 16 y no recomendamos el uso de valores mucho mayores, ya que hemos detectado problemas de rendimiento y memoria.

- Establezca el `AutoMinorVersionUpgrade` parámetro en sí y ElastiCache (Redis OSS) gestionará la actualización de la versión secundaria mediante actualizaciones de autoservicio. Esto se gestionará a través de canales de notificación estándar a los clientes mediante una campaña de actualización de autoservicio. Para obtener más información, consulte [Actualizaciones de autoservicio](#) en ElastiCache

El comportamiento de Redis OSS 5.0 y los cambios incompatibles con versiones anteriores

Para obtener una lista completa de cambios, consulte las notas de la [versión de Redis OSS 5.0](#).

- Los scripts se replican por efectos en lugar de volver a ejecutar el script en la réplica. Por lo general, esto mejora el rendimiento, pero puede aumentar la cantidad de datos replicados entre los principales y las réplicas. Existe una opción para volver al comportamiento anterior que solo está disponible en ElastiCache (Redis OSS) 5.0.
- Si está actualizando desde Redis OSS 4.0, algunos comandos de los scripts de LUA devolverán los argumentos en un orden diferente al de las versiones anteriores. En Redis OSS 4.0, Redis OSS ordenaba algunas respuestas de forma lexicográfica para hacerlas deterministas; este orden no se aplica cuando los scripts se replican mediante efectos.
- En Redis OSS 5.0.3 y versiones posteriores, ElastiCache (Redis OSS) transferirá parte del trabajo de E/S a núcleos en segundo plano en tipos de instancias con más de 4 CPU virtuales. Esto puede cambiar las características de rendimiento de Redis OSS y cambiar los valores de algunas métricas. Para obtener más información, consulte [¿Qué métricas debo monitorear?](#) para saber si necesita cambiar las métricas que ve.

El comportamiento de Redis OSS 4.0 y los cambios incompatibles con versiones anteriores

Para obtener una lista completa de los cambios, consulte las notas de la [versión de Redis OSS 4.0](#).

- El registro lento ahora registra dos argumentos adicionales, el nombre y la dirección del cliente. Este cambio debe ser compatible con versiones anteriores a menos que dependa explícitamente de cada entrada de registro lento que contenga 3 valores.
- El comando `CLUSTER NODES` ahora devuelve un formato ligeramente diferente, que no es compatible con versiones anteriores. Recomendamos que los clientes no usen este comando para

obtener información sobre los nodos presentes en un clúster y que, en su lugar, usen CLUSTER SLOTS.

EOL anterior

El comportamiento de Redis OSS 3.2 y los cambios incompatibles con versiones anteriores

Para obtener una lista completa de cambios, consulte las notas de la [versión 3.2 de Redis OSS](#).

- No hay cambios de compatibilidad que destacar para esta versión.

Para obtener más información, consulte [Calendario de fin de vida de las versiones de Redis OSS](#).

El comportamiento de Redis OSS 2.8 y los cambios incompatibles con versiones anteriores

Para obtener una lista completa de cambios, consulte las notas de la [versión 2.8 de Redis OSS](#).

- A partir de Redis OSS 2.8.22, el AOF de Redis OSS ya no es compatible con (Redis OSS). ElastiCache Recomendamos usar MemoryDB cuando sea necesario conservar los datos de forma duradera.
- A partir de Redis OSS 2.8.22, ElastiCache (Redis OSS) ya no permite adjuntar réplicas a las unidades principales alojadas en ellas. ElastiCache Durante la actualización, las réplicas externas se desconectarán y no podrán volver a conectarse. Recomendamos utilizar el almacenamiento en caché del lado del cliente, disponible en Redis OSS 6.0, como alternativa a las réplicas externas.
- Los comandos TTL y PTTL ahora devuelven -2 si la clave no existe y -1 si existe pero no tiene fecha de caducidad asociada. Redis OSS 2.6 y las versiones anteriores solían devolver -1 en ambas condiciones.
- SORT con ALPHA ahora ordena según la configuración regional de intercalación local si no se utiliza la opción STORE.

Para obtener más información, consulte [Calendario de fin de vida de las versiones de Redis OSS](#).

ElastiCache mejores prácticas y estrategias de almacenamiento en caché

A continuación, encontrarás las prácticas recomendadas para Amazon ElastiCache. Si observa estos procedimientos, mejorará el rendimiento y la fiabilidad de su caché.

Temas

- [Trabajando con Redis OSS](#)
- [Mejores prácticas con los clientes de Redis OSS](#)
- [Administración de la memoria reservada](#)
- [Procedimientos recomendados al trabajar con clústeres de autodiseño](#)
- [Mejores prácticas de Redis OSS](#)
- [Estrategias de almacenamiento en caché](#)

Trabajando con Redis OSS

A continuación encontrará información sobre la interfaz OSS de Redis que contiene. ElastiCache

Temas

- [Comandos de Redis OSS compatibles y restringidos](#)
- [Configuración y límites de Redis OSS](#)

Comandos de Redis OSS compatibles y restringidos

Comandos de Redis OSS compatibles

Comandos de Redis OSS compatibles

Las cachés sin servidor admiten los siguientes comandos de Redis OSS. Además de estos comandos, estos [Comandos JSON de Redis OSS compatibles](#) también son compatibles.

Comandos de mapa de bits

- BITCOUNT

Cuenta el número de bits establecidos (recuento de integrantes) de una cadena.

[Más información](#)

- BITFIELD

Realiza operaciones arbitrarias con enteros de campos de bits en cadenas.

[Más información](#)

- BITFIELD_R0

Realiza operaciones arbitrarias de solo lectura con enteros de campos de bits en cadenas.

[Más información](#)

- BITOP

Realiza operaciones bit a bit en varias cadenas y almacena el resultado.

[Más información](#)

- BITPOS

Busca el primer bit activo (1) o inactivo (0) en una cadena.

[Más información](#)

- GETBIT

Devuelve el valor de un bit en una posición determinada.

[Más información](#)

- SETBIT

Pone a 1 o 0 el bit de una posición determinada en el valor de la cadena. Crea la clave si esta no existe.

[Más información](#)

Comandos de administración de clústeres

- CLUSTER COUNTKEYSINSLOT

Devuelve el número de claves en un slot hash.

[Más información](#)

- CLUSTER GETKEYSINSLOT

Devuelve los nombres de las claves de un slot hash.

[Más información](#)

- CLUSTER INFO

Devuelve información sobre el estado de un nodo. En una caché sin servidor, devuelve el estado de la única “partición” virtual expuesta al cliente.

[Más información](#)

- CLUSTER KEYSLOT

Devuelve el slot hash de una clave.

[Más información](#)

- CLUSTER MYID

Devuelve el ID de un nodo. En una caché sin servidor, devuelve el estado de la única “partición” virtual expuesta al cliente.

[Más información](#)

- CLUSTER NODES

Devuelve la configuración del clúster de un nodo. En una caché sin servidor, devuelve el estado de la única “partición” virtual expuesta al cliente.

[Más información](#)

- CLUSTER REPLICAS

Muestra los nodos de réplica de un nodo maestro. En una caché sin servidor, devuelve el estado de la única “partición” virtual expuesta al cliente.

[Más información](#)

- CLUSTER SHARDS

Devuelve la asignación de slots del clúster a las particiones. En una caché sin servidor, devuelve el estado de la única “partición” virtual expuesta al cliente.

[Más información](#)

- CLUSTER SLOTS

Devuelve la asignación de slots del clúster a los nodos. En una caché sin servidor, devuelve el estado de la única “partición” virtual expuesta al cliente.

[Más información](#)

- READONLY

Habilita las consultas de solo lectura para una conexión a un nodo de réplica del clúster OSS de Redis.

[Más información](#)

- READWRITE

Habilita las consultas de lectura y escritura para una conexión a un nodo de réplica del clúster OSS de Redis.

[Más información](#)

Comandos de administración de conexiones

- AUTH

Autentica la conexión.

[Más información](#)

- CLIENT GETNAME

Devuelve el nombre de la conexión.

[Más información](#)

- CLIENT REPLY

Le dice al servidor si debe responder a los comandos.

[Más información](#)

- CLIENT SETNAME

Establece el nombre de la conexión.

[Más información](#)

- ECHO

Devuelve la cadena determinada.

[Más información](#)

- HELLO

Apretón de manos con el servidor OSS de Redis.

[Más información](#)

- PING

Devuelve la respuesta de vivacidad del servidor.

[Más información](#)

- QUIT

Cierra la conexión.

[Más información](#)

- RESET

Restablece la conexión.

[Más información](#)

- SELECT

Cambia la base de datos seleccionada.

[Más información](#)

Comandos genéricos

- COPY

Copia el valor de una clave en una nueva clave.

[Más información](#)

- DEL

Elimina una o varias claves.

[Más información](#)

- DUMP

Devuelve una representación serializada del valor almacenado en una clave.

[Más información](#)

- EXISTS

Determina si hay una o más claves.

[Más información](#)

- EXPIRE

Establece el tiempo de caducidad de una clave en segundos.

[Más información](#)

- EXPIREAT

Establece el tiempo de caducidad de una clave en una marca de tiempo de Unix.

[Más información](#)

- EXPIRETIME

Devuelve el tiempo de caducidad de una clave como una marca de tiempo de Unix.

[Más información](#)

- PERSIST

Elimina el tiempo de caducidad de una clave.

[Más información](#)

- PEXPIRE

Establece el tiempo de caducidad de una clave en segundos.

[Más información](#)

- PEXPIREAT

Establece el tiempo de caducidad de una clave como una marca de milisegundos de Unix.

[Más información](#)

- PEXPIRETIME

Devuelve el tiempo de caducidad de una clave como una marca de tiempo de milisegundos de Unix.

[Más información](#)

- PTTL

Devuelve el tiempo de caducidad de una clave en milisegundos.

[Más información](#)

- RANDOMKEY

Devuelve un nombre de clave aleatorio de la base de datos.

[Más información](#)

- RENAME

Cambia el nombre de una clave y sobrescribe el destino.

[Más información](#)

- RENAMENX

Cambia el nombre de una clave solo cuando el nombre de la clave de destino no existe.

[Más información](#)

- RESTORE

Crea una clave a partir de la representación serializada de un valor.

[Más información](#)

- SCAN

Hace una iteración sobre los nombres de claves en la base de datos.

[Más información](#)

- SORT

Ordena los elementos de una lista, un conjunto o un conjunto ordenado y, opcionalmente, almacena el resultado.

[Más información](#)

- SORT_RO

Devuelve los elementos ordenados de una lista, un conjunto o un conjunto ordenado.

[Más información](#)

- TOUCH

Devuelve el número de claves existentes entre las claves especificadas tras actualizar la hora en la que se accedió a ellas por última vez.

[Más información](#)

- TTL

Devuelve el tiempo de caducidad de una clave en segundos.

[Más información](#)

- TYPE

Determina el tipo de valor almacenado en una clave.

[Más información](#)

- UNLINK

Elimina de forma asíncrona una o más claves.

[Más información](#)

Comandos geoespaciales

- GEOADD

Añade uno o varios miembros a un índice geoespacial. La clave se crea si no existe.

[Más información](#)

- GEODIST

Devuelve la distancia entre dos miembros de un índice geoespacial.

[Más información](#)

- GEOHASH

Devuelve los miembros de un índice geoespacial como cadenas geohash.

[Más información](#)

- GEOPOS

Devuelve la longitud y la latitud de los miembros de un índice geoespacial.

[Más información](#)

- GEORADIUS

Consulta un índice geoespacial para ver los miembros que se encuentran a una distancia de una coordenada y, si lo desea, almacena el resultado.

[Más información](#)

- GEORADIUS_RO

Devuelve los miembros de un índice geoespacial que se encuentran a una determinada distancia de una coordenada.

[Más información](#)

- GEORADIUSBYMEMBER

Consulta un índice geoespacial para ver los miembros que se encuentran a una determinada distancia de un miembro y, si lo desea, almacena el resultado.

[Más información](#)

- GEORADIUSBYMEMBER_RO

Devuelve los miembros de un índice geoespacial que se encuentran a una determinada distancia de un miembro.

[Más información](#)

- GEOSEARCH

Consulta en un índice geoespacial a los miembros que se encuentran dentro del área de un cuadro o un círculo.

[Más información](#)

- GEOSEARCHSTORE

Consulta en un índice geoespacial a los miembros que se encuentran dentro del área de un cuadro o un círculo y, opcionalmente, almacena el resultado.

[Más información](#)

Comandos hash

- HDEL

Elimina uno o más campos (y sus valores) de un hash. Elimina el hash si no queda ningún campo.

[Más información](#)

- HEXISTS

Determina si hay un campo en un hash.

[Más información](#)

- HGET

Devuelve el valor de un campo en un hash.

[Más información](#)

- HGETALL

Devuelve todos los campos y valores de un hash.

[Más información](#)

- HINCRBY

Incrementa el valor entero de un campo en un hash por un número determinado. Usa 0 como valor inicial si el campo no existe.

[Más información](#)

- HINCRBYFLOAT

Incrementa el valor de punto flotante de un campo por un número determinado. Usa 0 como valor inicial si el campo no existe.

[Más información](#)

- HKEYS

Devuelve todos los campos de un hash.

[Más información](#)

- HLEN

Devuelve el número de campos en un hash.

[Más información](#)

- HMGET

Devuelve todos los campos y valores en un hash.

[Más información](#)

- HMSET

Establece los valores de varios campos.

[Más información](#)

- HRANDFIELD

Devuelve uno o más campos aleatorios de un hash.

[Más información](#)

- HSCAN

Hace iteraciones sobre los campos y valores de un hash.

[Más información](#)

- HSET

Crea o modifica el valor de un campo en un hash.

[Más información](#)

- HSETNX

Establece el valor de un campo en un hash solo cuando el campo no existe.

[Más información](#)

- HSTRLEN

Devuelve la longitud del valor de un campo.

[Más información](#)

- HVALS

Devuelve todos valores de un hash.

[Más información](#)

HyperLogLog Comandos

- PFADD

Añade elementos a una HyperLogLog clave. Crea la clave si esta no existe.

[Más información](#)

- PFCOUNT

Devuelve la cardinalidad aproximada de los conjuntos observados por las HyperLogLog claves.

[Más información](#)

- PFMERGE

Combina uno o más HyperLogLog valores en una sola clave.

[Más información](#)

Comandos de listas

- BLMOVE

Saca un elemento de una lista, lo coloca en otra y lo devuelve. De lo contrario, bloquea hasta que haya un elemento disponible. Elimina la lista si se ha movido el último elemento.

[Más información](#)

- BLMPOP

Saca el primer elemento de una de las múltiples listas. De lo contrario, bloquea hasta que haya un elemento disponible. Elimina la lista si se ha sacado el último elemento.

[Más información](#)

- BLPOP

Elimina y devuelve el primer elemento de una lista. De lo contrario, bloquea hasta que haya un elemento disponible. Elimina la lista si se ha sacado el último elemento.

[Más información](#)

- BRPOP

Elimina y devuelve el último elemento de una lista. De lo contrario, bloquea hasta que haya un elemento disponible. Elimina la lista si se ha sacado el último elemento.

[Más información](#)

- BRPOPLPUSH

Saca un elemento de una lista, lo coloca en otra y lo devuelve. De lo contrario, bloquea hasta que haya un elemento disponible. Elimina la lista si se ha sacado el último elemento.

[Más información](#)

- LINDEX

Devuelve un elemento de una lista por su índice.

[Más información](#)

- LINSERT

Inserta un elemento antes o después de otro elemento de una lista.

[Más información](#)

- LLEN

Devuelve la longitud de una lista.

[Más información](#)

- LMOVE

Devuelve un elemento después de sacarlo de una lista y pasarlo a otra. Elimina la lista si se ha movido el último elemento.

[Más información](#)

- LMOVE

Devuelve varios elementos de una lista después de eliminarlos. Elimina la lista si se ha sacado el último elemento.

[Más información](#)

- LPOP

Devuelve los primeros elementos de una lista después de eliminarla. Elimina la lista si se ha sacado el último elemento.

[Más información](#)

- LPOS

Devuelve el índice de los elementos coincidentes de una lista.

[Más información](#)

- LPUSH

Antepone uno o más elementos a una lista. Crea la clave si esta no existe.

[Más información](#)

- LPUSHX

Antepone uno o más elementos a una lista solo cuando la lista existe.

[Más información](#)

- LRANGE

Devuelve un rango de elementos de una lista.

[Más información](#)

- LREM

Elimina elementos de una lista. Elimina la lista si se ha eliminado el último elemento.

[Más información](#)

- LSET

Establece el valor de un elemento de una lista por su índice.

[Más información](#)

- LTRIM

Elimina los elementos de ambos extremos de la lista. Elimina la lista si se han recortado todos los elementos.

[Más información](#)

- RPOP

Elimina y devuelve los últimos elementos de una lista. Elimina la lista si se ha sacado el último elemento.

[Más información](#)

- RPOPLPUSH

Devuelve el último elemento de una lista después de extraerlo y colocarlo en otra lista. Elimina la lista si se ha sacado el último elemento.

[Más información](#)

- RPUSH

Antepone uno o más elementos a una lista. Crea la clave si esta no existe.

[Más información](#)

- RPUSHX

Añade un elemento a una lista solo cuando la lista existe.

[Más información](#)

Comandos Pub/Sub

Note

Los comandos PUBSUB utilizan internamente PUBSUB con particiones, por lo que los nombres de los canales se mezclarán.

- PUBLISH

Publica un mensaje en un canal.

[Más información](#)

- PUBSUB CHANNELS

Devuelve los canales activos.

[Más información](#)

- PUBSUB NUMSUB

Devuelve el recuento de suscriptores a los canales.

[Más información](#)

- PUBSUB SHARDCHANNELS

Devuelve los canales de particiones activos.

[PUBSUB-SHARDCHANNELS](#)

- PUBSUB SHARDNUMSUB

Devuelve el recuento de suscriptores a los canales de particiones.

[PUBSUB-SHARDNUMSUB](#)

- SPUBLISH

Publicación de un mensaje en un canal de partición

[Más información](#)

- SSUBSCRIBE

~~Escucha los mensajes publicados en los canales de particiones.~~

[Más información](#)

- SUBSCRIBE

Escucha los mensajes publicados en los canales.

[Más información](#)

- SUNSUBSCRIBE

Deja de escuchar los mensajes publicados en los canales de particiones.

[Más información](#)

- UNSUBSCRIBE

Deja de escuchar los mensajes publicados en los canales.

[Más información](#)

Comandos de scripts

- EVAL

Ejecuta un script de Lua en el lado del servidor.

[Más información](#)

- EVAL_RO

Ejecuta un script de Lua de solo lectura en el lado del servidor.

[Más información](#)

- EVALSHA

Ejecuta un script de Lua en el lado del servidor con el identificador SHA1 digest.

[Más información](#)

- EVALSHA_RO

Ejecuta un script de Lua de solo lectura en el lado del servidor con el identificador SHA1 digest.

[Más información](#)

- SCRIPT EXISTS

Determina si hay scripts de Lua del lado del servidor en la caché de scripts.

[Más información](#)

- SCRIPT FLUSH

Actualmente no está en funcionamiento y la caché del script está administrada por el servicio.

[Más información](#)

- SCRIPT LOAD

Carga un script de Lua del lado del servidor en la caché de scripts.

[Más información](#)

Comandos de administración de servidores

- ACL CAT

Muestra las categorías de ACL o los comandos dentro de una categoría.

[Más información](#)

- ACL GENPASS

Genera una contraseña segura y pseudoaleatoria que se puede utilizar para identificar a los usuarios de ACL.

[Más información](#)

- ACL GETUSER

Muestra las reglas de ACL de un usuario.

[Más información](#)

- ACL LIST

Coloca las reglas vigentes en formato de archivo ACL.

[Más información](#)

- ACL USERS

Muestra todos los usuarios de ACL.

[Más información](#)

- ACL WHOAMI

Devuelve el nombre de usuario autenticado de la conexión actual.

[Más información](#)

- DBSIZE

Devuelve el número de claves de la base de datos actualmente seleccionada. No se garantiza que esta operación sea integral en todos los slots.

[Más información](#)

- COMMAND

Devuelve información detallada sobre todos los comandos.

[Más información](#)

- COMMAND COUNT

Devuelve un recuento de comandos.

[Más información](#)

- COMMAND DOCS

Devuelve información documental sobre un comando, varios o todos ellos.

[Más información](#)

- COMMAND GETKEYS

Extrae los nombres de claves de un comando arbitrario.

[Más información](#)

- COMMAND GETKEYSANDFLAGS

Extrae los nombres de claves y los indicadores de acceso de un comando arbitrario.

[Más información](#)

- **COMMAND INFO**

Devuelve información sobre un comando, varios o todos ellos.

[Más información](#)

- **COMMAND LIST**

Devuelve una lista de nombres de comandos.

[Más información](#)

- **FLUSHALL**

Elimina todas las claves de todas las bases de datos. No se garantiza que esta operación sea integral en todos los slots.

[Más información](#)

- **FLUSHDB**

Elimina todas las claves de la base de datos actual. No se garantiza que esta operación sea integral en todos los slots.

[Más información](#)

- **INFO**

Devuelve información y estadísticas sobre el servidor.

[Más información](#)

- **LOLWUT**

Muestra imágenes de ordenador y la versión OSS de Redis.

[Más información](#)

- **ROLE**

Devuelve la función de réplica.

[Más información](#)

- **TIME**

Devuelve la hora del servidor.

[Más información](#)

Comandos para conjuntos

- SADD

Añade uno o varios miembros a un conjunto. Crea la clave si esta no existe.

[Más información](#)

- SCARDT

Devuelve el número de miembros de un conjunto.

[Más información](#)

- SDIFF

Devuelve la diferencia de varios conjuntos.

[Más información](#)

- SDIFFSTORE

Almacena la diferencia de varios conjuntos en una clave.

[Más información](#)

- SINTER

Devuelve la intersección de varios conjuntos.

[Más información](#)

- SINTERCARD

Devuelve el número de miembros de la intersección de varios conjuntos.

[Más información](#)

- SINTERSTORE

Almacena la intersección de varios conjuntos en una clave.

[Más información](#)

- SISMEMBER

Determina si un miembro pertenece a un conjunto.

[Más información](#)

- SMEMBERS

Devuelve todos los miembros de un conjunto.

[Más información](#)

- SMISMEMBER

Determina si varios miembros pertenecen a un conjunto.

[Más información](#)

- SMOVE

Mueve un miembro de un conjunto a otro.

[Más información](#)

- SPOP

Devuelve uno o más miembros aleatorios de un conjunto después de eliminarlos. Elimina la lista si se ha sacado el último miembro.

[Más información](#)

- SRANDMEMBER

Obtiene uno o varios miembros aleatorios de un conjunto

[Más información](#)

- SREM

Elimina uno o varios miembros de un conjunto. Elimina el conjunto si se ha extraído el último miembro.

[Más información](#)

- SSCAN

Hace iteraciones sobre los miembros de un conjunto.

[Más información](#)

- SUNION

Devuelve la unión de varios conjuntos.

[Más información](#)

- SUNIONSTORE

Almacena la unión de varios conjuntos en una clave.

[Más información](#)

Comandos de conjuntos ordenados

- BZMPOP

Elimina y devuelve un miembro por puntuación desde uno o más conjuntos ordenados. De lo contrario, bloquea hasta que haya un miembro disponible. Elimina el conjunto ordenado si se ha sacado el último elemento.

[Más información](#)

- BZPOPMAX

Elimina y devuelve el miembro con la puntuación más alta de uno o más conjuntos ordenados. De lo contrario, bloquea hasta que haya un miembro disponible. Elimina el conjunto ordenado si se ha sacado el último elemento.

[Más información](#)

- BZPOPMIN

Elimina y devuelve el miembro con la puntuación más baja de uno o más conjuntos ordenados. De lo contrario, bloquea hasta que haya un miembro disponible. Elimina el conjunto ordenado si se ha sacado el último elemento.

[Más información](#)

- ZADD

Añade uno o más miembros a un conjunto ordenado o actualiza sus puntuaciones. Crea la clave si esta no existe.

[Más información](#)

- ZCARD

Devuelve el número de miembros de un conjunto ordenado.

[Más información](#)

- ZCOUNT

Devuelve el recuento de miembros de un conjunto ordenado con puntuaciones dentro de un rango determinado.

[Más información](#)

- ZDIFF

Devuelve la diferencia entre varios conjuntos ordenados.

[Más información](#)

- ZDIFFSTORE

Almacena la diferencia de varios conjuntos en una clave.

[Más información](#)

- ZINCRBY

Incrementa la puntuación de un miembro en un conjunto ordenado.

[Más información](#)

- ZINTER

Devuelve la intersección de varios conjuntos ordenados.

[Más información](#)

- ZINTERCARD

Devuelve el número de miembros de la intersección de varios conjuntos ordenados.

[Más información](#)

- ZINTERSTORE

Almacena la intersección de varios conjuntos ordenados en una clave.

[Más información](#)

- ZLEXCOUNT

Devuelve el número de miembros de un conjunto ordenado dentro de un rango lexicográfico.

[Más información](#)

- ZMPOP

Devuelve los miembros con la puntuación más alta o más baja de uno o más conjuntos ordenados después de eliminarlos. Elimina el conjunto ordenado si se ha sacado el último miembro.

[Más información](#)

- ZMSCORE

Devuelve la puntuación de uno o más miembros de un conjunto ordenado.

[Más información](#)

- ZPOPMAX

Devuelve los miembros con la puntuación más alta de un conjunto ordenado después de eliminarlos. Elimina el conjunto ordenado si se ha sacado el último miembro.

[Más información](#)

- ZPOPMIN

Devuelve los miembros con la puntuación más baja de un conjunto ordenado después de eliminarlos. Elimina el conjunto ordenado si se ha sacado el último miembro.

[Más información](#)

- ZRANDMEMBER

Devuelve uno o más miembros aleatorios de un conjunto ordenado.

[Más información](#)

- ZRANGE

Devuelve los miembros de un conjunto ordenado dentro de un rango de índices.

[Más información](#)

- ZRANGEBYLEX

Devuelve los miembros de un conjunto ordenado dentro de un rango lexicográfico.

[Más información](#)

- ZRANGEBYSCORE

Devuelve los miembros de un conjunto ordenado dentro de un rango de puntuaciones.

[Más información](#)

- ZRANGESTORE

Almacena un rango de miembros de un conjunto ordenado en una clave.

[Más información](#)

- ZRANK

Devuelve el índice de un miembro en un conjunto ordenado por puntuaciones ascendentes.

[Más información](#)

- ZREM

Elimina uno o varios miembros de un conjunto ordenado. Elimina el conjunto ordenado si se han eliminado todos los miembros.

[Más información](#)

- ZREMRANGEBYLEX

Elimina los miembros de un conjunto ordenado dentro de un rango lexicográfico. Elimina el conjunto ordenado si se han eliminado todos los miembros.

[Más información](#)

- ZREMRANGEBYRANK

Elimina los miembros de un conjunto ordenado dentro de un rango de índices. Elimina el conjunto ordenado si se han eliminado todos los miembros.

[Más información](#)

- ZREMRANGEBYSCORE

Elimina los miembros de un conjunto ordenado dentro de un rango de puntuaciones. Elimina el conjunto ordenado si se han eliminado todos los miembros.

[Más información](#)

- ZREVRANGE

Devuelve los miembros de un conjunto ordenado dentro de un rango de índices en orden inverso.

[Más información](#)

- ZREVRANGEBYLEX

Devuelve los miembros de un conjunto ordenado dentro de un rango lexicográfico en orden inverso.

[Más información](#)

- ZREVRANGEBYSCORE

Devuelve los miembros de un conjunto ordenado dentro de un rango de puntuaciones en orden inverso.

[Más información](#)

- ZREVRANK

Devuelve el índice de un miembro en un conjunto ordenado por puntuaciones descendientes.

[Más información](#)

- ZSCAN

Hace iteraciones sobre los miembros y las puntuaciones de un conjunto ordenado.

[Más información](#)

- ZSCORE

Devuelve la puntuación de un miembro en un conjunto ordenado.

[Más información](#)

- ZUNION

Devuelve la unión de varios conjuntos ordenados.

[Más información](#)

- ZUNIONSTORE

Almacena la unión de varios conjuntos ordenados en una clave.

[Más información](#)

Comandos de transmisión

- XACK

Devuelve el número de mensajes confirmados correctamente por el miembro del grupo de consumidores de una transmisión.

[Más información](#)

- XADD

Añade un mensaje nuevo a una transmisión. Crea la clave si esta no existe.

[Más información](#)

- XAUTOCLAIM

Cambia o adquiere la propiedad de los mensajes de un grupo de consumidores, como si los mensajes se hubieran entregado como miembro del grupo de consumidores.

[Más información](#)

- XCLAIM

Cambia o adquiere la propiedad de un mensaje en un grupo de consumidores, como si el mensaje se hubiera entregado por parte de un miembro del grupo de consumidores.

[Más información](#)

- XDEL

Devuelve el número de mensajes después de eliminarlos de una transmisión.

[Más información](#)

- XGROUP CREATE

Crea un grupo de consumidores.

[Más información](#)

- XGROUP CREATECONSUMER

Crea un consumidor en un grupo de consumidores.

[Más información](#)

- XGROUP DELCONSUMER

Elimina un consumidor de un grupo de consumidores.

[Más información](#)

- XGROUP DESTROY

Destruye un grupo de consumidores.

[Más información](#)

- XGROUP SETID

Establece el último ID entregado de un grupo de consumidores.

[Más información](#)

- XINFO CONSUMERS

Devuelve una lista de los consumidores de un grupo de consumidores.

[Más información](#)

- XINFO GROUPS

Devuelve una lista de los grupos de consumidores de una transmisión.

- XINFO STREAM

Devuelve información sobre una transmisión.

[Más información](#)

- XLEN

Devuelve la cantidad de mensajes en una transmisión.

[Más información](#)

- XPENDING

Devuelve la información y las entradas de la lista de entradas pendientes de un grupo de consumidores de una transmisión.

[Más información](#)

- XRANGE

Devuelve los mensajes de una transmisión dentro de un rango de ID.

[Más información](#)

- XREAD

Devuelve los mensajes de varias transmisiones con ID superiores a los solicitados. De lo contrario, bloquea hasta que haya un mensaje disponible.

[Más información](#)

- XREADGROUP

Devuelve mensajes nuevos o históricos de una transmisión para un consumidor en un grupo. De lo contrario, bloquea hasta que haya un mensaje disponible.

[Más información](#)

- XREVRANGE

Devuelve los mensajes de una transmisión dentro de un rango de ID en orden inverso.

[Más información](#)

- XRTM

Elimina los mensajes desde el principio de una transmisión.

[Más información](#)

Comandos de cadenas

- APPEND

Añade una cadena al valor de una clave. Crea la clave si esta no existe.

[Más información](#)

- DECR

Disminuye en uno el valor entero de una clave. Utiliza 0 como valor inicial si la clave no existe.

[Más información](#)

- DECRBY

Disminuye un número del valor entero de una clave. Utiliza 0 como valor inicial si la clave no existe.

[Más información](#)

- GET

Devuelve el valor de cadena de una clave.

[Más información](#)

- GETDEL

Devuelve el valor de cadena de una clave tras eliminarla.

[Más información](#)

- GETEX

Devuelve el valor de cadena de una clave tras establecer su fecha de caducidad.

[Más información](#)

- GETRANGE

Devuelve una subcadena de la cadena almacenada en una clave.

[Más información](#)

- GETSET

Devuelve el valor de cadena anterior de una clave después de establecerla en un nuevo valor.

[Más información](#)

- INCR

Aumenta en uno el valor entero de una clave. Utiliza 0 como valor inicial si la clave no existe.

[Más información](#)

- INCRBY

Incrementa en un número determinado el valor entero de una clave. Utiliza 0 como valor inicial si la clave no existe.

[Más información](#)

- INCRBYFLOAT

Incrementa en un número determinado el valor de punto flotante de una clave. Utiliza 0 como valor inicial si la clave no existe.

[Más información](#)

- LCS

Busca la subcadena común más larga.

[Más información](#)

- MGET

Devuelve íntegramente los valores de cadena de una o más claves.

[Más información](#)

- MSET

Crea o modifica íntegramente los valores de cadena de una o más claves.

[Más información](#)

- MSETNX

Modifica íntegramente los valores de cadena de una o más claves solo cuando no existen todas las claves.

[Más información](#)

- PSETEX

Establece tanto el valor de la cadena como el tiempo de caducidad en milisegundos de una clave. La clave se crea si no existe.

[Más información](#)

- SET

Establece el valor de cadena de una clave e ignora el tipo. La clave se crea si no existe.

[Más información](#)

- SETEX

Establece el valor de la cadena y el tiempo de caducidad de una clave. Crea la clave si esta no existe.

[Más información](#)

- SETNX

Establece el valor de cadena de una clave solo cuando la clave no existe.

[Más información](#)

- SETRANGE

Sobrescribe una parte del valor de una cadena por otra a partir de una determinada posición. Crea la clave si esta no existe.

[Más información](#)

- STRLEN

Devuelve la longitud del valor de la cadena.

[Más información](#)

- SUBSTR

Devuelve una subcadena a partir de un valor de cadena.

[Más información](#)

Comandos de transacción

- DISCARD

Descarta una transacción.

[Más información](#)

- EXEC

Ejecuta todos los comandos de una transacción.

[Más información](#)

- MULTI

Inicia una transacción.

[Más información](#)

Comandos restringidos de Redis OSS

Para ofrecer una experiencia de servicio gestionado, ElastiCache restringe el acceso a determinados comandos específicos del motor de caché que requieren privilegios avanzados. En las cachés que ejecutan Redis, los siguientes comandos no están disponibles:

- `acl setuser`
- `acl load`
- `acl save`
- `acl deluser`
- `bgrewriteaof`
- `bgsave`

- `cluster addslot`
- `cluster addslotsrange`
- `cluster bumpepoch`
- `cluster delslot`
- `cluster delslotsrange`
- `cluster failover`
- `cluster flushslots`
- `cluster forget`
- `cluster links`
- `cluster meet`
- `cluster setslot`
- `config`
- `debug`
- `migrate`
- `psync`
- `replicaof`
- `save`
- `slaveof`
- `shutdown`
- `sync`

Además, los siguientes comandos no están disponibles para las cachés sin servidor:

- `acl log`
- `client caching`
- `client getredir`
- `client id`
- `client info`
- `client kill`
- `client list`

- `client no-evict`
- `client pause`
- `client tracking`
- `client trackinginfo`
- `client unblock`
- `client unpause`
- `cluster count-failure-reports`
- `fcall`
- `fcall_ro`
- `function`
- `function delete`
- `function dump`
- `function flush`
- `function help`
- `function kill`
- `function list`
- `function load`
- `function restore`
- `function stats`
- `keys`
- `lastsave`
- `latency`
- `latency doctor`
- `latency graph`
- `latency help`
- `latency histogram`
- `latency history`
- `latency latest`
- `latency reset`

- `memory`
- `memory doctor`
- `memory help`
- `memory malloc-stats`
- `memory purge`
- `memory stats`
- `memory usage`
- `monitor`
- `move`
- `object`
- `object encoding`
- `object freq`
- `object help`
- `object idletime`
- `object refcount`
- `pfdebug`
- `pfselftest`
- `punsubscribe`
- `pubsub numpat`
- `punsubscribe`
- `script kill`
- `slowlog`
- `slowlog get`
- `slowlog help`
- `slowlog len`
- `slowlog reset`
- `swapdb`
- `unwatch`
- `wait`
- `watch`

Configuración y límites de Redis OSS

El motor OSS de Redis proporciona una serie de parámetros de configuración, algunos de los cuales se pueden modificar en ElastiCache (Redis OSS) y otros no se pueden modificar para proporcionar un rendimiento y una fiabilidad estables.

Cachés sin servidor

En el caso de las cachés sin servidor, no se utilizan grupos de parámetros y no se puede modificar toda la configuración de Redis OSS. Están implementados los siguientes parámetros de Redis OSS:

Nombre	Detalles	Descripción
acl-pubsub-default	allchannels	Permisos de canal pubsub predeterminados para los usuarios de ACL en la caché.
client-output-buffer-limit	normal 0 0 0 pubsub 32mb 8mb 60	Los clientes normales no tienen límite de búfer. Los clientes de PUB/SUB se desconectarán si superan una acumulación de 32 MiB o una acumulación de 8 MiB durante 60 segundos.
client-query-buffer-limit	1 GiB	El tamaño máximo de un búfer de consulta de cliente. Además, los clientes no pueden emitir una solicitud con más de 4000 argumentos.
cluster-allow-pubsubshard-when-down	yes	Esto permite que la caché sirva el tráfico de pubsub mientras esté parcialmente inactiva.
cluster-allow-reads-when-down	yes	Esto permite que la caché sirva el tráfico de lectura mientras esté parcialmente inactiva.
cluster-enabled	yes	Todas las cachés sin servidor están habilitadas para el modo de clúster, lo que les permite dividir sus datos de forma transparente en

Nombre	Detalles	Descripción
		varias particiones de backend. Los clientes observan todos los slots como si perteneciesen a un único nodo virtual.
<code>cluster-require-full-coverage</code>	no	Cuando el espacio de claves esté parcialmente inactivo (es decir, hay, al menos, un slot hash inaccesible), la memoria caché seguirá aceptando consultas para la parte del espacio de claves que aún esté cubierta. El espacio de claves en su totalidad estará siempre “cubierto” por un único nodo virtual en <code>cluster slots</code> .
<code>lua-time-limit</code>	5000	<p>El tiempo máximo de ejecución de un script de Lua, en milisegundos, antes de que ElastiCache tome medidas para detener el script.</p> <p>Si <code>lua-time-limit</code> se supera, todos los comandos OSS de Redis pueden devolver un error del formato <code>____-BUSY</code>. Dado que este estado puede interferir con muchas operaciones esenciales del OSS de Redis, primero ElastiCache emitirá un comando <code>SCRIPT KILL</code>. Si esto no funciona, reiniciará ElastiCache Redis OSS por la fuerza.</p>
<code>maxclients</code>	65000	Número máximo de clientes que pueden conectarse a la vez. Las conexiones adicionales que se establezcan pueden realizarse correctamente o no.
<code>maxmemory-policy</code>	<code>volatile-lru</code>	Los elementos con un TTL configurado se expulsan tras una estimación <code>least-recently-used</code> (LRU) cuando se alcanza el límite de memoria de la caché.

Nombre	Detalles	Descripción
notify-keyspace-events	(una cadena vacía)	Actualmente, los eventos de espacio de claves no son compatibles con las cachés sin servidor.
port	Puerto principal: 6379 Puerto de lectura: 6380	Las cachés sin servidor muestran dos puertos con el mismo nombre de host. El puerto principal permite escribir y leer, mientras que el puerto de lectura permite lecturas coherentes posteriores de menor latencia mediante el comando READONLY.
proto-max-bulk-len	512 MiB	El tamaño máximo de una sola solicitud de elemento.
timeout	0	Los clientes no se desconectan de forma ineludible tras un tiempo de inactividad específico, pero es posible que se desconecten durante el estado estable para equilibrar la carga.

Además, existen los siguientes límites:

Nombre	Detalles	Descripción
Longitud del nombre de la clave	4 KiB	El tamaño máximo de una sola clave OSS de Redis o nombre de canal. Los clientes que hagan referencia a claves de un tamaño mayor a este valor recibirán un error.
El tamaño del script de Lua	4 MiB	El tamaño máximo de un único script Lua OSS de Redis. Los intentos de cargar un script de Lua con un tamaño superior a este recibirán un error.

Nombre	Detalles	Descripción
Tamaño del slot	32 GiB	El tamaño máximo de una sola ranura de hash de Redis OSS. Los clientes que intenten colocar más datos que estos en una sola ranura OSS de Redis activarán la política de desalojo en la ranura y, si no se puede extraer ninguna clave, recibirán un error de memoria insuficiente (<code>OOM</code>).

Clústeres de autodiseño

Con respecto a los clústeres de autodiseño, consulte [Parámetros específicos de Redis OSS](#) para obtener información sobre los valores predeterminados de los parámetros de configuración y para saber cuáles son configurables. Por lo general, se recomiendan los valores predeterminados, a menos que tenga un caso de uso específico que requiera anularlos.

Mejores prácticas con los clientes de Redis OSS

Conozca las prácticas recomendadas para situaciones comunes y siga los ejemplos de código de algunas de las bibliotecas cliente de código abierto más populares de Redis OSS (redis-py, phRedis y Lettuce).

Temas

- [Número elevado de conexiones](#)
- [Descubrimiento de clientes de clústeres OSS de Redis y retroceso exponencial](#)
- [Configurar un tiempo de espera del cliente](#)
- [Configurar un tiempo de inactividad en el servidor](#)
- [Secuencias de comandos Lua de Redis OSS](#)
- [Almacenamiento de elementos compuestos de gran tamaño](#)
- [Configuración del cliente de Lettuce](#)
- [Ejemplos de cliente de IPv6](#)

Número elevado de conexiones

Las cachés sin servidor y los nodos individuales ElastiCache (Redis OSS) admiten hasta 65 000 conexiones de clientes simultáneas. Sin embargo, para optimizar el rendimiento, recomendamos que las aplicaciones cliente no funcionen constantemente con ese volumen de conexiones. El OSS de Redis es un proceso de subproceso único basado en un bucle de eventos en el que las solicitudes entrantes de los clientes se gestionan de forma secuencial. Esto significa que el tiempo de respuesta de un cliente determinado se hace más largo a medida que aumenta el número de clientes conectados.

Puede tomar las siguientes medidas para evitar que se produzca un cuello de botella en la conexión del servidor OSS de Redis:

- Llevar a cabo las operaciones de lectura a partir de réplicas de lectura. Esto se puede hacer utilizando los puntos finales del ElastiCache lector en el modo de clúster desactivado o mediante réplicas para las lecturas en el modo de clúster activado, incluida una caché sin servidor.
- Distribuir el tráfico de escritura entre varios nodos principales. Puede hacerlo de dos formas. Puede utilizar un clúster de Redis OSS con varios fragmentos con un cliente compatible con el modo de clúster de Redis OSS. También puede escribir en varios nodos principales con el modo de clúster deshabilitado y con partición en el lado del cliente. Esto se hace automáticamente en una memoria caché sin servidor.
- Usar un grupo de conexiones cuando esté disponible en la biblioteca de su cliente.

En general, crear una conexión TCP es una operación costosa desde el punto de vista computacional en comparación con los comandos OSS típicos de Redis. Por ejemplo, gestionar una solicitud SET/GET es una orden de magnitud más rápido cuando se reutiliza una conexión existente. El uso de un grupo de conexiones de clientes con un tamaño finito reduce la sobrecarga en la administración de conexiones. También limita el número de conexiones entrantes simultáneas desde la aplicación cliente.

El siguiente ejemplo de código de PHPRedis muestra cómo se crea una nueva conexión para cada nueva solicitud de usuario:

```
$redis = new Redis();
if ($redis->connect($HOST, $PORT) != TRUE) {
    //ERROR: connection failed
    return;
}
```

```
$redis->set($key, $value);  
unset($redis);  
$redis = NULL;
```

Hemos comparado este código en bucle en una instancia de Amazon Elastic Compute Cloud (Amazon EC2) conectada a un nodo Graviton2 (m6g.2xlarge) (Redis OSS). ElastiCache Hemos colocado el cliente y el servidor dentro de la misma zona de disponibilidad. La latencia media de toda la operación fue de 2,82 milisegundos.

Cuando actualizamos el código y utilizamos conexiones persistentes y un grupo de conexiones, la latencia media de toda la operación fue de 0,21 milisegundos:

```
$redis = new Redis();  
if ($redis->pconnect($HOST, $PORT) != TRUE) {  
    // ERROR: connection failed  
    return;  
}  
$redis->set($key, $value);  
unset($redis);  
$redis = NULL;
```

Configuraciones de redis.ini obligatorias:

- `redis.pconnect.pooling_enabled=1`
- `redis.pconnect.connection_limit=10`

El siguiente código es un ejemplo de un [grupo de conexiones Redis-py](#):

```
conn = Redis(connection_pool=redis.BlockingConnectionPool(host=HOST,  
    max_connections=10))  
conn.set(key, value)
```

El siguiente código es un ejemplo de un [grupo de conexiones Lettuce](#):

```
RedisClient client = RedisClient.create(RedisURI.create(HOST, PORT));  
GenericObjectPool<StatefulRedisConnection> pool =  
    ConnectionPoolSupport.createGenericObjectPool(() -> client.connect(), new  
    GenericObjectPoolConfig());  
pool.setMaxTotal(10); // Configure max connections to 10  
try (StatefulRedisConnection connection = pool.borrowObject()) {  
    RedisCommands syncCommands = connection.sync();
```

```
syncCommands.set(key, value);  
}
```

Descubrimiento de clientes de clústeres OSS de Redis y retroceso exponencial

Al conectarse a un clúster ElastiCache (Redis OSS) con el modo de clúster activado, la biblioteca cliente de Redis OSS correspondiente debe ser compatible con los clústeres. Los clientes deben obtener un mapa de los slots hash de los nodos correspondientes del clúster para poder enviar las solicitudes a los nodos correctos, y evitar así la sobrecarga de rendimiento que supone gestionar las redirecciones del clúster. Como resultado, el cliente debe descubrir (o detectar) una lista completa de los slots y los nodos mapeados en dos situaciones diferentes:

- El cliente se inicializa y debe completar la configuración inicial de los slots.
- Se recibe una redirección MOVED del servidor; por ejemplo, en una conmutación por error, cuando la réplica se hace cargo de todos los slots atendidos por el nodo principal anterior, o en una nueva partición, cuando los slots se mueven del nodo principal de origen al nodo principal de destino.

Por lo general, la detección de clientes se realiza mediante la emisión de un comando CLUSTER SLOT o CLUSTER NODE al servidor OSS de Redis. Recomendamos el método CLUSTER SLOT porque devuelve al cliente el conjunto de rangos de slots, así como los nodos principales y de réplica asociados. Es un método que no requiere un análisis adicional por parte del cliente y es más eficaz.

En función de la topología del clúster, el tamaño de la respuesta al comando CLUSTER SLOT puede variar según el tamaño del clúster. Los clústeres más grandes y con más nodos producen una respuesta mayor. Por lo tanto, es importante asegurarse de que la cantidad de clientes que llevan a cabo la detección de la topología del clúster no aumente de forma ilimitada. Por ejemplo, cuando la aplicación cliente se inicia o pierde la conexión con el servidor y debe realizar una detección de clústeres, un error común es que la aplicación cliente desencadene varias solicitudes de reconexión y detección sin añadir un retroceso exponencial con el nuevo intento. Esto puede hacer que el servidor OSS de Redis deje de responder durante un período prolongado, con un uso de la CPU del 100%. La interrupción se prolonga si cada comando CLUSTER SLOT debe procesar una gran cantidad de nodos en el bus del clúster. Hemos observado varias interrupciones en los clientes en el pasado debido a este comportamiento en varios lenguajes diferentes, incluidos Python (redis-py-cluster) y Java (Lettuce y Redisson).

En una caché sin servidor, muchos de los problemas se mitigan automáticamente porque la topología de clúster expuesta es estática y consta de dos entradas: un punto de conexión de escritura y otro de lectura. La detección de clústeres también se distribuye automáticamente entre

varios nodos cuando se utiliza el punto de conexión de la caché. Sin embargo, las siguientes recomendaciones siguen siendo útiles.

A fin de mitigar el impacto causado por una entrada repentina de solicitudes de conexión y detección, recomendamos lo siguiente:

- Implemente un grupo de conexiones de cliente con un tamaño finito a fin de limitar el número de conexiones entrantes simultáneas desde la aplicación cliente.
- Cuando el cliente se desconecte del servidor debido al tiempo de espera, vuelva a intentarlo con retroceso exponencial y fluctuación. Esto evitará que varios clientes sobrecarguen el servidor al mismo tiempo.
- Utilice la guía en [Búsqueda de puntos de conexión](#) para encontrar el punto de conexión del clúster que necesitará a fin de realizar la detección del clúster. De este modo, distribuirá la carga de detección entre todos los nodos del clúster (hasta 90), en lugar de centrarse en unos pocos nodos raíz codificados del clúster.

A continuación, se muestran algunos ejemplos de código para la lógica de reintentos de retroceso exponencial en redis-py, PHPRedis y Lettuce.

Ejemplo 1 de lógica de retroceso: redis-py

Redis-py tiene un mecanismo de reintento incorporado: se lleva a cabo un reintento inmediatamente después de un error. Este mecanismo se puede activar mediante el `retry_on_timeout` argumento proporcionado al crear un objeto OSS de [Redis](#). Aquí mostramos un mecanismo de reintento personalizado con retroceso exponencial y fluctuación. Hemos enviado una solicitud de extracción para implementar de forma nativa el retroceso exponencial en [redis-py \(1494\)](#). En el futuro, puede que no sea necesario implementarlo manualmente.

```
def run_with_backoff(function, retries=5):
    base_backoff = 0.1 # base 100ms backoff
    max_backoff = 10 # sleep for maximum 10 seconds
    tries = 0
    while True:
        try:
            return function()
        except (ConnectionError, TimeoutError):
            if tries >= retries:
                raise
            backoff = min(max_backoff, base_backoff * (pow(2, tries) + random.random()))
            print(f"sleeping for {backoff:.2f}s")
```

```
sleep(backoff)
tries += 1
```

Luego, puede utilizar el siguiente código para establecer un valor:

```
client = redis.Redis(connection_pool=redis.BlockingConnectionPool(host=HOST,
    max_connections=10))
res = run_with_backoff(lambda: client.set("key", "value"))
print(res)
```

En función de la carga de trabajo, es posible que desee cambiar el valor del retroceso base (de 1 segundo a unas pocas decenas o cientos de milisegundos) para las cargas de trabajo sensibles a la latencia.

Ejemplo 2 de lógica de retroceso: PHPRedis

PHPRedis tiene un mecanismo de reintento incorporado: hay un número máximo de reintento de 10 veces (no configurable). Hay un retraso configurable entre los intentos (con una fluctuación a partir del segundo reintento). Para obtener más información, consulte el siguiente [código de muestra](#). Enviamos una solicitud de extracción para implementar de forma nativa el retroceso exponencial en [PHPRedis \(1986\)](#); desde entonces, se ha fusionado y [documentado](#). Si utiliza la última versión de PHPRedis, no será necesario implementarla manualmente, pero hemos incluido aquí la referencia para las versiones anteriores. De momento, aquí tiene un ejemplo de código para configurar el retraso del mecanismo de reintento:

```
$timeout = 0.1; // 100 millisecond connection timeout
$retry_interval = 100; // 100 millisecond retry interval
$client = new Redis();
if($client->pconnect($HOST, $PORT, $timeout, NULL, $retry_interval) != TRUE) {
    return; // ERROR: connection failed
}
$client->set($key, $value);
```

Ejemplo 3 de lógica de retroceso: Lettuce

Lettuce tiene mecanismos de reintento incorporados que emplean las estrategias de retroceso exponencial descritas en la publicación [Exponential Backoff and Jitter](#). A continuación, puede ver un fragmento de código con el método de fluctuación total:

```
public static void main(String[] args)
{
```

```
ClientResources resources = null;
RedisClient client = null;

try {
    resources = DefaultClientResources.builder()
        .reconnectDelay(Delay.fullJitter(
            Duration.ofMillis(100),    // minimum 100 millisecond delay
            Duration.ofSeconds(5),    // maximum 5 second delay
            100, TimeUnit.MILLISECONDS) // 100 millisecond base
        ).build();

    client = RedisClient.create(resources, RedisURI.create(HOST, PORT));
    client.setOptions(ClientOptions.builder()
        .socketOptions(SocketOptions.builder().connectTimeout(Duration.ofMillis(100)).build()) //
        100 millisecond connection timeout
        .timeoutOptions(TimeoutOptions.builder().fixedTimeout(Duration.ofSeconds(5)).build()) //
        5 second command timeout
        .build());

    // use the connection pool from above example
} finally {
    if (connection != null) {
        connection.close();
    }

    if (client != null){
        client.shutdown();
    }

    if (resources != null){
        resources.shutdown();
    }

}
}
```

Configurar un tiempo de espera del cliente

Configure el tiempo de espera del lado del cliente de la manera adecuada, a fin de que el servidor tenga tiempo suficiente para procesar la solicitud y generar la respuesta. Esto también le permitirá responder rápido a los errores si no se puede establecer la conexión con el servidor. Ciertos comandos OSS de Redis pueden ser más caros desde el punto de vista computacional que otros. Un ejemplo de esto son los scripts de Lua o las transacciones MULTI/EXEC con varios comandos que

deben ejecutarse de forma integral. Por lo general, es recomendable aumentar el tiempo de espera del cliente para evitar que este se quede sin tiempo antes de recibir la respuesta del servidor, lo que incluye lo siguiente:

- Ejecutar comandos en varias claves
- Ejecutar transacciones MULTI/EXEC o scripts de Lua que constan de varios comandos individuales de Redis OSS
- Leer valores grandes
- Realizar operaciones de bloqueo, como BLPOP

En el caso de una operación de bloqueo como BLPOP, el procedimiento recomendado es establecer el tiempo de espera del comando en un número inferior al tiempo de espera del socket.

A continuación, encontrará ejemplos de código para implementar un tiempo de espera del lado del cliente en redis-py, PHPRedis y Lettuce.

Ejemplo 1 de configuración de tiempo de espera: redis-py

A continuación, puede ver un ejemplo de código con redis-py:

```
# connect to Redis server with a 100 millisecond timeout
# give every Redis command a 2 second timeout
client = redis.Redis(connection_pool=redis.BlockingConnectionPool(host=HOST,
    max_connections=10,socket_connect_timeout=0.1,socket_timeout=2))

res = client.set("key", "value") # will timeout after 2 seconds
print(res)                       # if there is a connection error

res = client.blpop("list", timeout=1) # will timeout after 1 second
                                     # less than the 2 second socket timeout
print(res)
```

Ejemplo 2 de configuración de tiempo de espera: PHPRedis

A continuación, puede ver un ejemplo de código con PHPRedis:

```
// connect to Redis server with a 100ms timeout
// give every Redis command a 2s timeout
$client = new Redis();
$timeout = 0.1; // 100 millisecond connection timeout
```

```

$retry_interval = 100; // 100 millisecond retry interval
$client = new Redis();
if($client->pconnect($HOST, $PORT, 0.1, NULL, 100, $read_timeout=2) != TRUE){
    return; // ERROR: connection failed
}
$client->set($key, $value);

$res = $client->set("key", "value"); // will timeout after 2 seconds
print "$res\n"; // if there is a connection error

$res = $client->blpop("list", 1); // will timeout after 1 second
print "$res\n"; // less than the 2 second socket timeout

```

Ejemplo 3 de configuración de tiempo de espera: Lettuce

A continuación, puede ver un ejemplo de código con Lettuce:

```

// connect to Redis server and give every command a 2 second timeout
public static void main(String[] args)
{
    RedisClient client = null;
    StatefulRedisConnection<String, String> connection = null;
    try {
        client = RedisClient.create(RedisURI.create(HOST, PORT));
        client.setOptions(ClientOptions.builder()
            .socketOptions(SocketOptions.builder().connectTimeout(Duration.ofMillis(100)).build()) //
            100 millisecond connection timeout
            .timeoutOptions(TimeoutOptions.builder().fixedTimeout(Duration.ofSeconds(2)).build()) //
            2 second command timeout
            .build());

        // use the connection pool from above example

        commands.set("key", "value"); // will timeout after 2 seconds
        commands.blpop(1, "list"); // BLPOP with 1 second timeout
    } finally {
        if (connection != null) {
            connection.close();
        }

        if (client != null){
            client.shutdown();
        }
    }
}

```

```
}
```

Configurar un tiempo de inactividad en el servidor

Hemos observado casos en los que la aplicación de un cliente tiene un gran número de clientes inactivos conectados, pero no envía comandos de forma activa. En estos casos, se pueden agotar las 65 000 conexiones con un número elevado de clientes inactivos. Para evitar este tipo de situaciones, configure el tiempo de espera adecuadamente en el servidor mediante [Parámetros específicos de Redis OSS](#). Esto garantiza que el servidor tome la iniciativa en desconectar a los clientes inactivos para evitar un aumento en el número de conexiones. Esta configuración no está disponible en las caché sin servidor.

Secuencias de comandos Lua de Redis OSS

Redis OSS admite más de 200 comandos, incluidos los que ejecutan scripts de Lua. Sin embargo, en lo que respecta a los scripts de Lua, existen varios escollos que pueden afectar a la memoria y la disponibilidad del OSS de Redis.

Scripts de Lua no parametrizados

Cada script de Lua se almacena en caché en el servidor OSS de Redis antes de ejecutarse. Los scripts de Lua no parametrizados son únicos, lo que puede provocar que el servidor OSS de Redis almacene una gran cantidad de scripts de Lua y consuma más memoria. Para mitigar esta situación, asegúrese de que todos los scripts de Lua estén parametrizados y ejecute SCRIPT FLUSH con regularidad a fin de depurar los scripts de Lua almacenados en caché si es necesario.

En el siguiente ejemplo, se muestra cómo definir y utilizar parámetros: En primer lugar, tenemos el ejemplo de un método no parametrizado que da como resultado tres scripts de Lua en caché diferentes (no recomendable):

```
eval "return redis.call('set','key1','1')" 0
eval "return redis.call('set','key2','2')" 0
eval "return redis.call('set','key3','3')" 0
```

En lugar de esto, utilice el siguiente patrón para crear un único script que pueda aceptar los parámetros pasados:

```
eval "return redis.call('set',KEYS[1],ARGV[1])" 1 key1 1
eval "return redis.call('set',KEYS[1],ARGV[1])" 1 key2 2
eval "return redis.call('set',KEYS[1],ARGV[1])" 1 key3 3
```

Scripts de Lua de larga duración

Los scripts de Lua pueden ejecutar varios comandos de forma atómica, por lo que pueden tardar más en completarse que un comando normal de Redis OSS. Si el script de Lua solo ejecuta operaciones de solo lectura, puede detenerlo durante la ejecución. Sin embargo, cuando el script de Lua lleve a cabo operaciones de escritura, no se puede detener y debe ejecutarse hasta el final. Un script de Lua de ejecución prolongada que esté mutando puede provocar que el servidor OSS de Redis deje de responder durante mucho tiempo. Para mitigar este problema, evite el uso de scripts de Lua de larga duración y pruébelos en un entorno de preproducción.

Script de Lua con escrituras encubiertas

Hay varias maneras en las que un script de Lua puede seguir escribiendo nuevos datos en Redis OSS incluso cuando Redis OSS ha terminado: `maxmemory`

- El script se inicia cuando el servidor OSS de Redis está debajo `maxmemory` y contiene varias operaciones de escritura en su interior
- El primer comando de escritura del script no consume memoria (como `DEL`) y va seguido de más operaciones de escritura que sí consumen memoria.
- Puede mitigar este problema configurando una política de desalojo adecuada en un servidor OSS de Redis que no sea. `noeviction` Esto permite a Redis OSS expulsar elementos y liberar memoria entre los scripts de Lua.

Almacenamiento de elementos compuestos de gran tamaño

En algunos escenarios, una aplicación puede almacenar elementos compuestos de gran tamaño en Redis OSS (como un conjunto de datos hash de varios GB). No se recomienda esta práctica porque suele provocar problemas de rendimiento en Redis OSS. Por ejemplo, el cliente puede ejecutar un comando `HGETALL` para recuperar toda la colección de hash de varios GB. Esto puede generar una presión de memoria significativa para el servidor OSS de Redis que almacena en búfer el elemento grande del búfer de salida del cliente. Además, para la migración de ranuras en modo clúster, ElastiCache no migra las ranuras que contienen elementos con un tamaño serializado superior a 256 MB.

Para resolver los problemas implicados en la gestión de elementos de gran tamaño, recomendamos lo siguiente:

- Divida el elemento compuesto grande en varios elementos más pequeños. Algo que puede hacer es dividir una colección hash grande en campos clave-valor individuales con un esquema de

nombres de clave que refleje adecuadamente la colección; por ejemplo, puede utilizar un prefijo común en el nombre de la clave para identificar la colección de elementos. Si debe acceder a varios campos de la misma colección de forma integral, puede usar el comando MGET para recuperar varios valores clave en el mismo comando.

- Si ha evaluado todas las opciones y sigue sin poder dividir el conjunto de datos de gran tamaño de la colección, intente utilizar comandos que funcionen en un subconjunto de los datos de la colección, en lugar de en toda la colección. Evite los casos de uso que requieran recuperar de forma integral toda la colección de varios GB con el mismo comando. Un ejemplo de esto es el uso de los comandos HGET o HMGET, en lugar de HGETALL, en las colecciones de hash.

Configuración del cliente de Lettuce

En esta sección se describen las opciones de configuración recomendadas de Java y Lettuce y cómo se aplican a ElastiCache los clústeres.

Las recomendaciones de esta sección se probaron con la versión 6.2.2 de Lettuce.

Temas

- [Ejemplo: Configuración de Lettuce para el modo de clúster y TLS habilitado](#)
- [Ejemplo: Configuración de Lettuce para el modo de clúster y TLS desactivado](#)

TTL de la caché DNS de Java

La máquina virtual de Java (JVM) almacena en caché las búsquedas de nombres DNS. Cuando la JVM convierte un nombre de host en una dirección IP, almacena en caché la dirección IP durante un período de tiempo específico, conocido como TTL time-to-live.

La elección del valor de TTL implica un compromiso entre latencia y la capacidad de respuesta a los cambios. Con TTL más cortos, los solucionadores de DNS detectan las actualizaciones en el DNS del clúster con mayor rapidez. Esto puede hacer que la aplicación responda más rápido a las sustituciones u otros flujos de trabajo a los que se somete el clúster. Sin embargo, si el TTL es demasiado bajo, aumenta el volumen de consultas, lo que puede aumentar la latencia de la aplicación. Aunque no existe un valor de TTL correcto, vale la pena esperar a que surta efecto un cambio cuando se configura el valor de TTL.

Como ElastiCache los nodos utilizan entradas de nombres DNS que pueden cambiar, le recomendamos que configure la JVM con un TTL bajo, de 5 a 10 segundos. Con esto, se asegurará

de que, cuando cambie la dirección IP de un nodo, su aplicación pueda recibir y utilizar la nueva dirección IP del recurso volviendo a consultar la entrada de DNS.

En algunas configuraciones de Java, el TTL predeterminado de JVM está establecido de forma que nunca se actualicen las entradas DNS hasta que se reinicie la JVM.

Para obtener más información sobre cómo configurar el TTL de JVM, consulte [Cómo configurar el TTL de JVM](#).

Versión de Lettuce

Recomendamos usar la versión 6.2.2 o posterior de Lettuce.

Puntos de conexión

Cuando utilice clústeres habilitados para el modo de clúster, establezca `redisUri` en el punto de conexión de configuración del clúster. La búsqueda de DNS para este URI devuelve una lista de todos los nodos disponibles en el clúster y se resuelve aleatoriamente en uno de ellos durante la inicialización del clúster. Para obtener más información sobre cómo funciona la actualización de la topología, consulte `dynamicRefreshResources` más adelante en este tema.

SocketOption

Habilitar. [KeepAlive](#) Al habilitar esta opción, se reduce la necesidad de gestionar las conexiones erróneas durante el tiempo de ejecución del comando.

Asegúrese de configurar el [Tiempo de espera de la conexión](#) en función de los requisitos de la aplicación y la carga de trabajo. Para obtener más información, consulte la sección de tiempos de espera más adelante en este tema.

`ClusterClientOption`: Opciones de cliente habilitadas para el modo de clúster

Se activa [AutoReconnect](#) cuando se pierde la conexión.

Configurar [CommandTimeout](#). Para obtener más información, consulte la sección Tiempos de espera más adelante en este tema.

Establezca [nodeFilter](#) para filtrar los nodos con errores de la topología. Lettuce guarda todos los nodos que se encuentran en la salida de los “nodos del clúster” (incluidos los nodos con el estado PFAIL/FAIL) en las “particiones” del cliente. Durante el proceso de creación de la topología del

clúster, intenta conectarse a todos los nodos de partición. Este comportamiento de Lettuce de agregar nodos con errores puede provocar errores de conexión (o advertencias) cuando los nodos se sustituyen por cualquier motivo.

Por ejemplo, una vez que finaliza una conmutación por error y el clúster inicia el proceso de recuperación, mientras se actualiza la topología del clúster, el mapa de nodos del bus del clúster tiene un breve periodo de tiempo en el que el nodo inactivo se muestra como nodo FAIL, antes de que se elimine por completo de la topología. Durante este período, el cliente OSS de Lettuce Redis lo considera un nodo en buen estado y se conecta continuamente a él. Esto provoca un error cuando se agota el reintento.

Por ejemplo:

```
final ClusterClientOptions clusterClientOptions =
    ClusterClientOptions.builder()
        ... // other options
        .nodeFilter(it ->
            ! (it.is(RedisClusterNode.NodeFlag.FAIL)
                || it.is(RedisClusterNode.NodeFlag.EVENTUAL_FAIL)
                || it.is(RedisClusterNode.NodeFlag.HANDSHAKE)
                || it.is(RedisClusterNode.NodeFlag.NOADDR)))
        .validateClusterNodeMembership(false)
        .build();
redisClusterClient.setOptions(clusterClientOptions);
```

Note

El filtrado de nodos se utiliza mejor si se `DynamicRefreshSources` establece en `true`. De lo contrario, si la vista de topología se toma de un solo nodo raíz problemático, que ve que un nodo principal de alguna partición está produciendo un error, filtrará este nodo principal, lo que hará que las ranuras no queden cubiertas. Tener varios nodos iniciales (si `DynamicRefreshSources` es cierto) reduce la probabilidad de que se produzca este problema, ya que al menos algunos de los nodos iniciales deberían tener una vista de topología actualizada tras una conmutación por error con el nodo principal recién promocionado.

`ClusterTopologyRefreshOptions`: Opciones para controlar la actualización de la topología del clúster del cliente con el modo de clúster activado

Note

Los clústeres desactivados en modo de clúster no admiten los comandos de detección de clústeres y no son compatibles con la funcionalidad de detección de topología dinámica de todos los clientes.

El modo de clúster desactivado con ElastiCache no es compatible con el de Lettuce.

MasterSlaveTopologyRefresh En cambio, para el modo de clúster desactivado puede configurar StaticMasterReplicaTopologyProvider y proporcionar los puntos de conexión de lectura y escritura del clúster.

Para obtener más información acerca de la conexión a clústeres desactivados en modo de clúster, consulte [Búsqueda de los puntos finales de un clúster de Redis OSS \(modo de clúster desactivado\) \(consola\)](#).

Si desea utilizar la funcionalidad de detección de topología dinámica de Lettuce, puede crear un clúster habilitado para el modo de clúster con la configuración del mismo fragmento que el clúster existente. Sin embargo, para los clústeres habilitados para el modo de clúster, recomendamos configurar al menos 3 particiones con al menos 1 réplica para admitir una conmutación por error rápida.

Habilitar [enablePeriodicRefresh](#). Esto permite las actualizaciones periódicas de la topología del clúster para que el cliente actualice la topología del clúster en los intervalos del periodo de actualización (predeterminado: 60 segundos). Cuando están desactivadas, el cliente actualiza la topología del clúster solo cuando se producen errores al intentar ejecutar comandos en el clúster.

Con esta opción habilitada, puede reducir la latencia asociada a la actualización de la topología del clúster agregando este trabajo a una tarea que se esté ejecutando en segundo plano. Aunque la actualización de la topología se realiza en un trabajo que se esté ejecutando en segundo plano, puede resultar algo lenta para los clústeres con muchos nodos. Esto se debe a que se están consultando las vistas de todos los nodos para obtener la vista de clúster más actualizada. Si ejecuta un clúster grande, es posible que desee aumentar el periodo.

Habilitar [enableAllAdaptiveRefreshTriggers](#). Esto permite actualizar la topología adaptativa y utilizar todos los [disparadores](#): MOVED_REDIRECT, ASK_REDIRECT, PERSISTENT_RECONNECTS, UNCOVERED_SLOT, UNKNOWN_NODE. Los activadores de actualización adaptativos inician las actualizaciones de la vista de la topología en función de los eventos que se producen durante las operaciones del clúster de Redis OSS. Al habilitar esta opción, se actualiza inmediatamente la topología cuando se activa uno de los desencadenadores anteriores. Las actualizaciones desencadenadas adaptativas tienen un límite de velocidad mediante un tiempo de espera

porque los eventos se pueden producir a gran escala (tiempo de espera predeterminado entre actualizaciones: 30).

Habilitar [closeStaleConnections](#) Esto permite cerrar las conexiones obsoletas al actualizar la topología del clúster. Solo entra en vigor si [ClusterTopologyRefreshOptions.isPeriodicRefreshEnabled \(\)](#) es verdadero. Cuando se habilita, el cliente puede cerrar las conexiones obsoletas y crear otras nuevas en segundo plano. Esto reduce la necesidad de gestionar las conexiones erróneas durante el tiempo de ejecución del comando.

Habilitar [dynamicRefreshResources](#). Recomendamos habilitarlo `dynamicRefreshResources` para clústeres pequeños y deshabilitarlo para clústeres grandes. `dynamicRefreshResources` permite descubrir los nodos del clúster desde el nodo inicial proporcionado (por ejemplo, el punto final de configuración del clúster). Utiliza todos los nodos detectados como orígenes para actualizar la topología del clúster.

El uso de la actualización dinámica consulta todos los nodos detectados para la topología del clúster e intenta elegir la vista de clúster más precisa. Si se establece en falso, solo se utilizan los nodos raíz iniciales como orígenes para la detección de la topología y el número de clientes se obtiene solo para los nodos raíz iniciales. Cuando se desactiva, si el punto de conexión de la configuración del clúster se resuelve como un nodo con errores, se produce un error al intentar actualizar la vista del clúster y se producen excepciones. Este escenario se puede producir porque pasa algún tiempo hasta que la entrada de un nodo con errores se elimina del punto de conexión de la configuración del clúster. Por lo tanto, el punto de conexión de la configuración aún se puede resolver aleatoriamente en un nodo erróneo durante un breve periodo de tiempo.

Sin embargo, cuando se habilita, utilizamos todos los nodos del clúster que se reciben de la vista de clústeres para consultar la vista actual. Como filtramos los nodos con errores de esa vista, la actualización de la topología se realizará correctamente. Sin embargo, cuando `dynamicRefreshSources` es cierto, Lettuce consulta todos los nodos para obtener la vista del clúster y, a continuación, compara los resultados. Por lo tanto, puede resultar caro para los clústeres con muchos nodos. Le sugerimos que desactive esta característica para los clústeres con muchos nodos.

```
final ClusterTopologyRefreshOptions topologyOptions =
    ClusterTopologyRefreshOptions.builder()
        .enableAllAdaptiveRefreshTriggers()
        .enablePeriodicRefresh()
        .dynamicRefreshSources(true)
        .build();
```

ClientResources

Configure [DnsResolver](#) con [DirContextDnsResolver](#). El solucionador de DNS se basa en `com.sun.jndi.dns` de Java. `DnsContextFactory`.

Configure [reconnectDelay](#) con retroceso exponencial y fluctuación total. Lettuce tiene mecanismos de reintento integrados basados en las estrategias de retroceso exponencial. Para obtener más información, consulte [Retroceso exponencial y fluctuación](#) en el AWS blog de arquitectura. Para obtener más información sobre la importancia de contar con una estrategia de retroceso de los reintentos, consulte las secciones sobre lógica de retroceso de la entrada del blog sobre [mejores prácticas en el blog](#) de bases de datos. AWS

```
ClientResources clientResources = DefaultClientResources.builder()
    .dnsResolver(new DirContextDnsResolver())
    .reconnectDelay(
        Delay.fullJitter(
            Duration.ofMillis(100), // minimum 100 millisecond delay
            Duration.ofSeconds(10), // maximum 10 second delay
            100, TimeUnit.MILLISECONDS)) // 100 millisecond base
    .build();
```

Tiempos de espera

Utilice un valor de tiempo de espera de conexión inferior al tiempo de espera del comando. Lettuce utiliza un establecimiento de conexión diferida. Por lo tanto, si el tiempo de espera de conexión es superior al tiempo de espera del comando, puede producirse un periodo de error persistente tras una actualización de la topología si Lettuce intenta conectarse a un nodo en mal estado y siempre se supera el tiempo de espera del comando.

Utilice un tiempo de espera de comando dinámico para diferentes comandos. Le recomendamos que establezca el tiempo de espera del comando en función de la duración esperada del comando. Por ejemplo, utilice un tiempo de espera más largo para los comandos que se repiten en varias claves, como los scripts FLUSHDB, FLUSHALL, KEYS, SMEMBERS o Lua. Utilice tiempos de espera más cortos para los comandos de una sola clave, como SET, GET y HSET.

Note

Los tiempos de espera que se configuran en el siguiente ejemplo son para pruebas que ejecutaron comandos SET/GET con claves y valores de hasta 20 bytes de longitud. El tiempo de procesamiento puede ser mayor cuando los comandos son complejos o las claves y los

valores son más grandes. Debe establecer los tiempos de espera en función del caso de uso de la aplicación.

```
private static final Duration META_COMMAND_TIMEOUT = Duration.ofMillis(1000);
private static final Duration DEFAULT_COMMAND_TIMEOUT = Duration.ofMillis(250);
// Socket connect timeout should be lower than command timeout for Lettuce
private static final Duration CONNECT_TIMEOUT = Duration.ofMillis(100);
```

```
SocketOptions socketOptions = SocketOptions.builder()
    .connectTimeout(CONNECT_TIMEOUT)
    .build();
```

```
class DynamicClusterTimeout extends TimeoutSource {
    private static final Set<ProtocolKeyword> META_COMMAND_TYPES =
    ImmutableSet.<ProtocolKeyword>builder()
        .add(CommandType.FLUSHDB)
        .add(CommandType.FLUSHALL)
        .add(CommandType.CLUSTER)
        .add(CommandType.INFO)
        .add(CommandType.KEYS)
        .build();

    private final Duration defaultCommandTimeout;
    private final Duration metaCommandTimeout;

    DynamicClusterTimeout(Duration defaultTimeout, Duration metaTimeout)
    {
        defaultCommandTimeout = defaultTimeout;
        metaCommandTimeout = metaTimeout;
    }

    @Override
    public long getTimeout(RedisCommand<?, ?, ?> command) {
        if (META_COMMAND_TYPES.contains(command.getType())) {
            return metaCommandTimeout.toMillis();
        }
        return defaultCommandTimeout.toMillis();
    }
}

// Use a dynamic timeout for commands, to avoid timeouts during
```

```
// cluster management and slow operations.
TimeoutOptions timeoutOptions = TimeoutOptions.builder()
    .timeoutSource(
        new DynamicClusterTimeout(DEFAULT_COMMAND_TIMEOUT, META_COMMAND_TIMEOUT))
    .build();
```

Ejemplo: Configuración de Lettuce para el modo de clúster y TLS habilitado

Note

Los tiempos de espera del siguiente ejemplo son para pruebas que ejecutaron comandos SET/GET con claves y valores de hasta 20 bytes de longitud. El tiempo de procesamiento puede ser mayor cuando los comandos son complejos o las claves y los valores son más grandes. Debe establecer los tiempos de espera en función del caso de uso de la aplicación.

```
// Set DNS cache TTL
public void setJVMPProperties() {
    java.security.Security.setProperty("networkaddress.cache.ttl", "10");
}

private static final Duration META_COMMAND_TIMEOUT = Duration.ofMillis(1000);
private static final Duration DEFAULT_COMMAND_TIMEOUT = Duration.ofMillis(250);
// Socket connect timeout should be lower than command timeout for Lettuce
private static final Duration CONNECT_TIMEOUT = Duration.ofMillis(100);

// Create RedisURI from the cluster configuration endpoint
clusterConfigurationEndpoint = <cluster-configuration-endpoint> // TODO: add your
    cluster configuration endpoint
final RedisURI redisUriCluster =
    RedisURI.Builder.redis(clusterConfigurationEndpoint)
        .withPort(6379)
        .withSsl(true)
        .build();

// Configure the client's resources
ClientResources clientResources = DefaultClientResources.builder()
    .reconnectDelay(
        Delay.fullJitter(
            Duration.ofMillis(100), // minimum 100 millisecond delay
            Duration.ofSeconds(10), // maximum 10 second delay
            100, TimeUnit.MILLISECONDS)) // 100 millisecond base
```

```
.dnsResolver(new DirContextDnsResolver())
    .build();

// Create a cluster client instance with the URI and resources
RedisClusterClient redisClusterClient =
    RedisClusterClient.create(clientResources, redisUriCluster);

// Use a dynamic timeout for commands, to avoid timeouts during
// cluster management and slow operations.
class DynamicClusterTimeout extends TimeoutSource {
    private static final Set<ProtocolKeyword> META_COMMAND_TYPES =
    ImmutableSet.<ProtocolKeyword>builder()
        .add(CommandType.FLUSHDB)
        .add(CommandType.FLUSHALL)
        .add(CommandType.CLUSTER)
        .add(CommandType.INFO)
        .add(CommandType.KEYS)
        .build();

    private final Duration metaCommandTimeout;
    private final Duration defaultCommandTimeout;

    DynamicClusterTimeout(Duration defaultTimeout, Duration metaTimeout)
    {
        defaultCommandTimeout = defaultTimeout;
        metaCommandTimeout = metaTimeout;
    }

    @Override
    public long getTimeout(RedisCommand<?, ?, ?> command) {
        if (META_COMMAND_TYPES.contains(command.getType())) {
            return metaCommandTimeout.toMillis();
        }
        return defaultCommandTimeout.toMillis();
    }
}

TimeoutOptions timeoutOptions = TimeoutOptions.builder()
    .timeoutSource(new DynamicClusterTimeout(DEFAULT_COMMAND_TIMEOUT,
    META_COMMAND_TIMEOUT))
    .build();

// Configure the topology refreshment options
final ClusterTopologyRefreshOptions topologyOptions =
```

```
ClusterTopologyRefreshOptions.builder()
    .enableAllAdaptiveRefreshTriggers()
    .enablePeriodicRefresh()
    .dynamicRefreshSources(true)
    .build();

// Configure the socket options
final SocketOptions socketOptions =
    SocketOptions.builder()
        .connectTimeout(CONNECT_TIMEOUT)
        .keepAlive(true)
        .build();

// Configure the client's options
final ClusterClientOptions clusterClientOptions =
    ClusterClientOptions.builder()
        .topologyRefreshOptions(topologyOptions)
        .socketOptions(socketOptions)
        .autoReconnect(true)
        .timeoutOptions(timeoutOptions)
        .nodeFilter(it ->
            ! (it.is(RedisClusterNode.NodeFlag.FAIL)
                || it.is(RedisClusterNode.NodeFlag.EVENTUAL_FAIL)
                || it.is(RedisClusterNode.NodeFlag.NOADDR)))
        .validateClusterNodeMembership(false)
        .build();

redisClusterClient.setOptions(clusterClientOptions);

// Get a connection
final StatefulRedisClusterConnection<String, String> connection =
    redisClusterClient.connect();

// Get cluster sync/async commands
RedisAdvancedClusterCommands<String, String> sync = connection.sync();
RedisAdvancedClusterAsyncCommands<String, String> async = connection.async();
```

Ejemplo: Configuración de Lettuce para el modo de clúster y TLS desactivado

Note

Los tiempos de espera del siguiente ejemplo son para pruebas que ejecutaron comandos SET/GET con claves y valores de hasta 20 bytes de longitud. El tiempo de procesamiento

puede ser mayor cuando los comandos son complejos o las claves y los valores son más grandes. Debe establecer los tiempos de espera en función del caso de uso de la aplicación.

```
// Set DNS cache TTL
public void setJVMProperties() {
    java.security.Security.setProperty("networkaddress.cache.ttl", "10");
}

private static final Duration META_COMMAND_TIMEOUT = Duration.ofMillis(1000);
private static final Duration DEFAULT_COMMAND_TIMEOUT = Duration.ofMillis(250);
// Socket connect timeout should be lower than command timeout for Lettuce
private static final Duration CONNECT_TIMEOUT = Duration.ofMillis(100);

// Create RedisURI from the primary/reader endpoint
clusterEndpoint = <primary/reader-endpoint> // TODO: add your node endpoint
RedisURI redisUriStandalone =

    RedisURI.Builder.redis(clusterEndpoint).withPort(6379).withSsl(true).withDatabase(0).build();

ClientResources clientResources =
    DefaultClientResources.builder()
        .dnsResolver(new DirContextDnsResolver())
        .reconnectDelay(
            Delay.fullJitter(
                Duration.ofMillis(100), // minimum 100 millisecond delay
                Duration.ofSeconds(10), // maximum 10 second delay
                100,
                TimeUnit.MILLISECONDS)) // 100 millisecond base
        .build();

// Use a dynamic timeout for commands, to avoid timeouts during
// slow operations.
class DynamicTimeout extends TimeoutSource {
    private static final Set<ProtocolKeyword> META_COMMAND_TYPES =
    ImmutableSet.<ProtocolKeyword>builder()
        .add(CommandType.FLUSHDB)
        .add(CommandType.FLUSHALL)
        .add(CommandType.INFO)
        .add(CommandType.KEYS)
        .build();

    private final Duration metaCommandTimeout;
```

```

private final Duration defaultCommandTimeout;

DynamicTimeout(Duration defaultTimeout, Duration metaTimeout)
{
    defaultCommandTimeout = defaultTimeout;
    metaCommandTimeout = metaTimeout;
}

@Override
public long getTimeout(RedisCommand<?, ?, ?> command) {
    if (META_COMMAND_TYPES.contains(command.getType())) {
        return metaCommandTimeout.toMillis();
    }
    return defaultCommandTimeout.toMillis();
}
}

TimeoutOptions timeoutOptions = TimeoutOptions.builder()
    .timeoutSource(new DynamicTimeout(DEFAULT_COMMAND_TIMEOUT, META_COMMAND_TIMEOUT))
    .build();

final SocketOptions socketOptions =
    SocketOptions.builder().connectTimeout(CONNECT_TIMEOUT).keepAlive(true).build();

ClientOptions clientOptions =

    ClientOptions.builder().timeoutOptions(timeoutOptions).socketOptions(socketOptions).build();

RedisClient redisClient = RedisClient.create(clientResources, redisUriStandalone);
redisClient.setOptions(clientOptions);

```

Ejemplos de cliente de IPv6

Las siguientes son las mejores prácticas para interactuar con ElastiCache recursos habilitados para IPv6 con bibliotecas de clientes de código abierto de uso común. Puede consultar [las prácticas recomendadas actuales con las que interactuar y obtener](#) recomendaciones sobre ElastiCache la configuración de los clientes para ElastiCache los recursos. Sin embargo, hay algunas advertencias que merece la pena señalar al interactuar con recursos habilitados para IPv6.

Cientes validados

ElastiCache es compatible con Redis OSS de código abierto. Esto significa que los clientes Redis OSS de código abierto que admiten conexiones IPv6 deberían poder conectarse a clústeres

habilitados para IPv6 ElastiCache (Redis OSS). Además, varios de los clientes de Python y Java más populares se han probado y validado específicamente para que funcionen con todas las configuraciones de tipos de red compatibles (solo IPv4, solo IPv6 y doble pila).

Clientes validados:

- [Redis Py \(\)](#): [4.1.2](#)
- [Lettuce](#): [versión 6.1.6.RELEASE](#)
- [Jedis](#): [versión 3.6.0](#)

Configuración de un protocolo preferido para clústeres de doble pila

En el caso de los clústeres Redis OSS habilitados para el modo de clúster, puede controlar el protocolo que utilizarán los clientes para conectarse a los nodos del clúster con el parámetro IP Discovery. El parámetro de detección de IP se puede establecer en IPv4 o IPv6.

En el caso de los clústeres OSS de Redis, el parámetro de detección de IP establece el protocolo IP utilizado en la [salida de las ranuras \(\)](#), [los fragmentos de clúster \(\)](#) y [los nodos del clúster \(\)](#). Los clientes utilizan estos comandos para detectar la topología del clúster. Los clientes usan las IP de estos comandos para conectarse a los otros nodos del clúster.

Cambiar la detección de IP no provocará ningún tiempo de inactividad para los clientes conectados. Sin embargo, los cambios tardarán algún tiempo en propagarse. Para determinar si los cambios se han propagado por completo en un clúster de OSS de Redis, supervise el resultado de `cluster slots`. Una vez que todos los nodos devueltos por el comando de ranuras del clúster registren las IP con el nuevo protocolo, los cambios terminarán de propagarse.

Ejemplo con Redis-Py:

```
cluster = RedisCluster(host="xxxx", port=6379)
target_type = IPv6Address # Or IPv4Address if changing to IPv4

nodes = set()
while len(nodes) == 0 or not all((type(ip_address(host)) is target_type) for host in
nodes):
    nodes = set()

# This refreshes the cluster topology and will discovery any node updates.
# Under the hood it calls cluster slots
cluster.nodes_manager.initialize()
```

```

for node in cluster.get_nodes():
    nodes.add(node.host)
self.logger.info(nodes)

time.sleep(1)

```

Ejemplo con Lettuce:

```

RedisClusterClient clusterClient = RedisClusterClient.create(RedisURI.create("xxxx",
6379));

Class targetProtocolType = Inet6Address.class; // Or Inet4Address.class if you're
switching to IPv4

Set<String> nodes;

do {
    // Check for any changes in the cluster topology.
    // Under the hood this calls cluster slots
    clusterClient.refreshPartitions();
    Set<String> nodes = new HashSet<>();

    for (RedisClusterNode node : clusterClient.getPartitions().getPartitions()) {
        nodes.add(node.getUri().getHost());
    }

    Thread.sleep(1000);
} while (!nodes.stream().allMatch(node -> {
    try {
        return finalTargetProtocolType.isInstance(InetAddress.getByName(node));
    } catch (UnknownHostException ignored) {}
    return false;
})));

```

Clústeres de doble pila compatibles con TLS ElastiCache

Cuando el TLS está habilitado para ElastiCache los clústeres, las funciones de detección de clústeres (`cluster slotscluster shards`, `ycluster nodes`) devuelven nombres de host en lugar de direcciones IP. A continuación, se utilizan los nombres de host en lugar de las IP para conectarse al ElastiCache clúster y realizar un protocolo de enlace TLS. Esto significa que los clientes no se verán afectados por el parámetro de detección de IP. En el caso de los clústeres habilitados para TLS, el parámetro de detección de IP no tiene ningún efecto en el protocolo IP

preferido. En cambio, el protocolo IP utilizado se determinará según el protocolo IP que prefiera el cliente al resolver los nombres de host de DNS.

Cientes de Java

Al conectarse desde un entorno de Java que admite IPv4 e IPv6, Java preferirá de forma predeterminada IPv4 en lugar de IPv6 por motivos de compatibilidad con versiones anteriores. Sin embargo, la preferencia del protocolo IP se puede configurar mediante los argumentos de JVM. Para preferir IPv4, JVM acepta `-Djava.net.preferIPv4Stack=true` y para preferir IPv6 establece `-Djava.net.preferIPv6Stack=true`. La configuración de `-Djava.net.preferIPv4Stack=true` significa que JVM ya no realizará ninguna conexión IPv6. Incluidos los de otras aplicaciones de OSS que no son de Redis.

Preferencias de nivel de host

En general, si el cliente o el entorno de ejecución del cliente no ofrecen opciones de configuración para establecer una preferencia de protocolo IP, al realizar la resolución de DNS, el protocolo IP dependerá de la configuración del host. De forma predeterminada, la mayoría de los hosts prefieren IPv6 en lugar de IPv4, pero esta preferencia se puede establecer en el nivel de host. Esto afectará a todas las solicitudes de DNS de ese host, no solo a las dirigidas a los ElastiCache clústeres.

Hosts de Linux

Para Linux, se puede configurar una preferencia de protocolo IP modificando el archivo `gai.conf`. El archivo `gai.conf` se encuentra en `/etc/gai.conf`. Si no se especifica `gai.conf`, debería haber disponible un ejemplo en `/usr/share/doc/glibc-common-x.xx/gai.conf` que se pueda copiar a `/etc/gai.conf`. Además, la configuración predeterminada no debe estar comentada. Para actualizar la configuración para que prefiera el IPv4 al conectarse a un ElastiCache clúster, actualice la prioridad del rango CIDR que abarca las IP del clúster para que esté por encima de la prioridad de las conexiones IPv6 predeterminadas. De forma predeterminada, las conexiones IPv6 tienen una prioridad de 40. Por ejemplo, suponiendo que el clúster esté ubicado en una subred con el CIDR `172.31.0.0/16`, la siguiente configuración haría que los clientes prefirieran las conexiones IPv4 a ese clúster.

```
label ::1/128      0
label ::/0        1
label 2002::/16   2
label ::/96       3
label ::ffff:0:0/96 4
label fec0::/10   5
```

```

label fc00::/7      6
label 2001:0::/32  7
label ::ffff:172.31.0.0/112 8
#
# This default differs from the tables given in RFC 3484 by handling
# (now obsolete) site-local IPv6 addresses and Unique Local Addresses.
# The reason for this difference is that these addresses are never
# NATed while IPv4 site-local addresses most probably are. Given
# the precedence of IPv6 over IPv4 (see below) on machines having only
# site-local IPv4 and IPv6 addresses a lookup for a global address would
# see the IPv6 be preferred. The result is a long delay because the
# site-local IPv6 addresses cannot be used while the IPv4 address is
# (at least for the foreseeable future) NATed. We also treat Teredo
# tunnels special.
#
# precedence <mask> <value>
# Add another rule to the RFC 3484 precedence table. See section 2.1
# and 10.3 in RFC 3484. The default is:
#
precedence ::1/128      50
precedence ::/0         40
precedence 2002::/16    30
precedence ::/96        20
precedence ::ffff:0:0/96 10
precedence ::ffff:172.31.0.0/112 100

```

Puede encontrar más información disponible sobre `gai.conf` en la [página principal de Linux](#)

Hosts de Windows

El proceso para los hosts de Windows es similar. Para los hosts de Windows puede ejecutar `netsh interface ipv6 set prefix CIDR_CONTAINING_CLUSTER_IPS PRECEDENCE LABEL`. Esto tiene el mismo efecto que modificar el archivo `gai.conf` en los hosts de Linux.

Esto actualizará las políticas de preferencias, de modo que se prefieran las conexiones IPv4 en lugar de las conexiones IPv6 para el rango de CIDR especificado. Por ejemplo, suponiendo que el clúster esté en una subred con el CIDR `172.31.0.0/16`, ejecutar `netsh interface ipv6 set prefix ::ffff:172.31.0.0:0/112 100 15` generaría la siguiente tabla de prioridades, lo que haría que los clientes prefirieran IPv4 al conectarse al clúster.

```

C:\Users\Administrator>netsh interface ipv6 show prefixpolicies
Querying active state...

```

Precedence Label Prefix

```
-----  
100 15 ::ffff:172.31.0.0:0/112  
20 4 ::ffff:0:0/96  
50 0 ::1/128  
40 1 ::/0  
30 2 2002::/16  
5 5 2001::/32  
3 13 fc00::/7  
1 11 fec0::/10  
1 12 3ffe::/16  
1 3 ::/96
```

Administración de la memoria reservada

La memoria reservada es una memoria que se aparta del uso para los datos. Al realizar una copia de seguridad o una conmutación por error, Redis OSS utiliza la memoria disponible para grabar las operaciones de escritura en el clúster mientras se escriben los datos del clúster en el archivo.rdb. Si no dispone de suficiente memoria disponible para todas las operaciones de escritura, se produce un error en el proceso. A continuación, encontrará información sobre las opciones para administrar la memoria reservada para ElastiCache (Redis OSS) y cómo aplicarlas.

Temas

- [¿Cuánta memoria reservada necesita?](#)
- [Parámetros de administración de la memoria reservada](#)
- [Especificación del parámetro de administración de memoria reservada](#)

¿Cuánta memoria reservada necesita?

Si ejecuta una versión de Redis OSS anterior a la 2.8.22, reserve más memoria para las copias de seguridad y las conmutaciones por error que si ejecuta Redis OSS 2.8.22 o posterior. Este requisito se debe a las diferentes formas en que ElastiCache (Redis OSS) implementa el proceso de copia de seguridad. La regla general es reservar la mitad del `maxmemory` valor de un tipo de nodo para la sobrecarga de Redis OSS en las versiones anteriores a la 2.8.22 y una cuarta parte para las versiones 2.8.22 y posteriores de Redis OSS.

Debido a las diferentes formas de ElastiCache implementar el proceso de copia de seguridad y replicación, la regla general es reservar el 25% del valor de un tipo de nodo mediante el uso del parámetro `maxmemory reserved-memory-percent`. Este es el valor predeterminado y se recomienda en la mayoría de los casos.

Cuando los tipos de microinstancias y pequeñas con capacidad de fragmentación funcionan cerca de los `maxmemory` límites, es posible que sufran un uso de intercambio. Para mejorar la confiabilidad operativa de estos tipos de instancias durante el respaldo, la replicación y el tráfico intenso, recomendamos aumentar el valor del `reserved-memory-percent` parámetro hasta un 30% en los tipos de instancias pequeñas y hasta un 50% en los tipos de microinstancias.

Para cargas de trabajo con un uso intensivo de escritura en ElastiCache clústeres con almacenamiento en niveles de datos, recomendamos aumentar la memoria disponible del nodo `reserved-memory-percent` hasta un 50%.

Para más información, consulte los siguientes temas:

- [Asegurarse de tener suficiente memoria para crear una instantánea de Redis OSS](#)
- [Cómo se implementan la sincronización y la copia de seguridad](#)
- [Organización de datos en niveles](#)

Parámetros de administración de la memoria reservada

A partir del 16 de marzo de 2017, Amazon ElastiCache (Redis OSS) proporciona dos parámetros que se excluyen mutuamente para administrar la memoria de Redis OSS, `reserved-memory` y `reserved-memory-percent`. Ninguno de estos parámetros forma parte de la distribución de Redis OSS.

Dependiendo de cuándo se convirtió en ElastiCache cliente, uno u otro de estos parámetros es el parámetro de administración de memoria predeterminado. Este parámetro se aplica cuando se crea un nuevo clúster o grupo de replicación de Redis OSS y se utiliza un grupo de parámetros predeterminado.

- Para los clientes que comenzaron antes del 16 de marzo de 2017: cuando crea un clúster o grupo de replicación de Redis OSS con el grupo de parámetros predeterminado, su parámetro de administración de memoria es `reserved-memory`. En este caso, cero (0) bytes de memoria se encuentran reservados.
- Para los clientes que comenzaron el 16 de marzo de 2017 o después de esa fecha: al crear un clúster o grupo de replicación de Redis OSS con el grupo de parámetros predeterminado, su parámetro de administración de memoria es `reserved-memory-percent`. En este caso, el 25 % del valor `maxmemory` del nodo se encuentra reservado para fines no relacionados con datos.

Tras leer acerca de los dos parámetros de administración de memoria del OSS de Redis, puede que prefiera utilizar el que no sea el predeterminado o el que tenga valores no predeterminados. En este caso, puede cambiar al otro parámetro de administración de la memoria reservada.

Para cambiar el valor de ese parámetro, puede crear un grupo de parámetros personalizado y modificarlo a fin de utilizar el parámetro y el valor de administración de la memoria preferidos. A continuación, podrá utilizar el grupo de parámetros personalizado siempre que cree un nuevo clúster o grupo de replicación de Redis OSS. Para los clústeres o grupos de reproducción existentes, puede modificarlos para que usen su grupo de parámetros personalizado.

Para más información, consulte los siguientes temas:

- [Especificación del parámetro de administración de memoria reservada](#)
- [Creación de un grupo de parámetros](#)
- [Modificación de un grupo de parámetros](#)
- [Modificación de un ElastiCache clúster](#)
- [Modificación de un grupo de reproducción](#)

Parámetro reserved-memory

Antes del 16 de marzo de 2017, toda la administración de la memoria reservada ElastiCache (Redis OSS) se realizaba mediante el parámetro. `reserved-memory`. El valor predeterminado de `reserved-memory` es 0. Este valor predeterminado no reserva memoria para la sobrecarga de Redis OSS y permite que Redis OSS consuma toda la memoria de un nodo con datos.

La modificación del parámetro `reserved-memory` para tener suficiente memoria disponible para las copias de seguridad y las conmutaciones por error requiere crear un grupo de parámetros personalizado. En este grupo de parámetros personalizados, se establece `reserved-memory` un valor adecuado para la versión de Redis OSS que se ejecuta en el clúster y el tipo de nodo del clúster. Para obtener más información, consulte [¿Cuánta memoria reservada necesita?](#).

El parámetro `reserved-memory` es específico de ElastiCache (Redis OSS) y no forma parte de la distribución general de Redis OSS.

El siguiente procedimiento muestra cómo administrar la `reserved-memory` memoria del clúster de Redis OSS.

Para reservar memoria mediante `reserved-memory`

1. Cree un grupo de parámetros personalizado especificando la familia del grupo de parámetros que coincide con la versión del motor que está ejecutando, como, por ejemplo, la familia del grupo de parámetros `redis2.8`. Para obtener más información, consulte [Creación de un grupo de parámetros](#).

```
aws elasticache create-cache-parameter-group \  
  --cache-parameter-group-name redis6x-m3xl \  
  --description "Redis OSS 2.8.x for m3.xlarge node type" \  
  --cache-parameter-group-family redis6.x
```

2. Calcule cuántos bytes de memoria debe reservar para la sobrecarga de Redis OSS. Encontrará el valor de `maxmemory` para su tipo de nodo en [Parámetros específicos del tipo de nodo de Redis OSS](#).
3. Modifique el grupo de parámetros personalizado de modo que el parámetro `reserved-memory` coincida con el número de bytes que calculó en el paso anterior. En el siguiente AWS CLI ejemplo, se supone que está ejecutando una versión de Redis OSS anterior a la 2.8.22 y que necesita reservar la mitad de la de los nodos. `maxmemory` Para obtener más información, consulte [Modificación de un grupo de parámetros](#).

```
aws elasticache modify-cache-parameter-group \  
  --cache-parameter-group-name redis28-m3x1 \  
  --parameter-name-values "ParameterName=reserved-memory,  
  ParameterValue=7130316800"
```

Necesita un grupo de parámetros personalizado individual para cada tipo de nodo que utilice, ya que cada tipo de nodo tiene un valor de `maxmemory` distinto. Por lo tanto, cada tipo de nodo necesita un valor diferente para `reserved-memory`.

4. Modifique su clúster o grupo de replicación de Redis OSS para usar su grupo de parámetros personalizado.

En el ejemplo de la CLI siguiente se modifica el clúster `my-redis-cluster` de forma que comience a usar de inmediato el grupo de parámetros personalizado `redis28-m3x1`. Para obtener más información, consulte [Modificación de un ElastiCache clúster](#).

```
aws elasticache modify-cache-cluster \  
  --cache-cluster-id my-redis-cluster \  
  --cache-parameter-group-name redis28-m3x1 \  
  --apply-immediately
```

En el ejemplo de la CLI siguiente se modifica el grupo de reproducción `my-redis-repl-grp` de forma que comience a usar de inmediato el grupo de parámetros personalizado `redis28-m3x1`. Para obtener más información, [Modificación de un grupo de reproducción](#).

```
aws elasticache modify-replication-group \  
  --replication-group-id my-redis-repl-grp \  
  --cache-parameter-group-name redis28-m3x1 \  
  --apply-immediately
```

El parámetro reserved-memory-percent

El 16 de marzo de 2017, Amazon ElastiCache introdujo el parámetro `reserved-memory-percent` y lo puso a disposición en todas las versiones de ElastiCache (Redis OSS). El propósito del parámetro `reserved-memory-percent` es simplificar la administración de la memoria reservada en todos los clústeres. Esta simplificación se consigue al disponer de un único grupo de parámetros para cada familia de grupos de parámetros (como `redis2.8`) para administrar la memoria reservada de sus clústeres, sea cual sea el tipo de nodo. El valor de predeterminado para `reserved-memory-percent` es 25 (25 por ciento).

El parámetro `reserved-memory-percent` es específico de ElastiCache (Redis OSS) y no forma parte de la distribución general de Redis OSS.

Si el clúster utiliza un tipo de nodo de la familia `r6gd` y el uso de memoria alcanza el 75 por ciento, la organización de datos en niveles se activará automáticamente. Para obtener más información, consulte [Organización de datos en niveles](#).

Para reservar memoria mediante `reserved-memory-percent`

`reserved-memory-percent` Para administrar la memoria de su clúster ElastiCache (Redis OSS), realice una de las siguientes acciones:

- Si ejecuta Redis OSS 2.8.22 o una versión posterior, asigne el grupo de parámetros predeterminado al clúster. El valor predeterminado de 25 debe ser suficiente. De lo contrario, siga los pasos que se describen a continuación para cambiar el valor.
- Si ejecuta una versión de Redis OSS anterior a la 2.8.22, probablemente necesite reservar más memoria de la predeterminada, un 25 por ciento. `reserved-memory-percent` Para ello, siga el procedimiento que se indica a continuación.

Para cambiar el valor porcentual de `reserved-memory-percent`

1. Cree un grupo de parámetros personalizado especificando la familia del grupo de parámetros que coincide con la versión del motor que está ejecutando, como, por ejemplo, la familia del grupo de parámetros `redis2.8`. Necesita disponer de un grupo de parámetros personalizado porque no es posible modificar grupos de parámetros predeterminados. Para obtener más información, consulte [Creación de un grupo de parámetros](#).

```
aws elasticache create-cache-parameter-group \  
  --cache-parameter-group-name redis28-50 \  
  --parameter-names reserved-memory-percent
```

```
--description "Redis OSS 2.8.x 50% reserved" \  
--cache-parameter-group-family redis2.8
```

Dado que `reserved-memory-percent` se reserva la memoria como un porcentaje del valor `maxmemory` de un nodo, no necesita un grupo de parámetros personalizado para cada tipo de nodo.

2. Modifique el grupo de parámetros personalizados de modo que el parámetro `reserved-memory-percent` tenga el valor 50 (50 %). Para obtener más información, consulte [Modificación de un grupo de parámetros](#).

```
aws elasticache modify-cache-parameter-group \  
  --cache-parameter-group-name redis28-50 \  
  --parameter-name-values "ParameterName=reserved-memory-percent,  
  ParameterValue=50"
```

3. Utilice este grupo de parámetros personalizado para cualquier clúster de Redis OSS o grupo de replicación que ejecute una versión de Redis OSS anterior a la 2.8.22.

El siguiente ejemplo de CLI modifica el clúster OSS de Redis `my-redis-cluster` para usar el grupo de parámetros personalizado de forma inmediata `redis28-50`. Para obtener más información, consulte [Modificación de un ElastiCache clúster](#).

```
aws elasticache modify-cache-cluster \  
  --cache-cluster-id my-redis-cluster \  
  --cache-parameter-group-name redis28-50 \  
  --apply-immediately
```

El siguiente ejemplo de CLI modifica el grupo de replicación de Redis OSS `my-redis-repl-grp` para utilizar el grupo de parámetros personalizado de forma inmediata `redis28-50`. Para obtener más información, consulte [Modificación de un grupo de reproducción](#).

```
aws elasticache modify-replication-group \  
  --replication-group-id my-redis-repl-grp \  
  --cache-parameter-group-name redis28-50 \  
  --apply-immediately
```

Especificación del parámetro de administración de memoria reservada

Si era ElastiCache cliente actual el 16 de marzo de 2017, su parámetro de administración de memoria reservada predeterminado es `reserved-memory` con cero (0) bytes de memoria reservada. Si se convirtió en ElastiCache cliente después del 16 de marzo de 2017, su parámetro de administración de memoria reservada predeterminado es `reserved-memory-percent` tener el 25 por ciento de la memoria del nodo reservada. Esto es cierto independientemente de cuándo creó su clúster o grupo de replicación ElastiCache (Redis OSS). Sin embargo, puede cambiar el parámetro de administración de la memoria reservada mediante la ElastiCache API AWS CLI o.

Los parámetros `reserved-memory` y `reserved-memory-percent` son mutuamente excluyentes. Un grupo de parámetros siempre tiene un parámetro, pero nunca ambos. Puede cambiar el parámetro que usa el grupo de parámetros para la administración de memoria reservada modificando el grupo de parámetros. El grupo de parámetros debe ser personalizado, ya que no es posible modificar grupos de parámetros predeterminados. Para obtener más información, consulte [Creación de un grupo de parámetros](#).

Para especificar `reserved-memory-percent`

Para utilizar `reserved-memory-percent` como parámetro de administración de memoria reservada, modifique un grupo de parámetros personalizado mediante el comando `modify-cache-parameter-group`. Utilice el parámetro `parameter-name-values` para especificar `reserved-memory-percent` y un valor para ello.

El siguiente ejemplo de la CLI modifica el grupo de parámetros personalizados `redis32-cluster-on` para usar el parámetro `reserved-memory-percent` para administrar la memoria reservada. Se debe asignar un valor a `ParameterValue` para que el grupo de parámetros utilice el parámetro `ParameterName` a fin de administrar la memoria reservada. Para obtener más información, consulte [Modificación de un grupo de parámetros](#).

```
aws elasticache modify-cache-parameter-group \  
  --cache-parameter-group-name redis32-cluster-on \  
  --parameter-name-values "ParameterName=reserved-memory-percent, ParameterValue=25"
```

Para especificar `reserved-memory`

Para utilizar `reserved-memory` como parámetro de administración de memoria reservada, modifique un grupo de parámetros personalizado mediante el comando `modify-cache-parameter-group`. Utilice el parámetro `parameter-name-values` para especificar `reserved-memory` y un valor para ello.

El siguiente ejemplo de la CLI modifica el grupo de parámetros personalizados `redis32-m3x1` para usar el parámetro `reserved-memory` para administrar la memoria reservada. Se debe asignar un valor a `ParameterValue` para que el grupo de parámetros utilice el parámetro `ParameterName` a fin de administrar la memoria reservada. Dado que la versión del motor es posterior a la versión 2.8.22, establecemos el valor en `3565158400`, que es el 25 % del valor de `maxmemory` de `cache.m3.xlarge`. Para obtener más información, consulte [Modificación de un grupo de parámetros](#).

```
aws elasticache modify-cache-parameter-group \  
  --cache-parameter-group-name redis32-m3x1 \  
  --parameter-name-values "ParameterName=reserved-memory, ParameterValue=3565158400"
```

Procedimientos recomendados al trabajar con clústeres de autodiseño

Esta sección solo se aplica cuando decide diseñar sus propios clústeres de Redis OSS. Le recomendamos que revise y siga estos procedimientos recomendados.

Temas

- [Minimización del tiempo de inactividad con Multi-AZ](#)
- [Asegurarse de tener suficiente memoria para crear una instantánea de Redis OSS](#)
- [Cambio de tamaño de clústeres online](#)
- [Minimización del tiempo de inactividad durante el mantenimiento](#)

Minimización del tiempo de inactividad con Multi-AZ

Consulte [Minimizar el tiempo de inactividad en ElastiCache \(Redis OSS\) con Multi-AZ](#) para obtener más información sobre Multi-AZ y minimizar el tiempo de inactividad.

Asegurarse de tener suficiente memoria para crear una instantánea de Redis OSS

Instantáneas y sincronizaciones de Redis OSS en la versión 2.8.22 y posteriores

Redis OSS 2.8.22 presenta un proceso de guardado sencillo que le permite asignar una mayor cantidad de memoria al uso de la aplicación sin incurrir en un aumento del uso del intercambio durante las sincronizaciones y los guardados. Para obtener más información, consulte [Cómo se implementan la sincronización y la copia de seguridad](#).

Instantáneas y sincronizaciones de Redis OSS anteriores a la versión 2.8.22

Cuando trabaja con ElastiCache (Redis OSS), Redis OSS invoca un comando de escritura en segundo plano en varios casos:

- Al crear una instantánea para una copia de seguridad.
- Al sincronizar réplicas con la instancia principal en un grupo de reproducción.
- Al habilitar la función de solo adjuntar archivos (AOF) para Redis OSS.
- Al promocionar una réplica al nodo primario (lo que produce la sincronización del nodo primario/réplica).

Siempre que Redis OSS ejecute un proceso de escritura en segundo plano, debe tener suficiente memoria disponible para soportar la sobrecarga del proceso. Si no tiene suficiente memoria disponible, se produce un error en el proceso. Por ello, es importante elegir un tipo de instancia de nodo que tenga suficiente memoria al crear el clúster de Redis OSS.

Proceso de escritura en segundo plano y uso de memoria

Cada vez que se invoca un proceso de escritura en segundo plano, Redis OSS bifurca su proceso (recuerde que Redis es de un solo subproceso). Una bifurcación conserva los datos en el disco en un archivo de instantáneas .rdb de Redis OSS. El resto de los servicios de la ramificación son todas operaciones de lectura y escritura. Para garantizar que la instantánea sea una point-in-time instantánea, todas las actualizaciones y adiciones de datos se escriben en un área de memoria disponible separada del área de datos.

Siempre que tenga suficiente memoria disponible para registrar todas las operaciones de escritura mientras se almacenan los datos en disco, no debería tener problemas de memoria insuficiente. Es probable que experimente problemas de memoria insuficiente si se da alguna de las siguientes condiciones:

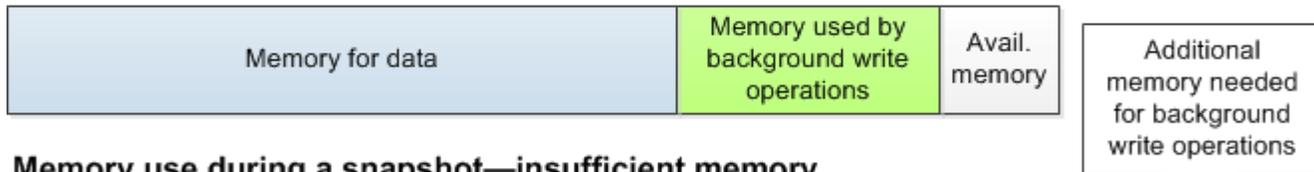
- Su aplicación realiza muchas operaciones de escritura, lo que requiere una gran cantidad de memoria disponible para aceptar los datos nuevos o actualizados.
- Tiene muy poca memoria disponible en la que agregar datos nuevos o datos.
- Dispone de un gran conjunto de datos que requiere mucho tiempo para guardar en disco, lo que exige un gran número de operaciones de escritura.

El siguiente diagrama ilustra el uso de memoria al ejecutar un proceso de escritura en segundo plano.

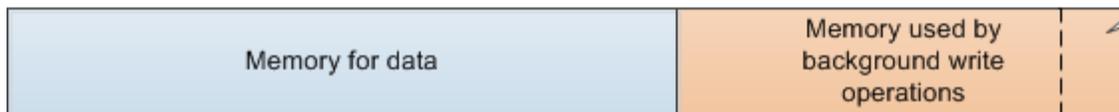
Memory use prior to a snapshot



Memory use during a snapshot—sufficient memory



Memory use during a snapshot—insufficient memory



Para obtener más información acerca del impacto en el rendimiento de hacer una copia de seguridad, consulte [El impacto en el rendimiento de las copias de seguridad de los clústeres de autodiseño](#).

Para obtener más información sobre cómo Redis OSS realiza las instantáneas, consulte <http://redis.io>.

Para obtener más información acerca de las regiones y zonas de disponibilidad, consulte [Elección de regiones y zonas de disponibilidad](#).

Evitar el agotamiento de la memoria al ejecutar una escritura en segundo plano

Siempre que se llame a un proceso de escritura en segundo plano, como BGSAVE o BGREWRITEAOF se invoque, para evitar que el proceso falle, debe tener más memoria disponible de la que consumirán las operaciones de escritura durante el proceso. El peor de los casos es que, durante la operación de escritura en segundo plano, se actualicen todos los registros del OSS de Redis y se agreguen algunos registros nuevos a la memoria caché. Por este motivo, le recomendamos que configure 50 (50 por ciento) `reserved-memory-percent` para las versiones de Redis OSS anteriores a la 2.8.22 o 25 (25 por ciento) para las versiones 2.8.22 de Redis OSS y posteriores.

El valor `maxmemory` indica la memoria disponible para sus datos y para la carga adicional operativa. Puesto que no puede modificar el parámetro `reserved-memory` en el grupo de parámetros predeterminado, debe crear un grupo de parámetros personalizado para el clúster. El valor predeterminado `reserved-memory` es 0, lo que permite a Redis OSS consumir toda la memoria máxima con datos, lo que podría dejar muy poca memoria para otros usos, como un proceso de

escritura en segundo plano. Para los valores `maxmemory` por tipo de instancia de nodo, consulte [Parámetros específicos del tipo de nodo de Redis OSS](#).

También puede utilizar el `reserved-memory` parámetro para reducir la cantidad de memoria que Redis OSS utiliza en la caja.

Para obtener más información sobre los parámetros específicos de Redis en, consulte. ElastiCache [Parámetros específicos de Redis OSS](#)

Para obtener información acerca de cómo crear y modificar grupos de parámetros, consulte [Creación de un grupo de parámetros](#) y [Modificación de un grupo de parámetros](#).

Cambio de tamaño de clústeres online

El cambio de las particiones implica agregar y eliminar particiones o nodos del clúster y redistribuir los espacios clave. Como resultado, varios aspectos influyen en la operación de cambio de las particiones, como la carga en el clúster, la utilización de memoria y el tamaño total de los datos. Para disfrutar de la mejor experiencia, recomendamos que siga las prácticas recomendadas de clúster global para una distribución uniforme del patrón de carga de trabajo. Además, recomendamos que siga los pasos que se detallan a continuación.

Antes de iniciar el cambio de las particiones, recomendamos lo siguiente:

- Probar la aplicación: si es posible, pruebe el comportamiento de la aplicación durante el cambio de las particiones en un entorno de ensayo.
- Recibir notificaciones anticipadas sobre problemas de escalado: el cambio de particiones es una operación que requiere mucho procesamiento. Por ello, recomendamos que mantenga el uso de la CPU por debajo del 80 por ciento en instancias de varios núcleos y en menos del 50 por ciento en instancias de un solo núcleo durante el cambio de particiones. Supervise las métricas ElastiCache (Redis OSS) e inicie la refragmentación antes de que su aplicación comience a observar los problemas de escalado. Las métricas de las que se puede realizar un seguimiento son `CPUUtilization`, `NetworkBytesIn`, `NetworkBytesOut`, `CurrConnections`, `NewConnections`, `FreeableMemory`, `SwapUsage` y `BytesUsedForCacheItems`.
- Comprobar que hay suficiente memoria libre disponible antes de la reducción horizontal: si va a realizar una reducción horizontal, asegúrese de que la memoria libre disponible en las particiones que se van a retener sea al menos 1,5 veces la memoria utilizada en las particiones que tiene previsto eliminar.
- Iniciar el cambio de las particiones durante las horas de menor actividad: esta práctica contribuye a reducir la latencia y el impacto en el rendimiento en el cliente durante la operación de cambio de

las particiones. También ayuda a completar el cambio de las particiones con mayor rapidez ya que se pueden usar más recursos para la redistribución de ranuras.

- Revisar el comportamiento de tiempo de espera de cliente: es posible que algunos clientes observen una latencia más alta durante el cambio de tamaño del clúster en línea. La configuración de la biblioteca de cliente con un tiempo de espera más alto puede ayudar a conceder al sistema tiempo para conectar incluso en condiciones de carga más altas en servidor. En algunos casos, es posible que abra un gran número de conexiones al servidor. En estos casos, considere la posibilidad de agregar retardo exponencial a la lógica de reconexión. Si lo hace, puede ayudar a evitar que llegue una ráfaga de conexiones nuevas al servidor al mismo tiempo.
- Cargue sus funciones en cada fragmento: al escalar el clúster, ElastiCache replicará automáticamente las funciones cargadas en uno de los nodos existentes (seleccionadas al azar) en los nuevos nodos. Si su clúster tiene Redis OSS 7.0 o superior y su aplicación usa [Redis OSS Functions](#), le recomendamos que cargue todas las funciones en todos los fragmentos antes de ampliarlos para que el clúster no acabe con funciones diferentes en fragmentos diferentes.

Después del cambio de las particiones, tenga en cuenta lo siguiente:

- La reducción horizontal se puede realizar parcialmente si no hay suficiente memoria disponible en las particiones de destino. Si se produce este resultado, revise la memoria disponible y, si es necesario, reintente la operación. Los datos de las particiones de destino no se eliminarán.
- Las ranuras con elementos grandes no se migran. En concreto, no se migran las ranuras con elementos que superen los 256 MB después de la serialización.
- No se admiten los comandos FLUSHALL y FLUSHDB en los scripts Lua dentro durante una operación de cambio de particiones. Antes de Redis OSS 6, el BRPOPLPUSH comando no era compatible si funcionaba en la ranura que se estaba migrando.

Minimización del tiempo de inactividad durante el mantenimiento

La configuración de modo de clúster presenta la mejor disponibilidad durante operaciones administradas y no administradas. Se recomienda utilizar un cliente compatible con modo de clúster que se conecte al punto de enlace de detección de clústeres. Para el modo de clúster deshabilitado, se recomienda utilizar el punto de enlace principal para todas las operaciones de escritura.

Para la actividad de lectura, las aplicaciones pueden conectarse también a cualquier nodo del clúster. A diferencia del punto de conexión principal, los puntos de conexión de nodo se resuelven en puntos de enlace específicos. Si realiza un cambio en su clúster, como añadir o eliminar una

réplica, debe actualizar los puntos de enlace del nodo en su aplicación. Por eso, si el modo de clúster está deshabilitado, le recomendamos que utilice el punto de conexión del lector para la actividad de lectura.

Si la conmutación por error automática está habilitada en el clúster, es posible que el nodo principal cambie. Por lo tanto, la aplicación debe confirmar el rol del nodo y actualizar todos los puntos de enlace de lectura. Esta operación ayuda a garantizar que no esté provocando una carga importante en el nodo principal. Con la conmutación por error automática desactivada, el rol del nodo no cambia. Sin embargo, el tiempo de inactividad en las operaciones administradas o no administradas es mayor en comparación con los clústeres con conmutación por error automática habilitada.

Evite dirigir las solicitudes de lectura a un único nodo de réplica de lectura, ya que su falta de disponibilidad podría provocar una interrupción de la lectura. Opte por la lectura desde el servidor principal o asegúrese de tener al menos dos réplicas de lectura para evitar cualquier interrupción de la lectura durante el mantenimiento.

Mejores prácticas de Redis OSS

Las siguientes son las mejores prácticas a la hora de utilizar Redis OSS para mejorar el rendimiento y la fiabilidad:

- Utilice configuraciones habilitadas para el modo de clúster: el modo de clúster habilitado permite que la caché se escale horizontalmente para lograr un mayor almacenamiento y rendimiento que una configuración en modo de clúster deshabilitado. ElastiCache La configuración sin servidor solo está disponible en una configuración habilitada para el modo de clúster.
- Utilice conexiones de larga duración: crear una nueva conexión es caro, y requiere tiempo y recursos de CPU de la memoria caché. Reutilice las conexiones siempre que sea posible (por ejemplo, mediante la agrupación de conexiones) para amortizar este coste a lo largo de muchos comandos.
- Lectura desde réplicas: si utiliza réplicas de lectura ElastiCache sin servidor o ha provisionado réplicas de lectura (clústeres de diseño propio), dirija las lecturas a las réplicas para lograr una mejor escalabilidad o una menor latencia. Las lecturas desde réplicas acaban siendo coherentes con la principal.

En un clúster de autodiseño, evite dirigir las solicitudes de lectura a una única réplica de lectura, ya que es posible que las lecturas no estén disponibles temporalmente si el nodo falla. Configure su cliente para que dirija las solicitudes de lectura al menos a dos réplicas de lectura, o dirija las lecturas a una única réplica y a la principal.

En los sistemas ElastiCache sin servidor, la lectura desde el puerto de réplica (6380) dirigirá las lecturas a la zona de disponibilidad local del cliente siempre que sea posible, lo que reducirá la latencia de recuperación. Cuando se produzca un error, volverá automáticamente a los demás nodos.

- Evitar los comandos costosos: evite ejecutar operaciones que hagan una utilización intensiva de procesamiento y de E/S, como los comandos KEYS y SMEMBERS. Recomendamos este enfoque porque estas operaciones aumentan la carga en el clúster e influyen en el rendimiento del clúster. En su lugar, utilice los comandos SCAN y SSCAN.
- Seguir las prácticas recomendadas de Lua: evite los scripts Lua de ejecución prolongada y siempre declare por adelantado las claves que utiliza en los scripts Lua. Recomendamos este enfoque para determinar que el script Lua no está utilizando comandos de ranura cruzada. Asegúrese de que las claves utilizadas en scripts Lua pertenezcan a la misma ranura.
- Utilice sharded pub/sub: cuando utilice Redis OSS para soportar cargas de trabajo de pub/sub con un alto rendimiento, le recomendamos que utilice [sharded](#) pub/sub (disponible con Redis OSS

7 o versiones posteriores). El pub/sub tradicional en los clústeres habilitados para el modo de clúster transmite mensajes a todos los nodos del clúster, lo que puede generar un nivel alto de `EngineCPUUtilization`. Tenga en cuenta que en los sistemas sin servidor, los comandos pub/sub tradicionales utilizan internamente comandos pub/sub fragmentados. ElastiCache

Estrategias de almacenamiento en caché

En el siguiente tema, encontrará estrategias para completar y mantener la caché.

Las estrategias que implemente para completar y mantener su caché dependen del tipo de datos que va a almacenar en su caché, así como de los patrones de acceso a dichos datos. Por ejemplo, probablemente no quiera utilizar la misma estrategia para una tabla de clasificación de los 10 mejores jugadores de un sitio de juegos y noticias de moda. En el resto de esta sección, se analizan las distintas estrategias comunes de mantenimiento de caché, junto con sus ventajas e inconvenientes.

Temas

- [Carga diferida](#)
- [Escritura indirecta](#)
- [Agregar TTL](#)
- [Temas relacionados de](#)

Carga diferida

Como su nombre indica, la carga diferida es una estrategia de almacenamiento en caché que carga los datos en la caché solo cuando es necesario. Funciona como se describe a continuación.

Amazon ElastiCache es un almacén de valores clave en memoria que se encuentra entre la aplicación y el almacén de datos (base de datos) al que accede. Siempre que la aplicación solicita datos, primero realiza la solicitud a la memoria caché. ElastiCache Si los datos existen en la memoria caché y están actualizados, los ElastiCache devuelve a la aplicación. Si los datos no existen en la caché o se han vencido, la aplicación solicita los datos del almacén de datos. A continuación, el almacén de datos devuelve los datos a su aplicación. Luego, su aplicación escribe los datos que recibió del almacén en la caché. De esta forma, se pueden recuperar más rápidamente la próxima vez que se los soliciten.

Un acierto de caché se produce cuando los datos se encuentran en la caché y no han vencido:

1. La aplicación solicita los datos a la caché.
2. La caché devuelve los datos a la aplicación.

Un error de caché se produce cuando los datos no se encuentran en la caché y han vencido:

1. La aplicación solicita los datos a la caché.
2. La caché no dispone de los datos solicitados, por lo que devuelve null.
3. La aplicación solicita los datos a la base de datos y los recibe.
4. La aplicación actualiza la caché con los datos nuevos.

Ventajas y desventajas de la carga diferida

Las ventajas de la carga diferida son las siguientes:

- Solo se almacenan en la caché los datos solicitados.

Dado que nunca se solicita la mayoría de los datos, la carga diferida evita completar la caché con datos que no se solicitan.

- Los errores de nodo no son fatales para su aplicación.

Cuando se produce un error en un nodo y se reemplaza por un nodo nuevo y vacío, la aplicación sigue funcionando, aunque con mayor latencia. Cuando se realizan solicitudes al nodo nuevo, cada error de caché da como resultado una consulta de la base de datos. Al mismo tiempo, la copia de datos se agrega a la caché para que las solicitudes posteriores se recuperen de la caché.

Las desventajas de la carga diferida son las siguientes:

- Existe una penalización de errores de caché. Cada error de caché genera tres acciones:
 1. Solicitud inicial de los datos a la caché
 2. Consulta de los datos en la base de datos
 3. Escritura de los datos en la caché

Estos errores pueden provocar un retraso significativo en la obtención de los datos en la aplicación.

- Datos obsoletos.

Si los datos se escriben en la caché solo cuando se produce un error de caché, los datos de la caché pueden quedar obsoletos. Este resultado se produce porque no hay actualizaciones en la caché cuando se cambian los datos en la base de datos. Para solucionar este problema, puede utilizar las estrategias [Escritura indirecta](#) y [Agregar TTL](#).

Ejemplo de pseudocódigo de carga diferida

El siguiente ejemplo es un pseudocódigo de lógica de carga diferida.

```
// *****  
// function that returns a customer's record.  
// Attempts to retrieve the record from the cache.  
// If it is retrieved, the record is returned to the application.  
// If the record is not retrieved from the cache, it is  
//   retrieved from the database,  
//   added to the cache, and  
//   returned to the application  
// *****  
get_customer(customer_id)  
  
    customer_record = cache.get(customer_id)  
    if (customer_record == null)  
  
        customer_record = db.query("SELECT * FROM Customers WHERE id = {0}",  
customer_id)  
        cache.set(customer_id, customer_record)  
  
    return customer_record
```

Para este ejemplo, el código de aplicación que obtiene los datos es el siguiente.

```
customer_record = get_customer(12345)
```

Escritura indirecta

La estrategia de escritura indirecta agrega o actualiza los datos de la caché siempre que se escriben datos en la base de datos.

Ventajas y desventajas de la escritura indirecta

Las ventajas de la escritura indirecta son las siguientes:

- Los datos de la caché nunca quedan obsoletos.

Dado que los datos de la caché se actualizan cada vez que se escriben en la base de datos, estos siempre se mantienen actualizados.

- Penalización de escritura frente a penalización de lectura.

Toda operación de escritura implica dos acciones:

1. Una operación de escritura en la caché
2. Una operación de escritura en la base de datos

Estas acciones añaden latencia al proceso. Dicho esto, los usuarios finales suelen ser más tolerantes con la latencia a la hora de actualizar datos que con la latencia a la hora de recuperar datos. Existe un sentido inherente que apunta a que las actualizaciones conllevan más trabajo y, por lo tanto, requieren mayor tiempo.

Las desventajas de la escritura indirecta son las siguientes:

- Pérdida de datos.

Si pone en marcha un nodo nuevo, ya sea debido a un error de nodo o a una operación de escalado horizontal, existen datos que se perderán. Estos datos siguen faltando hasta que se agregan o actualizan en la base de datos. Puede minimizar esto al implementar una [carga diferida](#) con escritura indirecta.

- Pérdida de caché.

La mayoría de los datos nunca se leen, lo cual es un desperdicio de recursos. Al [agregar un valor de periodo de vida \(TTL\)](#), puede minimizar el desperdicio de espacio.

Ejemplo de pseudocódigo de escritura indirecta

El siguiente ejemplo es un pseudocódigo de lógica de escritura indirecta.

```
// *****  
// function that saves a customer's record.  
// *****  
save_customer(customer_id, values)  
  
    customer_record = db.query("UPDATE Customers WHERE id = {0}", customer_id, values)  
    cache.set(customer_id, customer_record)  
    return success
```

Para este ejemplo, el código de aplicación que obtiene los datos es el siguiente.

```
save_customer(12345, {"address": "123 Main"})
```

Agregar TTL

La carga diferida da lugar a que los datos queden obsoletos, pero no falla con nodos vacíos. La escritura diferida mantiene los datos siempre actualizados, pero puede fallar con nodos vacíos y puede llenar la caché con datos superfluos. Al agregar un valor de periodo de vida (TTL) a cada escritura, puede tener las ventajas de cada estrategia. Al mismo tiempo, puede evitar en gran medida saturar la memoria caché con datos adicionales.

El periodo de vida (TTL) es un valor entero que especifica el número de segundos hasta que venza la clave. Redis OSS puede especificar segundos o milisegundos para este valor. Cuando una aplicación intenta leer una clave vencida, la trata como si no se encontrara la clave. La base de datos se consulta para la clave y se actualiza la caché. Este enfoque no garantiza que un valor no se encuentre obsoleto. Sin embargo, evita que los datos queden demasiado obsoletos y se actualizan los valores de la caché con frecuencia desde la base de datos.

Para obtener más información, consulte el [setcomando de Redis OSS](#).

Ejemplos de pseudocódigo de TTL

El siguiente ejemplo es un pseudocódigo de lógica de escritura indirecta con TTL.

```
// *****  
// function that saves a customer's record.  
// The TTL value of 300 means that the record expires  
//   300 seconds (5 minutes) after the set command  
//   and future reads will have to query the database.  
// *****  
save_customer(customer_id, values)  
  
    customer_record = db.query("UPDATE Customers WHERE id = {0}", customer_id, values)  
    cache.set(customer_id, customer_record, 300)  
  
    return success
```

El siguiente ejemplo es un pseudocódigo de lógica de carga diferida con TTL.

```
// *****  
// function that returns a customer's record.  
// Attempts to retrieve the record from the cache.
```

```
// If it is retrieved, the record is returned to the application.
// If the record is not retrieved from the cache, it is
//   retrieved from the database,
//   added to the cache, and
//   returned to the application.
// The TTL value of 300 means that the record expires
//   300 seconds (5 minutes) after the set command
//   and subsequent reads will have to query the database.
// *****
get_customer(customer_id)

    customer_record = cache.get(customer_id)

    if (customer_record != null)
        if (customer_record.TTL < 300)
            return customer_record          // return the record and exit function

    // do this only if the record did not exist in the cache OR
    //   the TTL was >= 300, i.e., the record in the cache had expired.
    customer_record = db.query("SELECT * FROM Customers WHERE id = {0}", customer_id)
    cache.set(customer_id, customer_record, 300) // update the cache
    return customer_record          // return the newly retrieved record and exit
function
```

Para este ejemplo, el código de aplicación que obtiene los datos es el siguiente.

```
save_customer(12345, {"address": "123 Main"})
```

```
customer_record = get_customer(12345)
```

Temas relacionados de

- [Almacenamiento de datos en memoria](#)
- [Elección de un motor y una versión](#)
- [Escalado ElastiCache \(Redis OSS\)](#)

Administración de un clúster de autodiseño

Esta sección contiene temas que lo ayudarán a gestionar sus clústeres de autodiseño.

Note

Estos temas no son aplicables en ElastiCache Serverless.

Temas

- [Clústeres de Auto Scaling ElastiCache \(Redis OSS\)](#)
- [Modificación del modo de clúster](#)
- [Replicación entre AWS regiones mediante almacenes de datos globales](#)
- [Alta disponibilidad a través de grupos de reproducción](#)
- [Administración del mantenimiento](#)
- [Configuración de los parámetros de motor mediante los grupos de parámetros](#)

Clústeres de Auto Scaling ElastiCache (Redis OSS)

Requisitos previos

ElastiCache (Redis OSS) Auto Scaling se limita a lo siguiente:

- Clústeres de Redis OSS (modo de clúster activado) que ejecutan el motor Redis OSS a partir de la versión 6.0
- Clústeres de datos por niveles (modo de clúster activado) que ejecutan el motor OSS de Redis a partir de la versión 7.0.7
- Tamaños de instancia: Large, XLarge, 2XLarge
- Familias de tipos de instancia: R7g, R6g, R6gd, R5, M7g, M6g, M5, C7gn
- Auto Scaling in ElastiCache (Redis OSS) no es compatible con los clústeres que se ejecutan en almacenes de datos globales, Outposts o Zonas Locales.

Gestión automática de la capacidad con Auto Scaling ElastiCache (Redis OSS)

ElastiCache El escalado automático (Redis OSS) es la capacidad de aumentar o disminuir automáticamente los fragmentos o réplicas deseados en su servicio ElastiCache (Redis OSS). ElastiCache (Redis OSS) aprovecha el servicio Application Auto Scaling para proporcionar esta funcionalidad. Para obtener más información, consulte [Auto Scaling de aplicaciones](#). Para utilizar

el escalado automático, debe definir y aplicar una política de escalado que utilice CloudWatch las métricas y los valores objetivo que usted asigne. ElastiCache (Redis OSS) el escalado automático utiliza la política para aumentar o disminuir la cantidad de instancias en respuesta a las cargas de trabajo reales.

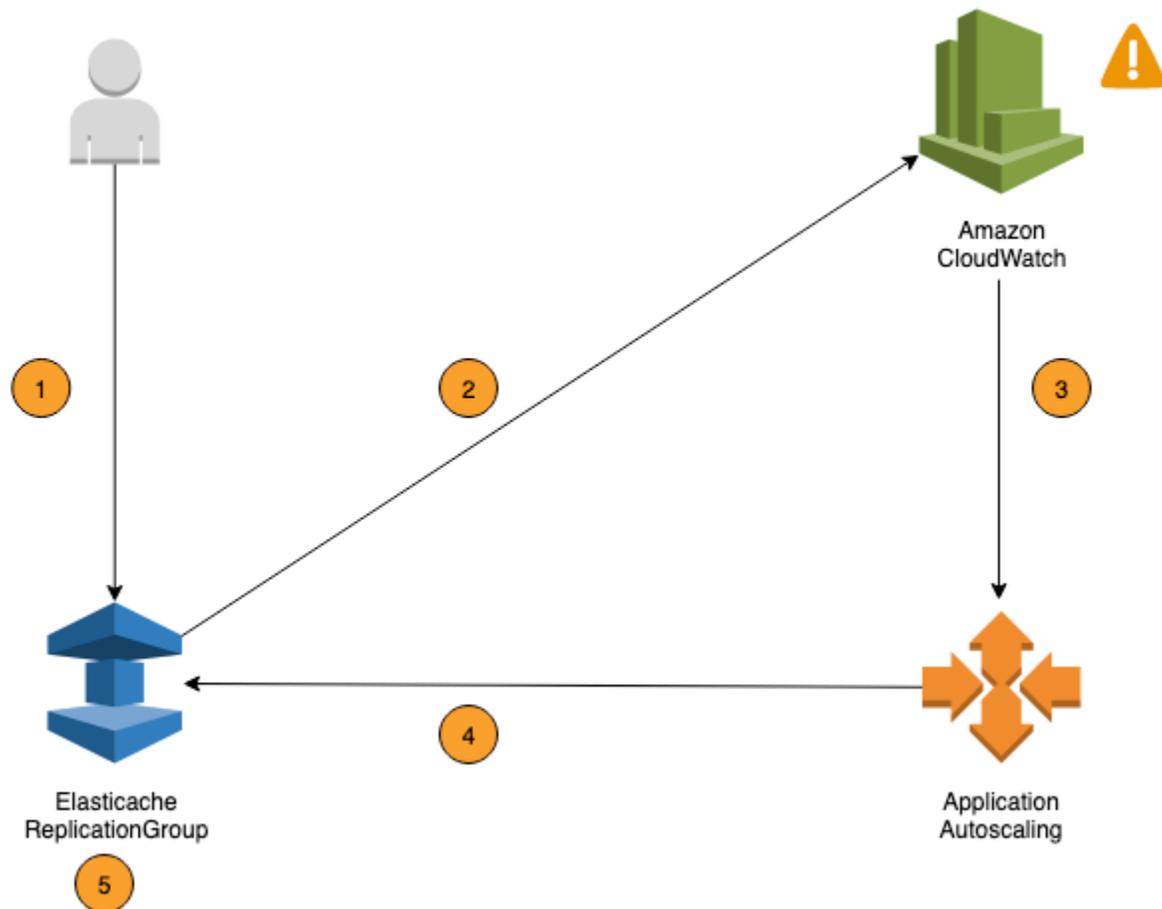
Puede utilizarla AWS Management Console para aplicar una política de escalado basada en una métrica predefinida. Se define una `predefined metric` en una enumeración, de manera que puede especificarla por el nombre en el código o utilizarla en la AWS Management Console. Las métricas personalizadas no se pueden seleccionar utilizando la AWS Management Console. Como alternativa, puede usar la API Application Auto Scaling AWS CLI o la API para aplicar una política de escalado basada en una métrica predefinida o personalizada.

ElastiCache (Redis OSS) admite el escalado de las siguientes dimensiones:

- **Particiones:** agregar/eliminar de forma automática particiones en el clúster de forma similar al cambio de particiones manual en línea. En este caso, el escalado automático ElastiCache (Redis OSS) activa el escalado en su nombre.
- **Réplicas:** agregar/eliminar de forma automática réplicas en el clúster de manera similar a las operaciones manuales de aumento/reducción de réplicas. ElastiCache (Redis OSS) el escalado automático agrega o elimina réplicas de manera uniforme en todos los fragmentos del clúster.

ElastiCache (Redis OSS) admite los siguientes tipos de políticas de escalado automático:

- [Políticas de escalado de seguimiento de destino](#): permite aumentar o reducir el número de particiones/réplicas que ejecuta el servicio en función del valor de destino para una métrica determinada. Se asemeja a los termostatos que se utilizan para mantener la temperatura del hogar. Se selecciona una temperatura y el termostato hace el resto.
- [Escalado automático programado para aplicaciones ElastiCache \(Redis OSS\)](#): aumente o disminuya la cantidad de fragmentos/réplicas que ejecuta su servicio en función de la fecha y la hora.



Los siguientes pasos resumen el proceso de escalado automático ElastiCache (Redis OSS), tal como se muestra en el diagrama anterior:

1. Usted crea una política de escalado automático ElastiCache (Redis OSS) para su grupo de replicación ElastiCache (Redis OSS).
2. ElastiCache (Redis OSS) el escalado automático crea un par de CloudWatch alarmas en su nombre. Cada par representa los límites superiores e inferiores de las métricas. Estas CloudWatch alarmas se activan cuando la utilización real del clúster se desvía de la utilización objetivo durante un período prolongado. Puede ver las alarmas en la consola.
3. Si el valor de la métrica configurada supera el uso objetivo (o cae por debajo del objetivo) durante un período de tiempo específico, CloudWatch activa una alarma que invoca el escalado automático ElastiCache (Redis OSS) para evaluar su política de escalado.
4. ElastiCache (Redis OSS) el escalado automático emite una solicitud de modificación para ajustar la capacidad del clúster.

5. ElastiCache (Redis OSS) procesa la solicitud de modificación y aumenta (o disminuye) de forma dinámica la capacidad de los fragmentos y réplicas del clúster para que se acerque al uso deseado.

Para entender cómo funciona Auto Scaling ElastiCache (Redis OSS), supongamos que tiene un clúster denominado `UsersCluster`. Al supervisar las CloudWatch métricas `UsersCluster`, se determinan los fragmentos máximos que necesita el clúster cuando el tráfico está en su punto máximo y los fragmentos mínimos cuando el tráfico está en su punto más bajo. También decide un valor objetivo para la utilización de la CPU para el clúster `UsersCluster`. ElastiCache (Redis OSS) el escalado automático utiliza su algoritmo de seguimiento de objetivos para garantizar que las particiones aprovisionadas se ajusten según sea necesario, de `UsersCluster` modo que la utilización se mantenga en el valor objetivo o cerca de él.

Note

El escalado puede llevar un tiempo considerable y requerirá recursos de clúster adicionales para que los fragmentos se reequilibren. ElastiCache (Redis OSS) Auto Scaling modifica la configuración de los recursos solo cuando la carga de trabajo real permanece elevada (o reducida) durante un período prolongado de varios minutos. El algoritmo de seguimiento de objetivos de escalado automático ElastiCache (Redis OSS) busca mantener la utilización del objetivo en el valor elegido o cerca de él a largo plazo.

Políticas de Auto Scaling

Una directiva de escalado tiene los siguientes componentes:

- Una métrica objetivo: la CloudWatch métrica que Auto Scaling ElastiCache (Redis OSS) usa para determinar cuándo y cuánto escalar.
- Capacidad mínima y máxima: el número mínimo y máximo de particiones o réplicas que se utilizarán para el escalado.

Important

Al crear la política de escalado automático, si la capacidad actual es superior a la capacidad máxima configurada, la aplicamos `MaxCapacity` durante la creación de la

política. Del mismo modo, si la capacidad actual es inferior a la capacidad mínima configurada, escalamos hacia fuera hasta la. MinCapacity

- Un periodo de recuperación: la cantidad de tiempo, en segundos, tras completarse una actividad de reducción o escalado horizontal antes de que pueda comenzar otra actividad de escalado horizontal.
- Un rol vinculado a un servicio: un rol de AWS Identity and Access Management (IAM) que está vinculado a un servicio específico. AWS Un rol vinculado a un servicio incluye todos los permisos que el servicio requiere para llamar a otros AWS servicios en tu nombre. ElastiCache (Redis OSS) Auto Scaling genera automáticamente este rol para usted. `AWSServiceRoleForApplicationAutoScaling_ElastiCacheRG`
- Habilitar o deshabilitar actividades de reducción horizontal: permite habilitar o deshabilitar actividades de reducción horizontal para una política.

Temas

- [Métrica de destino para Auto Scaling](#)
- [Capacidad mínima y máxima](#)
- [Periodo de recuperación](#)
- [Activar o desactivar actividades de escalado descendente](#)

Métrica de destino para Auto Scaling

En este tipo de política, se especifican una métrica predefinida o personalizada y un valor objetivo para la métrica en una configuración de política de escalado de seguimiento de objetivos. ElastiCache (Redis OSS) Auto Scaling crea y administra CloudWatch las alarmas que activan la política de escalado y calcula el ajuste de escalado en función de la métrica y el valor objetivo. La política de escalado agrega o elimina las particiones/réplicas en función de las necesidades para mantener la métrica en el valor objetivo especificado o en un valor próximo. Además de mantener la métrica próxima al valor de destino, la política de escalado de seguimiento de destino también se ajusta a las fluctuaciones de la métrica producidas por una carga de trabajo en constante cambio. Esta política también minimiza las fluctuaciones rápidas del número de particiones/réplicas disponibles de su clúster.

Por ejemplo, considere una política de escalado que utilice la métrica `ElastiCachePrimaryEngineCPUUtilization` media predefinida. Esta política puede mantener

la utilización de la CPU en el porcentaje de utilización especificado o en un valor próximo, como el 70 %.

Note

Para cada clúster, puede crear solo una política de Auto Scaling para cada métrica de destino.

Capacidad mínima y máxima

Particiones

Puede especificar la cantidad máxima de fragmentos a los que se puede escalar mediante el escalado automático ElastiCache (Redis OSS). Este valor debe ser igual o inferior a 250 con un mínimo de 1. También puede especificar la cantidad mínima de fragmentos que se administrarán mediante el escalado automático ElastiCache (Redis OSS). Este valor debe ser al menos 1, e igual o inferior al valor especificado para el máximo de 250 particiones.

Réplicas

Puede especificar el número máximo de réplicas que se administrarán mediante el autoescalado ElastiCache (Redis OSS). Este valor debe ser igual o inferior a 5. También puede especificar el número mínimo de réplicas que se administrarán mediante el autoescalado ElastiCache (Redis OSS). Este valor debe ser al menos 1, e igual o inferior al valor especificado para el máximo de 5 réplicas.

A fin de determinar el número mínimo y máximo de particiones/réplicas que necesite para el tráfico normal, pruebe la configuración de Auto Scaling con la tasa de tráfico esperada para su modelo.

Note

ElastiCache Las políticas de escalado automático (Redis OSS) aumentan la capacidad del clúster hasta que alcance el tamaño máximo definido o hasta que se apliquen los límites de servicio. Para solicitar un aumento del límite, consulte [AWS Service Limits](#) y elija el tipo de límite Nodes per cluster per instance type (Nodos por clúster por tipo de instancias).

⚠ Important

La reducción horizontal sucede cuando no hay tráfico. Si el tráfico de una variante se reduce a cero, ElastiCache (Redis OSS) se amplía automáticamente hasta el número mínimo de instancias especificado.

Periodo de recuperación

Puede ajustar la capacidad de respuesta de una política de escalado de seguimiento de destino al agregar periodos de recuperación que afecten al escalado de su clúster. Un periodo de recuperación bloquea solicitudes de escalado descendente o ascendente posteriores hasta que vence el periodo. Esto ralentiza la eliminación de fragmentos o réplicas en su clúster ElastiCache (Redis OSS) para las solicitudes de escalado interno y la creación de fragmentos/réplicas para las solicitudes de escalamiento horizontal. Puede especificar los siguientes periodos de recuperación:

- Una actividad de escalamiento horizontal reduce la cantidad de fragmentos o réplicas en su clúster (de Redis OSS). ElastiCache Un periodo de recuperación de escalado descendente especifica la cantidad de tiempo, en segundos, tras completarse una actividad de escalado descendente antes de que pueda comenzar otra actividad de escalado descendente.
- Una actividad de escalamiento horizontal aumenta la cantidad de fragmentos o réplicas en su clúster (de Redis OSS). ElastiCache Un periodo de recuperación de escalado ascendente especifica la cantidad de tiempo, en segundos, tras completarse una actividad de escalado ascendente antes de que pueda comenzar otra actividad de escalado ascendente.

Si no se especifica un periodo de recuperación de reducción o escalado horizontal, el valor predeterminado del escalado horizontal es de 600 segundos y de la reducción horizontal es de 900 segundos.

Activar o desactivar actividades de escalado descendente

Puede habilitar o deshabilitar actividades de escalado descendente para una política. La habilitación de actividades de reducción horizontal permite a la política de escalado eliminar particiones/réplicas. Al habilitarse actividades de escalado descendente, el periodo de recuperación de escalado descendente de la política de escalado se aplica a las actividades de escalado descendente. La deshabilitación de actividades de reducción horizontal impide a la política de escalado eliminar particiones/réplicas.

Note

Las actividades de escalado horizontal siempre están habilitadas para que la política de escalado pueda crear fragmentos/réplicas (Redis OSS) según sea necesario. ElastiCache

Se requieren permisos de IAM para Auto ElastiCache Scaling (Redis OSS)

ElastiCache (Redis OSS) Auto Scaling es posible gracias a una combinación de las API Auto Scaling ElastiCache (Redis OSS) y Application Auto Scaling. CloudWatch Los clústeres se crean y actualizan con ElastiCache (Redis OSS), las alarmas se crean con CloudWatch Application Auto Scaling y las políticas de escalado se crean con Application Auto Scaling. Además de los permisos de IAM estándar para crear y actualizar clústeres, el usuario de IAM que accede a la configuración de Auto Scaling ElastiCache (Redis OSS) debe tener los permisos adecuados para los servicios que admiten el escalado dinámico. Los usuarios de IAM deben contar con los permisos necesarios para utilizar las acciones que se muestran en la siguiente política de ejemplo:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "application-autoscaling:*",
        "elasticache:DescribeReplicationGroups",
        "elasticache:ModifyReplicationGroupShardConfiguration",
        "elasticache:IncreaseReplicaCount",
        "elasticache:DecreaseReplicaCount",
        "elasticache:DescribeCacheClusters",
        "elasticache:DescribeCacheParameters",
        "cloudwatch:DeleteAlarms",
        "cloudwatch:DescribeAlarmHistory",
        "cloudwatch:DescribeAlarms",
        "cloudwatch:DescribeAlarmsForMetric",
        "cloudwatch:GetMetricStatistics",
        "cloudwatch:ListMetrics",
        "cloudwatch:PutMetricAlarm",
        "cloudwatch:DisableAlarmActions",
        "cloudwatch:EnableAlarmActions",
        "iam:CreateServiceLinkedRole",
        "sns:CreateTopic",

```

```

        "sns:Subscribe",
        "sns:Get*",
        "sns:List*"
    ],
    "Resource": "arn:aws:iam::123456789012:role/autoscaling-roles-for-cluster"
}
]
}

```

Rol vinculado al servicio

El servicio de escalado automático ElastiCache (Redis OSS) también necesita permiso para describir sus clústeres y CloudWatch alarmas, y permisos para modificar su capacidad objetivo ElastiCache (Redis OSS) en su nombre. Si habilita Auto Scaling para su clúster ElastiCache (Redis OSS), se crea un rol vinculado a un servicio denominado `AWSServiceRoleForApplicationAutoScaling_ElastiCacheRG`. Esta función vinculada al servicio otorga a ElastiCache (Redis OSS) permiso de escalado automático para describir las alarmas de sus políticas, monitorear la capacidad actual de la flota y modificar la capacidad de la flota. La función vinculada al servicio es la función predeterminada para el escalado automático ElastiCache (Redis OSS). Para obtener más información, consulte [Funciones vinculadas a servicios para el escalado automático ElastiCache \(Redis OSS\) en la Guía del usuario de Application Auto Scaling](#).

Prácticas recomendadas de Auto Scaling

Antes de registrarse en Auto Scaling, recomendamos lo siguiente:

1. Utilizar solo una métrica de seguimiento: identifique si el clúster tiene CPU o cargas de trabajo intensivas de datos y utilice una métrica predefinida correspondiente para definir la política de escalado.
 - CPU del motor: `ElastiCachePrimaryEngineCPUUtilization` (dimensión de partición) o `ElastiCacheReplicaEngineCPUUtilization` (dimensión de réplica)
 - Uso de bases de datos: `ElastiCacheDatabaseCapacityUsageCountedForEvictPercentage` Esta política de escalado funciona mejor si la política `maxmemory` está configurada como `noeviction` en el clúster.

Le recomendamos que evite tener varias políticas por dimensión en el clúster. ElastiCache (Redis OSS) El escalado automático ampliará el objetivo escalable si hay alguna política de seguimiento

de objetivos preparada para ampliarla, pero solo se ampliará si todas las políticas de seguimiento de objetivos (con la parte de escalamiento interno habilitada) están listas para ampliarse. Si varias políticas indican al destino escalable que se realice una reducción o escalado horizontal al mismo tiempo, se escala en función de la política que proporciona la mayor capacidad para la reducción o el escalado horizontal.

2. Métricas personalizadas para el seguimiento de destinos: tenga cuidado al utilizar métricas personalizadas para el seguimiento de destinos, ya que el escalado automático es la característica más adecuada para el escalado horizontal en proporción a los cambios en las métricas elegidas para la política. Si esas métricas no cambian proporcionalmente a las acciones de escalado usadas para la creación de políticas, es posible que den lugar a acciones continuas de escalar horizontalmente o reducir horizontalmente que es posible que afecten a la disponibilidad o el costo.

En el caso de los clústeres de datos por niveles (tipos de instancias de la familia r6gd), evite utilizar métricas basadas en memoria para escalar.

3. Escalado programado: si identifica que su carga de trabajo es determinista (alcance alto/bajo en un momento específico), recomendamos utilizar el escalado programado y configurar su capacidad objetivo según la necesidad. El seguimiento de destino es el más adecuado para las cargas de trabajo no determinista y para que el clúster funcione con la métrica de destino requerida al escalar horizontalmente cuando necesite más recursos y reducir horizontalmente cuando necesite menos.
4. Deshabilitar la reducción horizontal: el Auto Scaling del seguimiento de destino es el más adecuado para clústeres con aumento/disminución gradual de las cargas de trabajo, ya que los picos o la disminución de las métricas pueden desencadenar oscilaciones consecutivas de reducción o escalado horizontal. Para evitar tales oscilaciones, puede comenzar con la reducción horizontal deshabilitada y luego siempre puede reducir horizontalmente de forma manual según sus necesidades.
5. Probar la aplicación: recomendamos que pruebe la aplicación con las cargas de trabajo mínimas y máximas estimadas a fin de determinar las particiones o réplicas mínimas y máximas necesarias para el clúster mientras crea políticas de escalado a fin de evitar problemas de disponibilidad. Auto Scaling puede escalar horizontalmente hasta el máximo y reducir horizontalmente en el umbral mínimo configurado para el destino.
6. Definición del valor objetivo: puede analizar CloudWatch las métricas correspondientes a la utilización del clúster durante un período de cuatro semanas para determinar el umbral del valor objetivo. Si aún no se encuentra seguro de qué valor elegir, se recomienda comenzar con el valor mínimo admitido de métrica predefinida.

7. AutoScaling on Target Tracking es ideal para clústeres con una distribución uniforme de las cargas de trabajo en todas las dimensiones de fragmentos o réplicas. Tener una distribución no uniforme puede conducir a:
- Escalado cuando no es necesario debido a un pico/caída de la carga de trabajo en algunas particiones o réplicas en almacenamiento activo.
 - Falta de escalado cuando sí es necesario debido a que la carga de trabajo media general se encuentra próxima al objetivo aunque tenga particiones o réplicas en el almacenamiento activo.

Note

Al ampliar el clúster, ElastiCache replicará automáticamente las funciones cargadas en uno de los nodos existentes (seleccionadas al azar) en los nuevos nodos. Si su clúster tiene Redis OSS 7.0 o superior y su aplicación usa [Redis OSS Functions](#), le recomendamos que cargue todas las funciones en todos los fragmentos antes de ampliarlos para que el clúster no acabe con funciones diferentes en los distintos fragmentos.

Tras registrarse, tenga en cuenta lo AutoScaling siguiente:

- Existen limitaciones en las configuraciones compatibles con Auto Scaling, por lo que recomendamos no cambiar la configuración de un grupo de reproducción que se haya registrado para Auto Scaling. A continuación se muestran algunos ejemplos:
 - Modificación manual del tipo de instancias a tipos no admitidos.
 - Asociación del grupo de reproducción a un almacén de datos global.
 - Cambio del parámetro `ReservedMemoryPercent`.
 - Aumento/disminución manual de particiones/réplicas más allá de la capacidad mínima y máxima configurada durante la creación de políticas.

Uso de Auto Scaling con particiones

A continuación, se proporcionan detalles sobre el seguimiento de objetivos y las políticas programadas y cómo aplicarlas mediante las API AWS Management Console AWS CLI y.

Temas

- [Políticas de escalado de seguimiento de destino](#)

- [Adición de una política de escalado](#)
- [Registro de un destino escalable](#)
- [Defining a scaling policy](#)
- [Edición de una política de escalado](#)
- [Eliminación de una política de escalado](#)
- [Uso de AWS CloudFormation para las políticas de Auto Scaling](#)
- [Escalado programado](#)

Políticas de escalado de seguimiento de destino

Las políticas de escalado de seguimiento de destino le permiten seleccionar una métrica y establecer un valor de destino. ElastiCache (Redis OSS) Auto Scaling crea y administra las CloudWatch alarmas que activan la política de escalado y calcula el ajuste de escalado en función de la métrica y el valor objetivo. La política de escalado agrega o elimina las particiones en función de las necesidades para mantener la métrica en el valor objetivo especificado o en un valor próximo. Además de mantener la métrica próxima al valor de destino, la política de escalado de seguimiento de destino también se ajusta a las fluctuaciones de la métrica producidas por patrones de carga fluctuante y minimiza las fluctuaciones rápidas de la capacidad de la flota.

Por ejemplo, considere una política de escalado que utilice la métrica `ElastiCachePrimaryEngineCPUUtilization` media predefinida con el valor objetivo configurado. Esta política puede mantener la utilización de la CPU en el valor objetivo especificado o en un valor próximo.

Métricas predefinidas

Una métrica predefinida es una estructura que hace referencia a un nombre, una dimensión y una estadística (`average`) específicos de una métrica determinada CloudWatch . La política de escalado automático define una de las siguientes métricas predefinidas para el clúster:

Nombre de métrica predefinido	CloudWatch Nombre de la métrica	CloudWatch Dimensión métrica	Tipos de instancia no aptos
ElastiCachePrimaryEngineCPU	EngineCPUUtilization	ReplicationGroupId	Ninguna

Nombre de métrica predefinido	CloudWatch Nombre de la métrica	CloudWatch Dimensión métrica	Tipos de instancia no aptos
Utilization		, Función = principal	
ElastiCacheDatabaseCapacityUsageCountedForEvictPercentage	DatabaseCapacityUsageCountedForEvictPercentage	Métricas del grupo de replicación de Redis OSS	Ninguna
ElastiCacheDatabaseMemoryUsageCountedForEvictPercentage	DatabaseMemoryUsageCountedForEvictPercentage	Métricas del grupo de replicación de Redis OSS	R6gd

Los tipos de instancias con niveles de datos no pueden utilizar `ElastiCacheDatabaseMemoryUsageCountedForEvictPercentage`, ya que estos tipos de instancias almacenan datos en la memoria y SSD. El caso de uso esperado para las instancias con niveles de datos es tener un uso de memoria del 100 por ciento y llenar el SSD según sea necesario.

Criterios de Auto Scaling para particiones

Cuando el servicio detecta que la métrica predefinida es igual o mayor que la configuración del objetivo, aumentará la capacidad de las particiones de forma automática. ElastiCache (Redis OSS) escala los fragmentos del clúster mediante un recuento igual al mayor de los dos números: variación porcentual con respecto al objetivo y 20 por ciento de los fragmentos actuales. En el caso de la ampliación, ElastiCache (Redis OSS) no se ampliará automáticamente a menos que el valor de la métrica general esté por debajo del 75 por ciento del objetivo definido.

Para un ejemplo de escalado horizontal, si tiene 50 particiones y

- Si su Target se infringe en un 30 por ciento, ElastiCache (Redis OSS) se amplía en un 30 por ciento, lo que da como resultado 65 fragmentos por clúster.
- Si su Target se infringe en un 10 por ciento, ElastiCache (Redis OSS) se amplía de forma predeterminada con un mínimo del 20 por ciento, lo que se traduce en 60 fragmentos por clúster.

Por ejemplo, si ha seleccionado un valor objetivo del 60 por ciento, ElastiCache (Redis OSS) no escalará automáticamente hasta que la métrica sea inferior o igual al 45 por ciento (un 25 por ciento inferior al 60 por ciento objetivo).

Consideraciones de Auto Scaling

Tenga en cuenta las siguientes consideraciones:

- En las políticas de escalado de seguimiento de destino, se presupone que el escalado ascendente se realiza cuando la métrica está por encima del valor objetivo. No puede utilizar una política de escalado de seguimiento de objetivos para escalar de forma horizontal cuando la métrica especificada esté por debajo del valor objetivo. ElastiCache (Redis OSS) escala los fragmentos con una desviación mínima del 20 por ciento con respecto al objetivo de los fragmentos existentes en el clúster.
- Las políticas de escalado de seguimiento de destino no realizan el escalado cuando la métrica especificada no tiene datos suficientes. No realiza la reducción horizontalmente porque la carencia de datos no se interpreta como una infrautilización de recursos.
- Es posible que haya diferencias entre el valor objetivo y los puntos de datos de la métrica real. Esto se debe a que Auto Scaling ElastiCache (Redis OSS) siempre actúa de forma conservadora al redondear hacia arriba o hacia abajo cuando determina la cantidad de capacidad que se debe añadir o eliminar. Con esto se evita que se agrega capacidad insuficiente o se elimine demasiada capacidad.
- Para garantizar la disponibilidad de la aplicación, el servicio se escala horizontalmente en proporción a la métrica tan rápido como puede, pero se reduce horizontalmente de forma más gradual.
- Puede tener varias políticas de escalado y seguimiento de objetivos para un clúster ElastiCache (Redis OSS), siempre que cada una de ellas utilice una métrica diferente. La intención de Auto Scaling ElastiCache (Redis OSS) es priorizar siempre la disponibilidad, por lo que su comportamiento varía según si las políticas de seguimiento de los objetivos están listas para ampliarse o ampliarse. Realizará un escalado horizontal del servicio si cualquiera de las políticas

de seguimiento de destino está lista para el escalado horizontal, pero solo realizará la reducción horizontal si todas las políticas de seguimiento de destino (que tienen la parte de reducción horizontal habilitada) están listas para la reducción horizontal.

- No edite ni elimine las CloudWatch alarmas que Auto Scaling ElastiCache (Redis OSS) administra para una política de escalado de seguimiento de objetivos. ElastiCache (Redis OSS) Auto Scaling elimina las alarmas automáticamente al eliminar la política de escalado.
- ElastiCache (Redis OSS) Auto Scaling no le impide modificar manualmente los fragmentos del clúster. Estos ajustes manuales no afectan a ninguna de CloudWatch las alarmas existentes asociadas a la política de escalado, pero pueden afectar a las métricas que podrían activar estas CloudWatch alarmas.
- Estas CloudWatch alarmas administradas por Auto Scaling se definen en la métrica de AVG en todos los fragmentos del clúster. Por lo tanto, tener particiones activas puede resultar en cualquiera de los siguientes escenarios:
 - escalando cuando no es necesario debido a la carga de algunos fragmentos calientes, lo que desencadena una alarma CloudWatch
 - no escalar cuando sí sea necesario debido a un AVG (Promedio) agregado en todas las particiones que causa que la alarma se interrumpa.
- ElastiCache (Redis OSS) se siguen aplicando los límites predeterminados de nodos por clúster. Por lo tanto, al optar por Auto Scaling y, si espera que los nodos máximos sean superiores al límite predeterminado, solicite un aumento del límite en [AWS Service Limits](#) y elija el tipo de límite Nodes per cluster per instance type (Nodos por clúster por tipo de instancias).
- Asegúrese de tener suficientes ENI (interfaces de red elásticas) disponibles en su VPC, que son necesarias durante el escalado horizontal. Para obtener más información, consulte [Interfaces de red elástica](#).
- Si no hay suficiente capacidad disponible en EC2, Auto Scaling ElastiCache (Redis OSS) no escalará y se retrasará hasta que la capacidad esté disponible.
- ElastiCache (Redis OSS) El Auto Scaling durante el escalado interno no eliminará los fragmentos con ranuras con un tamaño de elemento superior a 256 MB después de la serialización.
- Durante la reducción horizontal, no se eliminarán particiones si no se encuentra disponible la memoria suficiente en la configuración de particiones resultante.

Adición de una política de escalado

Puede añadir una política de escalado mediante AWS Management Console.

Para agregar una política de Auto Scaling a un ElastiCache clúster (Redis OSS)

1. Inicia sesión en la ElastiCache consola de Amazon AWS Management Console y ábrela en <https://console.aws.amazon.com/elasticache/>.
 2. En el panel de navegación, selecciona Redis OSS.
 3. Elija el clúster al que desea agregar una política (elija el nombre del clúster y no el botón a su izquierda).
 4. Elija la pestaña de Auto Scaling policies (Políticas de Auto Scaling).
 5. Elija add dynamic scaling (agregar escalado dinámico).
 6. En Policy name (Nombre de política), ingrese un nombre de política.
 7. En Scalable Dimension (Dimensión escalable), elija shards (particiones).
 8. Para la métrica de destino, elija una de las siguientes opciones:
 - Primary CPU Utilization (Utilización de CPU principal) para crear una política basada en la utilización promedio de la CPU.
 - Memory (Memoria) para crear una política basada en la memoria media de base de datos.
 - Capacidad para crear una política basada en el uso promedio de la capacidad de la base de datos. La métrica de capacidad incluye la utilización de memoria y SSD para las instancias con niveles de datos y la utilización de la memoria para todos los demás tipos de instancias.
 9. Para el valor objetivo, elija un valor mayor o igual a 35 y menor o igual a 70. El escalado automático mantendrá este valor para la métrica objetivo seleccionada en todos los ElastiCache fragmentos:
 - Utilización de la CPU principal: mantiene el valor objetivo de la métrica `EngineCPUUtilization` en los nodos principales.
 - Memoria: mantiene el valor objetivo de la métrica `DatabaseMemoryUsageCountedForEvictPercentage`
 - La capacidad mantiene el valor objetivo de la métrica `DatabaseCapacityUsageCountedForEvictPercentage`,
- Las particiones de clúster se agregan o quitan para mantener la métrica en un valor próximo al especificado.
10. (Opcional) La consola no admite periodos de recuperación de reducción o escalado horizontal. Usa el AWS CLI para modificar los valores de enfriamiento.

11. En Capacidad mínima, escriba el número mínimo de particiones que debe mantener la política de Auto Scaling ElastiCache (Redis OSS).
12. En Capacidad máxima, escriba el número máximo de particiones que debe mantener la política de Auto Scaling ElastiCache (Redis OSS). Este valor debe ser igual o inferior a 250.
13. Seleccione Crear.

Registro de un destino escalable

Antes de poder usar Auto Scaling con un clúster ElastiCache (Redis OSS), registre el clúster con el escalado automático ElastiCache (Redis OSS). Esto se hace para definir la dimensión de escalado y los límites que se van a aplicar a ese clúster. El escalado automático (Redis OSS) escala dinámicamente el clúster ElastiCache (Redis OSS) a lo largo de la dimensión `elasticache:replication-group:NodeGroups` escalable, que representa la cantidad de particiones del clúster.

Mediante el AWS CLI

Para registrar su clúster ElastiCache (Redis OSS), utilice el [register-scalable-target](#) comando con los siguientes parámetros:

- `--service-namespace` – establezca este valor en `elasticache`
- `--resource-id`— El identificador de recursos del clúster ElastiCache (Redis OSS). Para este parámetro, el tipo de recurso es `ReplicationGroup` y el identificador único es el nombre del clúster ElastiCache (Redis OSS), por ejemplo. `replication-group/myscalablecluster`
- `--scalable-dimension`: establezca este valor en `elasticache:replication-group:NodeGroups`.
- `--max-capacity` — El número máximo de fragmentos que se administrarán mediante el escalado automático ElastiCache (Redis OSS). Para obtener información sobre la relación entre `--min-capacity`, `--max-capacity` y el número de particiones del clúster, consulte [Capacidad mínima y máxima](#).
- `--min-capacity` — El número mínimo de fragmentos que se administrarán mediante el escalado automático ElastiCache (Redis OSS). Para obtener información sobre la relación entre `--min-capacity`, `--max-capacity` y el número de particiones del clúster, consulte [Capacidad mínima y máxima](#).

Example

En el siguiente ejemplo, se registra un clúster ElastiCache (Redis OSS) denominado `myscalablecluster`. El registro indica que el clúster debe escalarse de forma dinámica para tener de una a diez particiones.

Para Linux, macOS o Unix:

```
aws application-autoscaling register-scalable-target \  
  --service-namespace elasticache \  
  --resource-id replication-group/myscalablecluster \  
  --scalable-dimension elasticache:replication-group:NodeGroups \  
  --min-capacity 1 \  
  --max-capacity 10 \  

```

Para Windows:

```
aws application-autoscaling register-scalable-target ^  
  --service-namespace elasticache ^  
  --resource-id replication-group/myscalablecluster ^  
  --scalable-dimension elasticache:replication-group:NodeGroups ^  
  --min-capacity 1 ^  
  --max-capacity 10 ^  

```

Uso del API

Para registrar el ElastiCache clúster, utilice el [register-scalable-target](#) comando con los siguientes parámetros:

- `ServiceNamespace` — Establezca este valor en `elasticache`.
- `ResourceID`: el identificador de recursos del clúster. ElastiCache Para este parámetro, el tipo de recurso es `ReplicationGroup` y el identificador único es el nombre del clúster ElastiCache (Redis OSS), por ejemplo. `replication-group/myscalablecluster`
- `ScalableDimension` — Establezca este valor en. `elasticache:replication-group:NodeGroups`
- `MinCapacity` — El número mínimo de fragmentos que se administrarán mediante el escalado automático ElastiCache (Redis OSS). Para obtener información sobre la relación entre `--min-capacity`, `--max-capacity` y el número de réplicas del clúster, consulte [Capacidad mínima y máxima](#).

- **MaxCapacity** — El número máximo de fragmentos que se administrarán mediante el escalado automático ElastiCache (Redis OSS). Para obtener información sobre la relación entre `--min-capacity`, `--max-capacity` y el número de réplicas del clúster, consulte [Capacidad mínima y máxima](#).

Example

En el siguiente ejemplo, se registra un clúster ElastiCache (Redis OSS) denominado `myscalablecluster` con la API Application Auto Scaling. Este registro indica que el clúster debe escalarse de forma dinámica para tener de una a cinco réplicas.

```
POST / HTTP/1.1
Host: autoscaling.us-east-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 219
X-Amz-Target: AnyScaleFrontendService.RegisterScalableTarget
X-Amz-Date: 20160506T182145Z
User-Agent: aws-cli/1.10.23 Python/2.7.11 Darwin/15.4.0 botocore/1.4.8
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS
{
  "ServiceNamespace": "elasticache",
  "ResourceId": "replication-group/myscalablecluster",
  "ScalableDimension": "elasticache:replication-group:NodeGroups",
  "MinCapacity": 1,
  "MaxCapacity": 5
}
```

Defining a scaling policy

Una configuración de la política de escalado de seguimiento de destino está representada por un bloque JSON en el que se definen las métricas y los valores de destino. Puede guardar una configuración de la política de escalado como bloque JSON en un archivo de texto. Este archivo de texto se utiliza al invocar la API Application Auto Scaling AWS CLI o la API. Para obtener más información sobre la sintaxis de configuración de políticas, consulte [TargetTrackingScalingPolicyConfiguration](#) la Referencia de la API Application Auto Scaling.

Las siguientes opciones se encuentran disponibles para definir una configuración de la política de escalado de seguimiento de destino:

Temas

- [Uso de una métrica predefinida](#)
- [Uso de una métrica personalizada](#)
- [Uso de periodos de recuperación](#)
- [Desactivación de actividad de escalado descendente](#)
- [Aplicación de una política de escalado](#)

Uso de una métrica predefinida

Mediante el uso de métricas predefinidas, puede definir rápidamente una política de escalado de seguimiento de objetivos para un clúster ElastiCache (Redis OSS) que funcione con el seguimiento de objetivos en Auto Scaling ElastiCache (Redis OSS).

Actualmente, ElastiCache (Redis OSS) admite las siguientes métricas predefinidas en NodeGroup Auto ElastiCache Scaling (Redis OSS):

- `ElastiCachePrimaryEngineUtilización de la CPU`: el valor promedio de la `EngineCPUUtilization` métrica en todos los CloudWatch nodos principales del clúster ElastiCache (Redis OSS).
- `ElastiCacheDatabaseMemoryUsageCountedForEvictPercentage`— El valor promedio de la `DatabaseMemoryUsageCountedForEvictPercentage` métrica en todos los CloudWatch nodos principales del clúster ElastiCache (Redis OSS).
- `ElastiCacheDatabaseCapacityUsageCountedForEvictPercentage`— El valor medio de la `ElastiCacheDatabaseCapacityUsageCountedForEvictPercentage` métrica en CloudWatch todos los nodos principales del clúster ElastiCache (Redis OSS).

Para obtener más información sobre las métricas de `EngineCPUUtilization`, `DatabaseMemoryUsageCountedForEvictPercentage` y `DatabaseCapacityUsageCountedForEvictPercentage`, consulte [Monitoreo del uso con métricas de CloudWatch](#). Para usar una métrica predefinida en su política de escalado, puede crear una configuración de seguimiento de destino para su política de escalado. Esta configuración debe incluir un `PredefinedMetricSpecification` para la métrica predefinida y un `TargetValue` para el valor objetivo de esa métrica.

Example

El siguiente ejemplo describe una configuración de política típica para el escalado con seguimiento de objetivos para un clúster ElastiCache (Redis OSS). En esta configuración, la métrica

ElastiCachePrimaryEngineCPUUtilization predefinida se utiliza para ajustar el clúster ElastiCache (Redis OSS) en función de una utilización media de la CPU del 40 por ciento en todos los nodos principales del clúster.

```
{
  "TargetValue": 40.0,
  "PredefinedMetricSpecification":
  {
    "PredefinedMetricType": "ElastiCachePrimaryEngineCPUUtilization"
  }
}
```

Uso de una métrica personalizada

Mediante las métricas personalizadas, puede definir una política de escalado de seguimiento de destino que cumpla sus requisitos personalizados. Puede definir una métrica personalizada en función de cualquier ElastiCache métrica que cambie en proporción a la escala. No todas las ElastiCache métricas funcionan para el seguimiento de objetivos. La métrica debe ser una métrica de utilización válida y describir el nivel de actividad de una instancia. El valor de la métrica debe aumentar o reducirse en proporción al número de particiones del clúster. Este aumento o reducción proporcionales son necesarios para utilizar los datos de las métricas a fin de reducir o escalar horizontalmente de forma proporcional el número de particiones.

Example

En el siguiente ejemplo se describe una configuración de seguimiento de destino para una política de escalado. En esta configuración, una métrica personalizada ajusta un clúster ElastiCache (Redis OSS) en función de una utilización media de la CPU del 50 por ciento en todos los fragmentos de un clúster denominado `my-db-cluster`

```
{
  "TargetValue": 50,
  "CustomizedMetricSpecification":
  {
    "MetricName": "EngineCPUUtilization",
    "Namespace": "AWS/ElastiCache",
    "Dimensions": [
      {
        "Name": "RelicationGroup", "Value": "my-db-cluster"
      }
    ],
  },
}
```

```

        {
            "Name": "Role","Value": "PRIMARY"
        }
    ],
    "Statistic": "Average",
    "Unit": "Percent"
}
}

```

Uso de periodos de recuperación

Puede especificar un valor, en segundos, a fin de que `ScaleOutCooldown` agregue un periodo de recuperación para el escalado horizontal de su clúster. De forma similar, puede agregar un valor, en segundos, a fin de que `ScaleInCooldown` agregue un periodo de recuperación para el escalado vertical de su clúster. Para obtener más información, consulte la Referencia [TargetTrackingScalingPolicyConfiguration](#) de la API Application Auto Scaling.

En el siguiente ejemplo se describe una configuración de seguimiento de destino para una política de escalado. En esta configuración, la métrica `ElastiCachePrimaryEngineCPUUtilization` predefinida se utiliza para ajustar un clúster ElastiCache (Redis OSS) en función de una utilización media de la CPU del 40 por ciento en todos los nodos principales de ese clúster. La configuración proporciona un periodo de recuperación de escalado descendente de 10 minutos y un periodo de recuperación de escalado ascendente de 5 minutos.

```

{
    "TargetValue": 40.0,
    "PredefinedMetricSpecification":
    {
        "PredefinedMetricType": "ElastiCachePrimaryEngineCPUUtilization"
    },
    "ScaleInCooldown": 600,
    "ScaleOutCooldown": 300
}

```

Desactivación de actividad de escalado descendente

Puede evitar que la configuración de la política de escalado con seguimiento de objetivos se escale en su clúster ElastiCache (Redis OSS) deshabilitando la actividad de escalado interno. La deshabilitación de la actividad de escalado horizontal evita que la política de escalado elimine particiones, a la vez que permite a la política de escalado crearlas según sea necesario.

Puede especificar un valor booleano a fin de que `DisableScaleIn` habilite o deshabilite la actividad de reducción horizontal para su clúster. Para obtener más información, consulte la Referencia [TargetTrackingScalingPolicyConfiguration](#) de la API Application Auto Scaling.

En el siguiente ejemplo se describe una configuración de seguimiento de destino para una política de escalado. En esta configuración, la métrica `ElastiCachePrimaryEngineCPUUtilization` predefinida ajusta un clúster ElastiCache (Redis OSS) en función de una utilización media de la CPU del 40 por ciento en todos los nodos principales de ese clúster. La configuración deshabilita la actividad de escalado descendente para la política de escalado.

```
{
  "TargetValue": 40.0,
  "PredefinedMetricSpecification":
  {
    "PredefinedMetricType": "ElastiCachePrimaryEngineCPUUtilization"
  },
  "DisableScaleIn": true
}
```

Aplicación de una política de escalado

Tras registrar su clúster con el escalado automático ElastiCache (Redis OSS) y definir una política de escalado, aplique la política de escalado al clúster registrado. Para aplicar una política de escalado a un clúster ElastiCache (Redis OSS), puede utilizar la API Application Auto Scaling AWS CLI o la API Application Auto Scaling.

Aplicar una política de escalado mediante AWS CLI

Para aplicar una política de escalado a su clúster ElastiCache (Redis OSS), utilice el [put-scaling-policy](#) comando con los siguientes parámetros:

- `--policy-name`: el nombre de la política de escalado.
- `--policy-type`: establezca este valor en `TargetTrackingScaling`.
- `--resource-id`: el identificador de recursos del (Redis OSS ElastiCache). Para este parámetro, el tipo de recurso es `ReplicationGroup` y el identificador único es el nombre del clúster ElastiCache (Redis OSS), por ejemplo. `replication-group/myscalablecluster`
- `--service-namespace`: establezca este valor en `elasticache`.
- `--scalable-dimension`: establezca este valor en `elasticache:replication-group:NodeGroups`.

- --target-tracking-scaling-policy -configuración: configuración de la política de escalado de seguimiento de objetivos que se utilizará en el clúster ElastiCache (Redis OSS).

En el siguiente ejemplo, se aplica una política de escalado de seguimiento de objetivos denominada `myscalablepolicy` a un clúster ElastiCache (Redis OSS) denominado `myscalablecluster` con escalado automático ElastiCache (Redis OSS). Para ello, puede usar una configuración de la política guardada en un archivo denominado `config.json`.

Para Linux, macOS o Unix:

```
aws application-autoscaling put-scaling-policy \  
  --policy-name myscalablepolicy \  
  --policy-type TargetTrackingScaling \  
  --resource-id replication-group/myscalablecluster \  
  --service-namespace elasticache \  
  --scalable-dimension elasticache:replication-group:NodeGroups \  
  --target-tracking-scaling-policy-configuration file://config.json
```

Para Windows:

```
aws application-autoscaling put-scaling-policy ^  
  --policy-name myscalablepolicy ^  
  --policy-type TargetTrackingScaling ^  
  --resource-id replication-group/myscalablecluster ^  
  --service-namespace elasticache ^  
  --scalable-dimension elasticache:replication-group:NodeGroups ^  
  --target-tracking-scaling-policy-configuration file://config.json
```

Aplicación de una política de escalado mediante la API

Para aplicar una política de escalado a su clúster ElastiCache (Redis OSS), utilice el [PutScalingPolicy](#) AWS CLI comando con los siguientes parámetros:

- --policy-name: el nombre de la política de escalado.
- --resource-id: el identificador de recursos del (Redis OSS ElastiCache). Para este parámetro, el tipo de recurso es `ReplicationGroup` y el identificador único es el nombre del clúster ElastiCache (Redis OSS), por ejemplo. `replication-group/myscalablecluster`

- `--service-namespace`: establezca este valor en `elasticache`.
- `--scalable-dimension`: establezca este valor en `elasticache:replication-group:NodeGroups`.
- `--target-tracking-scaling-policy -configuración`: configuración de la política de escalado de seguimiento de objetivos que se utilizará en el clúster ElastiCache (Redis OSS).

En el siguiente ejemplo, se aplica una política de escalado de seguimiento de objetivos denominada `myscalablepolicy` a un clúster ElastiCache (Redis OSS) denominado `myscalablecluster` con escalado automático ElastiCache (Redis OSS). Puede usar una configuración de la política en función de la métrica predefinida `ElastiCachePrimaryEngineCPUUtilization`.

```
POST / HTTP/1.1
Host: autoscaling.us-east-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 219
X-Amz-Target: AnyScaleFrontendService.PutScalingPolicy
X-Amz-Date: 20160506T182145Z
User-Agent: aws-cli/1.10.23 Python/2.7.11 Darwin/15.4.0 botocore/1.4.8
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS
{
  "PolicyName": "myscalablepolicy",
  "ServiceNamespace": "elasticache",
  "ResourceId": "replication-group/myscalablecluster",
  "ScalableDimension": "elasticache:replication-group:NodeGroups",
  "PolicyType": "TargetTrackingScaling",
  "TargetTrackingScalingPolicyConfiguration": {
    "TargetValue": 40.0,
    "PredefinedMetricSpecification":
    {
      "PredefinedMetricType": "ElastiCachePrimaryEngineCPUUtilization"
    }
  }
}
```

Edición de una política de escalado

Puede editar una política de escalado mediante la AWS Management Console API Application Auto Scaling o la API Application Auto Scaling. AWS CLI

Edición de una política de escalado mediante la AWS Management Console

Para editar una política de Auto Scaling para un ElastiCache clúster (Redis OSS)

1. Inicia sesión en la ElastiCache consola de Amazon AWS Management Console y ábrela en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación, selecciona Redis OSS.
3. Elija el clúster al que desea agregar una política (elija el nombre del clúster y no el botón a su izquierda).
4. Elija la pestaña de Auto Scaling policies (Políticas de Auto Scaling).
5. En Scaling policies (Políticas de escalado), elija el botón a la izquierda de la política de Auto Scaling que desea cambiar y, a continuación, elija Modify (Modificar).
6. Realice los cambios solicitados en la política.
7. Elija Modificar.

Edición de una política de escalado mediante la API AWS CLI and

Puede usar la API Application Auto Scaling AWS CLI o la API para editar una política de escalado del mismo modo que aplica una política de escalado:

- Cuando utilice la AWS CLI, especifique el nombre de la política que desee editar en el `--policy-name` parámetro. Especifique nuevos valores para los parámetros que desea cambiar.
- Al usar la API de Auto Scaling de aplicaciones, especifique el nombre de la política que desea editar en el parámetro `PolicyName`. Especifique nuevos valores para los parámetros que desea cambiar.

Para obtener más información, consulte [Aplicación de una política de escalado](#).

Eliminación de una política de escalado

Puede eliminar una política de escalado mediante la AWS Management Console API Application Auto Scaling o la API Application Auto Scaling. AWS CLI

Eliminar una política de escalado mediante la AWS Management Console

Para eliminar una política de Auto Scaling para un ElastiCache clúster (Redis OSS)

1. Inicia sesión en la ElastiCache consola de Amazon AWS Management Console y ábrela en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación, selecciona Redis OSS.
3. Elija el clúster con la política de Auto Scaling que desea editar (elija el nombre del clúster, no el botón a su izquierda).
4. Elija la pestaña de Auto Scaling policies (Políticas de Auto Scaling).
5. En Scaling policies (Políticas de escalado), elija la política de Auto Scaling y, a continuación, elija Delete (Eliminar).

Eliminar una política de escalado mediante el AWS CLI

Para eliminar una política de escalado en su clúster ElastiCache (Redis OSS), utilice el [delete-scaling-policy](#) AWS CLI comando con los siguientes parámetros:

- `--policy-name`: el nombre de la política de escalado.
- `--resource-id`: el identificador de recursos del (Redis OSS ElastiCache). Para este parámetro, el tipo de recurso es `ReplicationGroup` y el identificador único es el nombre del clúster ElastiCache (Redis OSS), por ejemplo. `replication-group/myscalablecluster`
- `--service-namespace`: establezca este valor en `elasticache`.
- `--scalable-dimension`: establezca este valor en `elasticache:replication-group:NodeGroups`.

En el siguiente ejemplo, se elimina una política de escalado de seguimiento de objetivos denominada `myscalablepolicy` de un clúster ElastiCache (Redis OSS) denominado `myscalablecluster`

Para Linux, macOS o Unix:

```
aws application-autoscaling delete-scaling-policy \  
  --policy-name myscalablepolicy \  
  --resource-id replication-group/myscalablecluster \  
  --service-namespace elasticache \  
  --scalable-dimension elasticache:replication-group:NodeGroups
```

Para Windows:

```
aws application-autoscaling delete-scaling-policy ^
  --policy-name myscalablepolicy ^
  --resource-id replication-group/myscalablecluster ^
  --service-namespace elasticache ^
  --scalable-dimension elasticache:replication-group:NodeGroups
```

Eliminación de una política de escalado mediante la API

Para eliminar una política de escalado de su clúster ElastiCache (Redis OSS), utilice el [DeleteScalingPolicy](#) AWS CLI comando con los siguientes parámetros:

- `--policy-name`: el nombre de la política de escalado.
- `--resource-id`: el identificador de recursos del (Redis OSS ElastiCache). Para este parámetro, el tipo de recurso es `ReplicationGroup` y el identificador único es el nombre del clúster ElastiCache (Redis OSS), por ejemplo. `replication-group/myscalablecluster`
- `--service-namespace`: establezca este valor en `elasticache`.
- `--scalable-dimension`: establezca este valor en `elasticache:replication-group:NodeGroups`.

En el siguiente ejemplo, se elimina una política de escalado de seguimiento de objetivos denominada `myscalablepolicy` de un clúster ElastiCache (Redis OSS) denominado `myscalablecluster`

```
POST / HTTP/1.1
Host: autoscaling.us-east-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 219
X-Amz-Target: AnyScaleFrontendService.DeleteScalingPolicy
X-Amz-Date: 20160506T182145Z
User-Agent: aws-cli/1.10.23 Python/2.7.11 Darwin/15.4.0 botocore/1.4.8
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS
{
  "PolicyName": "myscalablepolicy",
  "ServiceNamespace": "elasticache",
  "ResourceId": "replication-group/myscalablecluster",
  "ScalableDimension": "elasticache:replication-group:NodeGroups"
}
```

Uso de AWS CloudFormation para las políticas de Auto Scaling

Este fragmento muestra cómo crear una política de seguimiento de destino y aplicarla a un recurso [AWS::ElastiCache::ReplicationGroup](#) mediante el recurso [AWS::ApplicationAutoScaling::ScalableTarget](#). Utiliza las funciones intrínsecas [Fn::Join](#) y [Ref](#) para construir la propiedad `ResourceId` con el nombre lógico del recurso `AWS::ElastiCache::ReplicationGroup` especificado en la misma plantilla.

```
ScalingTarget:
  Type: 'AWS::ApplicationAutoScaling::ScalableTarget'
  Properties:
    MaxCapacity: 3
    MinCapacity: 1
    ResourceId: !Sub replication-group/${logicalName}
    ScalableDimension: 'elasticache:replication-group:NodeGroups'
    ServiceNamespace: elasticache
    RoleARN: !Sub "arn:aws:iam::${AWS::AccountId}:role/aws-
service-role/elasticache.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_ElastiCacheRG"

ScalingPolicy:
  Type: "AWS::ApplicationAutoScaling::ScalingPolicy"
  Properties:
    ScalingTargetId: !Ref ScalingTarget
    ServiceNamespace: elasticache
    PolicyName: testpolicy
    PolicyType: TargetTrackingScaling
    ScalableDimension: 'elasticache:replication-group:NodeGroups'
    TargetTrackingScalingPolicyConfiguration:
      PredefinedMetricSpecification:
        PredefinedMetricType: ElastiCachePrimaryEngineCPUUtilization
      TargetValue: 40
```

Escalado programado

El escalado según una programación le permite escalar la aplicación en respuesta a los cambios previstos en la demanda. Para utilizar el escalado programado, debe crear acciones programadas que indiquen a ElastiCache (Redis OSS) que realice actividades de escalado en momentos específicos. Al crear una acción programada, se especifica un clúster existente ElastiCache (Redis OSS), el momento en que debe producirse la actividad de escalado, la capacidad mínima y la

capacidad máxima. Puede crear acciones programadas que realizan el escalado de forma puntual o periódica.

Solo puede crear una acción programada para los clústeres ElastiCache (Redis OSS) que ya existen. No puede crear una acción programada a la vez que crea un clúster.

A fin de obtener más información sobre la terminología de la creación, administración y eliminación de acciones programadas, consulte [Comandos de uso frecuente para la creación, administración y eliminación de acciones programadas](#).

Para crear de forma periódica:

1. Inicia sesión en la ElastiCache consola de Amazon AWS Management Console y ábrela en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación, selecciona Redis OSS.
3. Elija el clúster para el que desea agregar una política.
4. Elija Manage Auto Scaling policies (Administrar política de Auto Scaling) en el menú desplegable de Actions (Acciones).
5. Elija la pestaña de Auto Scaling policies (Políticas de Auto Scaling).
6. En la sección de Auto Scaling policies (Políticas de Auto Scaling), aparece el cuadro de diálogo de Add Scaling policy (Agregar política de escalado). Elija Scheduled scaling (Escalado programado).
7. En Policy name (Nombre de política), ingrese el nombre de la política.
8. En Scalable Dimension (Dimensión escalable), elija Shards (Particiones).
9. En Target Shards (Particiones de destino), elija el valor.
10. En Recurrence (Recurrencia), elija Recurring (Recurrente).
11. En Frequency (Frecuencia), elija el valor respectivo.
12. En Start Date (Fecha de inicio) y Start time (Hora de inicio), elija la hora a partir de la cual la política entrará en vigor.
13. Elija Add Policy (Agregar política).

Para crear una acción programada puntual:

1. Inicia sesión en la ElastiCache consola de Amazon AWS Management Console y ábrela en <https://console.aws.amazon.com/elasticache/>.

2. En el panel de navegación, selecciona Redis OSS.
3. Elija el clúster para el que desea agregar una política.
4. Elija Manage Auto Scaling policies (Administrar política de Auto Scaling) en el menú desplegable de Actions (Acciones).
5. Elija la pestaña de Auto Scaling policies (Políticas de Auto Scaling).
6. En la sección de Auto Scaling policies (Políticas de Auto Scaling), aparece el cuadro de diálogo de Add Scaling policy (Agregar política de escalado). Elija Scheduled scaling (Escalado programado).
7. En Policy name (Nombre de política), ingrese el nombre de la política.
8. En Scalable Dimension (Dimensión escalable), elija Shards (Particiones).
9. En Target Shards (Particiones de destino), elija el valor.
10. En Recurrence (Recurrencia), elija One Time (Una vez).
11. En Start Date (Fecha de inicio) y Start time (Hora de inicio), elija la hora a partir de la cual la política entrará en vigor.
12. En End Date (Fecha de finalización), elija la fecha hasta la que la política estará vigente.
13. Elija Add Policy (Agregar política).

Para eliminar una acción programada

1. Inicia sesión en la ElastiCache consola de Amazon AWS Management Console y ábrela en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación, selecciona Redis OSS.
3. Elija el clúster para el que desea agregar una política.
4. Elija Manage Auto Scaling policies (Administrar política de Auto Scaling) en el menú desplegable de Actions (Acciones).
5. Elija la pestaña de Auto Scaling policies (Políticas de Auto Scaling).
6. En la sección de Auto Scaling Policies (Políticas de Auto Scaling), elija la política de Auto Scaling y, a continuación, elija Delete (Eliminar) en el cuadro de diálogo de Actions (Acciones).

Para administrar el escalado programado mediante la AWS CLI

Utilice las siguientes API de escalado automático de aplicaciones:

- [put-scheduled-action](#)

- [describe-scheduled-actions](#)
- [delete-scheduled-action](#)

Utilice AWS CloudFormation para crear una acción programada

Este fragmento muestra cómo crear una política de seguimiento de objetivos y aplicarla a un [AWS::ElastiCache::ReplicationGroup](#) recurso mediante el [AWS::ApplicationAutoScaling::ScalableTarget](#) recurso. Utiliza las funciones intrínsecas [Fn::Join](#) y [Ref](#) para construir la propiedad `ResourceId` con el nombre lógico del recurso `AWS::ElastiCache::ReplicationGroup` especificado en la misma plantilla.

```
ScalingTarget:
  Type: 'AWS::ApplicationAutoScaling::ScalableTarget'
  Properties:
    MaxCapacity: 3
    MinCapacity: 1
    ResourceId: !Sub replication-group/${logicalName}
    ScalableDimension: 'elasticache:replication-group:NodeGroups'
    ServiceNamespace: elasticache
    RoleARN: !Sub "arn:aws:iam::${AWS::AccountId}:role/aws-
service-role/elasticache.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_ElastiCacheRG"
    ScheduledActions:
      - EndTime: '2020-12-31T12:00:00.000Z'
        ScalableTargetAction:
          MaxCapacity: '5'
          MinCapacity: '2'
          ScheduledActionName: First
          Schedule: 'cron(0 18 * * ? *)'
```

Uso de Auto Scaling con réplicas

A continuación se proporcionan detalles sobre el seguimiento de destinos y las políticas programadas, así como sobre cómo aplicarlas mediante la AWS Management Console, la AWS CLI y API.

Políticas de escalado de seguimiento de destino

Las políticas de escalado de seguimiento de destino le permiten seleccionar una métrica y establecer un valor de destino. ElastiCache (Redis OSS) AutoScaling crea y administra las CloudWatch alarmas

que activan la política de escalado y calcula el ajuste de escalado en función de la métrica y el valor objetivo. La política de escalado agrega o elimina las réplicas por igual en todas las particiones en función de las necesidades para mantener la métrica en el valor objetivo especificado o en un valor próximo. Además de mantener la métrica próxima al valor de destino, la política de escalado de seguimiento de destino también se ajusta a las fluctuaciones de la métrica producidas por patrones de carga fluctuante y minimiza las fluctuaciones rápidas de la capacidad de la flota.

Criterios de Auto Scaling para las réplicas

La política de Auto Scaling define la siguiente métrica predefinida para el clúster:

`ElastiCacheReplicaEngineCPUUtilization`: El umbral de utilización de la CPU del motor de AVG agregado en todas las réplicas que ElastiCache (Redis OSS) utiliza para activar una operación de autoscalamiento. Puede establecer el objetivo de utilización entre el 35 y el 70 %.

Cuando el servicio detecte que la `ElastiCacheReplicaEngineCPUUtilization` métrica es igual o superior a la configuración de `Target`, aumentará automáticamente las réplicas en todos los fragmentos. ElastiCache (Redis OSS) escala las réplicas del clúster en un cómputo igual al mayor de los dos números: variación porcentual con respecto a `Target` y una réplica. En el caso de la ampliación, ElastiCache (Redis OSS) no se ampliará automáticamente a menos que el valor de la métrica general esté por debajo del 75 por ciento del objetivo definido.

Para un ejemplo de escalado horizontal, si tiene 5 particiones y 1 réplica cada una:

Si su `Target` se infringe un 30 por ciento, ElastiCache (Redis OSS) amplía la escala en 1 réplica (máximo (0,3, valor predeterminado 1)) en todos los fragmentos, lo que da como resultado 5 fragmentos con 2 réplicas cada uno,

Por ejemplo, si ha seleccionado un valor objetivo del 60 por ciento, ElastiCache (Redis OSS) no escalará automáticamente hasta que la métrica sea inferior o igual al 45 por ciento (un 25 por ciento inferior al 60 por ciento objetivo).

Consideraciones de Auto Scaling

Tenga en cuenta las siguientes consideraciones:

- En las políticas de escalado de seguimiento de destino, se presupone que el escalado ascendente se realiza cuando la métrica está por encima del valor objetivo. No puede utilizar una política de escalado del seguimiento de objetivos para escalar de forma horizontal cuando la métrica especificada esté por debajo del valor objetivo. ElastiCache (Redis OSS) escala las réplicas como

máximo (% de desviación redondeado con respecto al objetivo, predeterminado 1) de las réplicas existentes en todos los fragmentos del clúster.

- Las políticas de escalado de seguimiento de destino no realizan el escalado cuando la métrica especificada no tiene datos suficientes. No realiza el escalado porque la carencia de datos no se interpreta como una infrutilización de recursos.
- Es posible que haya diferencias entre el valor objetivo y los puntos de datos de la métrica real. Esto se debe a que Auto Scaling ElastiCache (Redis OSS) siempre actúa de forma conservadora al redondear hacia arriba o hacia abajo cuando determina la cantidad de capacidad que se debe añadir o eliminar. Con esto se evita que se agregue capacidad insuficiente o se elimine demasiada capacidad.
- Para garantizar la disponibilidad de la aplicación, el servicio escalará horizontalmente en proporción a la métrica tan rápido como puede, pero se reduce horizontalmente de manera más gradual con un máximo de reducción horizontal de 1 réplica en las particiones del clúster.
- Puede tener varias políticas de escalado y seguimiento de objetivos para un clúster ElastiCache (Redis OSS), siempre que cada una de ellas utilice una métrica diferente. La intención de Auto Scaling ElastiCache (Redis OSS) es priorizar siempre la disponibilidad, por lo que su comportamiento varía según si las políticas de seguimiento de los objetivos están listas para ampliarse o ampliarse. Realizará un escalado horizontal del servicio si cualquiera de las políticas de seguimiento de destino está lista para el escalado horizontal, pero solo realizará la reducción horizontal si todas las políticas de seguimiento de destino (que tienen la parte de reducción horizontal habilitada) están listas para la reducción horizontal.
- No edite ni elimine las CloudWatch alarmas que Auto Scaling ElastiCache (Redis OSS) administra para una política de escalado de seguimiento de objetivos. ElastiCache (Redis OSS) Auto Scaling elimina las alarmas automáticamente al eliminar la política de escalado o al eliminar el clúster.
- ElastiCache (Redis OSS) Auto Scaling no le impide modificar manualmente las réplicas en los fragmentos. Estos ajustes manuales no afectan a ninguna de CloudWatch las alarmas existentes asociadas a la política de escalado, pero pueden afectar a las métricas que podrían activar estas alarmas. CloudWatch
- Estas CloudWatch alarmas administradas por Auto Scaling se definen en la métrica de AVG en todos los fragmentos del clúster. Por lo tanto, tener particiones activas puede resultar en cualquiera de los siguientes escenarios:
 - escalando cuando no es necesario debido a la carga de algunos fragmentos calientes, lo que desencadena una alarma CloudWatch
 - no escalar cuando sí sea necesario debido a un AVG (Promedio) agregado en todas las particiones que causa que la alarma se interrumpa.

- ElastiCache (Redis OSS) se siguen aplicando los límites predeterminados de nodos por clúster. Por lo tanto, al optar por Auto Scaling y, si espera que los nodos máximos sean superiores al límite predeterminado, solicite un aumento del límite en [AWS Service Limits](#) y elija el tipo de límite Nodes per cluster per instance type (Nodos por clúster por tipo de instancias).
- Asegúrese de tener suficientes ENI (interfaces de red elásticas) disponibles en su VPC, que son necesarias durante el escalado horizontal. Para obtener más información, consulte [Interfaces de red elástica](#).
- Si no hay suficiente capacidad disponible en EC2, Auto Scaling ElastiCache (Redis OSS) no se ampliará hasta que la capacidad esté disponible o si se modifica manualmente el clúster para adaptarlo a los tipos de instancias que tienen suficiente capacidad.
- ElastiCache (Redis OSS) Auto Scaling no admite el escalado de réplicas con un clúster que tenga ReservedMemoryPercent menos del 25 por ciento. Para obtener más información, consulte [Administración de memoria reservada](#).

Adición de una política de escalado

Puede añadir una política de escalado mediante AWS Management Console.

Añadir una política de escalado mediante el AWS Management Console

Para agregar una política de escalado automático a un ElastiCache (Redis OSS)

1. Inicia sesión en la ElastiCache consola de Amazon AWS Management Console y ábrela en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación, selecciona Redis OSS.
3. Elija el clúster al que desea agregar una política (elija el nombre del clúster y no el botón a su izquierda).
4. Elija la pestaña de Auto Scaling policies (Políticas de Auto Scaling).
5. Elija add dynamic scaling (agregar escalado dinámico).
6. En Scaling policies (Políticas de escalado), elija Add dynamic scaling (Agregar escalado dinámico).
7. En Policy name (Nombre de política), ingrese el nombre de la política.
8. En Scalable Dimension (Dimensión escalable), seleccione Replicas (Réplicas) en el cuadro de diálogo.

9. Para el valor objetivo, escriba el porcentaje medio de utilización de la CPU que desea mantener en ElastiCache las réplicas. El valor debe ser ≥ 35 y ≤ 70 . Las réplicas de clúster se agregan o quitan para mantener la métrica en un valor próximo al especificado.
10. (Opcional) La consola no admite periodos de recuperación de reducción o escalado horizontal. Utilice el AWS CLI para modificar los valores de enfriamiento.
11. En Capacidad mínima, escriba el número mínimo de réplicas que debe mantener la política de Auto Scaling ElastiCache (Redis OSS).
12. En Capacidad máxima, escriba el número máximo de réplicas que debe mantener la política de Auto Scaling ElastiCache (Redis OSS). Este valor debe ser ≥ 5 .
13. Seleccione Crear.

Registro de un destino escalable

Puede aplicar una política de escalado en función de una métrica predefinida o una personalizada. Para ello, puede utilizar la API Application Auto Scaling AWS CLI o la API. El primer paso es registrar su grupo de replicación ElastiCache (Redis OSS) con el escalado automático ElastiCache (Redis OSS).

Antes de poder usar el escalado automático ElastiCache (Redis OSS) con un clúster ElastiCache (Redis OSS), registre el clúster con el escalado automático ElastiCache (Redis OSS). Esto se hace para definir la dimensión de escalado y los límites que se van a aplicar a ese clúster. ElastiCache El escalado automático (Redis OSS) escala dinámicamente el clúster ElastiCache (Redis OSS) a lo largo de la dimensión `elasticache:replication-group:Replicas` escalable, que representa la cantidad de réplicas de clúster por fragmento.

Uso de la CLI

Para registrar el ElastiCache clúster, utilice el [register-scalable-target](#) comando con los siguientes parámetros:

- `--service-namespace`: establezca este valor en `elasticache`.
- `--resource-id`: el identificador de recursos del clúster. ElastiCache Para este parámetro, el tipo de recurso es `ReplicationGroup` y el identificador único es el nombre del clúster ElastiCache (Redis OSS), por ejemplo. `replication-group/myscalablecluster`
- `--scalable-dimension`: establezca este valor en `elasticache:replication-group:Replicas`.

- `--min-capacity`: la cantidad mínima de réplicas que se administrarán mediante el autoescalado ElastiCache (Redis OSS). Para obtener información sobre la relación entre `--min-capacity`, `--max-capacity` y el número de réplicas del clúster, consulte [Capacidad mínima y máxima](#).
- `--max-capacity`: la cantidad máxima de réplicas que se administrarán mediante el escalado automático ElastiCache (Redis OSS). Para obtener información sobre la relación entre `--min-capacity`, `--max-capacity` y el número de réplicas del clúster, consulte [Capacidad mínima y máxima](#).

Example

En el siguiente ejemplo, se registra un clúster ElastiCache (Redis OSS) denominado `myscalablecluster`. El registro indica que el clúster debe escalarse de forma dinámica para tener de una a cinco réplicas.

Para Linux, macOS o Unix:

```
aws application-autoscaling register-scalable-target \  
  --service-namespace elasticache \  
  --resource-id replication-group/myscalablecluster \  
  --scalable-dimension elasticache:replication-group:Replicas \  
  --min-capacity 1 \  
  --max-capacity 5 \  

```

Para Windows:

```
aws application-autoscaling register-scalable-target ^ \  
  --service-namespace elasticache ^ \  
  --resource-id replication-group/myscalablecluster ^ \  
  --scalable-dimension elasticache:replication-group:Replicas ^ \  
  --min-capacity 1 ^ \  
  --max-capacity 5 ^ \  

```

Uso del API

Para registrar el ElastiCache clúster, utilice el [register-scalable-target](#) comando con los siguientes parámetros:

- `ServiceNamespace` — Establezca este valor en `elasticache`.
- `ResourceID`: el identificador de recursos del clúster. ElastiCache Para este parámetro, el tipo de recurso es `ReplicationGroup` y el identificador único es el nombre del clúster ElastiCache (Redis OSS), por ejemplo, `replication-group/myscalablecluster`

- **ScalableDimension** — Establezca este valor en `elasticache:replication-group:Replicas`
- **MinCapacity** — El número mínimo de réplicas que se administrarán mediante el autoescalado ElastiCache (Redis OSS). Para obtener información sobre la relación entre `--min-capacity`, `--max-capacity` y el número de réplicas del clúster, consulte [Capacidad mínima y máxima](#).
- **MaxCapacity** — El número máximo de réplicas que se administrarán mediante el autoescalado ElastiCache (Redis OSS). Para obtener información sobre la relación entre `--min-capacity`, `--max-capacity` y el número de réplicas del clúster, consulte [Capacidad mínima y máxima](#).

Example

En el siguiente ejemplo, se registra un clúster ElastiCache (Redis OSS) denominado `myscalablecluster` con la API Application Auto Scaling. Este registro indica que el clúster debe escalarse de forma dinámica para tener de una a cinco réplicas.

```
POST / HTTP/1.1
Host: autoscaling.us-east-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 219
X-Amz-Target: AnyScaleFrontendService.RegisterScalableTarget
X-Amz-Date: 20160506T182145Z
User-Agent: aws-cli/1.10.23 Python/2.7.11 Darwin/15.4.0 botocore/1.4.8
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS
{
  "ServiceNamespace": "elasticache",
  "ResourceId": "replication-group/myscalablecluster",
  "ScalableDimension": "elasticache:replication-group:Replicas",
  "MinCapacity": 1,
  "MaxCapacity": 5
}
```

Defining a scaling policy

Una configuración de la política de escalado de seguimiento de destino está representada por un bloque JSON en el que se definen las métricas y los valores de destino. Puede guardar una configuración de la política de escalado como bloque JSON en un archivo de texto. Este archivo de texto se utiliza al invocar la API Application Auto Scaling AWS CLI o la API. Para obtener más información sobre la sintaxis de configuración de políticas, consulte [TargetTrackingScalingPolicyConfiguration](#) la Referencia de la API Application Auto Scaling.

Las siguientes opciones se encuentran disponibles para definir una configuración de la política de escalado de seguimiento de destino:

Temas

- [Uso de una métrica predefinida](#)
- [Edición de una política de escalado](#)
- [Eliminación de una política de escalado](#)
- [Uso de AWS CloudFormation para las políticas de Auto Scaling](#)
- [Escalado programado](#)

Uso de una métrica predefinida

Una configuración de la política de escalado de seguimiento de destino está representada por un bloque JSON en el que se definen las métricas y los valores de destino. Puede guardar una configuración de la política de escalado como bloque JSON en un archivo de texto. Este archivo de texto se utiliza al invocar la API Application Auto Scaling AWS CLI o la API. Para obtener más información sobre la sintaxis de configuración de políticas, consulte [TargetTrackingScalingPolicyConfiguration](#) la Referencia de la API Application Auto Scaling.

Las siguientes opciones se encuentran disponibles para definir una configuración de la política de escalado de seguimiento de destino:

Temas

- [Uso de una métrica predefinida](#)
- [Uso de una métrica personalizada](#)
- [Uso de periodos de recuperación](#)
- [Desactivación de actividad de escalado descendente](#)
- [Aplicar una política de escalado a un clúster ElastiCache \(Redis OSS\)](#)

Uso de una métrica predefinida

Mediante el uso de métricas predefinidas, puede definir rápidamente una política de escalado de seguimiento de objetivos para un clúster ElastiCache (Redis OSS) que funcione con el seguimiento de objetivos en Auto Scaling ElastiCache (Redis OSS). Actualmente, ElastiCache (Redis OSS) admite la siguiente métrica predefinida en ElastiCache Replicas Auto Scaling:

ElastiCacheReplicaEngineCPUUtilization— El valor medio de la métrica **EngineCPUUtilization** en todas las réplicas del clúster (CloudWatch Redis OSS). ElastiCache El valor promedio de la **EngineCPUUtilization** métrica en todas las réplicas del CloudWatch clúster (Redis OSS). ElastiCache Puede encontrar el valor agregado de la métrica en CloudWatch ElastiCache (Redis OSS) **ReplicationGroupId**, **Role** para las réplicas requeridas **ReplicationGroupId** y de rol.

Para usar una métrica predefinida en su política de escalado, puede crear una configuración de seguimiento de destino para su política de escalado. Esta configuración debe incluir **PredefinedMetricSpecification** para la métrica predefinida y **TargetValue** para el valor de destino de esa métrica.

Uso de una métrica personalizada

Mediante las métricas personalizadas, puede definir una política de escalado de seguimiento de destino que cumpla sus requisitos personalizados. Puede definir una métrica personalizada en función de cualquier métrica ElastiCache (Redis OSS) que cambie en proporción a la escala. No todas las métricas ElastiCache (Redis OSS) funcionan para el seguimiento de objetivos. La métrica debe ser una métrica de utilización válida y describir el nivel de actividad de una instancia. El valor de la métrica debe aumentar o reducirse en proporción al número de réplicas del clúster. Este aumento o reducción proporcionales son necesarios para utilizar los datos de las métricas a fin de ampliar o reducir proporcionalmente el número de réplicas.

Example

En el siguiente ejemplo se describe una configuración de seguimiento de destino para una política de escalado. En esta configuración, una métrica personalizada ajusta un clúster ElastiCache (Redis OSS) en función de una utilización media de la CPU del 50 por ciento en todas las réplicas de un clúster denominado `my-db-cluster`

```
{
  "TargetValue": 50,
  "CustomizedMetricSpecification":
  {
    "MetricName": "EngineCPUUtilization",
    "Namespace": "AWS/ElastiCache",
    "Dimensions": [
      {
        "Name": "ReplicationGroup",
        "Value": "my-db-cluster"
      },
      {
        "Name": "Role",
        "Value": "REPLICA"
      }
    ],
    "Statistic": "Average",
    "Unit": "Percent"
  }
}
```

```
}
```

Uso de periodos de recuperación

Puede especificar un valor, en segundos, a fin de que `ScaleOutCooldown` agregue un periodo de recuperación para el escalado horizontal de su clúster. De forma similar, puede agregar un valor, en segundos, a fin de que `ScaleInCooldown` agregue un periodo de recuperación para el escalado vertical de su clúster. Para obtener más información sobre `ScaleInCooldown` y `ScaleOutCooldown`, consulte la Referencia [TargetTrackingScalingPolicyConfiguration](#) de la API Application Auto Scaling. En el siguiente ejemplo se describe una configuración de seguimiento de destino para una política de escalado. En esta configuración, la métrica `ElastiCacheReplicaEngineCPUUtilization` predefinida se utiliza para ajustar un clúster ElastiCache (Redis OSS) en función de una utilización media de la CPU del 40 por ciento en todas las réplicas de ese clúster. La configuración proporciona un periodo de recuperación de escalado descendente de 10 minutos y un periodo de recuperación de escalado ascendente de 5 minutos.

```
{"TargetValue": 40.0,
  "PredefinedMetricSpecification":
  {"PredefinedMetricType": "ElastiCacheReplicaEngineCPUUtilization"},
  "ScaleInCooldown": 600,
  "ScaleOutCooldown": 300
}
```

Desactivación de actividad de escalado descendente

Puede evitar que la configuración de la política de escalado con seguimiento de objetivos se escale en su clúster ElastiCache (Redis OSS) deshabilitando la actividad de escalado interno. La deshabilitación de la actividad de escalado horizontal evita que la política de escalado elimine réplicas, a la vez que permite a la política de escalado agregarlas según sea necesario.

Puede especificar un valor booleano a fin de que `DisableScaleIn` habilite o deshabilite la actividad de reducción horizontal para su clúster. Para obtener más información al respecto `DisableScaleIn`, consulte [TargetTrackingScalingPolicyConfiguration](#) la Referencia de la API Application Auto Scaling.

Example

En el siguiente ejemplo se describe una configuración de seguimiento de destino para una política de escalado. En esta configuración, la métrica `ElastiCacheReplicaEngineCPUUtilization` predefinida ajusta un clúster ElastiCache (Redis OSS) en función de una utilización media de la CPU

del 40 por ciento en todas las réplicas de ese clúster. La configuración deshabilita la actividad de escalado descendente para la política de escalado.

```
{"TargetValue": 40.0,
  "PredefinedMetricSpecification":
  {"PredefinedMetricType": "ElastiCacheReplicaEngineCPUUtilization"
  },
  "DisableScaleIn": true
}
```

Aplicar una política de escalado a un clúster ElastiCache (Redis OSS)

Tras registrar su clúster con el escalado automático ElastiCache (Redis OSS) y definir una política de escalado, aplique la política de escalado al clúster registrado. Para aplicar una política de escalado a un clúster ElastiCache (Redis OSS), puede utilizar la API Application Auto Scaling AWS CLI o la API Application Auto Scaling.

Usando la AWS CLI

Para aplicar una política de escalado a su clúster ElastiCache (Redis OSS), utilice el [put-scaling-policy](#) comando con los siguientes parámetros:

- `--policy-name`: el nombre de la política de escalado.
- `--policy-type`: establezca este valor en `TargetTrackingScaling`.
- `--resource-id`: el identificador de recursos del clúster ElastiCache (Redis OSS). Para este parámetro, el tipo de recurso es `ReplicationGroup` y el identificador único es el nombre del clúster ElastiCache (Redis OSS), por ejemplo. `replication-group/myscalablecluster`
- `--service-namespace`: establezca este valor en `elasticache`.
- `--scalable-dimension`: establezca este valor en `elasticache:replication-group:Replicas`.
- `--target-tracking-scaling-policy-configuration`: configuración de la política de escalado de seguimiento de objetivos que se utilizará en el clúster ElastiCache (Redis OSS).

Example

En el siguiente ejemplo, se aplica una política de escalado de seguimiento de objetivos denominada `myscalablepolicy` a un clúster ElastiCache (Redis OSS) denominado `myscalablecluster` con escalado automático ElastiCache (Redis OSS). Para ello, puede usar una configuración de la política guardada en un archivo denominado `config.json`.

Para Linux, macOS o Unix:

```
aws application-autoscaling put-scaling-policy \  
  --policy-name myscalablepolicy \  
  --policy-type TargetTrackingScaling \  
  --resource-id replication-group/myscalablecluster \  
  --service-namespace elasticache \  
  --scalable-dimension elasticache:replication-group:Replicas \  
  --target-tracking-scaling-policy-configuration file://config.json
```

```
{"TargetValue": 40.0,  
  "PredefinedMetricSpecification":  
  {"PredefinedMetricType": "ElastiCacheReplicaEngineCPUUtilization"  
  },  
  "DisableScaleIn": true  
}
```

Para Windows:

```
aws application-autoscaling put-scaling-policy ^  
  --policy-name myscalablepolicy ^  
  --policy-type TargetTrackingScaling ^  
  --resource-id replication-group/myscalablecluster ^  
  --service-namespace elasticache ^  
  --scalable-dimension elasticache:replication-group:Replicas ^  
  --target-tracking-scaling-policy-configuration file://config.json
```

Uso del API

Para aplicar una política de escalado a su clúster ElastiCache (Redis OSS) con la API Application Auto Scaling, utilice la operación [PutScalingPolicy](#) Application Auto Scaling API con los siguientes parámetros:

- **PolicyName** — El nombre de la política de escalado.
- **PolicyType** — Establezca este valor en `TargetTrackingScaling`.
- **ResourceID**: el identificador de recursos del clúster (Redis ElastiCache OSS). Para este parámetro, el tipo de recurso es `ReplicationGroup` y el identificador único es el nombre del clúster ElastiCache (Redis OSS), por ejemplo. `replication-group/myscalablecluster`

- **ServiceNamespace** — Establezca este valor en elasticache.
- **ScalableDimension** — Defina este valor en. `elasticache:replication-group:Replicas`
- **TargetTrackingScalingPolicyConfiguration** — La configuración de la política de escalado de seguimiento de objetivos que se utilizará en el clúster ElastiCache (Redis OSS).

Example

En el siguiente ejemplo, se aplica una política de escalado de seguimiento de objetivos denominada `myscalablepolicy` a un clúster ElastiCache (Redis OSS) denominado `myscalablecluster` con escalado automático ElastiCache (Redis OSS). Puede usar una configuración de la política en función de la métrica predefinida `ElastiCacheReplicaEngineCPUUtilization`.

```
POST / HTTP/1.1
Host: autoscaling.us-east-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 219
X-Amz-Target: AnyScaleFrontendService.PutScalingPolicy
X-Amz-Date: 20160506T182145Z
User-Agent: aws-cli/1.10.23 Python/2.7.11 Darwin/15.4.0 botocore/1.4.8
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS
{
  "PolicyName": "myscalablepolicy",
  "ServiceNamespace": "elasticache",
  "ResourceId": "replication-group/myscalablecluster",
  "ScalableDimension": "elasticache:replication-group:Replicas",
  "PolicyType": "TargetTrackingScaling",
  "TargetTrackingScalingPolicyConfiguration": {
    "TargetValue": 40.0,
    "PredefinedMetricSpecification":
    {
      "PredefinedMetricType": "ElastiCacheReplicaEngineCPUUtilization"
    }
  }
}
```

Edición de una política de escalado

Puede editar una política de escalado mediante la AWS Management Console API Application Auto Scaling o la API Application Auto Scaling. AWS CLI

Edición de una política de escalado mediante AWS Management Console

Solo puede editar políticas con el tipo Predefined metrics (Métricas predefinidas) mediante la AWS Management Console

1. Inicia sesión en la ElastiCache consola de Amazon AWS Management Console y ábrela en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación, selecciona Redis OSS
3. Elija el clúster al que desea agregar una política (elijá el nombre del clúster y no el botón a su izquierda).
4. Elija la pestaña de Auto Scaling policies (Políticas de Auto Scaling).
5. En Scaling policies (Políticas de escalado), elija el botón a la izquierda de la política de Auto Scaling que desea cambiar y, a continuación, elija Modify (Modificar).
6. Realice los cambios solicitados en la política.
7. Elija Modificar.
8. Realice cambios en la política.
9. Elija Modificar.

Edición de una política de escalado mediante la API Application Auto Scaling AWS CLI o la API

Puede usar la API Application Auto Scaling AWS CLI o la API para editar una política de escalado del mismo modo que aplica una política de escalado:

- Al usar la API de Auto Scaling de aplicaciones, especifique el nombre de la política que desea editar en el parámetro `PolicyName`. Especifique nuevos valores para los parámetros que desea cambiar.

Para obtener más información, consulte [Aplicar una política de escalado a un clúster ElastiCache \(Redis OSS\)](#).

Eliminación de una política de escalado

Puede eliminar una política de escalado mediante la AWS Management Console API Application Auto Scaling AWS CLI o la API Application Auto Scaling.

Eliminar una política de escalado mediante la AWS Management Console

Solo puede editar políticas con el tipo Predefined metrics (Métricas predefinidas) mediante la AWS Management Console

1. Inicia sesión en la ElastiCache consola de Amazon AWS Management Console y ábrela en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación, selecciona Redis OSS
3. Elija el clúster con la política de Auto Scaling que desea eliminar.
4. Elija la pestaña de Auto Scaling policies (Políticas de Auto Scaling).
5. En Scaling policies (Políticas de escalado), elija la política de Auto Scaling y, a continuación, elija Delete (Eliminar).

Eliminar una política de escalado mediante la API Application Auto Scaling AWS CLI o la API

Puede usar la API Application Auto Scaling AWS CLI o la API para eliminar una política de escalado de un ElastiCache clúster.

CLI

Para eliminar una política de escalado de su clúster ElastiCache (Redis OSS), utilice el [delete-scaling-policy](#) comando con los siguientes parámetros:

- `--policy-name`: el nombre de la política de escalado.
- `--resource-id`: el identificador de recursos del clúster ElastiCache (Redis OSS). Para este parámetro, el tipo de recurso es `ReplicationGroup` y el identificador único es el nombre del ElastiCache clúster, por ejemplo. `replication-group/myscalablecluster`
- `--service-namespace`: establezca este valor en `elasticache`.
- `--scalable-dimension`: establezca este valor en `elasticache:replication-group:Replicas`.

Example

En el siguiente ejemplo, elimina una política de escalado de seguimiento de destino denominada `myscalablepolicy` de un clúster de ELC denominado `myscalablecluster`.

Para Linux, macOS o Unix:

```
aws application-autoscaling delete-scaling-policy \  
  --policy-name myscalablepolicy \  
  --resource-id replication-group/myscalablecluster \  
  --service-namespace elasticache \  
  --scalable-dimension elasticache:replication-group:Replicas \  

```

Para Windows:

```
aws application-autoscaling delete-scaling-policy ^  
  --policy-name myscalablepolicy ^  
  --resource-id replication-group/myscalablecluster ^  
  --service-namespace elasticache ^  
  --scalable-dimension elasticache:replication-group:Replicas ^  

```

API

Para eliminar una política de escalado de su clúster ElastiCache (Redis OSS), utilice la operación API [DeleteScalingPolicy](#) Application Auto Scaling con los siguientes parámetros:

- **PolicyName** — El nombre de la política de escalado.
- **ResourceID**: el identificador de recursos del clúster (Redis ElastiCache OSS). Para este parámetro, el tipo de recurso es `ReplicationGroup` y el identificador único es el nombre del ElastiCache clúster, por ejemplo. `replication-group/myscalablecluster`
- **ServiceNamespace** — Establezca este valor en `elasticache`.
- **ScalableDimension** — Defina este valor en `elasticache:replication-group:Replicas`

En el siguiente ejemplo, se elimina una política de escalado de seguimiento de objetivos denominada `myscalablepolicy` de un clúster ElastiCache (Redis OSS) denominado `myscalablecluster` con la API Application Auto Scaling.

```
POST / HTTP/1.1  
>>>>>> mainline  
Host: autoscaling.us-east-2.amazonaws.com  
Accept-Encoding: identity  
Content-Length: 219  
X-Amz-Target: AnyScaleFrontendService.DeleteScalingPolicy  
X-Amz-Date: 20160506T182145Z
```

```
User-Agent: aws-cli/1.10.23 Python/2.7.11 Darwin/15.4.0 botocore/1.4.8
Content-Type: application/x-amz-json-1.1
Authorization: AUTHPARAMS
{
  "PolicyName": "myscalablepolicy",
  "ServiceNamespace": "elasticache",
  "ResourceId": "replication-group/myscalablecluster",
  "ScalableDimension": "elasticache:replication-group:Replicas"
}
```

Uso de AWS CloudFormation para las políticas de Auto Scaling

Este fragmento muestra cómo crear una acción programada y aplicarla a un recurso [AWS::ElastiCache::ReplicationGroup](#) mediante el recurso [AWS::ApplicationAutoScaling::ScalableTarget](#). Utiliza las funciones intrínsecas [Fn::Join](#) y [Ref](#) para construir la propiedad `ResourceId` con el nombre lógico del recurso `AWS::ElastiCache::ReplicationGroup` especificado en la misma plantilla.

```
ScalingTarget:
  Type: 'AWS::ApplicationAutoScaling::ScalableTarget'
  Properties:
    MaxCapacity: 0
    MinCapacity: 0
    ResourceId: !Sub replication-group/${logicalName}
    ScalableDimension: 'elasticache:replication-group:Replicas'
    ServiceNamespace: elasticache
    RoleARN: !Sub "arn:aws:iam::${AWS::AccountId}:role/aws-
service-role/elasticache.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_ElastiCacheRG"

ScalingPolicy:
  Type: "AWS::ApplicationAutoScaling::ScalingPolicy"
  Properties:
    ScalingTargetId: !Ref ScalingTarget
    ServiceNamespace: elasticache
    PolicyName: testpolicy
    PolicyType: TargetTrackingScaling
    ScalableDimension: 'elasticache:replication-group:Replicas'
    TargetTrackingScalingPolicyConfiguration:
      PredefinedMetricSpecification:
        PredefinedMetricType: ElastiCacheReplicaEngineCPUUtilization
      TargetValue: 40
```

Escalado programado

El escalado según una programación le permite escalar la aplicación en respuesta a los cambios previstos en la demanda. Para utilizar el escalado programado, debe crear acciones programadas que indiquen a ElastiCache (Redis OSS) que realice actividades de escalado en momentos específicos. Al crear una acción programada, se especifica un clúster existente ElastiCache (Redis OSS), el momento en que debe producirse la actividad de escalado, la capacidad mínima y la capacidad máxima. Puede crear acciones programadas que realizan el escalado de forma puntual o periódica.

Solo puede crear una acción programada para los clústeres ElastiCache (Redis OSS) que ya existen. No puede crear una acción programada a la vez que crea un clúster.

A fin de obtener más información sobre la terminología de la creación, administración y eliminación de acciones programadas, consulte [Comandos de uso frecuente para la creación, administración y eliminación de acciones programadas](#).

Para crear una acción programada puntual:

Similar a la dimensión Shard (Partición). Consulte [Escalado programado](#) .

Para eliminar una acción programada

Similar a la dimensión Shard (Partición). Consulte [Escalado programado](#) .

Para administrar el escalado programado mediante la AWS CLI

Utilice las siguientes API de escalado automático de aplicaciones:

- [put-scheduled-action](#)
- [describe-scheduled-actions](#)
- [delete-scheduled-action](#)

Se usa AWS CloudFormation para crear políticas de Auto Scaling

Este fragmento muestra cómo crear una acción programada y aplicarla a un [AWS::ElastiCache::ReplicationGroup](#) recurso mediante el [AWS::ApplicationAutoScaling::ScalableTarget](#) recurso. Utiliza las funciones intrínsecas [Fn::Join](#) y [Ref](#) para construir la propiedad `ResourceId` con el nombre lógico del recurso `AWS::ElastiCache::ReplicationGroup` especificado en la misma plantilla.

```

ScalingTarget:
  Type: 'AWS::ApplicationAutoScaling::ScalableTarget'
  Properties:
    MaxCapacity: 0
    MinCapacity: 0
    ResourceId: !Sub replication-group/${logicalName}
    ScalableDimension: 'elasticache:replication-group:Replicas'
    ServiceNamespace: elasticache
    RoleARN: !Sub "arn:aws:iam::${AWS::AccountId}:role/aws-
service-role/elasticache.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_ElastiCacheRG"
  ScheduledActions:
    - EndTime: '2020-12-31T12:00:00.000Z'
      ScalableTargetAction:
        MaxCapacity: '5'
        MinCapacity: '2'
        ScheduledActionName: First
        Schedule: 'cron(0 18 * * ? *)'

```

Modificación del modo de clúster

Redis OSS es una base de datos distribuida en memoria que admite la fragmentación y la replicación. ElastiCache Los clústeres (Redis OSS) son la implementación distribuida de Redis OSS que permite dividir los datos en varios nodos de Redis OSS. Un clúster ElastiCache (Redis OSS) tiene dos modos de funcionamiento: el modo de clúster activado (CME) y el modo de clúster desactivado (CMD). En CME, Redis OSS funciona como una base de datos distribuida con varios fragmentos y nodos, mientras que en CMD, Redis OSS funciona como un nodo único.

Antes de migrar de CMD a CME, deben cumplirse las siguientes condiciones:

Important

La configuración del modo de clúster solo se puede cambiar del modo de clúster desactivado al modo de clúster habilitado. No es posible revertir esta configuración.

- Es posible que el clúster solo tenga claves en la base de datos 0.

- Las aplicaciones deben usar un cliente OSS de Redis que sea capaz de usar el protocolo Cluster y usar un punto final de configuración.
- La conmutación por error automática debe estar habilitada en el clúster con un mínimo de 1 réplica.
- La versión mínima del motor OSS de Redis necesaria para la migración es la 7.0.

Para migrar de CMD a CME, la configuración del modo de clúster solo se puede cambiar del modo de clúster desactivado al modo de clúster habilitado. Se trata de un procedimiento de dos pasos que garantiza la disponibilidad del clúster durante el proceso de migración.

Note

Debe proporcionar un grupo de parámetros con una configuración habilitada para el clúster, es decir, el parámetro habilitado para el clúster se establece como yes. Si utiliza un grupo de parámetros predeterminado, ElastiCache (Redis OSS) seleccionará automáticamente el grupo de parámetros predeterminado correspondiente con una configuración habilitada para clústeres. El valor del parámetro habilitado para el clúster se establece en no para un clúster de CMD. A medida que el clúster pasa al modo compatible, el valor del parámetro habilitado para el clúster se actualiza a yes como parte de la acción de modificación.

Para obtener más información, consulte [Configuración de los parámetros de motor mediante los grupos de parámetros](#).

1. Prepárese: cree un clúster de CME de prueba y asegúrese de que su pila esté lista para funcionar con él. ElastiCache (Redis OSS) no tiene forma de verificar si está preparado. Para obtener más información, consulte [Creación de un clúster](#).
2. Modifique la configuración del clúster CMD existente para que sea compatible con el modo de clúster: en este modo, se implementará un único fragmento y ElastiCache (Redis OSS) funcionará como un solo nodo, pero también como un clúster de un solo fragmento. El modo compatible significa que la aplicación cliente puede usar cualquiera de los protocolos para comunicarse con el clúster. En este modo, las aplicaciones deben reconfigurarse para empezar a utilizar el protocolo y el punto final de configuración del clúster OSS de Redis. Para cambiar el modo de clúster OSS de Redis a uno compatible con el modo de clúster, siga los pasos que se indican a continuación:

Note

En el modo compatible, no se permiten otras operaciones de modificación, como el escalado y la versión del motor, para el clúster. Además, los parámetros (excluidoscacheparameterGroupName) no se pueden modificar al definir el parámetro en modo de clúster en la solicitud. [ModifyReplicationGroup](#)

- a. Con el AWS Management Console, consulte [Modificación de un grupo de reproducción](#) y establezca el modo de clúster en Compatible
- b. Con la API, consulte [ModifyReplicationGroup](#) actualice el ClusterMode parámetro a compatible.
- c. Mediante AWS CLI, consulte [modify-replication-group](#) actualice el cluster-mode parámetro a compatible.

Tras cambiar el modo de clúster de Redis OSS a uno compatible con el modo de clúster, la [DescribeReplicationGroups](#) API devolverá el punto final de configuración del clúster ElastiCache (Redis OSS). El punto de conexión de configuración del clúster es un punto de conexión único que las aplicaciones pueden utilizar para conectarse al clúster. Para obtener más información, consulte [Búsqueda de puntos de conexión](#).

3. Modificar la configuración del clúster al modo de clúster habilitado: una vez que el modo de clúster esté configurado como compatible con el modo de clúster, el segundo paso consiste en modificar la configuración del clúster para que esté habilitado el modo de clúster. En este modo, se ejecuta una única partición y los clientes ahora pueden escalar los clústeres o modificar otras configuraciones de clústeres.

Para cambiar el modo de clúster a Habilitado, siga estos pasos:

Antes de empezar, asegúrese de que sus clientes de Redis OSS hayan migrado para utilizar el protocolo de clúster y de que el punto final de configuración del clúster no esté en uso.

- a. Con el AWS Management Console, consulte [Modificación de un grupo de reproducción](#) y configure el modo de clúster como Habilitado.
- b. Con la API, consulte [ModifyReplicationGroup](#) actualice el ClusterMode parámetro a enabled.

- c. Mediante AWS CLI, consulte [modify-replication-group](#) actualice el `cluster-mode` parámetro a `enabled`.

Tras cambiar el modo de clúster a activado, los puntos finales se configurarán según la especificación del clúster OSS de Redis. La [DescribeReplicationGroups](#) API devolverá el parámetro del modo de clúster `enabled` y los puntos finales del clúster que ahora están disponibles para que las aplicaciones los utilicen para conectarse al clúster.

Tenga en cuenta que los puntos de conexión del clúster cambiarán una vez que el modo de clúster se cambie a habilitado. Asegúrese de actualizar las aplicaciones con los nuevos puntos de conexión.

También puede elegir la opción de volver al modo de clúster desactivado (CMD) desde el modo de clúster compatible y conservar las configuraciones originales.

Modificar la configuración del clúster al modo de clúster desactivado desde el modo de clúster compatible

1. Con él AWS Management Console, consulte [Modificación de un grupo de reproducción](#) y establezca el modo de clúster en Desactivado
2. Con la API, consulte [ModifyReplicationGroup](#) actualice el `ClusterMode` parámetro a `disabled`.
3. Mediante AWS CLI, consulte [modify-replication-group](#) actualice el `cluster-mode` parámetro a `disabled`.

Tras cambiar el modo de clúster a deshabilitado, la [DescribeReplicationGroups](#) API devolverá el parámetro del modo de clúster `disabled`.

Replicación entre AWS regiones mediante almacenes de datos globales

Note

Actualmente, el almacén de datos global solo está disponible para clústeres autodiseñados.

Al utilizar la función Global Datastore for Redis OSS, puede trabajar con una replicación totalmente gestionada, rápida, fiable y segura en todas las regiones. AWS Con esta función, puede crear

clústeres de réplicas de lectura entre regiones para ElastiCache (Redis OSS) a fin de permitir las lecturas de baja latencia y la recuperación ante desastres en todas las regiones. AWS

En las secciones siguientes, puede encontrar una descripción de cómo trabajar con almacenes de datos globales.

Temas

- [Información general](#)
- [Requisitos previos y limitaciones](#)
- [Uso de almacenes de datos globales \(consola\)](#)
- [Uso de almacenes de datos globales \(CLI\)](#)

Información general

Cada almacén de datos global es un conjunto de uno o más clústeres que se replican entre sí.

Un almacén de datos global consta de lo siguiente:

- **Clúster principal (activo):** un clúster principal acepta escrituras que se replican en todos los clústeres dentro del almacén de datos global. Un clúster principal también acepta además solicitudes de lectura.
- **Clúster secundario (pasivo):** un clúster secundario solo acepta solicitudes de lectura y replica las actualizaciones de datos a partir de un clúster principal. Un clúster secundario debe estar en una AWS región diferente a la del clúster principal.

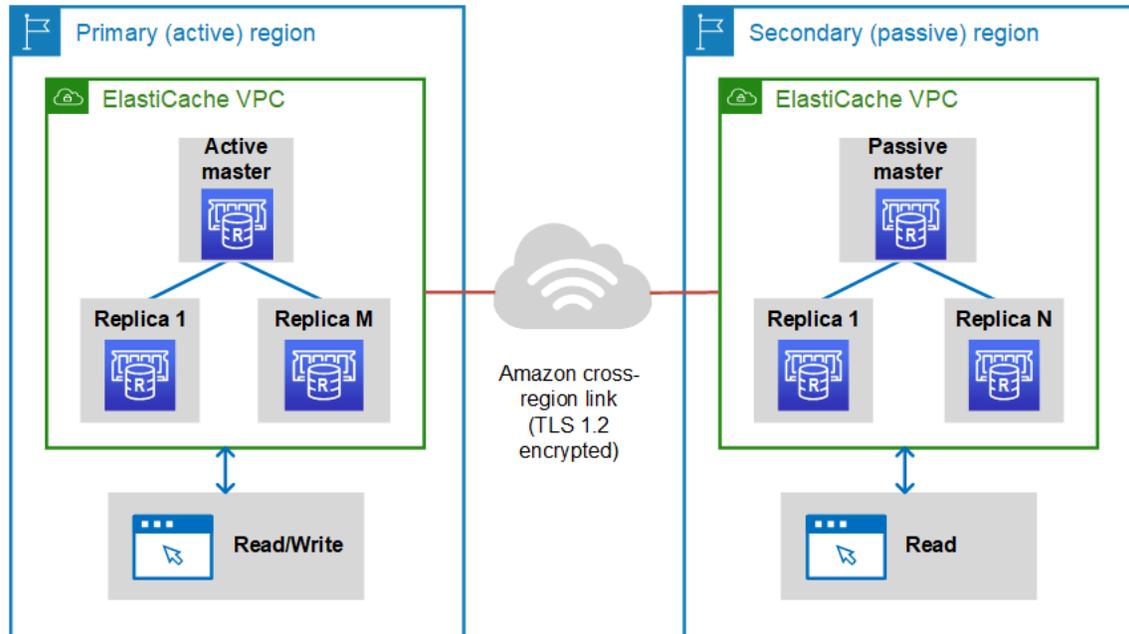
Al crear un almacén de datos global en ElastiCache, ElastiCache (Redis OSS) replica automáticamente los datos del clúster principal al clúster secundario. Elija la AWS región en la que se deben replicar los datos de Redis OSS y, a continuación, cree un clúster secundario en esa región. AWS ElastiCache a continuación, configura y gestiona la replicación automática y asíncrona de los datos entre los dos clústeres.

El uso de un almacén de datos global para Redis OSS ofrece las siguientes ventajas:

- **Rendimiento geolocal:** al configurar clústeres de réplicas remotas en AWS regiones adicionales y sincronizar los datos entre ellas, puede reducir la latencia del acceso a los datos en esa región. AWS Un almacén de datos global puede ayudar a aumentar la capacidad de respuesta de su aplicación al ofrecer lecturas geolocales de baja latencia en todas las regiones. AWS

- Recuperación de desastres: si el clúster principal de un almacén de datos global experimenta una reducción, puede promocionar un clúster secundario como su clúster principal nuevo. Puede hacerlo conectándose a cualquier AWS región que contenga un clúster secundario.

El siguiente diagrama muestra cómo pueden funcionar los almacenes de datos globales.



Requisitos previos y limitaciones

Antes de comenzar con almacenes de datos globales, tenga en cuenta lo siguiente:

- Los almacenes de datos globales son compatibles con las siguientes AWS regiones: Asia-Pacífico (Seúl, Tokio, Singapur, Sídney, Bombay y Osaka), Europa (Fráncfort, París, Londres, Irlanda y Estocolmo), EE.UU. Este (Norte de Virginia y Ohio), EE.UU. Oeste (Norte de California y Oregón), Sudamérica (São Paulo), AWS GovCloud (US-Oeste y US-Este), Región de Canadá (Central), China (Pekín y Ningxia)
- Todos los clústeres (principal y secundario) del almacén de datos global deben tener el mismo número de nodos primarios, tipo de nodo, versión del motor y número de particiones (en caso de que el modo clúster se encuentre habilitado). Cada clúster del almacén de datos global puede tener un número distinto de réplicas de lectura para acomodar el tráfico de lectura local a dicho clúster.

La reproducción debe estar habilitada si tiene previsto utilizar un clúster de nodo único existente.

- Los almacenes de datos globales no se admiten en instancias anteriores a m5 o r5.

- Puedes configurar la replicación de un clúster principal de una AWS región a un clúster secundario en un máximo de otras dos regiones. AWS

 Note

La excepción a esto son las regiones de China (Pekín) y China (Ningxia), donde la reproducción solo puede ocurrir entre las dos regiones.

- Solo puede trabajar con almacenes de datos globales en clústeres de VPC. Para obtener más información, consulte [Patrones de acceso para acceder a una ElastiCache caché en una Amazon VPC](#). Los almacenes de datos globales no se admiten cuando se utiliza EC2-Classic. Para obtener más información, consulte [EC2-Classic](#) en la Guía del usuario de Amazon EC2.

 Note

Por el momento, no puede utilizar almacenes de datos globales en [Uso de zonas locales con ElastiCache](#).

- ElastiCache no admite la conmutación automática por error de una región a otra. AWS Cuando sea necesario, puede promover manualmente un clúster secundario. Para ver un ejemplo, consulte [Promoción del clúster secundario a principal](#).
- Para iniciar desde datos existentes, utilice un clúster existente como principal para crear un almacén de datos global. No se admite la adición de un clúster existente como secundario. El proceso de agregar el clúster como secundario borra datos, lo que puede provocar la pérdida de datos.
- Las actualizaciones de parámetros se aplican a todos los clústeres cuando se modifica un grupo de parámetros local de un clúster que pertenece a un almacén de datos global.
- Puede escalar los clústeres regionales tanto vertical (aumento y disminución) como horizontalmente (escalado ascendente y descendente). Puede escalar los clústeres modificando el almacén de datos global. A continuación, todos los clústeres regionales del almacén de datos global se escalan sin interrupción. Para obtener más información, consulte [Escalado ElastiCache \(Redis OSS\)](#).
- [Los almacenes de datos globales admiten el cifrado en reposo, el cifrado en tránsito y la autenticación OSS de Redis.](#)
- Los almacenes de datos globales no admiten la versión 6 del Protocolo de Internet (IPv6).

- Los almacenes de datos globales admiten claves. AWS KMS Para obtener más información, consulte [Conceptos de servicios administrados de claves de AWS](#) en la Guía para desarrolladores de AWS Key Management Service .

Note

Los almacenes de datos globales admiten [mensajes de publicación/suscripción](#) con las siguientes condiciones:

- Para el modo de clúster deshabilitado, se admite por completo la publicación/suscripción. Los eventos publicados en el clúster principal de la AWS región principal se propagan a las regiones secundarias. AWS
- Para el modo de clúster habilitado, se aplica lo siguiente:
 - En el caso de los eventos publicados que no estén en un espacio de claves, solo los suscriptores de la misma AWS región los reciben.
 - En el caso de los eventos de keyspace publicados, los suscriptores de todas AWS las regiones reciben los eventos.

Uso de almacenes de datos globales (consola)

Para crear un almacén de datos global mediante la consola, siga este proceso de dos pasos:

1. Cree un clúster principal, ya sea mediante un clúster existente o creando un clúster nuevo. El motor debe ser Redis OSS 5.0.6 o posterior.
2. Agregue hasta dos clústeres secundarios en AWS regiones diferentes, utilizando de nuevo el motor Redis OSS 5.0.6 o posterior.

Los siguientes procedimientos le indican cómo crear un almacén de datos global para Redis OSS y realizar otras operaciones con la consola ElastiCache (Redis OSS).

Temas

- [Creación de un almacén de datos global mediante un clúster existente](#)
- [Creación de un almacén de datos global nuevo mediante un clúster principal nuevo](#)
- [Visualización de detalles del almacén de datos global](#)
- [Adición de una región a un almacén de datos global](#)

- [Modificación de un almacén de datos global](#)
- [Promoción del clúster secundario a principal](#)
- [Eliminación de una región de un almacén de datos global](#)
- [Eliminación de un almacén de datos global](#)

Creación de un almacén de datos global mediante un clúster existente

En esta situación, utilice un clúster existente que sirva como principal del nuevo almacén de datos global. A continuación, creará un clúster secundario de solo lectura en una región de AWS independiente. Este clúster secundario recibe actualizaciones automáticas y asincrónicas del clúster principal.

Important

El clúster existente debe usar el motor Redis OSS 5.0.6 o posterior.

Para crear un almacén de datos global mediante un clúster existente

1. [Inicie sesión AWS Management Console y abra la ElastiCache consola en https://console.aws.amazon.com/elasticache/.](https://console.aws.amazon.com/elasticache/)
2. En el panel de navegación, elija Almacenes de datos globales y, a continuación, elija Crear almacén de datos global.
3. En la página de configuración del clúster principal, haga lo siguiente:
 - En el campo de información del almacén de datos global, introduzca un nombre para el nuevo almacén de datos global.
 - (Opcional) Introduzca un valor Description (Descripción).
4. En Clúster regional, selecciona Usar clúster regional existente.
5. En Clúster existente, selecciona el clúster existente que desee usar.
6. Mantenga las siguientes opciones tal como están. Se rellenan previamente para que coincidan con la configuración del clúster principal, no se pueden cambiar.
 - Engine version (Versión del motor)
 - Tipo de nodo
 - Grupo de parámetros

Note

ElastiCache genera automáticamente un nuevo grupo de parámetros a partir de los valores del grupo de parámetros proporcionado y aplica el nuevo grupo de parámetros al clúster. Utilice este nuevo grupo de parámetros para modificar parámetros en un almacén de datos global. Cada grupo de parámetros generado automáticamente está asociado a un solo clúster y, por lo tanto, solo a un almacén de datos global.

- Número de fragmentos
- Encryption at rest (Cifrado en reposo): permite el cifrado de los datos almacenados en el disco. Para obtener más información, consulte [Cifrado en reposo](#).

Note

Para proporcionar una clave de cifrado diferente, seleccione la clave AWS KMS gestionada por el cliente y elija la clave. Para obtener más información, consulte [Uso de claves AWS KMS administradas por el cliente](#).

- Encryption in-transit (Cifrado en tránsito): permite el cifrado de datos del cable. Para obtener más información, consulte [Cifrado en tránsito](#). Para el motor OSS de Redis a partir de la versión 6.0, si habilita el cifrado en tránsito, se le pedirá que especifique una de las siguientes opciones de control de acceso:
 - No Access Control (Sin control de acceso): este es el valor predeterminado. Esto indica que no hay restricciones.
 - Lista de control de acceso (ACL) de grupos de usuarios: elija un grupo de usuarios con un conjunto definido de usuarios y permisos para las operaciones disponibles. Para obtener más información, consulte [Administración de grupos de usuarios con la consola y la CLI](#).
 - Usuario predeterminado de AUTH de Redis OSS: mecanismo de autenticación para el servidor OSS de Redis. Para obtener más información, consulte [Redis OSS AUTH](#).
7. (Opcional) Según sea necesario, actualice la configuración del clúster secundario restante. Estos se rellenan previamente con los mismos valores que el clúster principal, pero puede actualizarlos para que cumplan los requisitos específicos de ese clúster.
- Puerto
 - Número de réplicas

- Subnet group (Grupo de subredes)
 - Zonas de disponibilidad preferidas
 - Grupos de seguridad
 - Gestionado por el cliente (clave AWS KMS)
 - Token de autenticación OSS de Redis
 - Habilitar copias de seguridad automáticas
 - Backup retention period (Periodo de retención de copia de seguridad)
 - Backup target (Intervalo de copia de seguridad)
 - Periodo de mantenimiento
 - Tema para la notificación de SNS
8. Seleccione Crear. Al hacerlo, se establece el estado del almacén de datos global en Creating (Creando). El estado pasa a Modifying (Modificando) después de que el clúster principal esté asociado al almacén de datos global y el clúster secundario esté en estado Associating (Asociando).

Una vez que el clúster principal y los clústeres secundarios se asocian al almacén de datos global, el estado cambia a Available (Disponible). En este punto, tiene un clúster principal que acepta lecturas y escrituras y clústeres secundarios que aceptan lecturas replicadas desde el clúster principal.

La página OSS de Redis se actualiza para indicar si un clúster forma parte de un almacén de datos global, lo que incluye:

- Almacén de datos global: el nombre del almacén de datos global al que pertenece el clúster.
- Rol del almacén de datos global: el rol del clúster, ya sea principal o secundario.

Puede añadir hasta un clúster secundario adicional en una región diferente. AWS Para obtener más información, consulte [Adición de una región a un almacén de datos global](#).

Creación de un almacén de datos global nuevo mediante un clúster principal nuevo

Si decide crear un almacén de datos global con un clúster nuevo, utilice el siguiente procedimiento.

1. Inicie sesión AWS Management Console y abra la ElastiCache consola en <https://console.aws.amazon.com/elasticache/>.

2. En el panel de navegación, elija Almacenes de datos globales y, a continuación, elija Crear almacén de datos global.
3. En Primary cluster settings (Configuración de clúster principal), realice la siguiente operación:
 - a. Para Cluster mode (Modo clúster), elija Enabled (Habilitado) o Disabled (Desactivado).
 - b. Para obtener información sobre el almacén de datos global, introduzca un valor para el nombre. ElastiCache usa el sufijo para generar un nombre único para el almacén de datos global. Puede buscar el almacén de datos global utilizando el sufijo que especifique aquí.
 - c. (Opcional) Escriba un valor para Global Datastore Description (Descripción de almacén de datos global).
4. En Regional cluster (Clúster regional):
 - a. En Región, elija una región disponible. AWS
 - b. Elija Create new regional cluster (Crear un nuevo clúster regional) o Use existing regional cluster (Utilizar el clúster regional existente)
 - c. Si elige Create new regional cluster (Crear un nuevo clúster regional), en Cluster info (Información sobre clúster), introduzca un nombre y una descripción opcional del clúster.
 - d. En Location (Ubicación), le recomendamos que acepte la configuración predeterminada de Multi-AZ y Auto-failover (Conmutación por error automática).
5. En Cluster settings (Configuración del clúster)
 - a. Para Engine version (Versión del motor), elija una versión disponible, que sea 5.0.6 o una versión posterior.
 - b. En Port (Puerto), utilice el puerto predeterminado, 6379. Si tiene un motivo para utilizar un puerto diferente, escriba el número de puerto.
 - c. En Grupo de parámetros, elija un grupo de parámetros o cree uno nuevo. Los grupos de parámetros controlan los parámetros de tiempo de ejecución de su clúster. Para obtener más información acerca de los grupos de parámetros, consulte [Parámetros específicos de Redis OSS](#) y [Creación de un grupo de parámetros](#).

 Note

Cuando se selecciona un grupo de parámetros para definir los valores de configuración del motor, dicho grupo de parámetros se aplica a todos los clústeres del almacén de datos global. En la página Parameter Groups (Grupos de

parámetros), el atributo Global sí/no, indica si un grupo de parámetros forma parte de un almacén de datos global.

- d. Para Node type, elija la flecha hacia abajo (▼).

En el cuadro de diálogo Cambiar tipo de nodo, elija un valor para Familia de instancias para el tipo de nodo que desea. A continuación, elija el tipo de nodo que desea utilizar para este clúster y, a continuación, elija Guardar.

Para obtener más información, consulte [Elección del tamaño del nodo](#).

Si elige un tipo de nodo r6gd, la organización de datos en niveles se activará automáticamente. Para obtener más información, consulte [Organización de datos en niveles](#).

- e. Si va a crear un clúster de Redis OSS (modo de clúster desactivado):

Para Number of replicas (Número de réplicas), elija el número de réplicas que desea para este clúster.

- f. Si va a crear un clúster de Redis OSS (modo de clúster activado):

- i. En Número de fragmentos, elija el número de fragmentos (particiones/grupos de nodos) que desee para este clúster de Redis OSS (habilitado para el modo de clúster).

En algunas versiones de Redis OSS (modo de clúster activado), puede cambiar el número de fragmentos del clúster de forma dinámica:

- Redis OSS 3.2.10 y versiones posteriores: si su clúster ejecuta Redis OSS 3.2.10 o versiones posteriores, puede cambiar la cantidad de fragmentos del clúster de forma dinámica. Para obtener más información, consulte [Escalar clústeres en Redis OSS \(modo de clúster habilitado\)](#).
- Otras versiones de Redis OSS: si su clúster ejecuta una versión de Redis OSS anterior a la versión 3.2.10, existe otro enfoque. Para cambiar el número de particiones del clúster en este caso, cree un nuevo clúster con el nuevo número de particiones. Para obtener más información, consulte [Restauración desde una copia de seguridad a una nueva caché](#).

- ii. En Réplicas por partición, elija el número de nodos de réplica de lectura que desea en cada partición.

Existen las siguientes restricciones para Redis OSS (modo de clúster activado).

- Si tiene habilitado Multi-AZ, asegúrese de tener al menos una réplica por partición.
- El número de réplicas es el mismo para cada fragmento al crear el clúster utilizando la consola.
- El número de réplicas de lectura por fragmento es fijo y no puede modificarse. Si se da cuenta de que necesita más o menos réplicas por fragmento (API/CLI: grupo de nodos), debe crear un nuevo clúster con el nuevo número de réplicas. Para obtener más información, consulte [Inicialización de datos en un nuevo clúster de autodiseño con una copia de seguridad creada externamente](#).

6. En la configuración del grupo de subredes, elija la subred que desee aplicar a este clúster. ElastiCache proporciona un grupo de subredes IPv4 predeterminado o puede elegir crear uno nuevo. Para IPv6, debe crear un grupo de subred con un bloque de CIDR IPv6. Si elige dual stack (doble pila), debe seleccionar un tipo de IP de detección, IPv6 o IPv4.

Para obtener más información, consulte [Creación de una subred en la VPC](#).

7. En Availability zone placements Ubicaciones de las zonas de disponibilidad), tiene dos opciones:
 - Sin preferencia: ElastiCache elige la zona de disponibilidad.
 - Specify availability zones (Especificar zonas de disponibilidad): especifique la zona de disponibilidad para cada clúster.

Si decide especificar las zonas de disponibilidad, para cada clúster en cada fragmento, elija la zona de disponibilidad de la lista.

Para obtener más información, consulte [Elección de regiones y zonas de disponibilidad](#).

Slots and keyspaces: Custom distribution

Availability zone(s): Specify availability zones

	Slots/Keyspaces	Primary	Replica 1
NodeGroup 1	0-1234	us-east-1a	us-east-1a
NodeGroup 2		us-east-1b	us-east-1a
NodeGroup 3		us-east-1a	us-east-1a

Especificación de espacios de nombres y zonas de disponibilidad

8. Elija Next (Siguiente).

9. En la configuración avanzada de Redis OSS

- Para Security (Seguridad):
 - i. Para cifrar sus datos, tiene las siguientes opciones:
 - Encryption at rest (Cifrado en reposo): permite el cifrado de los datos almacenados en el disco. Para obtener más información, consulte [Cifrado en reposo](#).

Note

Tiene la opción de proporcionar una clave de cifrado diferente; para ello, seleccione la clave AWS KMS gestionada por el cliente y elija la clave. Para obtener más información, consulte [Uso de claves de AWS KMS administradas por el cliente](#).

- Encryption in-transit (Cifrado en tránsito): permite el cifrado de datos del cable. Para obtener más información, consulte [Cifrado en tránsito](#). Para la versión 6.0 y superior del motor OSS de Redis, si habilita el cifrado en tránsito, se le pedirá que especifique una de las siguientes opciones de control de acceso:

- No Access Control (Sin control de acceso): este es el valor predeterminado. Esto indica que no hay restricciones en el acceso de los usuarios al clúster.
- User Group Access Control List (Lista de control de acceso del grupo de usuarios): seleccione un grupo de usuarios con un conjunto definido de usuarios que pueden acceder al clúster. Para obtener más información, consulte [Administración de grupos de usuarios con la consola y la CLI](#).
- Usuario predeterminado de AUTH de Redis OSS: mecanismo de autenticación para el servidor OSS de Redis. Para obtener más información, consulte [Redis OSS AUTH](#).
- Redis OSS AUTH: un mecanismo de autenticación para el servidor Redis OSS. Para obtener más información, consulte [Redis OSS AUTH](#).

 Note

Para las versiones de Redis OSS desde la 3.2.6 en adelante, excepto la versión 3.2.10, la única opción es Redis OSS AUTH.

- ii. En Grupos de seguridad, elija los grupos de seguridad que desea para este clúster. Un grupo de seguridad actúa como un firewall para controlar el acceso de red al clúster. Puede utilizar el grupo de seguridad predeterminado para la VPC o crear uno nuevo.

Para obtener más información sobre los grupos de seguridad, consulte [Grupos de seguridad de su VPC](#) en la Guía del usuario de Amazon VPC.

10. En las copias de seguridad automáticas programadas regularmente, seleccione Enable automatic backups (Habilitar copias de seguridad automáticas) y, a continuación, escriba el número de días que desea conservar cada copia de seguridad automática antes de que se elimine automáticamente. Si no quiere tener copias de seguridad automáticas programadas regularmente, desactive la casilla de verificación Enable automatic backups (Habilitar copias de seguridad automáticas). En cualquier caso, siempre tendrá la opción de crear copias de seguridad manuales.

Para obtener más información sobre la copia de seguridad y la restauración de Redis OSS, consulte [Instantánea y restauración](#)

11. (Opcional) Especifique un periodo de mantenimiento. El período de mantenimiento es el tiempo, que suele durar una hora cada semana, en el que se ElastiCache programa el mantenimiento del sistema del clúster. Puede ElastiCache elegir el día y la hora del período de mantenimiento

(sin preferencia) o puede elegir el día, la hora y la duración usted mismo (especifique el período de mantenimiento). Si elige Specify maintenance window, elija Start day, Start time y Duration (en horas) de las listas para el periodo de mantenimiento. Todas las horas se indican en UCT.

Para obtener más información, consulte [Administración del mantenimiento](#).

12. (Opcional) Para Logs (Registros):

- En Log format (Formato de registro), elija Text o JSON.
- En Tipo de destino, elija CloudWatch Logs o Kinesis Firehose.
- En Destino del registro, elija Crear nuevo e introduzca el nombre del grupo de CloudWatch registros de Logs o el nombre de la transmisión de Firehose, o bien elija Seleccionar existente y, a continuación, elija el nombre del grupo de CloudWatch registros de Logs o el nombre de la transmisión de Firehose,

13. En el caso de las etiquetas, para ayudarte a gestionar tus clústeres y otros ElastiCache recursos, puedes asignar tus propios metadatos a cada recurso en forma de etiquetas. Para obtener más información, consulte [Etiquetado de los recursos de ElastiCache](#).

14. Revise todas las entradas y opciones y, a continuación, realice todos los cambios necesarios. Cuando esté preparado para continuar, seleccione Next (Siguiente).

15. Una vez que haya configurado el clúster en los pasos anteriores, configure los detalles de clúster secundario.

16. En Clúster regional, selecciona la AWS región en la que se encuentra el clúster.

17. En Cluster info (Información sobre clúster), introduzca un nombre y una descripción opcional del clúster.

18. Las siguientes opciones se rellenan previamente para que coincidan con la configuración del clúster principal y no se pueden cambiar:

- Ubicación
- Engine version (Versión del motor)
- Tipo de instancia
- Tipo de nodo
- Número de fragmentos
- Grupo de parámetros

 Note

ElastiCache genera automáticamente un nuevo grupo de parámetros a partir de los valores del grupo de parámetros proporcionado y aplica el nuevo grupo de parámetros al clúster. Utilice este nuevo grupo de parámetros para modificar parámetros en un almacén de datos global. Cada grupo de parámetros generado automáticamente está asociado a un solo clúster y, por lo tanto, solo a un almacén de datos global.

- Encryption at rest (Cifrado en reposo): permite el cifrado de los datos almacenados en el disco. Para obtener más información, consulte [Cifrado en reposo](#).

 Note

Para proporcionar una clave de cifrado diferente, seleccione la clave AWS KMS gestionada por el cliente y elija la clave. Para obtener más información, consulte [Uso de claves AWS KMS administradas por el cliente](#).

- Encryption in-transit (Cifrado en tránsito): permite el cifrado de datos del cable. Para obtener más información, consulte [Cifrado en tránsito](#). Para la versión 6.4 y superior del motor OSS de Redis, si habilita el cifrado en tránsito, se le pedirá que especifique una de las siguientes opciones de control de acceso:
 - No Access Control (Sin control de acceso): este es el valor predeterminado. Esto indica que no hay restricciones en el acceso de los usuarios al clúster.
 - User Group Access Control List (Lista de control de acceso del grupo de usuarios): elija un grupo de usuarios con un conjunto definido de usuarios que pueden acceder al clúster. Para obtener más información, consulte [Administración de grupos de usuarios con la consola y la CLI](#).
 - Usuario predeterminado de AUTH de Redis OSS: mecanismo de autenticación para el servidor OSS de Redis. Para obtener más información, consulte [Redis OSS AUTH](#).

 Note

Para las versiones de Redis OSS entre la 4.0.2, cuando se admitió por primera vez el cifrado en tránsito, y la 6.0.4, la autenticación de Redis OSS es la única opción.

La configuración de clúster secundario restante se rellena previamente con los mismos valores que el clúster principal, pero se puede actualizar lo siguiente para cumplir los requisitos específicos de dicho clúster:

- Puerto
- Número de réplicas
- Subnet group (Grupo de subredes)
- Zonas de disponibilidad preferidas
- Grupos de seguridad
- Gestionado por el cliente (clave KMS)AWS
- Token de autenticación OSS de Redis
- Habilitar copias de seguridad automáticas
- Backup retention period (Periodo de retención de copia de seguridad)
- Backup target (Intervalo de copia de seguridad)
- Periodo de mantenimiento
- Tema para la notificación de SNS

19. Seleccione Crear. Esto establece el estado del almacén de datos global en Creating (Creando). Una vez que el clúster principal y los clústeres secundarios se asocian al almacén de datos global, el estado cambia a Available (Disponible). Tiene un clúster principal que acepta lecturas y escrituras y un clúster secundario que acepta lecturas replicadas desde el clúster principal.

La página OSS de Redis también se actualiza para indicar si un clúster forma parte de un almacén de datos global, lo que incluye lo siguiente:

- Almacén de datos global: el nombre del almacén de datos global al que pertenece el clúster.
- Rol del almacén de datos global: el rol del clúster, ya sea principal o secundario.

Puede añadir hasta un clúster secundario adicional en una región diferente. AWS Para obtener más información, consulte [Adición de una región a un almacén de datos global](#).

Visualización de detalles del almacén de datos global

Puede ver los detalles de los almacenes de datos globales existentes y también modificarlos en la página Almacenes de datos globales.

Para ver los detalles del almacén de datos global

1. [Inicie sesión en la consola AWS Management Console y ábrala en https://console.aws.amazon.com/elasticache/](https://console.aws.amazon.com/elasticache/). ElastiCache
2. En el panel de navegación, elija Almacenes de datos globales y, a continuación, elija un almacén de datos global disponible.

A continuación, puede examinar las siguientes propiedades globales del almacén de datos:

- Global Datastore Name (Nombre del almacén de datos global): el nombre del almacén de datos global
- Description (Descripción): una descripción del almacén de datos global
- Status (Estado): las opciones son:
 - Creación
 - Modificar
 - Disponible
 - Eliminar
 - Solo principal: este estado indica que el almacén de datos global contiene solo un clúster principal. Todos los clústeres secundarios se eliminan o no se crean correctamente.
- Cluster Mode (Modo de clúster): habilitado o deshabilitado
- Versión del motor OSS de Redis: la versión del motor OSS de Redis que ejecuta el almacén de datos global
- Instance Node Type (Tipo de nodo de instancia): el tipo de nodo utilizado para el almacén de datos global
- Encryption at-rest (Cifrado en reposo): habilitado o deshabilitado
- Encryption in-transit (Cifrado en tránsito): habilitado o deshabilitado
- Autenticación de Redis OSS: habilitada o deshabilitada

Puede realizar los siguientes cambios en el almacén de datos global:

- [Adición de una región a un almacén de datos global](#)
- [Eliminación de una región de un almacén de datos global](#)
- [Promoción del clúster secundario a principal](#)

- [Modificación de un almacén de datos global](#)

La página Global Datastore también muestra los clústeres individuales que componen el almacén de datos global y las siguientes propiedades para cada uno de ellos:

- Región: la AWS región en la que se almacena el clúster
- Role (Rol): primario o secundario
- Cluster name (Nombre del clúster): el nombre del clúster
- Status (Estado): las opciones son:
 - Associating (Asociando): el clúster está en proceso de asociarse al almacén de datos global
 - Associated (Asociado): el clúster está asociado al almacén de datos global
 - Disassociating (Desasociando): proceso de eliminación de un clúster secundario del almacén de datos global mediante el nombre del almacén de datos global. Después de esto, el clúster secundario ya no recibe actualizaciones del clúster principal, sino que permanece como un clúster independiente en esa AWS región.
 - Desasociado: el clúster secundario se ha eliminado del almacén de datos global y ahora es un clúster independiente en su región de AWS .
- Retraso de réplica del almacén de datos global: muestra un valor por AWS región secundaria del almacén de datos global. Es el retraso entre el nodo primario de la región secundaria y el nodo primario de la región principal. En el caso de Redis OSS activado en modo clúster, el retraso indica el retraso máximo en segundos entre los fragmentos.

Adición de una región a un almacén de datos global

Puede añadir hasta una AWS región adicional a un almacén de datos global existente. En este escenario, está creando un clúster de solo lectura en una AWS región independiente que recibe actualizaciones automáticas y asíncronas del clúster principal.

Para agregar una región a un almacén de datos global AWS

1. [Inicie sesión AWS Management Console y abra la ElastiCache consola en https://console.aws.amazon.com/elasticache/.](https://console.aws.amazon.com/elasticache/)
2. En el panel de navegación, elija Almacenes de datos globales y, a continuación, seleccione un almacén de datos global existente.
3. Elija Agregar clúster regional y elija la AWS región en la que residirá el clúster secundario.

4. En Información del clúster, introduzca un valor para el nombre y, si lo desea, para la descripción del clúster.
5. Mantenga las siguientes opciones tal como están. Se rellenan previamente para que coincidan con la configuración del clúster principal y no se pueden cambiar.

- Engine version (Versión del motor)
- Tipo de instancia
- Tipo de nodo
- Número de fragmentos
- Grupo de parámetros

 Note

ElastiCache genera automáticamente un nuevo grupo de parámetros a partir de los valores del grupo de parámetros proporcionado y aplica el nuevo grupo de parámetros al clúster. Utilice este nuevo grupo de parámetros para modificar parámetros en un almacén de datos global. Cada grupo de parámetros generado automáticamente está asociado a un solo clúster y, por lo tanto, solo a un almacén de datos global.

- Cifrado en reposo

 Note

Para proporcionar una clave de cifrado diferente, seleccione la clave `AWS KMS` gestionada por el cliente y elija la clave.

- Cifrado en tránsito
 - AUTENTICACIÓN OSS DE REDIS
6. (Opcional) Actualice la configuración del clúster secundario restante. Esta se rellena previamente con los mismos valores que el clúster principal, pero puede actualizarlos para que cumplan los requisitos específicos de ese clúster:
 - Puerto
 - Número de réplicas
 - Subnet group (Grupo de subredes)
 - Zonas de disponibilidad preferidas

- Grupos de seguridad
- AWS KMS gestionado por el cliente (clave)
- Token de autenticación OSS de Redis
- Habilitar copias de seguridad automáticas
- Backup retention period (Periodo de retención de copia de seguridad)
- Backup target (Intervalo de copia de seguridad)
- Periodo de mantenimiento
- Tema para la notificación de SNS

7. Elija Añadir.

Modificación de un almacén de datos global

Puede modificar las propiedades de los clústeres regionales. Solo puede haber una operación de modificación en curso en un almacén de datos global, con la excepción de la promoción de un clúster secundario a principal. Para obtener más información, consulte [Promoción del clúster secundario a principal](#).

Para modificar un almacén de datos global

1. [Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en https://console.aws.amazon.com/elasticache/](https://console.aws.amazon.com/elasticache/).
2. En el panel de navegación, elija Almacenes de datos globales y, a continuación, en Nombre del almacén de datos global, elija un almacén de datos global.
3. Elija Modify (Modificar) y elija entre las siguientes opciones:
 - Modificar descripción: actualice la descripción del almacén de datos global
 - Modificar la versión del motor: solo está disponible la versión 5.0.6 o posterior del motor Redis OSS.
 - Modificar tipo de nodo: escale los clústeres regionales tanto de forma vertical (reducción y escalado vertical) como horizontal (escalado y reducción horizontal). Las opciones incluyen las familias de nodos R5 y M5. Para obtener más información acerca de los tipos de nodos, consulte [Tipos de nodos compatibles](#).
 - Modificar la conmutación por error automática: habilite o deshabilite la conmutación por error automática. Si habilita la conmutación por error y los nodos principales de los clústeres regionales se cierran inesperadamente, se realiza una conmutación por ElastiCache error a

una de las réplicas regionales. Para obtener más información, consulte [Conmutación por error automática](#).

Para los clústeres OSS de Redis con el modo de clúster activado:

- Agregar particiones: ingrese el número de particiones que desea agregar y, opcionalmente, especifique una o varias zonas de disponibilidad.
- Eliminar fragmentos: elija los fragmentos que desee eliminar en cada región. AWS
- Particiones de reequilibrio: reequilibre la distribución de las ranuras para garantizar una distribución uniforme entre las particiones existentes en el clúster.

Para modificar los parámetros de un almacén de datos global, modifique el grupo de parámetros de cualquier clúster miembro del almacén de datos global. ElastiCache aplica este cambio automáticamente a todos los clústeres de ese almacén de datos global. Para modificar el grupo de parámetros de ese clúster, utilice la consola OSS de Redis o la operación de la [ModifyCacheClusterAPI](#). Para obtener más información, consulte [Modificación de un grupo de parámetros](#). Cuando modifica el grupo de parámetros de cualquier clúster contenido en un almacén de datos global, se aplica a todos los clústeres dentro de ese almacén de datos global.

Para restablecer un grupo de parámetros completo o parámetros específicos, utilice la operación de [ResetCacheParameterGroupAPI](#).

Promoción del clúster secundario a principal

Si el clúster principal o AWS la región dejan de estar disponibles o tienen problemas de rendimiento, puede convertir un clúster secundario en principal. La promoción está permitida en cualquier momento, incluso si hay otras modificaciones en curso. También puede emitir varias promociones en paralelo y el almacén de datos global se resuelve al final en uno principal. Si promociona varios clústeres secundarios simultáneamente, ElastiCache (Redis OSS) no garantiza cuál de ellos pasará a ser el principal.

Para promover un clúster secundario a principal

1. Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación, elija Almacenes de datos globales.
3. Elija el nombre del almacén de datos global para ver los detalles.

4. Elija el clúster Secondary (Secundario).
5. Elija Promote to primary (Promocionar a principal).

A continuación, se pedirá que confirme su decisión con la siguiente advertencia: Promoting a region to primary will make the cluster in this region as read/writable. Are you sure you want to promote the *secondary* cluster to primary?

The current primary cluster in *primary region* will become secondary and will stop accepting writes after this operation completes. Please ensure you update your application stack to direct traffic to the new primary region.

6. Seleccione Confirm (Confirmar) si desea continuar la promoción o Cancel (Cancelar) si no lo desea.

Si opta por confirmar, el almacén de datos global pasará a un estado Modifying (Modificando) y no estará disponible hasta que se complete la promoción.

Eliminación de una región de un almacén de datos global

Puede eliminar una AWS región de un almacén de datos global mediante el siguiente procedimiento.

Para eliminar una AWS región de un almacén de datos global

1. [Inicie sesión AWS Management Console y abra la ElastiCache consola en https://console.aws.amazon.com/elasticache/.](https://console.aws.amazon.com/elasticache/)
2. En el panel de navegación, elija Almacenes de datos globales.
3. Elija un almacén de datos global.
4. Seleccione la Region (Región) que desea eliminar.
5. Elija Remove region (Quitar región).

 Note

Esta opción solo está disponible para clústeres secundarios.

A continuación, se le pedirá que confirme su decisión con la siguiente advertencia: Removing the region will remove your only available cross region replica for the primary cluster. Your primary cluster will no longer be set up for disaster recovery and improved read latency in remote region. Are you sure you want to remove the selected region from the global datastore?

6. Seleccione Confirm (Confirmar) si desea continuar la promoción o Cancel (Cancelar) si no lo desea.

Si elige confirmar, la AWS región se elimina y el clúster secundario deja de recibir las actualizaciones de replicación.

Eliminación de un almacén de datos global

Para eliminar un almacén de datos global, quite primero todos los clústeres secundarios. Para obtener más información, consulte [Eliminación de una región de un almacén de datos global](#). Al hacerlo, el almacén de datos global se encuentra en estado primary-only (solo principal).

Para eliminar un almacén de datos global

1. Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación, elija Almacenes de datos globales.
3. En Global Datastore Name (Nombre del almacén de datos global), elija el almacén de datos que desea eliminar y, a continuación, elija Delete (Eliminar).

A continuación, se le pedirá que confirme su decisión con la siguiente advertencia: Are you sure you want to delete this Global Datastore?

4. Elija Eliminar.

El almacén de datos global pasa al estado Deleting (Eliminando).

Uso de almacenes de datos globales (CLI)

Puede usar AWS Command Line Interface (AWS CLI) para controlar varios AWS servicios desde la línea de comandos y automatizarlos mediante scripts. Puede usar la AWS CLI para operaciones ad hoc (únicas).

Descargar y configurar el AWS CLI

Se AWS CLI ejecuta en Windows, macOS o Linux. Utilice el siguiente procedimiento para descargarlo y configurarlo.

Para descargar, instalar y configurar la CLI

1. Descargue la AWS CLI en la página web de la [interfaz de línea de AWS comandos](#).
2. Siga las instrucciones para instalar la AWS CLI y configurar la AWS CLI de la Guía del AWS Command Line Interface usuario.

Uso de la AWS CLI con almacenes de datos globales

Utilice las siguientes operaciones de la CLI para trabajar con almacenes de datos globales:

- [create-global-replication-group](#)

```
aws elasticache create-global-replication-group \
  --global-replication-group-id-suffix my global datastore \
  --primary-replication-group-id sample-repl-group \
  --global-replication-group-description an optional description of the global
  datastore
```

Amazon aplica ElastiCache automáticamente un prefijo al ID del almacén de datos global cuando se crea. Cada AWS región tiene su propio prefijo. Por ejemplo, un ID de almacén de datos global creado en la región de oeste de EE. UU. (Norte de California) comienza con “virxk” junto con el nombre del sufijo que proporcione. El sufijo, en conjunto con el prefijo autogenerado, garantiza la exclusividad del nombre del almacén de datos global en varias regiones.

En la siguiente tabla se muestra cada AWS región y su prefijo de ID de almacén de datos global.

Nombre de la región/Región	Prefix
Región del Este de EE. UU. (Ohio) us-east-2	fpkhr
Región del Este de EE. UU (Norte de Virginia)	ldgnf

Nombre de la región/Región	Prefix
us-east-1	
Región del Oeste de EE. UU (Norte de California)	virxk
us-west-1	
Región del Oeste de EE. UU (Oregón)	sgau1
us-west-2	
Región de Canadá (centro)	bxodz
ca-central-1	
Región de Asia-Pacífico (Bombay)	erpgt
ap-south-1	
Asia Pacífico (Tokio)	qusw
ap-northeast-1	
Región de Asia-Pacífico (Seúl)	lfqnh
ap-northeast-2	
Región Asia-Pacífico (Osaka)	n1apn
ap-northeast-3	
Región de Asia-Pacífico (Singapur)	v1qxn
ap-southeast-1	
Región de Asia-Pacífico (Sídney)	vbgxd
ap-southeast-2	

Nombre de la región/Región	Prefix
Región de Europa (Fráncfort) eu-central-1	iudkw
Región de Europa (Irlanda) eu-west-1	gxeiz
Región de Europa (Londres) eu-west-2	okuqm
Región EU (París) eu-west-3	fgjhi
Región de América del Sur (São Paulo) sa-east-1	juxlw
Región China (Pekín) cn-north-1	emvgo
Región China (Ningxia) cn-northwest-1	ckbem
Región de Asia-Pacífico (Hong Kong) ap-east-1	knjmp
AWS GovCloud (EE. UU.-Oeste) us-gov-west-1	sgwui

- [create-replication-group](#)— Utilice esta operación para crear clústeres secundarios para un almacén de datos global proporcionando el nombre del almacén de datos global al parámetro. `--global-replication-group-id`

```
aws elasticache create-replication-group \
  --replication-group-id secondary replication group name \
  --replication-group-description "Replication group description" \
  --global-replication-group-id global datastore name
```

Al llamar a esta operación y pasar un `--global-replication-group-id` valor, ElastiCache (Redis OSS) deducirá los valores del grupo de replicación principal del grupo de replicación global para los siguientes parámetros. No ingrese valores para estos parámetros:

```
"PrimaryClusterId",
"AutomaticFailoverEnabled",
"NumNodeGroups",
"CacheParameterGroupName",
"CacheNodeType",
"Engine",
"EngineVersion",
"CacheSecurityGroupNames",
"EnableTransitEncryption",
"AtRestEncryptionEnabled",
"SnapshotArns",
"SnapshotName"
```

- [describe-global-replication-groups](#)

```
aws elasticache describe-global-replication-groups \
  --global-replication-group-id my global datastore \
  --show-member-info an optional parameter that returns a list of the primary and secondary clusters that make up the global datastore
```

- [modify-global-replication-group](#)

```
aws elasticache modify-global-replication-group \
  --global-replication-group-id my global datastore \
  --automatic-failover-enabled \
  --cache-node-type node type \
  --cache-parameter-group-name parameter group name \
  --engine-version engine version \
  --apply-immediately \
  --global-replication-group-description description
```

- [delete-global-replication-group](#)

```
aws elasticache delete-global-replication-group \
  --global-replication-group-id my global datastore \
  --retain-primary-replication-group defaults to true
```

- [disassociate-global-replication-group](#)

```
aws elasticache disassociate-global-replication-group \
  --global-replication-group-id my global datastore \
  --replication-group-id my secondary cluster \
  --replication-group-region the AWS Region in which the secondary cluster resides
```

- [failover-global-replication-group](#)

```
aws elasticache failover-replication-group \
  --global-replication-group-id my global datastore \
  --primary-region The AWS Region of the primary cluster \
  --primary-replication-group-id The name of the global datastore, including the suffix.
```

- [increase-node-groups-in-global-replication-group](#)

```
aws elasticache increase-node-groups-in-global-replication-group \
  --apply-immediately yes \
  --global-replication-group-id global-replication-group-name \
  --node-group-count 3
```

- [decrease-node-groups-in-global-replication-group](#)

```
aws elasticache decrease-node-groups-in-global-replication-group \
  --apply-immediately yes \
```

```
--global-replication-group-id global-replication-group-name \  
--node-group-count 3
```

- [rebalance-shards-in-global-grupo de replicación](#)

```
aws elasticache rebalance-shards-in-global-replication-group \  
--apply-immediately yes \  
--global-replication-group-id global-replication-group-name
```

Utilice la ayuda para enumerar todos los comandos disponibles ElastiCache (Redis OSS).

```
aws elasticache help
```

También puede utilizar la ayuda para describir un comando específico y obtener más información sobre su uso:

```
aws elasticache create-global-replication-group help
```

Alta disponibilidad a través de grupos de reproducción

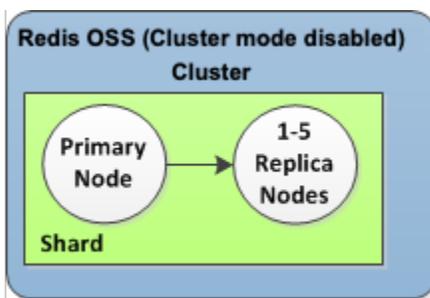
Los clústeres ElastiCache OSS de Amazon Redis de un solo nodo son entidades en memoria con servicios de protección de datos (AOF) limitados. Si el clúster produce un error por cualquier motivo, se perderán todos los datos del clúster. Sin embargo, si utiliza el motor OSS de Redis, puede agrupar de 2 a 6 nodos en un clúster con réplicas en el que de 1 a 5 nodos de solo lectura contengan datos replicados del único nodo principal de lectura/escritura del grupo. En esta situación, si solo se produce un error en un nodo por cualquier motivo, no perderá todos los datos, ya que estos se replican en otro u otros nodos. Debido a la latencia de reproducción, es posible que se pierdan algunos datos si el error se produce en el nodo principal de lectura/escritura.

Como se ve en el siguiente gráfico, la estructura de replicación está contenida en un fragmento (denominado grupo de nodos en la API/CLI) que se encuentra dentro de un clúster de Redis OSS. Los clústeres de Redis OSS (modo de clúster desactivado) siempre tienen un fragmento. Los clústeres OSS de Redis (modo de clúster activado) pueden tener hasta 500 fragmentos y los datos del clúster se dividen entre ellos. Puede crear un clúster con un mayor número de particiones y un menor número de réplicas con un total de hasta 90 nodos por clúster. Esta configuración de clúster puede variar desde 90 particiones y 0 réplicas hasta 15 particiones y 5 réplicas, que es el número máximo de réplicas permitido.

El límite de nodos o fragmentos se puede aumentar hasta un máximo de 500 por clúster si la versión del motor OSS de Redis es 5.0.6 o superior. Por ejemplo, puede elegir configurar un clúster de 500 nodos que oscila entre 83 particiones (uno primario y 5 réplicas por partición) y 500 particiones (único primario y sin réplicas). Asegúrese de que hay suficientes direcciones IP disponibles para acomodar el aumento. Algunos problemas comunes incluyen que las subredes del grupo de subredes tienen un rango CIDR demasiado pequeño o que otros clústeres comparten y utilizan considerablemente las subredes. Para obtener más información, consulte [Creación de un grupo de subredes](#).

Para las versiones inferiores a 5.0.6, el límite es de 250 por clúster.

Para solicitar un aumento del límite, consulte [AWS Service Limits](#) y elija el tipo de límite Nodes per cluster per instance type (Nodos por clúster por tipo de instancias).



El clúster OSS de Redis (modo de clúster desactivado) tiene un fragmento y de 0 a 5 nodos de réplica

Si el clúster con réplicas tiene habilitado Multi-AZ y el nodo principal produce un error, el principal realizará una conmutación por error a una réplica de lectura. Dado que los datos se actualizan en los nodos de réplica de forma asíncrona, es posible que haya algunas pérdidas de datos debido a la latencia en la actualización de los nodos de réplica. Para obtener más información, consulte [Mitigar los errores al ejecutar Redis OSS](#).

Temas

- [Descripción de la replicación de Redis OSS](#)
- [Replicación: Redis OSS \(modo de clúster desactivado\) frente a Redis OSS \(modo de clúster activado\)](#)
- [Minimizar el tiempo de inactividad en ElastiCache \(Redis OSS\) con Multi-AZ](#)
- [Cómo se implementan la sincronización y la copia de seguridad](#)
- [Creación de un grupo de replicación de Redis OSS](#)
- [Visualización de detalles de un grupo de reproducción](#)

- [Búsqueda de puntos de enlace del grupo de reproducción](#)
- [Modificación de un grupo de reproducción](#)
- [Eliminación de un grupo de reproducción](#)
- [Cambio del número de réplicas](#)
- [Convertir una réplica de lectura en principal, para grupos de replicación de Redis OSS \(modo de clúster desactivado\)](#)

Descripción de la replicación de Redis OSS

Redis OSS implementa la replicación de dos maneras:

- Con un único fragmento que contiene todos los datos del clúster en cada nodo: Redis OSS (modo de clúster desactivado)
- Con los datos divididos en hasta 500 fragmentos: Redis OSS (modo de clúster activado)

Cada partición de un grupo de reproducción tiene un nodo primario de lectura/escritura y hasta 5 nodos de réplica de solo lectura. Puede crear un clúster con un mayor número de particiones y un menor número de réplicas con un total de hasta 90 nodos por clúster. Esta configuración de clúster puede variar desde 90 particiones y 0 réplicas hasta 15 particiones y 5 réplicas, que es el número máximo de réplicas permitido.

El límite de nodos o fragmentos se puede aumentar hasta un máximo de 500 por clúster si la versión del motor OSS de Redis es 5.0.6 o superior. Por ejemplo, puede elegir configurar un clúster de 500 nodos que oscila entre 83 particiones (uno primario y 5 réplicas por partición) y 500 particiones (único primario y sin réplicas). Asegúrese de que hay suficientes direcciones IP disponibles para acomodar el aumento. Algunos problemas comunes incluyen que las subredes del grupo de subredes tienen un rango CIDR demasiado pequeño o que otros clústeres comparten y utilizan considerablemente las subredes. Para obtener más información, consulte [Creación de un grupo de subredes](#).

Para las versiones inferiores a 5.0.6, el límite es de 250 por clúster.

Para solicitar un aumento del límite, consulte [AWS Service Limits](#) y elija el tipo de límite Nodes per cluster per instance type (Nodos por clúster por tipo de instancias).

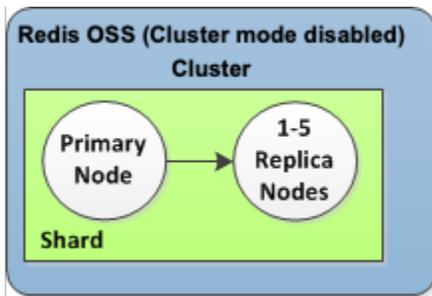
Temas

- [Redis OSS \(modo de clúster desactivado\)](#)
- [Redis OSS \(modo de clúster activado\)](#)

Redis OSS (modo de clúster desactivado)

Un clúster de Redis OSS (modo de clúster desactivado) tiene una sola partición, dentro de la cual hay un conjunto de nodos de Redis OSS: un nodo principal de lectura/escritura y hasta cinco nodos de réplica secundarios de solo lectura. Cada réplica de lectura mantiene una copia de los datos del nodo principal del clúster. Se utilizan mecanismos de replicación asíncronos para mantener las réplicas de lectura sincronizadas con el principal. Las aplicaciones pueden leer de cualquier nodos

del clúster. Las aplicaciones pueden escribir únicamente en el nodo principal. Las réplicas de lectura mejoran el rendimiento de lectura y evitan la pérdida de datos en caso de error en un nodo.



Clúster de Redis OSS (modo de clúster desactivado) con una sola partición y nodos de réplica

Puede utilizar clústeres de Redis OSS (modo de clúster desactivado) con nodos de réplica para escalar su solución de OSS de Redis ElastiCache a fin de gestionar aplicaciones de lectura intensiva o admitir un gran número de clientes que lean simultáneamente desde el mismo clúster.

Todos los nodos de un clúster de Redis OSS (modo de clúster desactivado) deben residir en la misma región.

Al agregar una réplica de lectura a un clúster, todos los datos de la instancia principal se copian al nuevo nodo. A partir de entonces, siempre que se escriben datos en el nodo principal, los cambios se propagarán de forma asíncrona a todas las réplicas de lectura.

Para mejorar la tolerancia a errores y reducir el tiempo de inactividad de escritura, habilite Multi-AZ con conmutación por error automática para su clúster de Redis OSS (modo de clúster desactivado) con réplicas. Para obtener más información, consulte [Minimizar el tiempo de inactividad en ElastiCache \(Redis OSS\) con Multi-AZ](#).

Puede cambiar las funciones de los nodos del clúster OSS de Redis (modo de clúster desactivado), de forma que la réplica principal y una de las réplicas intercambien funciones. Puede que decida hacerlo por motivos de ajuste del rendimiento. Por ejemplo, con una aplicación web que tenga una gran actividad de escritura, puede elegir el nodo con la menor latencia de red. Para obtener más información, consulte [Convertir una réplica de lectura en principal, para grupos de replicación de Redis OSS \(modo de clúster desactivado\)](#).

Redis OSS (modo de clúster activado)

Un clúster OSS (modo de clúster habilitado) de Redis se compone de 1 a 500 fragmentos (API/CLI: grupos de nodos). Cada partición tiene un nodo principal y hasta 5 nodos de réplica de solo lectura. La configuración puede variar desde 90 particiones y 0 réplicas hasta 15 particiones y 5 réplicas, que es el número máximo de réplicas permitido.

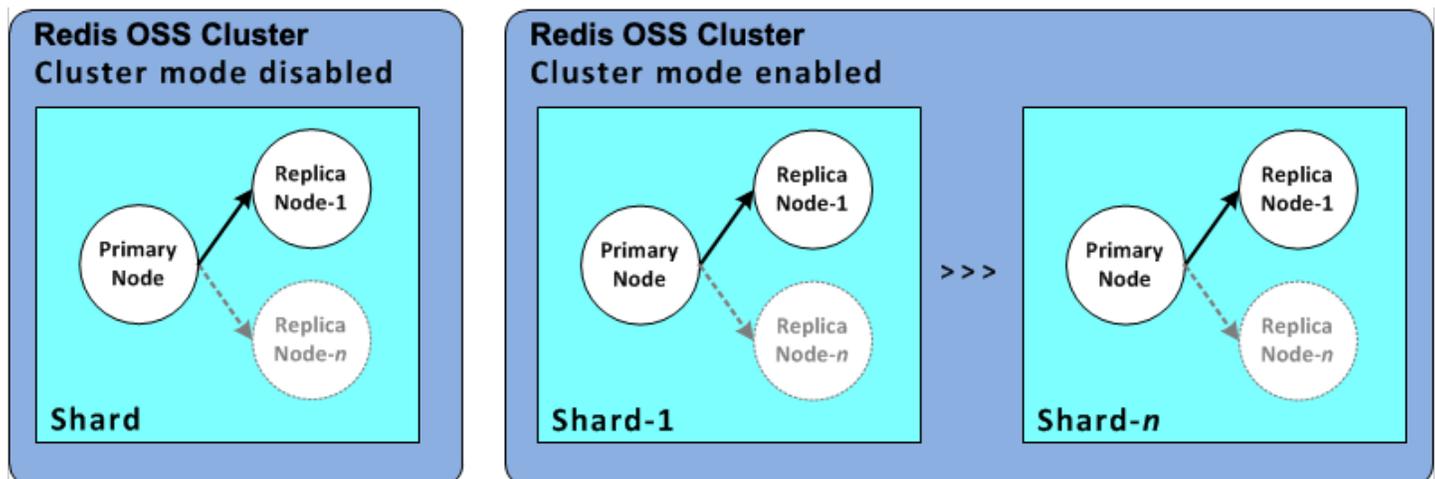
El límite de nodos o fragmentos se puede aumentar hasta un máximo de 500 por clúster si la versión del motor OSS de Redis es 5.0.6 o superior. Por ejemplo, puede elegir configurar un clúster de 500 nodos que oscila entre 83 particiones (uno primario y 5 réplicas por partición) y 500 particiones (único primario y sin réplicas). Asegúrese de que hay suficientes direcciones IP disponibles para acomodar el aumento. Algunos problemas comunes incluyen que las subredes del grupo de subredes tienen un rango CIDR demasiado pequeño o que otros clústeres comparten y utilizan considerablemente las subredes. Para obtener más información, consulte [Creación de un grupo de subredes](#).

Para las versiones inferiores a 5.0.6, el límite es de 250 por clúster.

Para solicitar un aumento del límite, consulte [AWS Service Limits](#) y elija el tipo de límite Nodes per cluster per instance type (Nodos por clúster por tipo de instancias).

Cada réplica de lectura de un fragmento mantiene una copia de los datos del principal del fragmento. Se utilizan mecanismos de replicación asíncronos para mantener las réplicas de lectura sincronizadas con el principal. Las aplicaciones pueden leer de cualquier nodos del clúster. Las aplicaciones pueden escribir únicamente en los nodos. Las réplicas de lectura mejoran la escalabilidad de lectura y evitan la pérdida de datos. Los datos se dividen entre los fragmentos de un clúster de Redis OSS (habilitado para el modo de clúster).

Las aplicaciones utilizan el punto final de configuración del clúster OSS de Redis (modo de clúster habilitado) para conectarse con los nodos del clúster. Para obtener más información, consulte [Búsqueda de puntos de conexión](#).



Clúster de Redis OSS (modo de clúster activado) con varios fragmentos y nodos de réplica

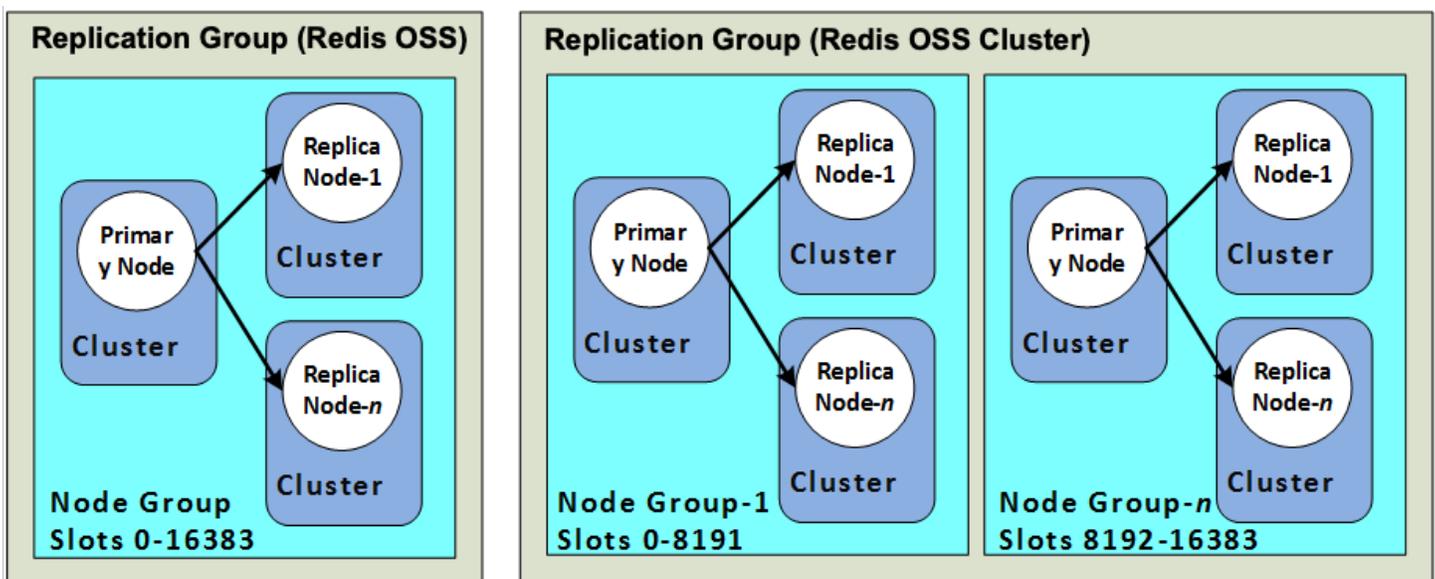
Todos los nodos de un clúster OSS de Redis (modo de clúster habilitado) deben residir en la misma región. Para mejorar la tolerancia a errores, puede aprovisionar tanto a los principales como a las réplicas de lectura en varias zonas de disponibilidad dentro de esa región.

Actualmente, en Redis OSS (modo de clúster habilitado), existen algunas limitaciones.

- No puede promocionar manualmente ningún nodo de réplica a principal.

Replicación: Redis OSS (modo de clúster desactivado) frente a Redis OSS (modo de clúster activado)

A partir de la versión 3.2 de Redis OSS, puede crear uno de los dos tipos distintos de clústeres de Redis OSS (API/CLI: grupos de replicación). Un clúster OSS (modo de clúster desactivado) de Redis siempre tiene un único fragmento (API/CLI: grupo de nodos) con hasta 5 nodos de réplica de lectura. Un clúster OSS de Redis (habilitado para el modo de clúster) tiene hasta 500 fragmentos con 1 a 5 nodos de réplica de lectura en cada uno.



Clústeres de Redis OSS (modo de clúster desactivado) y Redis OSS (modo de clúster activado)

En la siguiente tabla se resumen las diferencias importantes entre los clústeres de Redis OSS (modo de clúster desactivado) y de Redis OSS (modo de clúster activado).

Comparación de los clústeres de Redis OSS (modo de clúster desactivado) y Redis OSS (modo de clúster activado)

Característica	Redis OSS (modo de clúster desactivado)	Redis OSS (modo de clúster activado)
Modificable	Sí. Admite la adición y eliminación de nodos de	Limitado. Para obtener más información, consulte

Característica	Redis OSS (modo de clúster desactivado)	Redis OSS (modo de clúster activado)
	réplica, así como la ampliación del tipo de nodo.	Versiones del motor y actualizaciones y Escalar clústeres en Redis OSS (modo de clúster habilitado) .
Particiones de datos	No	Sí
Particiones	1	1 a 500
Réplicas de lectura	De 0 a 5	0 a 5 por partición.
	 Important Si no tiene réplicas y el nodo falla, experimentará la pérdida total de datos.	 Important Si no tiene réplicas y el nodo falla, experimentará la pérdida de todos los datos del fragmento.
Multi-AZ	Sí, con al menos 1 réplica. Opcional. Activado de forma predeterminada.	Sí Opcional. Activado de forma predeterminada.
Instantáneas (copias de seguridad)	Sí, creando un único archivo .rdb.	Sí, creando un único archivo .rdb para cada fragmento.
Restaurar	Sí, utilizando un único archivo.rdb de un clúster de Redis OSS (modo de clúster desactivado).	Sí, se utilizan archivos.rdb de un clúster de Redis OSS (modo de clúster desactivado) o de Redis OSS (modo de clúster activado).
Compatible con	Todas las versiones de Redis OSS	Redis OSS 3.2 y siguientes

Característica	Redis OSS (modo de clúster desactivado)	Redis OSS (modo de clúster activado)
Motor actualizable	Sí, con algunos límites. Para obtener más información, consulte Versiones del motor y actualizaciones .	Sí, con algunos límites. Para obtener más información, consulte Versiones del motor y actualizaciones .
Cifrado	Versiones 3.2.6 (programada para su fin de vida útil; consulte el calendario de fin de vida de las versiones de Redis OSS) y 4.0.10 y posteriores.	Versiones 3.2.6 (programada para la EOL; consulte el calendario de fin de vida de las versiones de Redis OSS) y 4.0.10 y posteriores.
Cumplimiento de requisitos de HIPAA	Versiones 3.2.6 (programada para la EOL; consulte el calendario de fin de vida de las versiones de Redis OSS) y 4.0.10 y posteriores.	Versiones 3.2.6 (programada para la EOL; consulte el calendario de fin de vida de las versiones de Redis OSS) y 4.0.10 y posteriores.
Compatible con PCI/DSS	Versiones 3.2.6 (programada para la EOL; consulte el calendario de fin de vida de las versiones de Redis OSS) y 4.0.10 y posteriores.	Versiones 3.2.6 (programada para la EOL; consulte el calendario de fin de vida de las versiones de Redis OSS) y 4.0.10 y posteriores.
Cambios de los fragmentos online	N/A	Versión 3.2.10 (programada para la EOL; consulte el calendario de fin de vida de las versiones de Redis OSS) y posteriores.

¿Cuál debería elegir?

Al elegir entre Redis OSS (modo de clúster desactivado) o Redis OSS (modo de clúster activado), tenga en cuenta los siguientes factores:

- Escalado frente a partición: las necesidades empresariales cambian. Debe aprovisionar los picos de demanda o escalar conforme cambie la demanda. Redis OSS (modo de clúster desactivado) admite el escalado. Puede escalar la capacidad de lectura añadiendo o eliminando nodos de réplica, o bien puede escalar la capacidad ampliando a un tipo de nodo más grande. Ambas operaciones requieren tiempo. Para obtener más información, consulte [Escalado de clústeres de Redis OSS \(modo de clúster desactivado\) con nodos de réplica](#).

Redis OSS (modo de clúster activado) permite dividir los datos en hasta 500 grupos de nodos. Puede cambiar el número de fragmentos de forma dinámica a medida que cambien sus necesidades de negocio. Una ventaja de la partición es que puede repartir la carga entre un mayor número de puntos de enlace, lo que reduce los cuellos de botella de acceso durante los picos de demanda. Además, puede alojar un mayor conjunto de datos, ya que los datos se pueden distribuir en varios servidores. Para obtener información sobre cómo escalar las particiones, consulte [Escalar clústeres en Redis OSS \(modo de clúster habilitado\)](#)

- Tamaño de nodo en comparación con número de nodos: dado que un clúster OSS (modo de clúster desactivado) de Redis solo tiene un fragmento, el tipo de nodo debe ser lo suficientemente grande como para alojar todos los datos del clúster, además de la sobrecarga necesaria. Por otro lado, dado que puede particionar los datos en varios fragmentos cuando utiliza un clúster de Redis OSS (habilitado para el modo de clúster), los tipos de nodos pueden ser más pequeños, aunque necesite más. Para obtener más información, consulte [Elección del tamaño del nodo](#).
- Lecturas y escrituras: si la carga principal del clúster son las aplicaciones que leen datos, puede escalar un clúster de Redis OSS (modo de clúster desactivado) añadiendo y eliminando réplicas de lectura. Sin embargo, hay un máximo de 5 réplicas de lectura. Si la carga de su clúster requiere un uso intensivo de escritura, puede aprovechar los puntos finales de escritura adicionales de un clúster OSS (habilitado para el modo de clúster) de Redis con varios fragmentos.

Independientemente del tipo de clúster que elija para implementar, debe asegurarse de elegir un tipo de nodo que se ajuste a sus necesidades actuales y futuras.

Minimizar el tiempo de inactividad en ElastiCache (Redis OSS) con Multi-AZ

Hay varios casos en los que ElastiCache (Redis OSS) puede necesitar reemplazar un nodo principal; estos incluyen ciertos tipos de mantenimiento planificado y el improbable caso de que se produzca un fallo en el nodo principal o en la zona de disponibilidad.

Este reemplazo produce un tiempo de inactividad para el clúster, pero si Multi-AZ se encuentra habilitado, el tiempo de inactividad es mínimo. El rol del nodo primario tendrá una conmutación por error automática en una de las réplicas de lectura. No es necesario crear ni aprovisionar un nuevo nodo principal, ya que ElastiCache lo gestionará de forma transparente. Esta conmutación por error y promoción de réplica garantizan la posibilidad de reanudar la escritura en la réplica principal tan pronto como se complete la promoción.

ElastiCache también propaga el nombre del Servicio de nombres de dominio (DNS) de la réplica promocionada. Lo hace así porque, en ese caso, si su aplicación escribe en el punto de enlace principal, no se requiere ningún cambio de punto de conexión en su aplicación. Si lee desde puntos de conexión individuales, asegúrese de cambiar el punto de enlace de lectura de la réplica promovida a principal en el punto de enlace de la nueva réplica.

En caso de que se inicien reemplazos de nodos planificados debido a actualizaciones de mantenimiento o actualizaciones de autoservicio, tenga en cuenta lo siguiente:

- En el ElastiCache caso del clúster (Redis OSS), las sustituciones de nodos planificadas se completan mientras el clúster atiende las solicitudes de escritura entrantes.
- En el caso de los clústeres OSS de Redis deshabilitados con Multi-AZ habilitados y que se ejecutan en el motor 5.0.6 o posterior, los reemplazos de nodos planificados se completan mientras el clúster atiende las solicitudes de escritura entrantes.
- En el caso de los clústeres desactivados en el modo OSS Cluster de Redis con Multi-AZ activado y que se ejecutan en el motor 4.0.10 o anterior, es posible que se produzca una breve interrupción de escritura asociada a las actualizaciones de DNS. Esta interrupción es posible que tarde unos segundos. Este proceso es mucho más rápido que el de volver a crear y aprovisionar una réplica principal nueva, que es el proceso que se realiza en caso de no habilitar Multi-AZ.

Puede habilitar las zonas de disponibilidad múltiples mediante la consola de ElastiCache administración, la o la AWS CLI API. ElastiCache

La activación de ElastiCache Multi-AZ en su clúster de OSS de Redis (en la API y la CLI, grupo de replicación) mejora su tolerancia a los errores. Esto es cierto especialmente en los casos en que el

nodo principal de lectura/escritura del clúster deja de estar accesible o de funcionar por cualquier motivo. El modo Multi-AZ solo se admite en los clústeres de OSS de Redis que tienen más de un nodo en cada fragmento.

Temas

- [Habilitación de Multi-AZ](#)
- [Escenarios de error con respuestas de Multi-AZ](#)
- [Prueba de la conmutación por error automática](#)
- [Limitaciones de Redis OSS Multi-AZ](#)

Habilitación de Multi-AZ

Puede habilitar Multi-AZ al crear o modificar un clúster (API o CLI, grupo de replicación) mediante la ElastiCache consola o la ElastiCache API. AWS CLI

Puede habilitar Multi-AZ solo en los clústeres de Redis OSS (modo de clúster desactivado) que tengan al menos una réplica de lectura disponible. Los clústeres sin réplicas de lectura no ofrecen alta disponibilidad ni tolerancia a errores. Para obtener información acerca de la creación de clústeres con reproducción, consulte [Creación de un grupo de replicación de Redis OSS](#). Para obtener información acerca de la adición de réplicas de lectura a un clúster con reproducción, consulte [Agregar una réplica de lectura, para grupos de replicación de Redis OSS \(modo de clúster deshabilitado\)](#).

Temas

- [Habilitación de Multi-AZ \(consola\)](#)
- [Habilitación de Multi-AZ \(AWS CLI\)](#)
- [Habilitación de Multi-AZ \(ElastiCache API\)](#)

Habilitación de Multi-AZ (consola)

Puede activar Multi-AZ mediante la ElastiCache consola al crear un nuevo clúster de OSS de Redis o modificando un clúster de OSS de Redis existente mediante replicación.

La zona de disponibilidad múltiple está habilitada de forma predeterminada en los clústeres de Redis OSS (modo de clúster activado).

⚠ Important

ElastiCache activará automáticamente las zonas de disponibilidad múltiples solo si el clúster contiene al menos una réplica en una zona de disponibilidad diferente de la principal en todos los fragmentos.

Habilitar Multi-AZ al crear un clúster mediante la consola ElastiCache

Para obtener más información acerca de este proceso, consulte [Creación de un clúster de Redis OSS \(modo de clúster desactivado\) \(consola\)](#). Asegúrese de tener una o más réplicas y habilitar Multi-AZ.

Habilitación de Multi-AZ en un clúster existente (consola)

Para obtener más información sobre este proceso, consulte [Modificación de un clúster](#) [Uso del AWS Management Console](#).

Habilitación de Multi-AZ (AWS CLI)

En el siguiente ejemplo de código, se utiliza AWS CLI para habilitar la zona de disponibilidad múltiple para el grupo de replicación. `redis12`

⚠ Important

El grupo de reproducción `redis12` debe existir y tener al menos una réplica de lectura disponible.

Para Linux, macOS o Unix:

```
aws elasticache modify-replication-group \  
  --replication-group-id redis12 \  
  --automatic-failover-enabled \  
  --multi-az-enabled \  
  --apply-immediately
```

Para Windows:

```
aws elasticache modify-replication-group ^
```

```
--replication-group-id redis12 ^
--automatic-failover-enabled ^
--multi-az-enabled ^
--apply-immediately
```

La salida JSON de este comando debería tener un aspecto similar al siguiente.

```
{
  "ReplicationGroup": {
    "Status": "modifying",
    "Description": "One shard, two nodes",
    "NodeGroups": [
      {
        "Status": "modifying",
        "NodeGroupMembers": [
          {
            "CurrentRole": "primary",
            "PreferredAvailabilityZone": "us-west-2b",
            "CacheNodeId": "0001",
            "ReadEndpoint": {
              "Port": 6379,
              "Address":
"redis12-001.v5r9dc.0001.usw2.cache.amazonaws.com"
            },
            "CacheClusterId": "redis12-001"
          },
          {
            "CurrentRole": "replica",
            "PreferredAvailabilityZone": "us-west-2a",
            "CacheNodeId": "0001",
            "ReadEndpoint": {
              "Port": 6379,
              "Address":
"redis12-002.v5r9dc.0001.usw2.cache.amazonaws.com"
            },
            "CacheClusterId": "redis12-002"
          }
        ],
        "NodeGroupId": "0001",
        "PrimaryEndpoint": {
          "Port": 6379,
          "Address": "redis12.v5r9dc.ng.0001.usw2.cache.amazonaws.com"
        }
      }
    ]
  }
}
```

```
    }
  ],
  "ReplicationGroupId": "redis12",
  "SnapshotRetentionLimit": 1,
  "AutomaticFailover": "enabling",
  "MultiAZ": "enabled",
  "SnapshotWindow": "07:00-08:00",
  "SnapshottingClusterId": "redis12-002",
  "MemberClusters": [
    "redis12-001",
    "redis12-002"
  ],
  "PendingModifiedValues": {}
}
```

Para obtener más información, consulte los temas siguientes en la Referencia de los comandos de la AWS CLI :

- [create-cache-cluster](#)
- [create-replication-group](#)
- [modify-replication-group](#) en la Referencia de AWS CLI comandos.

Habilitación de Multi-AZ (ElastiCache API)

El siguiente ejemplo de código usa la ElastiCache API para habilitar las zonas de disponibilidad múltiples para el grupo de replicación. `redis12`

Note

Para usar este ejemplo, el grupo de reproducción `redis12` debe existir y tener al menos una réplica de lectura disponible.

```
https://elasticache.us-west-2.amazonaws.com/
?Action=ModifyReplicationGroup
&ApplyImmediately=true
&AutoFailover=true
&MultiAZEnabled=true
&ReplicationGroupId=redis12
```

```
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20140401T192317Z
&X-Amz-Credential=<credential>
```

Para obtener más información, consulte estos temas en la referencia de la ElastiCache API:

- [CreateCacheCluster](#)
- [CreateReplicationGroup](#)
- [ModifyReplicationGroup](#)

Escenarios de error con respuestas de Multi-AZ

Antes de la introducción de Multi-AZ, ElastiCache detectaba y sustituía los nodos defectuosos de un clúster mediante la recreación y el reaprovisionamiento del nodo defectuoso. Si habilita Multi-AZ, un nodo principal que produce error conmuta por error a la réplica con el menor retraso de reproducción. La réplica seleccionada se promocionará automáticamente a la principal, lo cual es mucho más rápido que crear y reaprovisionar un nuevo nodo principal. Este proceso suele tardar tan solo unos segundos hasta que se puede escribir de nuevo en el clúster.

Cuando la Multi-AZ está habilitada, supervisa ElastiCache continuamente el estado del nodo principal. Si se produce un error en el nodo principal, se realiza una de las siguientes acciones en función del tipo de error.

Temas

- [Escenarios de error cuando solo se produce un error en el nodo principal](#)
- [Escenarios de error cuando el nodo primario y algunas réplicas de lectura producen un error](#)
- [Escenarios de error cuando se produce un error en todo el clúster](#)

Escenarios de error cuando solo se produce un error en el nodo principal

Si solo se produce un error en el nodo principal, la réplica de lectura con el menor retardo de reproducción se promociona al clúster principal. A continuación, se crea una réplica de lectura de reemplazo y se aprovisiona en la misma zona de disponibilidad que el principal ha producido un error.

Cuando solo falla el nodo principal, ElastiCache Multi-AZ hace lo siguiente:

1. El nodo principal con error se desconecta (sin conexión).
2. La réplica de lectura con el mínimo retardo de reproducción se promociona a nodo principal.

Las operaciones de escritura se pueden reanudar tan pronto como se haya completado el proceso de promoción, por lo general, en tan solo unos segundos. Si su aplicación está escribiendo en el punto final principal, no necesita cambiar el punto final de escritura o lectura. ElastiCachepropaga el nombre DNS de la réplica promocionada.

3. Una réplica de lectura de reemplazo se lanza y aprovisiona.

La réplica de lectura de reemplazo se lanza en la zona de disponibilidad en la que estaba el nodo principal con error, por lo que se mantiene la distribución de los nodos.

4. Las réplicas se sincronizan con el nuevo nodo principal.

Una vez que la nueva réplica esté disponible, tenga en cuenta estos efectos:

- Punto de enlace principal: no debe realizar cambios en su aplicación, ya que el nombre de DNS del nodo primario nuevo se propagará al punto de conexión principal.
- Punto de enlace de lectura: el punto de conexión del lector se actualiza de forma automática para apuntar a los nodos de réplica nuevos.

Para obtener información acerca de la búsqueda de los puntos de conexión de un clúster, consulte los temas siguientes:

- [Búsqueda de los puntos finales de un clúster de Redis OSS \(modo de clúster desactivado\) \(consola\)](#)
- [Búsqueda de los puntos de enlace de grupos de reproducción \(AWS CLI\)](#)
- [Búsqueda de puntos finales para grupos de replicación \(\) ElastiCache API](#)

Escenarios de error cuando el nodo primario y algunas réplicas de lectura producen un error

Si se produce un error en el nodo principal y en al menos una réplica, la réplica disponible con el menor retardo de reproducción se promocionará al clúster principal. Las nuevas réplicas de lectura también se crean y se aprovisionan en las mismas zonas de disponibilidad que las de los nodos con error y que la réplica que se promocionó a nodo principal.

Cuando el nodo principal y algunas réplicas de lectura fallan, ElastiCache Multi-AZ hace lo siguiente:

1. El nodo principal y las réplicas de lectura con error se desconectan.
2. La réplica disponible con el mínimo retardo de reproducción se promociona a nodo principal.

Las operaciones de escritura se pueden reanudar tan pronto como se haya completado el proceso de promoción, por lo general, en tan solo unos segundos. Si su aplicación está escribiendo en el punto final principal, no es necesario cambiar el punto final para las escrituras. ElastiCache propaga el nombre DNS de la réplica promocionada.

3. Las réplicas de reemplazo se crean y se aprovisionan.

Las réplicas de reemplazo se crean en las zonas de disponibilidad de los nodos con error para, de este modo, conservar la distribución de los nodos.

4. Todos los clústeres se sincronizan con el nodo principal.

Debe realizar los siguientes cambios en su aplicación una vez que los nuevos nodos estén disponibles:

- Punto de conexión principal: no realice cambios en su aplicación. El nombre de DNS del nuevo nodo principal se propaga al punto de conexión principal.
- Punto de conexión de lectura: el punto de enlace de lectura se actualiza de forma automática para apuntar a los nodos de réplica nuevos.

Para obtener información acerca de la búsqueda de los puntos de enlace de un grupo de replicación, consulte los temas siguientes:

- [Búsqueda de los puntos finales de un clúster de Redis OSS \(modo de clúster desactivado\) \(consola\)](#)
- [Búsqueda de los puntos de enlace de grupos de reproducción \(AWS CLI\)](#)
- [Búsqueda de puntos finales para grupos de replicación \(\) ElastiCache API](#)

Escenarios de error cuando se produce un error en todo el clúster

Si el error es general, todos los nodos se volverán a crear y a aprovisionar en las mismas zonas de disponibilidad que las de los nodos originales.

En esta situación, se perderán todos los datos del clúster debido al error de todos los nodos del clúster. Este tipo de error no suele producirse con frecuencia.

Cuando se produce un error en todo el clúster, ElastiCache Multi-AZ hace lo siguiente:

1. El nodo principal y las réplicas de lectura se desconectan.
2. Se crea y se aprovisiona un nodo principal de reemplazo.
3. Las réplicas de reemplazo se crean y se aprovisionan.

Los reemplazos se crean en las zonas de disponibilidad de los nodos con error para, de este modo, conservar la distribución de los nodos.

Puesto que el error ha afectado a la totalidad del clúster, los datos se perderán y los nuevos nodos se crean vacíos.

Puesto que cada uno de los nodos de reemplazo tendrán el mismo punto de conexión que el nodo al que reemplacen, no es necesario realizar ningún cambio de punto de conexión en su aplicación.

Para obtener información acerca de la búsqueda de los puntos de enlace de un grupo de replicación, consulte los temas siguientes:

- [Búsqueda de los puntos finales de un clúster de Redis OSS \(modo de clúster desactivado\) \(consola\)](#)
- [Búsqueda de los puntos de enlace de grupos de reproducción \(AWS CLI\)](#)
- [Búsqueda de puntos finales para grupos de replicación \(\) ElastiCache API](#)

Recomendamos que cree el nodo principal y las réplicas de lectura en distintas zonas de disponibilidad para incrementar el nivel de tolerancia a errores.

Prueba de la conmutación por error automática

Después de habilitar la conmutación por error automática, puede probarla mediante la ElastiCache consola AWS CLI, la y la ElastiCache API.

Cuando realice las pruebas, tenga en cuenta lo siguiente:

- Puedes usar esta operación para probar la conmutación por error automática en hasta 15 fragmentos (denominados grupos de nodos en la ElastiCache API AWS CLI) en cualquier período continuo de 24 horas.
- Si realiza una llamada a esta operación en fragmentos de distintos clústeres (denominados grupos de reproducción en la API y la CLI), podrá realizar las llamadas de forma simultánea.
- En algunos casos, puede realizar esta operación varias veces en diferentes fragmentos del mismo grupo de replicación de Redis OSS (habilitado para el modo de clúster). En tales casos, la sustitución del primer nodo debe completarse antes de que se pueda realizar una llamada posterior.
- Para determinar si se ha completado la sustitución del nodo, compruebe los eventos mediante la ElastiCache consola de Amazon AWS CLI, la o la ElastiCache API. Busque los eventos relacionados con la conmutación por error automática que se indican a continuación por orden de incidencia:
 1. Mensaje del grupo de replicación: `Test Failover API called for node group <node-group-id>`
 2. Mensaje del clúster de caché: `Failover from primary node <primary-node-id> to replica node <node-id> completed`
 3. Mensaje del grupo de replicación: `Failover from primary node <primary-node-id> to replica node <node-id> completed`
 4. Mensaje del clúster de caché: `Recovering cache nodes <node-id>`
 5. Mensaje del clúster de caché: `Finished recovery for cache nodes <node-id>`

Para más información, consulte los siguientes temas:

- [Visualización de eventos de ElastiCache](#) en la Guía del usuario de ElastiCache .
- [DescribeEvents](#) en la Referencia de la API de ElastiCache
- [describe-events](#) en la Referencia de los comandos de la AWS CLI .
- Esta API está diseñada para probar el comportamiento de la aplicación en caso de ElastiCache conmutación por error. No está diseñado para ser una herramienta operativa para iniciar una

conmutación por error para solucionar un problema con el clúster. Además, en determinadas condiciones, como eventos operativos a gran escala, AWS puede bloquear esta API.

Temas

- [Probar la conmutación por error automática mediante el AWS Management Console](#)
- [Prueba de la conmutación por error automática mediante el AWS CLI](#)
- [Probar la conmutación por error automática mediante la API ElastiCache](#)

Probar la conmutación por error automática mediante el AWS Management Console

Utilice el procedimiento siguiente para probar la conmutación por error automática con la consola.

Para probar la conmutación por error automática

1. Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación, elija Redis OSS.
3. En la lista de clústeres de Redis OSS, seleccione la casilla situada a la izquierda del clúster que desee probar. El clúster debe tener al menos un nodo de réplica de lectura.
4. En el área Details, asegúrese de que este clúster tiene habilitadas Multi-AZ. Si el clúster no tiene habilitado Multi-AZ, elija un clúster distinto o modifique este clúster para habilitar Multi-AZ. Para obtener más información, consulte [Uso del AWS Management Console](#).



5. Para Redis OSS (modo de clúster desactivado), elija el nombre del clúster.

Para Redis OSS (modo de clúster activado), haga lo siguiente:

- a. Elija el nombre del clúster.
 - b. En la página Shards, elija el nombre del fragmento (denominado grupo de nodos en la API y la CLI) en el que desea probar la conmutación por error.
6. En la página Nodos, elija Failover Primary.
 7. Elija Continue para realizar la conmutación por error al nodo principal, o bien Cancel para cancelar la operación y no realizar la conmutación por error al nodo principal.

Durante el proceso de conmutación por error, la consola seguirá mostrando el estado del nodo como disponible. Para realizar un seguimiento del progreso de la prueba de la conmutación por error, elija Events en el panel de navegación de la consola. En la pestaña Eventos, consulte los eventos que indican que la conmutación por error se ha iniciado (`Test Failover API called`) y completado (`Recovery completed`).

Prueba de la conmutación por error automática mediante el AWS CLI

Puede probar la conmutación por error automática en cualquier clúster habilitado para zonas de disponibilidad múltiples mediante esta operación. AWS CLI `test-failover`

Parámetros

- `--replication-group-id`: obligatorio. Grupo de reproducción (en la consola, clúster) que se va a comprobar.
- `--node-group-id`: obligatorio. Nombre del grupo de nodos en el que desea probar la conmutación por error automática. Puede probar un máximo de 15 grupos de nodos en un período continuo de 24 horas.

En el siguiente ejemplo, se utiliza AWS CLI para probar la conmutación por error automática en el grupo `redis00-0003` de nodos del clúster OSS (modo de clúster activado) de Redis. `redis00`

Example Pruebe la conmutación por error automática

Para Linux, macOS o Unix:

```
aws elasticache test-failover \
```

```
--replication-group-id redis00 \  
--node-group-id redis00-0003
```

Para Windows:

```
aws elasticache test-failover ^  
--replication-group-id redis00 ^  
--node-group-id redis00-0003
```

La salida del comando anterior es similar a la siguiente.

```
{  
  "ReplicationGroup": {  
    "Status": "available",  
    "Description": "1 shard, 3 nodes (1 + 2 replicas)",  
    "NodeGroups": [  
      {  
        "Status": "available",  
        "NodeGroupMembers": [  
          {  
            "CurrentRole": "primary",  
            "PreferredAvailabilityZone": "us-west-2c",  
            "CacheNodeId": "0001",  
            "ReadEndpoint": {  
              "Port": 6379,  
              "Address":  
"redis1x3-001.7ekv3t.0001.usw2.cache.amazonaws.com"  
            },  
            "CacheClusterId": "redis1x3-001"  
          },  
          {  
            "CurrentRole": "replica",  
            "PreferredAvailabilityZone": "us-west-2a",  
            "CacheNodeId": "0001",  
            "ReadEndpoint": {  
              "Port": 6379,  
              "Address":  
"redis1x3-002.7ekv3t.0001.usw2.cache.amazonaws.com"  
            },  
            "CacheClusterId": "redis1x3-002"  
          },  
          {  
            "CurrentRole": "replica",
```

```

        "PreferredAvailabilityZone": "us-west-2b",
        "CacheNodeId": "0001",
        "ReadEndpoint": {
            "Port": 6379,
            "Address":
"redis1x3-003.7ekv3t.0001.usw2.cache.amazonaws.com"
        },
        "CacheClusterId": "redis1x3-003"
    }
],
"NodeGroupId": "0001",
"PrimaryEndpoint": {
    "Port": 6379,
    "Address": "redis1x3.7ekv3t.ng.0001.usw2.cache.amazonaws.com"
}
}
],
"ClusterEnabled": false,
"ReplicationGroupId": "redis1x3",
"SnapshotRetentionLimit": 1,
"AutomaticFailover": "enabled",
"MultiAZ": "enabled",
"SnapshotWindow": "11:30-12:30",
"SnapshottingClusterId": "redis1x3-002",
"MemberClusters": [
    "redis1x3-001",
    "redis1x3-002",
    "redis1x3-003"
],
"CacheNodeType": "cache.m3.medium",
"DataTiering": "disabled",
"PendingModifiedValues": {}
}
}

```

Para realizar un seguimiento del progreso de la conmutación por error, utilice la operación. AWS CLI `describe-events`

Para más información, consulte los siguientes temas:

- [test-failover](#) en la Referencia de los comandos de la AWS CLI .
- [describe-events](#) en la Referencia de los comandos de la AWS CLI .

Probar la conmutación por error automática mediante la API ElastiCache

Puede probar la conmutación por error automática en cualquier clúster habilitado con Multi-AZ mediante la operación de ElastiCache API. `TestFailover`

Parámetros

- `ReplicationGroupId`: obligatorio. El grupo de reproducción (en la consola, clúster) que se va a comprobar.
- `NodeGroupId`: obligatorio. Nombre del grupo de nodos en el que desea probar la conmutación por error automática. Puede probar un máximo de 15 grupos de nodos en un período continuo de 24 horas.

El ejemplo siguiente comprueba la conmutación por error automática en el grupo de nodos `redis00-0003` del grupo de reproducción (clúster, en la consola) `redis00`.

Example Prueba de la conmutación por error automática

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=TestFailover  
  &NodeGroupId=redis00-0003  
  &ReplicationGroupId=redis00  
  &Version=2015-02-02  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Timestamp=20140401T192317Z  
  &X-Amz-Credential=<credential>
```

Para realizar un seguimiento del progreso de la conmutación por error, utiliza la operación de la ElastiCache `DescribeEvents` API.

Para más información, consulte los siguientes temas:

- [TestFailover](#) en la referencia de la ElastiCache API
- [DescribeEvents](#) en la referencia ElastiCache de la API

Limitaciones de Redis OSS Multi-AZ

Tenga en cuenta las siguientes limitaciones de Redis OSS Multi-AZ:

- Multi-AZ es compatible con la versión 2.8.6 y versiones posteriores de Redis OSS.
- Los tipos de nodos T1 no admiten Redis OSS Multi-AZ.
- La replicación de Redis OSS es asíncrona. Por lo tanto, cuando un nodo principal realiza una conmutación por error a una réplica, se puede perder una pequeña cantidad de datos debido al retraso de reproducción.

Al elegir la réplica que se va a convertir en principal, ElastiCache (Redis OSS) elige la réplica con el menor retraso de replicación. En otras palabras, elija la réplica más actual. Esto ayuda a minimizar la cantidad de datos perdidos. La réplica que tiene el menor retardo de reproducción puede estar en la misma zona de disponibilidad que el nodo principal con error o en otra.

- Al pasar manualmente las réplicas de lectura a principales en Redis OSS (modo de clúster desactivado), solo podrá hacerlo cuando las multizona de disponibilidad y la conmutación por error automática estén deshabilitadas. Para promocionar una réplica de lectura a principal, siga estos pasos:
 1. Deshabilite Multi-AZ en el clúster.
 2. Deshabilite la conmutación por error automática en el clúster. Para ello, utilice la consola OSS de Redis desactivando la casilla de verificación Conmutación automática por error del grupo de replicación. Para ello, puede configurar la AWS CLI `AutomaticFailoverEnabled` propiedad en `false` al llamar a la `ModifyReplicationGroup` operación.
 3. Promocione la réplica de lectura a principal.
 4. Vuelva a habilitar Multi-AZ.
- ElastiCache (Redis OSS) Los archivos multizona de disponibilidad y los archivos solo adjuntos (AOF) se excluyen mutuamente. Si habilita una opción, no puede habilitar la otra.
- El error de un nodo puede ser provocado por el improbable caso de que deje de funcionar una zona de disponibilidad completa. En este caso, la réplica que sustituye a la principal con error se creará únicamente si hay una copia de seguridad de la zona de disponibilidad. Por ejemplo, considere la posibilidad de un grupo de reproducción con el principal en AZ-a y las réplicas en AZ-b y AZ-c. Si el principal falla, la réplica con el menor retardo de reproducción se promociona al clúster principal. A continuación, ElastiCache crea una nueva réplica en AZ-A (donde se encontraba el archivo principal que falló) solo cuando AZ-a esté disponible de nuevo.
- Un reinicio iniciado por un cliente de un principal no desencadena la conmutación por error automática. Otros reinicios y errores sí activan la conmutación por error automática.

- Cuando se reinicia el principal, sus datos se borran cuando vuelve a estar en línea. Cuando las réplicas de lectura ven el clúster principal borrado, borran sus copias de los datos, lo que provoca una pérdida de datos.
- Después de la promoción de una réplica de lectura, las otras réplicas se sincronizan con el nuevo principal. Después de la sincronización inicial, el contenido de las réplicas se elimina y sincronizan los datos del nuevo principal. Este proceso de sincronización provoca una breve interrupción, durante la cual no se puede acceder a las réplicas. El proceso de sincronización también provoca un aumento de carga temporal en el principal mientras se sincroniza con las réplicas. Este comportamiento es nativo del OSS de Redis y no es exclusivo de Multi-AZ. ElastiCache Para obtener más información sobre este comportamiento de Redis OSS, consulte [Replicación en el sitio web](#) de Redis OSS.

 Important

En el caso de la versión 2.8.22 y posteriores de Redis OSS, no puede crear réplicas externas.

Para las versiones de Redis OSS anteriores a la 2.8.22, le recomendamos que no conecte una réplica externa de Redis OSS a un clúster ElastiCache (Redis OSS) que esté habilitado para Multi-AZ. Esta configuración no compatible puede crear problemas que impidan ElastiCache realizar correctamente la recuperación y la conmutación por error. Para conectar una réplica externa de Redis OSS a un ElastiCache clúster, asegúrese de que Multi-AZ no esté activado antes de realizar la conexión.

Cómo se implementan la sincronización y la copia de seguridad

Todas las versiones compatibles de Redis OSS admiten la copia de seguridad y la sincronización entre los nodos principal y de réplica. Sin embargo, la forma en que se implementan la copia de seguridad y la sincronización varía según la versión de Redis OSS.

Redis OSS versión 2.8.22 y posteriores

La replicación de Redis OSS, en las versiones 2.8.22 y posteriores, permite elegir entre dos métodos. Para obtener más información, consulte [Versiones de Redis OSS anteriores a la 2.8.22 y Instantánea y restauración](#).

Durante el proceso sin ramificaciones, si la carga de escritura es elevada, las operaciones de escritura en el clúster se retrasan para garantizar que no se acumulan demasiados cambios que impidan el éxito del proceso de instantánea.

Versiones de Redis OSS anteriores a la 2.8.22

La copia de seguridad y la sincronización de Redis OSS en las versiones anteriores a la 2.8.22 son un proceso de tres pasos.

1. Las ramificaciones y en el proceso en segundo plano, serializan los datos del clúster en disco. Esto crea una instantánea. point-in-time
2. En primer plano, se acumula un log de cambios en el búfer de salida del cliente.

Important

Si el log de cambios supera el tamaño del búfer de salida del cliente, la copia de seguridad o la sincronización fallarán. Para obtener más información, consulte [Asegurarse de tener suficiente memoria para crear una instantánea de Redis OSS](#).

3. Por último, transmita los datos de caché y, a continuación, el log de cambios al nodo de réplica.

Creación de un grupo de replicación de Redis OSS

Dispone de las siguientes opciones para crear un clúster con nodos de réplica. Una de ellas se aplica cuando ya tiene un clúster de Redis OSS (modo de clúster desactivado) disponible que no esté asociado a ningún clúster que tenga réplicas para utilizarlas como nodo principal. El otro se aplica cuando necesita crear un nodo principal con el clúster y réplicas de lectura. Actualmente, se debe crear un clúster de Redis OSS (modo de clúster habilitado) desde cero.

Opción 1: [Creación de un grupo de replicación mediante un clúster de Redis OSS \(modo de clúster desactivado\) disponible](#)

Utilice esta opción para aprovechar un clúster Redis OSS (modo de clúster desactivado) existente de un solo nodo. Podrá especificar este nodo existente como el nodo primario en el clúster nuevo y, a continuación, agregar individualmente de 1 a 5 réplicas de lectura al clúster. Si el clúster existente está activo, las réplicas de lectura se sincronizan con él conforme se creen. Consulte [Creación de un grupo de replicación mediante un clúster de Redis OSS \(modo de clúster desactivado\) disponible](#).

Important

No puede crear un clúster de Redis OSS (habilitado para el modo de clúster) con un clúster existente. Para crear un clúster de Redis OSS (modo de clúster habilitado) (API/CLI: grupo de replicación) mediante la ElastiCache consola, consulte. [Creación de un clúster de Redis OSS \(modo de clúster activado\) \(consola\)](#)

Opción 2: [Creación de un grupo de replicación de Redis OSS desde cero](#)

Utilice esta opción si aún no tiene un clúster de Redis OSS (modo de clúster desactivado) disponible para usarlo como nodo principal del clúster, o si desea crear un clúster de Redis OSS (modo de clúster activado). Consulte [Creación de un grupo de replicación de Redis OSS desde cero](#).

Creación de un grupo de replicación mediante un clúster de Redis OSS (modo de clúster desactivado) disponible

Un clúster disponible es un clúster de Redis OSS existente de un solo nodo. Actualmente, Redis OSS (modo de clúster activado) no admite la creación de un clúster con réplicas mediante un clúster de nodo único disponible. Si desea crear un clúster de Redis OSS (modo de clúster activado), consulte. [Creación de un clúster de Redis OSS \(modo de clúster activado\) \(consola\)](#)

El siguiente procedimiento solo se puede utilizar si tiene un clúster de un solo nodo de Redis OSS (modo de clúster desactivado). Este nodo de clúster pasa a ser el nodo principal en el nuevo clúster. Si no tiene un clúster de Redis OSS (modo de clúster desactivado) que pueda utilizar como principal del nuevo clúster, consulte. [Creación de un grupo de replicación de Redis OSS desde cero](#)

Creación de un grupo de replicación mediante un clúster de OSS de Redis disponible (consola)

Consulte el tema [Uso de AWS Management Console](#).

Crear un grupo de replicación mediante un clúster de caché de Redis OSS disponible (AWS CLI)

Hay dos pasos para crear un grupo de replicación con réplicas de lectura cuando se utiliza un clúster de caché OSS de Redis disponible para el principal cuando se utiliza el AWS CLI

Al utilizar el AWS CLI, se crea un grupo de replicación que especifica el nodo independiente disponible como nodo principal del clúster `--primary-cluster-id` y la cantidad de nodos que desea en el clúster mediante el comando CLI `create-replication-group`. Incluya los siguientes parámetros.

`--replication-group-id`

El nombre del grupo de replicación que va a crear. El valor de este parámetro se usa como la base de los nombres de los nodos añadidos con un número de tres dígitos secuencial añadido al final de `--replication-group-id`. Por ejemplo, `sample-repl-group-001`.

Las restricciones de denominación de los grupos de replicación de Redis OSS (modo de clúster desactivado) son las siguientes:

- Deben contener entre 1 y 40 caracteres alfanuméricos o guiones.
- Deben comenzar por una letra.
- No pueden contener dos guiones consecutivos.
- No pueden terminar con un guion.

--replication-group-description

Descripción del grupo de replicación.

--num-node-groups

El número de nodos que desea en este clúster. Este valor incluye el nodo principal. Este parámetro tiene un valor máximo de seis.

--primary-cluster-id

Nombre del nodo del clúster de Redis OSS (modo de clúster desactivado) disponible que desea que sea el nodo principal de este grupo de replicación.

El siguiente comando crea el grupo de replicación `sample-repl-group` utilizando el clúster Redis OSS (modo de clúster desactivado) disponible `redis01` como nodo principal del grupo de replicación. Crea dos nuevos nodos que son réplicas de lectura. La configuración de `redis01` (es decir, grupo de parámetros, grupo de seguridad, tipo de nodo, versión del motor, etc.) se aplicará a todos los nodos del grupo de reproducción.

Para Linux, macOS o Unix:

```
aws elasticache create-replication-group \  
  --replication-group-id sample-repl-group \  
  --replication-group-description "demo cluster with replicas" \  
  --num-cache-clusters 3 \  
  --primary-cluster-id redis01
```

Para Windows:

```
aws elasticache create-replication-group ^  
  --replication-group-id sample-repl-group ^  
  --replication-group-description "demo cluster with replicas" ^  
  --num-cache-clusters 3 ^  
  --primary-cluster-id redis01
```

Para obtener información y parámetros adicionales que desee utilizar, consulte el AWS CLI [tema `create-replication-group`](#).

A continuación, añada réplicas de lectura al grupo de replicación.

Después de haber creado el grupo de reproducción, agregue de una a cinco réplicas de lectura con el comando `create-cache-cluster`, asegurándose de incluir los siguientes parámetros.

`--cache-cluster-id`

El nombre del clúster que va a añadir al grupo de replicación.

Las restricciones para la asignación de nombres de clúster son las siguientes:

- Deben contener entre 1 y 40 caracteres alfanuméricos o guiones.
- Deben comenzar por una letra.
- No pueden contener dos guiones consecutivos.
- No pueden terminar con un guion.

`--replication-group-id`

El nombre del grupo de replicación al que va a añadir este clúster de caché.

Repita este comando para cada réplica de lectura que desee agregar al grupo de reproducción, cambiando únicamente el valor del parámetro `--cache-cluster-id`.

 Note

Recuerde que un grupo de reproducción no puede tener más de cinco réplicas de lectura. Si intenta agregar una réplica de lectura a un grupo de reproducción que ya tenga cinco réplicas de lectura hará que la operación falle.

El siguiente código añade la réplica de lectura `my-replica01` al grupo de replicación `sample-repl-group`. La configuración del clúster principal (grupo de parámetros, grupo de seguridad, tipo de nodo, etc.) se aplicará a los nodos que se agreguen al grupo de reproducción.

Para Linux, macOS o Unix:

```
aws elasticache create-cache-cluster \  
  --cache-cluster-id my-replica01 \  
  --replication-group-id sample-repl-group
```

Para Windows:

```
aws elasticache create-cache-cluster ^
--cache-cluster-id my-replica01 ^
--replication-group-id sample-repl-group
```

El resultado de este comando será algo similar a lo que se muestra a continuación.

```
{
  "ReplicationGroup": {
    "Status": "creating",
    "Description": "demo cluster with replicas",
    "ClusterEnabled": false,
    "ReplicationGroupId": "sample-repl-group",
    "SnapshotRetentionLimit": 1,
    "AutomaticFailover": "disabled",
    "SnapshotWindow": "00:00-01:00",
    "SnapshottingClusterId": "redis01",
    "MemberClusters": [
      "sample-repl-group-001",
      "sample-repl-group-002",
      "redis01"
    ],
    "CacheNodeType": "cache.m4.large",
    "DataTiering": "disabled",
    "PendingModifiedValues": {}
  }
}
```

Para obtener información adicional, consulte los AWS CLI temas:

- [create-replication-group](#)
- [modify-replication-group](#)

Añadir réplicas a un clúster (API) de Redis OSS (modo de clúster desactivado) independiente ElastiCache

Al utilizar la ElastiCache API, se crea un grupo de replicación que especifica el nodo independiente disponible como nodo principal del clúster `PrimaryClusterId` y la cantidad de nodos que desea que haya en el clúster mediante el comando CLI `CreateReplicationGroup`. Incluya los siguientes parámetros.

ReplicationGroupId

El nombre del grupo de replicación que va a crear. El valor de este parámetro se usa como la base de los nombres de los nodos añadidos con un número de tres dígitos secuencial añadido al final de `ReplicationGroupId`. Por ejemplo, `sample-repl-group-001`.

Las restricciones de denominación de los grupos de replicación de Redis OSS (modo de clúster desactivado) son las siguientes:

- Deben contener entre 1 y 40 caracteres alfanuméricos o guiones.
- Deben comenzar por una letra.
- No pueden contener dos guiones consecutivos.
- No pueden terminar con un guion.

ReplicationGroupDescription

Descripción del clúster con réplicas.

NumCacheClusters

El número de nodos que desea en este clúster. Este valor incluye el nodo principal. Este parámetro tiene un valor máximo de seis.

PrimaryClusterId

El nombre del clúster de Redis OSS (modo de clúster desactivado) disponible que desea que sea el nodo principal de este clúster.

El siguiente comando crea el clúster con réplicas `sample-repl-group` utilizando el clúster Redis OSS (modo de clúster desactivado) disponible `redis01` como nodo principal del grupo de replicación. Crea dos nuevos nodos que son réplicas de lectura. La configuración de `redis01` (es decir, grupo de parámetros, grupo de seguridad, tipo de nodo, versión del motor, etc.) se aplicará a todos los nodos del grupo de reproducción.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=CreateReplicationGroup  
&Engine=redis  
&EngineVersion=6.0  
&ReplicationGroupDescription=Demo%20cluster%20with%20replicas  
&ReplicationGroupId=sample-repl-group  
&PrimaryClusterId=redis01  
&Version=2015-02-02
```

```
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>
```

Para obtener información adicional, consulte los temas de ElastiCache APL:

- [CreateReplicationGroup](#)
- [ModifyReplicationGroup](#)

A continuación, añada réplicas de lectura al grupo de replicación.

Después de haber creado el grupo de reproducción, agregue de una a cinco réplicas de lectura con la operación `CreateCacheCluster`, asegurándose de incluir los siguientes parámetros.

CacheClusterId

El nombre del clúster que va a añadir al grupo de replicación.

Las restricciones para la asignación de nombres de clúster son las siguientes:

- Deben contener entre 1 y 40 caracteres alfanuméricos o guiones.
- Deben comenzar por una letra.
- No pueden contener dos guiones consecutivos.
- No pueden terminar con un guion.

ReplicationGroupId

El nombre del grupo de replicación al que va a añadir este clúster de caché.

Repita esta operación para cada réplica de lectura que desee agregar al grupo de reproducción, cambiando únicamente el valor del parámetro `CacheClusterId`.

El siguiente código agrega la réplica de lectura `myReplica01` al grupo de reproducción `myRep1Group`. La configuración del clúster principal (grupo de parámetros, grupo de seguridad, tipo de nodo, etc.) se aplicará a los nodos que se agreguen al grupo de reproducción.

```
https://elasticache.us-west-2.amazonaws.com/
?Action=CreateCacheCluster
&CacheClusterId=myReplica01
```

```
&ReplicationGroupId=myReplGroup
&SignatureMethod=HmacSHA256
&SignatureVersion=4
&Version=2015-02-02
&X-Amz-Algorithm=&AWS;4-HMAC-SHA256
&X-Amz-Credential=[your-access-key-id]/20150202/us-west-2/elasticache/aws4_request
&X-Amz-Date=20150202T170651Z
&X-Amz-SignedHeaders=content-type;host;user-agent;x-amz-content-sha256;x-amz-date
&X-Amz-Signature=[signature-value]
```

Para obtener información y parámetros adicionales que quizás quieras usar, consulta el tema sobre la ElastiCache API [CreateCacheCluster](#).

Creación de un grupo de replicación de Redis OSS desde cero

A continuación, encontrará información sobre cómo crear un grupo de replicación de Redis OSS sin utilizar un clúster OSS de Redis existente como principal. Puede crear un grupo de replicación de Redis OSS (modo de clúster desactivado) o Redis OSS (modo de clúster habilitado) desde cero mediante la ElastiCache consola, la API o la API AWS CLI. ElastiCache

Antes de continuar, decida si quiere crear un grupo de replicación de Redis OSS (modo de clúster desactivado) o uno de Redis OSS (modo de clúster activado). Si desea obtener orientación para tomar su decisión, consulte [Replicación: Redis OSS \(modo de clúster desactivado\) frente a Redis OSS \(modo de clúster activado\)](#).

Temas

- [Creación de un grupo de replicación de Redis OSS \(modo de clúster desactivado\) desde cero](#)
- [Crear un grupo de replicación en Redis OSS \(modo de clúster habilitado\) desde cero](#)

Creación de un grupo de replicación de Redis OSS (modo de clúster desactivado) desde cero

Puede crear un grupo de replicación de Redis OSS (modo de clúster desactivado) desde cero mediante la ElastiCache consola AWS CLI, la o la ElastiCache API. Un grupo de replicación de Redis OSS (modo de clúster desactivado) siempre tiene un grupo de nodos, un clúster principal y hasta cinco réplicas de lectura. Los grupos de replicación de Redis OSS (modo de clúster desactivado) no admiten la partición de los datos.

Note

El límite de nodos o de particiones puede aumentarse hasta un máximo de 500 por clúster. Para solicitar un aumento del límite, consulte [AWS Service Limits](#) e incluya el tipo de instancias en la solicitud.

Para crear un grupo de replicación de Redis OSS (modo de clúster desactivado) desde cero, siga uno de los siguientes enfoques:

Crear un grupo de replicación de Redis OSS (modo de clúster desactivado) desde cero ()AWS CLI

El siguiente procedimiento crea un grupo de replicación de Redis OSS (modo de clúster desactivado) mediante el AWS CLI.

Cuando crea un grupo de replicación de Redis OSS (modo de clúster desactivado) desde cero, crea el grupo de replicación y todos sus nodos con una sola llamada al AWS CLI `create-replication-group` comando. Incluya los siguientes parámetros.

`--replication-group-id`

El nombre del grupo de replicación que va a crear.

Las restricciones de denominación de los grupos de replicación de Redis OSS (modo de clúster desactivado) son las siguientes:

- Deben contener entre 1 y 40 caracteres alfanuméricos o guiones.
- Deben comenzar por una letra.
- No pueden contener dos guiones consecutivos.
- No pueden terminar con un guion.

`--replication-group-description`

Descripción del grupo de replicación.

--num-cache-clusters

El número de nodos que desea crear con este grupo de reproducción, combinando el principal y las réplicas de lectura.

Si habilita las Multi-AZ (--automatic-failover-enabled), el valor de --num-cache-clusters debe ser de al menos 2.

--cache-node-type

El tipo de nodo para cada nodo en el grupo de replicación.

ElastiCache admite los siguientes tipos de nodos. En general, los tipos de la generación actual ofrecen más capacidad de memoria y potencia de computación a menor costo comparados con sus equivalentes de generaciones previas.

Para obtener más información sobre los detalles del rendimiento de cada tipo de nodo, consulte [Tipos de instancia de Amazon EC2](#).

--data-tiering-enabled

Establezca este parámetro si está utilizando un tipo de nodo r6gd. Si no quiere la organización de datos en niveles, defina --no-data-tiering-enabled. Para obtener más información, consulte [Organización de datos en niveles](#).

--cache-parameter-group

Especifique un grupo de parámetros que corresponda a su versión de motor. Si ejecuta Redis OSS 3.2.4 o una versión posterior, especifique el grupo de parámetros o un grupo de default.redis3.2 parámetros del que se deriva default.redis3.2 para crear un grupo de replicación de Redis OSS (modo de clúster desactivado). Para obtener más información, consulte [Parámetros específicos de Redis OSS](#).

--network-type

ipv4, ipv6 o dual-stack. Si elige la doble pila, debe establecer el parámetro --IpDiscovery en ipv4 o ipv6.

--engine

redis

--engine-version

Para conseguir el conjunto de características más completo, elija la última versión de motor.

Los nombres de los nodos se derivarán del nombre del grupo de reproducción anexando `-00#` al nombre del grupo de reproducción. Por ejemplo, al utilizar el nombre de grupo de reproducción `myReplGroup`, el nombre del principal será `myReplGroup-001`, y el de las réplicas de lectura, de `myReplGroup-002` a `myReplGroup-006`.

Si desea habilitar el cifrado en tránsito o en reposo en este grupo de replicación, agregue uno o estos dos parámetros, `--transit-encryption-enabled` y `--at-rest-encryption-enabled`, y ajústese a las siguientes condiciones.

- Su grupo de replicación debe ejecutar Redis OSS versión 3.2.6 o 4.0.10.
- El grupo de reproducción debe crearse en una Amazon VPC.
- Debe incluirse también el parámetro `--cache-subnet-group`.
- Debe incluirse el parámetro `--auth-token` con el valor de cadena especificado por el cliente para el token AUTH (contraseña) que es necesario para realizar las operaciones en este grupo de replicación.

La siguiente operación crea un grupo de replicación de Redis OSS (modo de clúster desactivado) `sample-repl-group` con tres nodos, uno principal y dos réplicas.

Para Linux, macOS o Unix:

```
aws elasticache create-replication-group \  
  --replication-group-id sample-repl-group \  
  --replication-group-description "Demo cluster with replicas" \  
  --num-cache-clusters 3 \  
  --cache-node-type cache.m4.large \  
  --engine redis
```

Para Windows:

```
aws elasticache create-replication-group ^  
  --replication-group-id sample-repl-group ^  
  --replication-group-description "Demo cluster with replicas" ^  
  --num-cache-clusters 3 ^  
  --cache-node-type cache.m4.large ^  
  --engine redis
```

La salida de este comando es similar a la siguiente.

```
{
  "ReplicationGroup": {
    "Status": "creating",
    "Description": "Demo cluster with replicas",
    "ClusterEnabled": false,
    "ReplicationGroupId": "sample-repl-group",
    "SnapshotRetentionLimit": 0,
    "AutomaticFailover": "disabled",
    "SnapshotWindow": "01:30-02:30",
    "MemberClusters": [
      "sample-repl-group-001",
      "sample-repl-group-002",
      "sample-repl-group-003"
    ],
    "CacheNodeType": "cache.m4.large",
    "DataTiering": "disabled",
    "PendingModifiedValues": {}
  }
}
```

Para obtener información y parámetros adicionales que desee utilizar, consulte el AWS CLI tema.

[create-replication-group](#)

Crear un grupo de replicación de Redis OSS (modo de clúster desactivado) desde cero (ElastiCache API)

El siguiente procedimiento crea un grupo de replicación de Redis OSS (modo de clúster desactivado) mediante la ElastiCache API.

Cuando crea un grupo de replicación de Redis OSS (modo de clúster desactivado) desde cero, crea el grupo de replicación y todos sus nodos con una sola llamada a la operación de la ElastiCache `APICreateReplicationGroup`. Incluya los siguientes parámetros.

ReplicationGroupId

El nombre del grupo de replicación que va a crear.

Las restricciones de denominación de los grupos de replicación de Redis OSS (modo de clúster activado) son las siguientes:

- Deben contener entre 1 y 40 caracteres alfanuméricos o guiones.
- Deben comenzar por una letra.

- No pueden contener dos guiones consecutivos.
- No pueden terminar con un guion.

ReplicationGroupDescription

Su descripción del grupo de reproducción.

NumCacheClusters

El número total de nodos que desea crear con este grupo de reproducción, combinando el principal y las réplicas de lectura.

Si habilita las Multi-AZ (`AutomaticFailoverEnabled=true`), el valor de `NumCacheClusters` debe ser de al menos 2.

CacheNodeType

El tipo de nodo para cada nodo en el grupo de replicación.

ElastiCache admite los siguientes tipos de nodos. En general, los tipos de la generación actual ofrecen más capacidad de memoria y potencia de computación a menor costo comparados con sus equivalentes de generaciones previas.

Para obtener más información sobre los detalles del rendimiento de cada tipo de nodo, consulte [Tipos de instancia de Amazon EC2](#).

--data-tiering-enabled

Establezca este parámetro si está utilizando un tipo de nodo `r6gd`. Si no quiere la organización de datos en niveles, defina `--no-data-tiering-enabled`. Para obtener más información, consulte [Organización de datos en niveles](#).

CacheParameterGroup

Especifique un grupo de parámetros que corresponda a su versión de motor. Si ejecuta Redis OSS 3.2.4 o una versión posterior, especifique el grupo de parámetros o un grupo de `default.redis3.2` parámetros del que se deriva `default.redis3.2` para crear un grupo de replicación de Redis OSS (modo de clúster desactivado). Para obtener más información, consulte [Parámetros específicos de Redis OSS](#).

--network-type

`ipv4`, `ipv6` o `dual-stack`. Si elige la doble pila, debe establecer el parámetro `--IpDiscovery` en `ipv4` o `ipv6`.

Motor

redis

EngineVersion

6.0

Los nombres de los nodos se derivarán del nombre del grupo de reproducción anexando `-00#` al nombre del grupo de reproducción. Por ejemplo, al utilizar el nombre de grupo de reproducción `myReplGroup`, el nombre del principal será `myReplGroup-001`, y el de las réplicas de lectura, de `myReplGroup-002` a `myReplGroup-006`.

Si desea habilitar el cifrado en tránsito o en reposo en este grupo de replicación, agregue uno o estos dos parámetros, `TransitEncryptionEnabled=true` y `AtRestEncryptionEnabled=true`, y ajústese a las siguientes condiciones.

- Su grupo de replicación debe ejecutar Redis OSS versión 3.2.6 o 4.0.10.
- El grupo de reproducción debe crearse en una Amazon VPC.
- Debe incluirse también el parámetro `CacheSubnetGroup`.
- Debe incluirse el parámetro `AuthToken` con el valor de cadena especificado por el cliente para el token AUTH (contraseña) que es necesario para realizar las operaciones en este grupo de replicación.

La siguiente operación crea el grupo de replicación Redis OSS (modo de clúster desactivado) `myReplGroup` con tres nodos, uno principal y dos réplicas.

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=CreateReplicationGroup  
  &CacheNodeType=cache.m4.large  
  &CacheParameterGroup=default.redis6.x  
  &Engine=redis  
  &EngineVersion=6.0  
  &NumCacheClusters=3  
  &ReplicationGroupDescription=test%20group  
  &ReplicationGroupId=myReplGroup  
  &Version=2015-02-02  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Timestamp=20150202T192317Z
```

```
&X-Amz-Credential=<credential>
```

Para obtener información y parámetros adicionales que desee utilizar, consulte el tema de la ElastiCache API. [CreateReplicationGroup](#)

Crear un grupo de replicación en Redis OSS (modo de clúster habilitado) desde cero

Puede crear un clúster de Redis OSS (modo de clúster habilitado) (API/CLI: grupo de replicación) mediante la ElastiCache consola AWS CLI, la o la API. ElastiCache Un grupo de replicación de Redis OSS (modo de clúster habilitado) tiene de 1 a 500 fragmentos (API/CLI: grupos de nodos), un nodo principal en cada fragmento y hasta 5 réplicas de lectura en cada fragmento. Puede crear un clúster con un mayor número de particiones y un menor número de réplicas con un total de hasta 90 nodos por clúster. Esta configuración de clúster puede variar desde 90 particiones y 0 réplicas hasta 15 particiones y 5 réplicas, que es el número máximo de réplicas permitido.

El límite de nodos o fragmentos se puede aumentar hasta un máximo de 500 por clúster si la versión del motor OSS de Redis es 5.0.6 o superior. Por ejemplo, puede elegir configurar un clúster de 500 nodos que oscila entre 83 particiones (uno primario y 5 réplicas por partición) y 500 particiones (único primario y sin réplicas). Asegúrese de que hay suficientes direcciones IP disponibles para acomodar el aumento. Algunos problemas comunes incluyen que las subredes del grupo de subredes tienen un rango CIDR demasiado pequeño o que otros clústeres comparten y utilizan considerablemente las subredes. Para obtener más información, consulte [Creación de un grupo de subredes](#).

Para las versiones inferiores a 5.0.6, el límite es de 250 por clúster.

Para solicitar un aumento del límite, consulte [AWS Service Limits](#) y elija el tipo de límite Nodes per cluster per instance type (Nodos por clúster por tipo de instancias).

Creación de un clúster en Redis OSS (modo de clúster activado)

- [Creación de un clúster de Redis OSS \(modo de clúster activado\) \(consola\)](#)
- [Crear un grupo de replicación de Redis OSS \(modo de clúster activado\) desde cero \(AWS CLI\)](#)
- [Crear un grupo de replicación en Redis OSS \(modo de clúster activado\) desde cero \(ElastiCache API\)](#)

Creación de un clúster de Redis OSS (modo de clúster activado) (consola)

Para crear un clúster de Redis OSS (modo de clúster activado), consulte [Creación de un clúster de Redis OSS \(modo de clúster activado\) \(consola\)](#). Asegúrese de habilitar el modo de clúster, Cluster Mode enabled (Scale Out) y especifique al menos dos particiones y un nodo de réplica en cada uno.

Crear un grupo de replicación de Redis OSS (modo de clúster activado) desde cero (AWS CLI)

El siguiente procedimiento crea un grupo de replicación de Redis OSS (modo de clúster activado) mediante AWS CLI

Al crear un grupo de replicación de Redis OSS (habilitado para el modo de clúster) desde cero, se crean el grupo de replicación y todos sus nodos con una sola llamada al AWS CLI `create-replication-group` comando. Incluya los siguientes parámetros.

`--replication-group-id`

El nombre del grupo de replicación que va a crear.

Las restricciones de denominación de los grupos de replicación de Redis OSS (modo de clúster activado) son las siguientes:

- Deben contener entre 1 y 40 caracteres alfanuméricos o guiones.
- Deben comenzar por una letra.
- No pueden contener dos guiones consecutivos.
- No pueden terminar con un guion.

`--replication-group-description`

Descripción del grupo de replicación.

`--cache-node-type`

El tipo de nodo para cada nodo en el grupo de replicación.

ElastiCache admite los siguientes tipos de nodos. En general, los tipos de la generación actual ofrecen más capacidad de memoria y potencia de computación a menor costo comparados con sus equivalentes de generaciones previas.

Para obtener más información sobre los detalles del rendimiento de cada tipo de nodo, consulte [Tipos de instancia de Amazon EC2](#).

`--data-tiering-enabled`

Establezca este parámetro si está utilizando un tipo de nodo `r6gd`. Si no quiere la organización de datos en niveles, defina `--no-data-tiering-enabled`. Para obtener más información, consulte [Organización de datos en niveles](#).

`--cache-parameter-group`

Especifique el grupo de `default.redis6.x.cluster.on` parámetros o un grupo de parámetros del que se deriva `default.redis6.x.cluster.on` para crear un grupo de replicación de Redis OSS (habilitado para el modo de clúster). Para obtener más información, consulte [Cambios en los parámetros de Redis OSS 6.x](#).

`--engine`

redis

`--engine-version`

3.2.4

`--num-node-groups`

El número de grupos de nodo de este grupo de reproducción. Los valores válidos son de 1 a 500.

 Note

El límite de nodos o de particiones puede aumentarse hasta un máximo de 500 por clúster. Para solicitar un aumento del límite, consulte [AWS Service Limits](#) y seleccione el tipo de límite “Nodes per cluster per instance type” (Nodos por clúster por tipo de instancias).

`--replicas-per-node-group`

El número de nodos de réplica de cada grupo de nodos. Los valores válidos son de 0 a 5.

`--network-type`

ipv4, ipv o dual-stack. Si elige la doble pila, debe establecer el parámetro `--IpDiscovery` en ipv4 o ipv6.

Si desea habilitar el cifrado en tránsito o en reposo en este grupo de replicación, agregue uno o estos dos parámetros, `--transit-encryption-enabled` y `--at-rest-encryption-enabled`, y ajústese a las siguientes condiciones.

- Su grupo de replicación debe ejecutar la versión 3.2.6 o 4.0.10 de Redis OSS.
- El grupo de reproducción debe crearse en una Amazon VPC.
- Debe incluirse también el parámetro `--cache-subnet-group`.
- Debe incluirse el parámetro `--auth-token` con el valor de cadena especificado por el cliente para el token AUTH (contraseña) que es necesario para realizar las operaciones en este grupo de replicación.

La siguiente operación crea el grupo de replicación de Redis OSS (habilitado para el modo de clúster) `sample-repl-group` con tres grupos/fragmentos de nodos (`--num-node-groups`), cada uno con tres nodos, una réplica principal y dos réplicas de lectura (`--`). `replicas-per-node-group`

Para Linux, macOS o Unix:

```
aws elasticache create-replication-group \
  --replication-group-id sample-repl-group \
  --replication-group-description "Demo cluster with replicas" \
  --num-node-groups 3 \
  --replicas-per-node-group 2 \
  --cache-node-type cache.m4.large \
  --engine redis \
  --security-group-ids SECURITY_GROUP_ID \
  --cache-subnet-group-name SUBNET_GROUP_NAME>
```

Para Windows:

```
aws elasticache create-replication-group ^
  --replication-group-id sample-repl-group ^
  --replication-group-description "Demo cluster with replicas" ^
  --num-node-groups 3 ^
  --replicas-per-node-group 2 ^
  --cache-node-type cache.m4.large ^
  --engine redis ^
  --security-group-ids SECURITY_GROUP_ID ^
  --cache-subnet-group-name SUBNET_GROUP_NAME>
```

El comando anterior genera la siguiente salida.

```
{
  "ReplicationGroup": {
    "Status": "creating",
    "Description": "Demo cluster with replicas",
    "ReplicationGroupId": "sample-repl-group",
    "SnapshotRetentionLimit": 0,
    "AutomaticFailover": "enabled",
    "SnapshotWindow": "05:30-06:30",
    "MemberClusters": [
      "sample-repl-group-0001-001",
```

```

        "sample-repl-group-0001-002",
        "sample-repl-group-0001-003",
        "sample-repl-group-0002-001",
        "sample-repl-group-0002-002",
        "sample-repl-group-0002-003",
        "sample-repl-group-0003-001",
        "sample-repl-group-0003-002",
        "sample-repl-group-0003-003"
    ],
    "PendingModifiedValues": {}
}
}

```

Al crear un grupo de replicación de Redis OSS (habilitado para el modo de clúster) desde cero, puede configurar cada fragmento del clúster mediante el `--node-group-configuration` parámetro que se muestra en el siguiente ejemplo, que configura dos grupos de nodos (consola: fragmentos). La primera partición tiene dos nodos, un principal y una réplica de lectura. La segunda partición tiene tres nodos, un principal y dos réplicas de lectura.

`--node-group-configuration`

La configuración para cada grupo de nodos. El parámetro `--node-group-configuration` consta de los siguientes campos.

- **PrimaryAvailabilityZone:** la zona de disponibilidad en la que se encuentra el nodo primario de este grupo de nodos. Si se omite este parámetro, ElastiCache elige la zona de disponibilidad para el nodo principal.

Por ejemplo: `us-west-2a`.

- **ReplicaAvailabilityZones:** una lista separada por comas de las zonas de disponibilidad en las que se encuentran las réplicas de lectura. El número de zonas de disponibilidad de esta lista debe coincidir con el valor de `ReplicaCount`. Si se omite este parámetro, ElastiCache elige las zonas de disponibilidad para los nodos de réplica.

Ejemplo: `"us-west-2a,us-west-2b,us-west-2c"`

- **ReplicaCount:** el número de nodos de réplica de este grupo de nodos.
- **Slots:** una cadena que especifica el espacio de claves para el grupo de nodos. La cadena está en formato `startKey-endKey`. Si se omite este parámetro, ElastiCache asigna las claves por igual entre los grupos de nodos.

Ejemplo: "0-4999"

La siguiente operación crea el grupo de replicación Redis OSS (habilitado para el modo de clúster) `new-group` con dos grupos/fragmentos de nodos (`()`). `--num-node-groups` A diferencia del ejemplo anterior, cada grupo de nodos se configura de forma distinta a los demás grupos de nodos (`--node-group-configuration`).

Para Linux, macOS o Unix:

```
aws elasticache create-replication-group \
  --replication-group-id new-group \
  --replication-group-description "Sharded replication group" \
  --engine redis \
  --snapshot-retention-limit 8 \
  --cache-node-type cache.m4.medium \
  --num-node-groups 2 \
  --node-group-configuration \
    "ReplicaCount=1,Slots=0-8999,PrimaryAvailabilityZone=us-east-1c',ReplicaAvailabilityZones='us-east-1b'" \
    "ReplicaCount=2,Slots=9000-16383,PrimaryAvailabilityZone='us-east-1a',ReplicaAvailabilityZones='us-east-1a','us-east-1c'"
```

Para Windows:

```
aws elasticache create-replication-group ^
  --replication-group-id new-group ^
  --replication-group-description "Sharded replication group" ^
  --engine redis ^
  --snapshot-retention-limit 8 ^
  --cache-node-type cache.m4.medium ^
  --num-node-groups 2 ^
  --node-group-configuration \
    "ReplicaCount=1,Slots=0-8999,PrimaryAvailabilityZone='us-east-1c',ReplicaAvailabilityZones='us-east-1b'" \
    "ReplicaCount=2,Slots=9000-16383,PrimaryAvailabilityZone='us-east-1a',ReplicaAvailabilityZones='us-east-1a','us-east-1c'"
```

La operación anterior genera la siguiente salida.

```
{
  "ReplicationGroup": {
    "Status": "creating",
    "Description": "Sharded replication group",
    "ReplicationGroupId": "rc-rg",
    "SnapshotRetentionLimit": 8,
    "AutomaticFailover": "enabled",
    "SnapshotWindow": "10:00-11:00",
    "MemberClusters": [
      "rc-rg-0001-001",
      "rc-rg-0001-002",
      "rc-rg-0002-001",
      "rc-rg-0002-002",
      "rc-rg-0002-003"
    ],
    "PendingModifiedValues": {}
  }
}
```

Para obtener información y parámetros adicionales que desee utilizar, consulte el tema. [AWS CLI create-replication-group](#)

Crear un grupo de replicación en Redis OSS (modo de clúster activado) desde cero (ElastiCache API)

El siguiente procedimiento crea un grupo de replicación de Redis OSS (modo de clúster activado) mediante la ElastiCache API.

Al crear un grupo de replicación de Redis OSS (habilitado para el modo de clúster) desde cero, se crean el grupo de replicación y todos sus nodos con una sola llamada a la operación de la ElastiCache API `CreateReplicationGroup`. Incluya los siguientes parámetros.

ReplicationGroupId

El nombre del grupo de replicación que va a crear.

Las restricciones de denominación de los grupos de replicación de Redis OSS (modo de clúster activado) son las siguientes:

- Deben contener entre 1 y 40 caracteres alfanuméricos o guiones.
- Deben comenzar por una letra.

- No pueden contener dos guiones consecutivos.
- No pueden terminar con un guion.

ReplicationGroupDescription

Descripción del grupo de replicación.

NumNodeGroups

El número de grupos de nodo que desea crear con este grupo de reproducción. Los valores válidos son de 1 a 500.

ReplicasPerNodeGroup

El número de nodos de réplica de cada grupo de nodos. Los valores válidos son de 1 a 5.

NodeGroupConfiguration

La configuración para cada grupo de nodos. El parámetro `NodeGroupConfiguration` consta de los siguientes campos.

- `PrimaryAvailabilityZone`: la zona de disponibilidad en la que se encuentra el nodo primario de este grupo de nodos. Si se omite este parámetro, ElastiCache elige la zona de disponibilidad para el nodo principal.

Por ejemplo: `us-west-2a`.

- `ReplicaAvailabilityZones`: una lista de las zonas de disponibilidad en las que se encuentran las réplicas de lectura. El número de zonas de disponibilidad de esta lista debe coincidir con el valor de `ReplicaCount`. Si se omite este parámetro, ElastiCache elige las zonas de disponibilidad para los nodos de réplica.
- `ReplicaCount`: el número de nodos de réplica de este grupo de nodos.
- `Slots`: una cadena que especifica el espacio de claves para el grupo de nodos. La cadena está en formato `startKey-endKey`. Si se omite este parámetro, ElastiCache asigna las claves por igual entre los grupos de nodos.

Ejemplo: `"0-4999"`

CacheNodeType

El tipo de nodo para cada nodo en el grupo de replicación.

ElastiCache admite los siguientes tipos de nodos. En general, los tipos de la generación actual ofrecen más capacidad de memoria y potencia de computación a menor costo comparados con sus equivalentes de generaciones previas.

Para obtener más información sobre los detalles del rendimiento de cada tipo de nodo, consulte [Tipos de instancia de Amazon EC2](#).

--data-tiering-enabled

Establezca este parámetro si está utilizando un tipo de nodo r6gd. Si no quiere la organización de datos en niveles, defina --no-data-tiering-enabled. Para obtener más información, consulte [Organización de datos en niveles](#).

CacheParameterGroup

Especifique el grupo de `default.redis6.x.cluster.on` parámetros o un grupo de parámetros del que se deriva `default.redis6.x.cluster.on` para crear un grupo de replicación de Redis OSS (habilitado para el modo de clúster). Para obtener más información, consulte [Cambios en los parámetros de Redis OSS 6.x](#).

--network-type

`ipv4`, `ipv6` o `dual-stack`. Si elige la doble pila, debe establecer el parámetro --IpDiscovery en `ipv4` o `ipv6`.

Motor

`redis`

EngineVersion

`6.0`

Si desea habilitar el cifrado en tránsito o en reposo en este grupo de replicación, agregue uno o estos dos parámetros, `TransitEncryptionEnabled=true` y `AtRestEncryptionEnabled=true`, y ajústese a las siguientes condiciones.

- Su grupo de replicación debe ejecutar la versión 3.2.6 o 4.0.10 de Redis OSS.
- El grupo de reproducción debe crearse en una Amazon VPC.
- Debe incluirse también el parámetro `CacheSubnetGroup`.
- Debe incluirse el parámetro `AuthToken` con el valor de cadena especificado por el cliente para el token AUTH (contraseña) que es necesario para realizar las operaciones en este grupo de replicación.

Se agregan saltos de línea para facilitar la lectura.

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=CreateReplicationGroup  
  &CacheNodeType=cache.m4.large  
  &CacheParameterGroup=default.redis6.xcluster.on  
  &Engine=redis  
  &EngineVersion=6.0  
  &NumNodeGroups=3  
  &ReplicasPerNodeGroup=2  
  &ReplicationGroupDescription=test%20group  
  &ReplicationGroupId=myReplGroup  
  &Version=2015-02-02  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Timestamp=20150202T192317Z  
  &X-Amz-Credential=<credential>
```

Para obtener información y parámetros adicionales que desee utilizar, consulte el tema de la API.

ElastiCache [CreateReplicationGroup](#)

Visualización de detalles de un grupo de reproducción

Puede que a veces desee ver los detalles de un grupo de reproducción. Puede usar la ElastiCache consola, el AWS CLI for ElastiCache o la ElastiCache API. El proceso de la consola es diferente para Redis OSS (modo de clúster desactivado) y Redis OSS (modo de clúster activado).

Visualización de detalles de un grupo de reproducción

- [Visualización de los detalles de un OSS de Redis \(modo de clúster desactivado\) con réplicas](#)
 - [Visualización de los detalles de un grupo de replicación de Redis OSS \(modo de clúster desactivado\) \(consola\)](#)
 - [Visualización de los detalles de un grupo de replicación de Redis OSS \(modo de clúster desactivado\) \(AWS CLI\)](#)
 - [Visualización de los detalles de un grupo de replicación \(API\) de Redis OSS \(modo de clúster desactivado\) ElastiCache](#)
- [Visualización de los detalles de un grupo de replicación: Redis OSS \(modo de clúster activado\)](#)
 - [Visualización de los detalles de un clúster de Redis OSS \(modo de clúster activado\) \(consola\)](#)
 - [Visualización de los detalles de un clúster de Redis OSS \(modo de clúster activado\) \(AWS CLI\)](#)

- [Visualización de los detalles de un clúster \(API\) de Redis OSS \(modo de clúster activado\) ElastiCache](#)
- [Visualización de detalles de un grupo de reproducción \(AWS CLI\)](#)
- [Visualización de detalles de un grupo de reproducción \(API de ElastiCache\)](#)

Visualización de los detalles de un OSS de Redis (modo de clúster desactivado) con réplicas

Puede ver los detalles de un clúster de Redis OSS (modo de clúster deshabilitado) con réplicas (API/ CLI: grupo de replicación) mediante la ElastiCache consola, el AWS CLI for ElastiCache o la API. ElastiCache

Visualización de los detalles de un clúster de Redis OSS (modo de clúster desactivado)

- [Visualización de los detalles de un grupo de replicación de Redis OSS \(modo de clúster desactivado\) \(consola\)](#)
- [Visualización de los detalles de un grupo de replicación de Redis OSS \(modo de clúster desactivado\) \(AWS CLI\)](#)
- [Visualización de los detalles de un grupo de replicación \(API\) de Redis OSS \(modo de clúster desactivado\) ElastiCache](#)

Visualización de los detalles de un grupo de replicación de Redis OSS (modo de clúster desactivado) (consola)

Para ver los detalles de un clúster de Redis OSS (modo de clúster desactivado) con réplicas que utilizan la ElastiCache consola, consulte el tema. [Visualización de los detalles de un clúster de Redis OSS \(modo de clúster desactivado\) \(consola\)](#)

Visualización de los detalles de un grupo de replicación de Redis OSS (modo de clúster desactivado) (AWS CLI)

Para ver un AWS CLI ejemplo que muestre los detalles de un grupo de replicación de Redis OSS (modo de clúster desactivado), consulte [Visualización de detalles de un grupo de reproducción \(AWS CLI\)](#).

Visualización de los detalles de un grupo de replicación (API) de Redis OSS (modo de clúster desactivado) ElastiCache

Para ver un ejemplo de ElastiCache API que muestre los detalles de un grupo de replicación de Redis OSS (modo de clúster desactivado), consulte [Visualización de detalles de un grupo de reproducción \(API de ElastiCache\)](#).

Visualización de los detalles de un grupo de replicación: Redis OSS (modo de clúster activado)

Visualización de los detalles de un clúster de Redis OSS (modo de clúster activado) (consola)

Para ver los detalles de un clúster de Redis OSS (habilitado para el modo de clúster) mediante la ElastiCache consola, consulte [Visualización de los detalles de un clúster de Redis OSS \(modo de clúster activado\) \(consola\)](#).

Visualización de los detalles de un clúster de Redis OSS (modo de clúster activado) (AWS CLI)

Para ver un ejemplo de ElastiCache CLI que muestre los detalles de un grupo de replicación de Redis OSS (modo de clúster habilitado), consulte [Visualización de detalles de un grupo de reproducción \(AWS CLI\)](#).

Visualización de los detalles de un clúster (API) de Redis OSS (modo de clúster activado) ElastiCache

Para ver un ejemplo de ElastiCache API que muestre los detalles de un grupo de replicación de Redis OSS (modo de clúster habilitado), consulte [Visualización de detalles de un grupo de reproducción \(API de ElastiCache\)](#).

Visualización de detalles de un grupo de reproducción (AWS CLI)

Puede ver los detalles de un grupo de reproducción utilizando el comando AWS CLI de la `describe-replication-groups`. Utilice los siguientes parámetros opcionales para perfeccionar las listas. La omisión de los parámetros devuelve detalles de hasta 100 grupos de reproducción.

Parámetros opcionales

- `--replication-group-id`: utilice este parámetro para enumerar los detalles de un grupo de reproducción específico. En caso de que el grupo de reproducción especificado tenga más de un grupo de nodos, los resultados se devuelven agrupados por grupo de nodos.
- `--max-items`: utilice este parámetro para limitar el número de grupos de reproducción mostrado. El valor de `--max-items` no puede ser inferior a 20 ni superior a 100.

Example

El siguiente código enumera los detalles de hasta 100 grupos de reproducción.

```
aws elasticache describe-replication-groups
```

El siguiente código enumera los detalles de `sample-repl-group`.

```
aws elasticache describe-replication-groups --replication-group-id sample-repl-group
```

El siguiente código enumera los detalles de `sample-repl-group`.

```
aws elasticache describe-replication-groups --replication-group-id sample-repl-group
```

El siguiente código enumera los detalles de hasta 25 grupos de reproducción.

```
aws elasticache describe-replication-groups --max-items 25
```

La salida de esta operación anterior tendrá un aspecto similar al siguiente (formato JSON).

```
{
  "ReplicationGroups": [
    {
      "Status": "available",
      "Description": "test",
      "NodeGroups": [
        {
          "Status": "available",
          "NodeGroupMembers": [
            {
              "CurrentRole": "primary",
              "PreferredAvailabilityZone": "us-west-2a",
              "CacheNodeId": "0001",
              "ReadEndpoint": {
                "Port": 6379,
                "Address": "rg-name-001.1abc4d.0001.usw2.cache.amazonaws.com"
              },
              "CacheClusterId": "rg-name-001"
            },
            {
              "CurrentRole": "replica",
              "PreferredAvailabilityZone": "us-west-2b",
```

```

        "CacheNodeId": "0001",
        "ReadEndpoint": {
            "Port": 6379,
            "Address": "rg-name-002.1abc4d.0001.usw2.cache.amazonaws.com"
        },
        "CacheClusterId": "rg-name-002"
    },
    {
        "CurrentRole": "replica",
        "PreferredAvailabilityZone": "us-west-2c",
        "CacheNodeId": "0001",
        "ReadEndpoint": {
            "Port": 6379,
            "Address": "rg-name-003.1abc4d.0001.usw2.cache.amazonaws.com"
        },
        "CacheClusterId": "rg-name-003"
    }
],
"NodeGroupId": "0001",
"PrimaryEndpoint": {
    "Port": 6379,
    "Address": "rg-name.1abc4d.ng.0001.usw2.cache.amazonaws.com"
}
}
],
"ReplicationGroupId": "rg-name",
"AutomaticFailover": "enabled",
"SnapshottingClusterId": "rg-name-002",
"MemberClusters": [
    "rg-name-001",
    "rg-name-002",
    "rg-name-003"
],
"PendingModifiedValues": {}
},
{
    ... some output omitted for brevity
}
]
}

```

Para obtener más información, consulte la AWS CLI para el tema de ElastiCache [describe-replication-groups](#).

Visualización de detalles de un grupo de reproducción (API de ElastiCache)

Puede ver los detalles de una reproducción utilizando la operación AWS CLI de la `DescribeReplicationGroups`. Utilice los siguientes parámetros opcionales para perfeccionar las listas. La omisión de los parámetros devuelve detalles de hasta 100 grupos de reproducción.

Parámetros opcionales

- `ReplicationGroupId`: utilice este parámetro para enumerar los detalles de un grupo de reproducción específico. En caso de que el grupo de reproducción especificado tenga más de un grupo de nodos, los resultados se devuelven agrupados por grupo de nodos.
- `MaxRecords`: utilice este parámetro para limitar el número de grupos de reproducción mostrado. El valor de `MaxRecords` no puede ser inferior a 20 ni superior a 100. El valor predeterminado es 100.

Example

El siguiente código enumera los detalles de hasta 100 grupos de reproducción.

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=DescribeReplicationGroups  
  &Version=2015-02-02  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Timestamp=20150202T192317Z  
  &X-Amz-Credential=<credential>
```

El siguiente código enumera los detalles de `myReplGroup`.

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=DescribeReplicationGroups  
  &ReplicationGroupId=myReplGroup  
  &Version=2015-02-02  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Timestamp=20150202T192317Z  
  &X-Amz-Credential=<credential>
```

El siguiente código enumera los detalles de hasta 25 clústeres.

```
https://elasticache.us-west-2.amazonaws.com/
```

```
?Action=DescribeReplicationGroups  
&MaxRecords=25  
&Version=2015-02-02  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20150202T192317Z  
&X-Amz-Credential=<credential>
```

Para obtener más información, consulte el tema de referencia de la API de ElastiCache [DescribeReplicationGroups](#).

Búsqueda de puntos de enlace del grupo de reproducción

Una aplicación puede conectarse a cualquier nodo en un grupo de reproducción, siempre que tenga el punto de conexión de DNS y el número de puerto de ese nodo. Dependiendo de si está ejecutando un grupo de replicación de Redis OSS (modo de clúster deshabilitado) o un grupo de replicación de Redis OSS (modo de clúster habilitado), le interesarán diferentes puntos finales.

Redis OSS (modo de clúster desactivado)

Los clústeres de Redis OSS (modo de clúster desactivado) con réplicas tienen tres tipos de puntos finales: el punto final principal, el punto final del lector y los puntos finales del nodo. El punto de conexión principal es un nombre de DNS que siempre se resuelve en el nodo principal del clúster. El punto de conexión principal es inmune a los cambios en el clúster, como la promoción de una réplica de lectura a la función principal. Para una actividad de escritura, recomendamos que sus aplicaciones se conecten al punto de conexión principal.

Un punto final de lectura dividirá en partes iguales las conexiones entrantes al punto final entre todas las réplicas de lectura de un clúster ElastiCache (Redis OSS). Factores adicionales como cuando la aplicación crea las conexiones o cómo la aplicación (re)utiliza las conexiones determinará la distribución del tráfico. Los puntos de enlace de lector se mantienen actualizados con los cambios del clúster en tiempo real cuando se añaden o eliminan réplicas. Puede colocar las múltiples réplicas de lectura de su clúster ElastiCache (Redis OSS) en diferentes zonas de AWS disponibilidad (AZ) para garantizar una alta disponibilidad de los puntos finales de los lectores.

Note

Un punto de conexión del lector no es un equilibrador de carga. Se trata de un registro de DNS que se resolverá en una dirección IP de uno de los nodos de réplica en forma de turno rotativo.

Para la actividad de lectura, las aplicaciones pueden conectarse también a cualquier nodo del clúster. A diferencia del punto de conexión principal, los puntos de conexión de nodo se resuelven en puntos de enlace específicos. Si realiza un cambio en su clúster, como añadir o eliminar una réplica, debe actualizar los puntos de enlace del nodo en su aplicación.

Redis OSS (modo de clúster activado)

Los clústeres de Redis OSS (modo de clúster habilitado) con réplicas, ya que tienen varios fragmentos (API/CLI: grupos de nodos), lo que significa que también tienen varios nodos principales,

tienen una estructura de punto final diferente a la de los clústeres de Redis OSS (modo de clúster deshabilitado). El OSS de Redis (modo de clúster activado) tiene un punto final de configuración que «conoce» todos los puntos finales principales y de nodo del clúster. La aplicación se conecta al punto de conexión de configuración. Cada vez que la aplicación escribe o lee desde el punto final de configuración del clúster, Redis OSS determina, entre bastidores, a qué fragmento pertenece la clave y qué punto final de ese fragmento debe utilizar. Todo es bastante transparente para su aplicación.

Puede encontrar los puntos finales de un clúster mediante la ElastiCache consola, la API o la API AWS CLI. ElastiCache

Búsqueda de puntos de enlace del grupo de reproducción

Para encontrar los puntos de enlace de su grupo de reproducción, consulte uno de los siguientes temas:

- [Búsqueda de los puntos finales de un clúster de Redis OSS \(modo de clúster desactivado\) \(consola\)](#)
- [Búsqueda de puntos finales para un clúster de Redis OSS \(modo de clúster activado\) \(consola\)](#)
- [Búsqueda de los puntos de enlace de grupos de reproducción \(AWS CLI\)](#)
- [Búsqueda de puntos finales para grupos de replicación \(\) ElastiCache API](#)

Modificación de un grupo de reproducción

Restricciones importantes

- Actualmente, ElastiCache admite modificaciones limitadas de un grupo de replicación de Redis OSS (modo de clúster habilitado), por ejemplo, cambiar la versión del motor mediante la operación API `ModifyReplicationGroup` (CLI: `modify-replication-group`). Puede modificar la cantidad de fragmentos (grupos de nodos) en un clúster de Redis OSS (modo de clúster habilitado) con la operación de API (`ModifyReplicationGroupShardConfiguration` CLI: `modify-replication-group-shard-configuration`). Para obtener más información, consulte [Escalar clústeres en Redis OSS \(modo de clúster habilitado\)](#).

Para realizar otras modificaciones en un clúster de Redis OSS (habilitado para el modo de clúster), es necesario crear un clúster en el que el nuevo clúster incorpore los cambios.

- Puede actualizar los clústeres y grupos de replicación de Redis OSS (modo de clúster desactivado) y Redis OSS (modo de clúster habilitado) a versiones de motor más recientes. Sin embargo, no se puede volver a versiones del motor anteriores, salvo si se elimina el clúster o el motor de reproducción existente y se vuelve a crear. Para obtener más información, consulte [Versiones del motor y actualizaciones](#).
- Puede actualizar un clúster existente ElastiCache (Redis OSS) que utilice el modo de clúster desactivado para que utilice el modo de clúster activado mediante la consola, la `ModifyReplicationGroup` API o el comando `modify-replication-group` CLI, como se muestra en el siguiente ejemplo. También puede seguir los pasos que se indican en [Modificación del modo de clúster](#).

Puede modificar la configuración de un clúster de Redis OSS (modo de clúster desactivado) mediante la ElastiCache consola AWS CLI, la API o la ElastiCache API. Actualmente, ElastiCache admite un número limitado de modificaciones en un grupo de replicación de Redis OSS (habilitado para el modo de clúster). Otras modificaciones requieren que cree una copia de seguridad del grupo de replicación actual y, a continuación, utilice esa copia de seguridad para iniciar un nuevo grupo de replicación de Redis OSS (habilitado para el modo de clúster).

Temas

- [Uso del AWS Management Console](#)

- [Uso del AWS CLI](#)
- [Uso de la API ElastiCache](#)

Uso del AWS Management Console

Para modificar un clúster de Redis OSS (modo de clúster desactivado), consulte [Modificación de un ElastiCache clúster](#).

Uso del AWS CLI

A continuación se AWS CLI muestran algunos ejemplos del `modify-replication-group` comando. Puede utilizar el mismo comando para realizar otras modificaciones en un grupo de reproducción.

Habilite Multi-AZ en un grupo de replicación de Redis OSS existente:

Para Linux, macOS o Unix:

```
aws elasticache modify-replication-group \  
  --replication-group-id myReplGroup \  
  --multi-az-enabled = true
```

Para Windows:

```
aws elasticache modify-replication-group ^  
  --replication-group-id myReplGroup ^  
  --multi-az-enabled
```

Modifique el modo de clúster de desactivado a habilitado:

Para modificar el modo de clúster de desactivado a habilitado, primero debe establecer el modo de clúster en compatible. El modo compatible permite que sus clientes OSS de Redis se conecten utilizando el modo de clúster activado y el modo de clúster desactivado. Después de migrar todos los clientes OSS de Redis para que utilicen el modo de clúster activado, podrá completar la configuración del modo de clúster y configurarlo como habilitado.

Para Linux, macOS o Unix:

Establezca el modo de clúster en compatible.

```
aws elasticache modify-replication-group \  
  --replication-group-id myReplGroup \  
  --cache-parameter-group-name myParameterGroupName \  
  --cluster-mode compatible
```

Establezca el modo de clúster en habilitado.

```
aws elasticache modify-replication-group \  
  --replication-group-id myReplGroup \  
  --cluster-mode enabled
```

Para Windows:

Establezca el modo de clúster en compatible.

```
aws elasticache modify-replication-group ^  
  --replication-group-id myReplGroup ^  
  --cache-parameter-group-name myParameterGroupName ^  
  --cluster-mode compatible
```

Establezca el modo de clúster en habilitado.

```
aws elasticache modify-replication-group ^  
  --replication-group-id myReplGroup ^  
  --cluster-mode enabled
```

Para obtener más información sobre el AWS CLI `modify-replication-group` comando, consulte [modify-replication-group](#) o [Modificación del modo de clúster](#) en la Guía del usuario ElastiCache (Redis OSS).

Uso de la API ElastiCache

La siguiente operación de ElastiCache API habilita la opción Multi-AZ en un grupo de replicación de Redis OSS existente. Puede utilizar la misma operación para realizar otras modificaciones en un grupo de reproducción.

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=ModifyReplicationGroup  
  &AutomaticFailoverEnabled=true  
  &Mutli-AZEnabled=true
```

```
&ReplicationGroupId=myReplGroup
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20141201T220302Z
&Version=2014-12-01
&X-Amz-Algorithm=&AWS;4-HMAC-SHA256
&X-Amz-Date=20141201T220302Z
&X-Amz-SignedHeaders=Host
&X-Amz-Expires=20141201T220302Z
&X-Amz-Credential=<credential>
&X-Amz-Signature=<signature>
```

Para obtener más información sobre el `ModifyReplicationGroup` funcionamiento de la ElastiCache API, consulte. [ModifyReplicationGroup](#)

Eliminación de un grupo de reproducción

Si ya no necesita uno de sus clústeres con réplicas (denominados grupos de replicación en API/CLI), puede eliminarlo. Al eliminar un grupo de replicación, ElastiCache elimina todos los nodos de ese grupo.

Una vez que haya iniciado esta operación, no se puede interrumpir ni cancelar.

Warning

- Al eliminar un clúster ElastiCache (RedisOSS), se conservan las instantáneas manuales. También tendrá la opción de crear una instantánea final antes de la eliminación del clúster. Por el contrario, las instantáneas de caché automáticas no se conservan.
- `CreateSnapshotse` necesita permiso para crear una instantánea final. Sin este permiso, la API llamada fallará con una `Access Denied` excepción.

Eliminación de un grupo de reproducción (consola)

Para eliminar un clúster que tiene réplicas, consulte [Eliminación de un clúster](#).

Eliminación de un grupo de reproducción de (AWS CLI)

Use el comando [delete-replication-group](#) para eliminar un grupo de reproducción.

```
aws elasticache delete-replication-group --replication-group-id my-repgroup
```

Aparecerá una pregunta que pedirá que confirme su decisión. Escriba `y` (sí) para iniciar la operación de forma inmediata. Después de que comience el proceso, este será irreversible.

```
After you begin deleting this replication group, all of its nodes will be deleted as well.
```

```
Are you sure you want to delete this replication group? [Ny]y
```

```
REPLICATIONGROUP my-repgroup My replication group deleting
```

Eliminar un grupo de replicación (ElastiCache API)

Llame a [DeleteReplicationGroup](#) con el parámetro `ReplicationGroup`.

Example

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=DeleteReplicationGroup  
&ReplicationGroupId=my-repgroup  
&Version=2014-12-01  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20141201T220302Z  
&X-Amz-Algorithm=&AWS;4-HMAC-SHA256  
&X-Amz-Date=20141201T220302Z  
&X-Amz-SignedHeaders=Host  
&X-Amz-Expires=20141201T220302Z  
&X-Amz-Credential=<credential>  
&X-Amz-Signature=<signature>
```

Note

Si establece el parámetro `RetainPrimaryCluster` en `true`, todas las réplicas de lectura se eliminarán, pero el clúster principal se conservará.

Cambio del número de réplicas

Puede aumentar o disminuir dinámicamente el número de réplicas de lectura en su grupo de replicación de Redis OSS mediante la AWS Management Console AWS CLI, la o la ElastiCache API. Si su grupo de replicación es un grupo de replicación de Redis OSS (habilitado para el modo de clúster), puede elegir qué fragmentos (grupos de nodos) desea aumentar o disminuir el número de réplicas.

Para cambiar dinámicamente el número de réplicas de su grupo de replicación de Redis OSS, elija la operación de la siguiente tabla que se adapte a su situación.

Para hacer esto	Para Redis OSS (modo de clúster activado)	Para Redis OSS (modo de clúster desactivado)
Agregar réplicas	Aumento del número de réplicas de una partición	Aumento del número de réplicas de una partición Agregar una réplica de lectura, para grupos de replicación de Redis OSS (modo de clúster deshabilitado)
Eliminar réplicas	Reducción del número de réplicas de una partición	Reducción del número de réplicas de una partición Eliminar una réplica de lectura, para grupos de replicación de Redis OSS (modo de clúster deshabilitado)

Aumento del número de réplicas de una partición

Puede aumentar el número de réplicas en un fragmento de Redis OSS (modo de clúster activado) o en un grupo de replicación de Redis OSS (modo de clúster desactivado) hasta un máximo de cinco. Puede hacerlo mediante la AWS Management Console, la o la API AWS CLI. ElastiCache

Temas

- [Usando la AWS Management Console](#)
- [Usando el AWS CLI](#)
- [Uso de la API ElastiCache](#)

Usando la AWS Management Console

El siguiente procedimiento utiliza la consola para aumentar el número de réplicas en un grupo de replications de Redis OSS (modo de clúster habilitado).

Para aumentar el número de réplicas en los fragmentos de OSS de Redis

1. [Inicie sesión AWS Management Console y abra la ElastiCache consola en https://console.aws.amazon.com/elasticache/.](https://console.aws.amazon.com/elasticache/)
2. En el panel de navegación, elija Redis OSS y, a continuación, elija el nombre del grupo de replicación al que quiere añadir las réplicas.
3. Elija el cuadro correspondiente a cada fragmento al que desea agregar réplicas.
4. Elija Add replicas (Agregar réplicas).
5. Complete la página Add Replicas to Shards (Agregar réplicas a las particiones):
 - En New number of replicas/shard (Nuevo número de réplicas por partición), introduzca el número de réplicas que desea que tengan todos los fragmentos seleccionados. Este valor debe ser mayor o igual que Current Number of Replicas per shard (Número actual de réplicas por fragmento) y menor o igual que cinco. Recomendamos dos réplicas como mínimo.
 - En el caso de las zonas de disponibilidad, elija Sin preferencia para ElastiCache elegir una zona de disponibilidad para cada nueva réplica o Especificar zonas de disponibilidad para elegir una zona de disponibilidad para cada nueva réplica.

Si elige Specify Availability Zones (Especificar las zonas de disponibilidad), utilice la lista para especificar una zona de disponibilidad para cada réplica nueva.
6. Elija Add (Agregar) para agregar las réplicas o Cancel (Cancelar) para cancelar la operación.

Usando el AWS CLI

Para aumentar el número de réplicas en un fragmento OSS de Redis, utilice el `increase-replica-count` comando con los siguientes parámetros:

- `--replication-group-id`: obligatorio. Identifica el grupo de replicación en el que desea aumentar el número de réplicas.
- `--apply-immediately` o `--no-apply-immediately`: obligatorios. Especifica si se debe aumentar el número de réplicas inmediatamente (`--apply-immediately`) o en el periodo de mantenimiento siguiente (`--no-apply-immediately`). Actualmente, no se admite `--no-apply-immediately`.
- `--new-replica-count`: opcional. Especifica el número de nodos de réplica que se desea al terminar, hasta un máximo de cinco. Utilice este parámetro para los grupos de replicación de Redis OSS (modo de clúster desactivado) en los que solo haya un grupo de nodos o un grupo de Redis OSS (modo de clúster habilitado), o en los que desee que todos los grupos de nodos tengan el mismo número de réplicas. Si este valor no es superior al número actual de réplicas del grupo de nodos, la llamada produce una excepción.
- `--replica-configuration`: opcional. Le permite establecer el número de réplicas y de zonas de disponibilidad para cada grupo de nodos de forma independiente. Utilice este parámetro para los grupos de Redis OSS (modo de clúster activado) en los que desee configurar cada grupo de nodos de forma independiente.

`--replica-configuration` tiene tres miembros opcionales:

- `NodeGroupId`: el ID de cuatro dígitos del grupo de nodos que está configurando. En el caso de los grupos de replicación de Redis OSS (modo de clúster desactivado), el identificador del fragmento es siempre `0001`. Para encontrar el ID de un grupo de nodos (fragmentos) de Redis OSS (modo de clúster activado), consulte [Búsqueda del ID de una partición](#)
- `NewReplicaCount`: el número de réplicas que desea incluir en este grupo de nodos al final de la operación. El valor debe ser superior al número actual de réplicas, hasta un máximo de cinco. Si este valor no es superior al número actual de réplicas del grupo de nodos, la llamada produce una excepción.
- `PreferredAvailabilityZones`: una lista de cadenas `PreferredAvailabilityZone` que especifica en qué zonas de disponibilidad se encuentran los nodos del grupo de reproducción. El número de valores `PreferredAvailabilityZone` debe ser igual al valor de `NewReplicaCount` más 1 para incluir el nodo principal. Si `--replica-configuration` se

omite este miembro de, ElastiCache (Redis OSS) elige la zona de disponibilidad para cada una de las réplicas nuevas.

Important

En la llamada, debe incluir el parámetro `--new-replica-count` o `--replica-configuration`, pero no ambos.

Example

En el siguiente ejemplo, se aumenta el número de réplicas del grupo de reproducción `sample-repl-group` a tres. Al finalizar el ejemplo, existirán tres réplicas en cada grupo de nodos. Este número se aplica tanto si se trata de un grupo de Redis OSS (modo de clúster desactivado) con un solo grupo de nodos como de un grupo de Redis OSS (modo de clúster activado) con varios grupos de nodos.

Para Linux, macOS o Unix:

```
aws elasticache increase-replica-count \  
  --replication-group-id sample-repl-group \  
  --new-replica-count 3 \  
  --apply-immediately
```

Para Windows:

```
aws elasticache increase-replica-count ^  
  --replication-group-id sample-repl-group ^  
  --new-replica-count 3 ^  
  --apply-immediately
```

En el siguiente ejemplo, se aumenta el número de réplicas del grupo de replicación `sample-repl-group` al valor indicado para los dos grupos de nodos especificados. Dado que hay varios grupos de nodos, se trata de un grupo de replicación de Redis OSS (modo de clúster activado). Al especificar el valor opcional `PreferredAvailabilityZones`, el número de zonas de disponibilidad indicadas debe ser igual al valor de `NewReplicaCount` más 1. Este enfoque incluye el nodo principal del grupo identificado por `NodeGroupId`.

Para Linux, macOS o Unix:

```
aws elasticache increase-replica-count \
  --replication-group-id sample-repl-group \
  --replica-configuration \
    NodeGroupId=0001,NewReplicaCount=2,PreferredAvailabilityZones=us-east-1a,us-east-1c,us-east-1b \
    NodeGroupId=0003,NewReplicaCount=3,PreferredAvailabilityZones=us-east-1a,us-east-1b,us-east-1c,us-east-1c \
  --apply-immediately
```

Para Windows:

```
aws elasticache increase-replica-count ^
  --replication-group-id sample-repl-group ^
  --replica-configuration ^
    NodeGroupId=0001,NewReplicaCount=2,PreferredAvailabilityZones=us-east-1a,us-east-1c,us-east-1b ^
    NodeGroupId=0003,NewReplicaCount=3,PreferredAvailabilityZones=us-east-1a,us-east-1b,us-east-1c,us-east-1c \
  --apply-immediately
```

Para obtener más información sobre cómo aumentar el número de réplicas mediante la CLI, consulte [increase-replica-count](#) Amazon ElastiCache Command Line Reference.

Uso de la API ElastiCache

Para aumentar el número de réplicas en un fragmento OSS de Redis, utilice la `IncreaseReplicaCount` acción con los siguientes parámetros:

- `ReplicationGroupId`: obligatorio. Identifica el grupo de replicación en el que desea aumentar el número de réplicas.
- `ApplyImmediately`: obligatorio. Especifica si se debe aumentar el número de réplicas inmediatamente (`ApplyImmediately=True`) o en el periodo de mantenimiento siguiente (`ApplyImmediately=False`). Actualmente, no se admite `ApplyImmediately=False`.
- `NewReplicaCount`: opcional. Especifica el número de nodos de réplica que se desea al terminar, hasta un máximo de cinco. Utilice este parámetro para los grupos de replicación de Redis OSS (modo de clúster desactivado) en los que solo haya un grupo de nodos, o para grupos de Redis OSS (modo de clúster activado) en los que desee que todos los grupos de nodos tengan el mismo número de réplicas. Si este valor no es superior al número actual de réplicas del grupo de nodos, la llamada produce una excepción.

- **ReplicaConfiguration**: opcional. Le permite establecer el número de réplicas y de zonas de disponibilidad para cada grupo de nodos de forma independiente. Utilice este parámetro para los grupos OSS de Redis (modo de clúster activado) en los que desee configurar cada grupo de nodos de forma independiente.

ReplicaConfiguration tiene tres miembros opcionales:

- **NodeGroupId**: el ID de cuatro dígitos del grupo de nodos que está configurando. En el caso de los grupos de replicación de Redis OSS (modo de clúster desactivado), el ID del grupo de nodos (fragmento) es siempre. 0001 Para encontrar el ID de un grupo de nodos (fragmento) de Redis OSS (modo de clúster activado), consulte. [Búsqueda del ID de una partición](#)
- **NewReplicaCount**: el número de réplicas que desea incluir en este grupo de nodos al final de la operación. El valor debe ser superior al número actual de réplicas, hasta un máximo de cinco. Si este valor no es superior al número actual de réplicas del grupo de nodos, la llamada produce una excepción.
- **PreferredAvailabilityZones**: una lista de cadenas **PreferredAvailabilityZone** que especifica en qué zonas de disponibilidad se encuentran los nodos del grupo de reproducción. El número de valores **PreferredAvailabilityZone** debe ser igual al valor de **NewReplicaCount** más 1 para incluir el nodo principal. Si **ReplicaConfiguration** se omite este miembro de, ElastiCache (Redis OSS) elige la zona de disponibilidad para cada una de las réplicas nuevas.

Important

En la llamada, debe incluir el parámetro **NewReplicaCount** o **ReplicaConfiguration**, pero no ambos.

Example

En el siguiente ejemplo, se aumenta el número de réplicas del grupo de reproducción `sample-repl-group` a tres. Al finalizar el ejemplo, existirán tres réplicas en cada grupo de nodos. Este número se aplica tanto si se trata de un grupo de Redis OSS (modo de clúster desactivado) con un solo grupo de nodos como de un grupo de Redis OSS (modo de clúster activado) con varios grupos de nodos.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=IncreaseReplicaCount
```

```
&ApplyImmediately=True
&NewReplicaCount=3
&ReplicationGroupId=sample-repl-group
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>
```

En el siguiente ejemplo, se aumenta el número de réplicas del grupo de replicación `sample-repl-group` al valor indicado para los dos grupos de nodos especificados. Dado que hay varios grupos de nodos, se trata de un grupo de replicación de Redis OSS (modo de clúster activado). Al especificar el valor opcional `PreferredAvailabilityZones`, el número de zonas de disponibilidad indicadas debe ser igual al valor de `NewReplicaCount` más 1. Este enfoque incluye el nodo principal del grupo identificado por `NodeGroupId`.

```
https://elasticache.us-west-2.amazonaws.com/
  ?Action=IncreaseReplicaCount
  &ApplyImmediately=True
  &ReplicaConfiguration.ConfigureShard.1.NodeGroupId=0001
  &ReplicaConfiguration.ConfigureShard.1.NewReplicaCount=2

  &ReplicaConfiguration.ConfigureShard.1.PreferredAvailabilityZones.PreferredAvailabilityZone.1=
east-1a

  &ReplicaConfiguration.ConfigureShard.1.PreferredAvailabilityZones.PreferredAvailabilityZone.2=
east-1c

  &ReplicaConfiguration.ConfigureShard.1.PreferredAvailabilityZones.PreferredAvailabilityZone.3=
east-1b
  &ReplicaConfiguration.ConfigureShard.2.NodeGroupId=0003
  &ReplicaConfiguration.ConfigureShard.2.NewReplicaCount=3

  &ReplicaConfiguration.ConfigureShard.2.PreferredAvailabilityZones.PreferredAvailabilityZone.1=
east-1a

  &ReplicaConfiguration.ConfigureShard.2.PreferredAvailabilityZones.PreferredAvailabilityZone.2=
east-1b

  &ReplicaConfiguration.ConfigureShard.2.PreferredAvailabilityZones.PreferredAvailabilityZone.3=
east-1c
```

```
&ReplicaConfiguration.ConfigureShard.2.PreferredAvailabilityZones.PreferredAvailabilityZone.4=  
east-1c  
  &ReplicationGroupId=sample-repl-group  
  &Version=2015-02-02  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Timestamp=20150202T192317Z  
  &X-Amz-Credential=<credential>
```

Para obtener más información sobre cómo aumentar el número de réplicas que utilizan la API, consulte la referencia [IncreaseReplicaCount](#) de la ElastiCache API de Amazon.

Reducción del número de réplicas de una partición

Puede reducir el número de réplicas en un fragmento para Redis OSS (modo de clúster activado) o en un grupo de réplicas para Redis OSS (modo de clúster desactivado):

- En el caso de Redis OSS (modo de clúster desactivado), puede reducir el número de réplicas a una si la opción Multi-AZ está habilitada y a cero si no está habilitada.
- En el caso de Redis OSS (modo de clúster activado), puede reducir el número de réplicas a cero. Sin embargo, no es posible realizar la conmutación por error a una réplica si se produce un error en el nodo principal.

Puede utilizar la AWS Management Console, la AWS CLI o la ElastiCache API para reducir el número de réplicas en un grupo de nodos (fragmento) o un grupo de replications.

Temas

- [Usando la AWS Management Console](#)
- [Usando el AWS CLI](#)
- [Uso de la API ElastiCache](#)

Usando la AWS Management Console

El siguiente procedimiento utiliza la consola para reducir el número de réplicas en un grupo de replications de Redis OSS (modo de clúster habilitado).

Para reducir el número de réplicas en un fragmento de OSS de Redis

1. [Inicie sesión AWS Management Console y abra la ElastiCache consola en https://console.aws.amazon.com/elasticache/.](https://console.aws.amazon.com/elasticache/)
2. En el panel de navegación, elija Redis OSS y, a continuación, elija el nombre del grupo de replicación del que quiere eliminar las réplicas.
3. Elija el cuadro correspondiente a cada fragmento del que desea eliminar un nodo de réplica.
4. Elija Delete replicas (Eliminar réplicas).
5. Complete la página Delete Replicas from Shards (Eliminar réplicas de particiones):
 - a. En New number of replicas/shard (Nuevo número de réplicas por partición), introduzca el número de réplicas que desea que tengan los fragmentos seleccionados. Este número debe ser mayor o igual que 1. Recomendamos dos réplicas por fragmento como mínimo.

- b. Elija Delete (Eliminar) para eliminar las réplicas o Cancel (Cancelar) para cancelar la operación.

Important

- Si no especifica los nodos de réplica que se van a eliminar, ElastiCache (Redis OSS) selecciona automáticamente los nodos de réplica para eliminarlos. Mientras lo hace, ElastiCache (Redis OSS) intenta conservar la arquitectura Multi-AZ para su grupo de replicación y, a continuación, conserva las réplicas con un retraso de replicación mínimo con respecto al principal.
- No puede eliminar los nodos primarios o el nodo primario de un grupo de reproducción. Si especifica la eliminación del nodo principal, la operación producirá un error con un evento que indica que el nodo principal se ha seleccionado para su eliminación.

Usando el AWS CLI

Para reducir el número de réplicas en un fragmento OSS de Redis, utilice el `decrease-replica-count` comando con los siguientes parámetros:

- `--replication-group-id`: obligatorio. Identifica el grupo de replicación en el que se desea reducir el número de réplicas.
- `--apply-immediately` o `--no-apply-immediately`: obligatorios. Especifica si se debe reducir el número de réplicas inmediatamente (`--apply-immediately`) o en el periodo de mantenimiento siguiente (`--no-apply-immediately`). Actualmente, no se admite `--no-apply-immediately`.
- `--new-replica-count`: opcional. Especifica el número de nodos de réplica que se desean. El valor de `--new-replica-count` debe ser un valor válido inferior al número actual de réplicas de los grupos de nodos. Para conocer los valores mínimos permitidos, consulte [Reducción del número de réplicas de una partición](#). Si el valor de `--new-replica-count` no cumple este requisito, la llamada produce un error.
- `--replicas-to-remove`: opcional. Contiene una lista de los ID de nodo que especifican los nodos de réplica que se van a eliminar.
- `--replica-configuration`: opcional. Le permite establecer el número de réplicas y de zonas de disponibilidad para cada grupo de nodos de forma independiente. Utilice este parámetro para

los grupos OSS de Redis (modo de clúster activado) en los que desee configurar cada grupo de nodos de forma independiente.

`--replica-configuration` tiene tres miembros opcionales:

- `NodeId`: el ID de cuatro dígitos del grupo de nodos que está configurando. En el caso de los grupos de replicación de Redis OSS (modo de clúster desactivado), el identificador del fragmento es siempre `0001`. Para encontrar el ID de un grupo de nodos (fragmentos) de Redis OSS (modo de clúster activado), consulte [Búsqueda del ID de una partición](#)
- `NewReplicaCount`: un parámetro opcional que especifica el número de nodos de réplica que desea. El valor de `NewReplicaCount` debe ser un valor válido inferior al número actual de réplicas de los grupos de nodos. Para conocer los valores mínimos permitidos, consulte [Reducción del número de réplicas de una partición](#). Si el valor de `NewReplicaCount` no cumple este requisito, la llamada produce un error.
- `PreferredAvailabilityZones`: una lista de cadenas `PreferredAvailabilityZone` que especifica en qué zonas de disponibilidad se encuentran los nodos del grupo de reproducción. El número de valores `PreferredAvailabilityZone` debe ser igual al valor de `NewReplicaCount` más 1 para incluir el nodo principal. Si `--replica-configuration` se omite este miembro de, ElastiCache (Redis OSS) elige la zona de disponibilidad para cada una de las réplicas nuevas.

Important

Debe incluir únicamente uno de los parámetros `--new-replica-count`, `--replicas-to-remove` o `--replica-configuration`.

Example

En el siguiente ejemplo, se utiliza `--new-replica-count` para reducir el número de réplicas del grupo de reproducción `sample-repl-group` a una. Al finalizar el ejemplo, existirá una réplica en cada grupo de nodos. Este número se aplica tanto si se trata de un grupo de Redis OSS (modo de clúster desactivado) con un solo grupo de nodos como de un grupo de Redis OSS (modo de clúster activado) con varios grupos de nodos.

Para Linux, macOS o Unix:

```
aws elasticache decrease-replica-count
```

```
--replication-group-id sample-repl-group \  
--new-replica-count 1 \  
--apply-immediately
```

Para Windows:

```
aws elasticache decrease-replica-count ^  
--replication-group-id sample-repl-group ^  
--new-replica-count 1 ^  
--apply-immediately
```

En el siguiente ejemplo, se reduce el número de réplicas del grupo de reproducción `sample-repl-group` eliminando dos réplicas especificadas (`0001` y `0003`) del grupo de nodos.

Para Linux, macOS o Unix:

```
aws elasticache decrease-replica-count \  
--replication-group-id sample-repl-group \  
--replicas-to-remove 0001,0003 \  
--apply-immediately
```

Para Windows:

```
aws elasticache decrease-replica-count ^  
--replication-group-id sample-repl-group ^  
--replicas-to-remove 0001,0003 \  
--apply-immediately
```

En el siguiente ejemplo, se utiliza `--replica-configuration` para reducir el número de réplicas del grupo de reproducción `sample-repl-group` al valor indicado para los dos grupos de nodos especificados. Dado que hay varios grupos de nodos, se trata de un grupo de replicación de Redis OSS (modo de clúster activado). Al especificar el valor opcional `PreferredAvailabilityZones`, el número de zonas de disponibilidad indicadas debe ser igual al valor de `NewReplicaCount` más 1. Este enfoque incluye el nodo principal del grupo identificado por `NodeGroupId`.

Para Linux, macOS o Unix:

```
aws elasticache decrease-replica-count \  
--replication-group-id sample-repl-group \  
--replica-configuration \  
--apply-immediately
```

```

NodeGroupId=0001,NewReplicaCount=1,PreferredAvailabilityZones=us-east-1a,us-
east-1c \
NodeGroupId=0003,NewReplicaCount=2,PreferredAvailabilityZones=us-east-1a,us-
east-1b,us-east-1c \
--apply-immediately

```

Para Windows:

```

aws elasticache decrease-replica-count ^
--replication-group-id sample-repl-group ^
--replica-configuration ^
NodeGroupId=0001,NewReplicaCount=2,PreferredAvailabilityZones=us-east-1a,us-
east-1c ^
NodeGroupId=0003,NewReplicaCount=3,PreferredAvailabilityZones=us-east-1a,us-
east-1b,us-east-1c \
--apply-immediately

```

Para obtener más información sobre cómo reducir el número de réplicas mediante la CLI, consulte [decrease-replica-count](#) la Amazon ElastiCache Command Line Reference.

Uso de la API ElastiCache

Para reducir el número de réplicas en un fragmento OSS de Redis, utilice la `DecreaseReplicaCount` acción con los siguientes parámetros:

- `ReplicationGroupId`: obligatorio. Identifica el grupo de replicación en el que se desea reducir el número de réplicas.
- `ApplyImmediately`: obligatorio. Especifica si se debe reducir el número de réplicas inmediatamente (`ApplyImmediately=True`) o en el periodo de mantenimiento siguiente (`ApplyImmediately=False`). Actualmente, no se admite `ApplyImmediately=False`.
- `NewReplicaCount`: opcional. Especifica el número de nodos de réplica que se desean. El valor de `NewReplicaCount` debe ser un valor válido inferior al número actual de réplicas de los grupos de nodos. Para conocer los valores mínimos permitidos, consulte [Reducción del número de réplicas de una partición](#). Si el valor de `--new-replica-count` no cumple este requisito, la llamada produce un error.
- `ReplicasToRemove`: opcional. Contiene una lista de los ID de nodo que especifican los nodos de réplica que se van a eliminar.
- `ReplicaConfiguration`: opcional. Contiene una lista de grupos de nodos que permite establecer el número de réplicas y de zonas de disponibilidad para cada grupo de nodos de forma

independiente. Utilice este parámetro para los grupos OSS de Redis (modo de clúster activado) en los que desee configurar cada grupo de nodos de forma independiente.

ReplicaConfiguraion tiene tres miembros opcionales:

- **NodeId:** el ID de cuatro dígitos del grupo de nodos que está configurando. En el caso de los grupos de replicación de Redis OSS (modo de clúster desactivado), el ID del grupo de nodos es siempre. 0001 Para encontrar el ID de un grupo de nodos (fragmento) de Redis OSS (modo de clúster activado), consulte. [Búsqueda del ID de una partición](#)
- **NewReplicaCount:** el número de réplicas que desea incluir en este grupo de nodos al final de la operación. El valor debe ser inferior al número actual de réplicas hasta un mínimo de 1 si se ha habilitado Multi-AZ o de 0 si Multi-AZ con conmutación por error automática no está habilitado. Si este valor no es menor que el número actual de réplicas del grupo de nodos, la llamada produce una excepción.
- **PreferredAvailabilityZones:** una lista de cadenas PreferredAvailabilityZone que especifica en qué zonas de disponibilidad se encuentran los nodos del grupo de reproducción. El número de valores PreferredAvailabilityZone debe ser igual al valor de NewReplicaCount más 1 para incluir el nodo principal. Si ReplicaConfiguration se omite este miembro de, ElastiCache (Redis OSS) elige la zona de disponibilidad para cada una de las réplicas nuevas.

Important

Debe incluir únicamente uno de los parámetros NewReplicaCount, ReplicasToRemove o ReplicaConfiguration.

Example

En el siguiente ejemplo, se utiliza NewReplicaCount para reducir el número de réplicas del grupo de reproducción sample-repl-group a una. Al finalizar el ejemplo, existirá una réplica en cada grupo de nodos. Este número se aplica tanto si se trata de un grupo de Redis OSS (modo de clúster desactivado) con un solo grupo de nodos como de un grupo de Redis OSS (modo de clúster activado) con varios grupos de nodos.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=DecreaseReplicaCount  
&ApplyImmediately=True
```

```
&NewReplicaCount=1
&ReplicationGroupId=sample-repl-group
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>
```

En el siguiente ejemplo, se reduce el número de réplicas del grupo de reproducción `sample-repl-group` eliminando dos réplicas especificadas (`0001` y `0003`) del grupo de nodos.

```
https://elasticache.us-west-2.amazonaws.com/
?Action=DecreaseReplicaCount
&ApplyImmediately=True
&ReplicasToRemove.ReplicaToRemove.1=0001
&ReplicasToRemove.ReplicaToRemove.2=0003
&ReplicationGroupId=sample-repl-group
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>
```

En el siguiente ejemplo, se utiliza `ReplicaConfiguration` para reducir el número de réplicas del grupo de reproducción `sample-repl-group` al valor indicado para los dos grupos de nodos especificados. Dado que hay varios grupos de nodos, se trata de un grupo de replicación de Redis OSS (modo de clúster activado). Al especificar el valor opcional `PreferredAvailabilityZones`, el número de zonas de disponibilidad indicadas debe ser igual al valor de `NewReplicaCount` más 1. Este enfoque incluye el nodo principal del grupo identificado por `NodeGroupId`.

```
https://elasticache.us-west-2.amazonaws.com/
?Action=DecreaseReplicaCount
&ApplyImmediately=True
&ReplicaConfiguration.ConfigureShard.1.NodeGroupId=0001
&ReplicaConfiguration.ConfigureShard.1.NewReplicaCount=1

&ReplicaConfiguration.ConfigureShard.1.PreferredAvailabilityZones.PreferredAvailabilityZone.1=
east-1a

&ReplicaConfiguration.ConfigureShard.1.PreferredAvailabilityZones.PreferredAvailabilityZone.2=
east-1c
```

```
&ReplicaConfiguration.ConfigureShard.2.NodeGroupId=0003
&ReplicaConfiguration.ConfigureShard.2.NewReplicaCount=2

&ReplicaConfiguration.ConfigureShard.2.PreferredAvailabilityZones.PreferredAvailabilityZone.1=
east-1a

&ReplicaConfiguration.ConfigureShard.2.PreferredAvailabilityZones.PreferredAvailabilityZone.2=
east-1b

&ReplicaConfiguration.ConfigureShard.2.PreferredAvailabilityZones.PreferredAvailabilityZone.4=
east-1c
  &ReplicationGroupId=sample-repl-group
  &Version=2015-02-02
  &SignatureVersion=4
  &SignatureMethod=HmacSHA256
  &Timestamp=20150202T192317Z
  &X-Amz-Credential=<credential>
```

Para obtener más información sobre cómo reducir el número de réplicas que utilizan la API, consulta la referencia [DecreaseReplicaCount](#) de la ElastiCache API de Amazon.

Agregar una réplica de lectura, para grupos de replicación de Redis OSS (modo de clúster deshabilitado)

La información del siguiente tema se aplica únicamente a los grupos de replicación de Redis OSS (modo de clúster desactivado).

A medida que aumentan las operaciones de lectura, es recomendable que las distribuya entre más nodos para reducir así la presión de lectura en estos. En este tema, encontrará información sobre cómo añadir una réplica de lectura a un clúster de Redis OSS (modo de clúster desactivado).

Un grupo de replicación de Redis OSS (modo de clúster desactivado) puede tener un máximo de cinco réplicas de lectura. Si intenta agregar una réplica de lectura a un grupo de reproducción que ya tenga cinco réplicas de lectura, la operación produce un error.

Para obtener información sobre cómo agregar réplicas a un grupo de replicación de Redis OSS (modo de clúster habilitado), consulte lo siguiente:

- [Escalar clústeres en Redis OSS \(modo de clúster habilitado\)](#)
- [Aumento del número de réplicas de una partición](#)

Puede añadir una réplica de lectura a un clúster de Redis OSS (modo de clúster desactivado) mediante la ElastiCache consola AWS CLI, la o la API. ElastiCache

Temas relacionados de

- [Adición de nodos a un clúster](#)
- [Adición de una réplica de lectura a un grupo de reproducción \(AWS CLI\)](#)
- [Adición de una réplica de lectura a un grupo de reproducción mediante la API](#)

Adición de una réplica de lectura a un grupo de reproducción (AWS CLI)

Para añadir una réplica de lectura a un grupo de replicación de Redis OSS (modo de clúster desactivado), utilice el AWS CLI `create-cache-cluster` comando con el parámetro `--replication-group-id` para especificar a qué grupo de replicación añadir el clúster (nodo).

El siguiente ejemplo crea el clúster `my-read-replica` y lo añade al grupo de replicación `my-replication-group`. Los tipos de nodos, grupos de parámetros, grupos de seguridad, periodos de mantenimiento y otros ajustes de la réplica de lectura serán los mismos que los de los demás nodos de `my-replication-group`.

Para Linux, macOS o Unix:

```
aws elasticache create-cache-cluster \  
  --cache-cluster-id my-read-replica \  
  --replication-group-id my-replication-group
```

Para Windows:

```
aws elasticache create-cache-cluster ^  
  --cache-cluster-id my-read-replica ^  
  --replication-group-id my-replication-group
```

Para obtener más información sobre cómo añadir una réplica de lectura mediante la CLI, consulte [create-cache-cluster](#) la Amazon ElastiCache Command Line Reference.

Adición de una réplica de lectura a un grupo de reproducción mediante la API

Para añadir una réplica de lectura a un grupo de replicación de Redis OSS (modo de clúster desactivado), utilice la ElastiCache `CreateCacheCluster` operación con el parámetro `ReplicationGroupId` para especificar a qué grupo de replicación añadir el clúster (nodo).

El siguiente ejemplo crea el clúster `myReadReplica` y lo añade al grupo de replicación `myReplicationGroup`. Los tipos de nodos, grupos de parámetros, grupos de seguridad, periodos de mantenimiento y otros ajustes de la réplica de lectura serán los mismos que los de los demás nodos de `myReplicationGroup`.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=CreateCacheCluster  
&CacheClusterId=myReadReplica  
&ReplicationGroupId=myReplicationGroup  
&Version=2015-02-02  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20150202T192317Z  
&X-Amz-Credential=<credential>
```

Para obtener más información sobre cómo añadir una réplica de lectura mediante la API, consulta [CreateCacheCluster](#) la referencia de la ElastiCache API de Amazon.

Eliminar una réplica de lectura, para grupos de replicación de Redis OSS (modo de clúster deshabilitado)

La información del siguiente tema se aplica únicamente a los grupos de replicación de Redis OSS (modo de clúster desactivado).

A medida que cambia el tráfico de lectura del grupo de replicación de Redis OSS, puede que desee añadir o eliminar réplicas de lectura. Eliminar un nodo de un grupo de replicación de Redis OSS (modo de clúster desactivado) equivale a eliminar un clúster, aunque existen restricciones:

- No puede eliminar el nodo principal de un grupo de reproducción. Si desea eliminar el nodo principal, haga lo siguiente:
 1. Promocione una réplica de lectura a principal. Para obtener más información acerca de cómo promocionar una réplica de lectura a principal, consulte [Convertir una réplica de lectura en principal, para grupos de replicación de Redis OSS \(modo de clúster desactivado\)](#).
 2. Elimine el principal anterior. Consulte el siguiente punto para conocer una restricción de este método.
- Si se ha habilitado Multi-AZ en un grupo de reproducción, no se puede eliminar la última réplica de lectura del grupo de reproducción. En este caso, haga lo siguiente:

1. Modifique el grupo de reproducción deshabilitando las Multi-AZ. Para obtener más información, consulte [Modificación de un grupo de reproducción](#).
2. Elimine la réplica de lectura.

Puede eliminar una réplica de lectura de un grupo de replicación de Redis OSS (modo de clúster desactivado) mediante la ElastiCache consola, el AWS CLI for ElastiCache o la ElastiCache API.

Para obtener instrucciones sobre cómo eliminar un clúster de un grupo de replicación de Redis OSS, consulte lo siguiente:

- [Usando el AWS Management Console](#)
- [Usando el AWS CLI](#)
- [Uso del ElastiCache API](#)
- [Escalar clústeres en Redis OSS \(modo de clúster habilitado\)](#)
- [Reducción del número de réplicas de una partición](#)

Convertir una réplica de lectura en principal, para grupos de replicación de Redis OSS (modo de clúster desactivado)

La información del siguiente tema se aplica únicamente a los grupos de replicación de Redis OSS (modo de clúster desactivado).

Puede convertir una réplica de lectura de Redis OSS (modo de clúster deshabilitado) en principal mediante la AWS Management Console, la AWS CLI, o la ElastiCache API. No puede convertir una réplica de lectura en principal mientras la opción Multi-AZ con conmutación por error automática esté habilitada en el grupo de replicación de Redis OSS (modo de clúster desactivado). Para convertir una réplica de Redis OSS (modo de clúster deshabilitado) en principal en un grupo de replicación multizona de disponibilidad habilitado, haga lo siguiente:

1. Modifique el grupo de reproducción para deshabilitar las Multi-AZ (esto no requiere que todos los clústeres estén en la misma zona de disponibilidad). Para obtener más información, consulte [Modificación de un grupo de reproducción](#).
2. Promocione la réplica de lectura a principal.
3. Modifique el grupo de reproducción para volver a habilitar Multi-AZ.

La opción Multi-AZ no está disponible en los grupos de replicación que ejecutan Redis OSS 2.6.13 o una versión anterior.

Utilización del AWS Management Console

El siguiente procedimiento utiliza la consola para promocionar un nodo de réplica a principal.

Para promocionar una réplica de lectura a nodo primario (consola)

1. Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en <https://console.aws.amazon.com/elasticache/>.
2. Si la réplica que desea promocionar es miembro de un grupo de replicación de Redis OSS (modo de clúster desactivado) en el que la Multi-AZ está habilitada, modifique el grupo de replicación para deshabilitar la Multi-AZ antes de continuar. Para obtener más información, consulte [Modificación de un grupo de reproducción](#).
3. Elija Redis OSS y, a continuación, en la lista de clústeres, elija el grupo de replicación que desee modificar. Este grupo de reproducción debe ejecutar el motor "Redis", no el motor "Redis en clúster", y debe tener al menos dos nodos.

4. En la lista de nodos, elija el nodo de réplica que desea promocionar a principal y, a continuación, en Actions (Acciones), elija Promote (Promocionar).
5. En el cuadro de diálogo Promote Read Replica (Promocionar réplica de lectura), haga lo siguiente:
 - a. En Apply Immediately (Aplicar inmediatamente), elija Yes (Sí) para promocionar la réplica de lectura inmediatamente, o No para promocionarla en el próximo periodo de mantenimiento del clúster.
 - b. Elija Promote para promocionar la réplica de lectura o Cancel para cancelar la operación.
6. Si el clúster tenía Multi-AZ habilitadas antes de comenzar el proceso de promoción, espere hasta que el estado del grupo de reproducción sea available (disponible) y, a continuación, modifique el clúster para volver a habilitar las Multi-AZ. Para obtener más información, consulte [Modificación de un grupo de reproducción](#).

Usando el AWS CLI

No puede promocionar una réplica de lectura a principal si el grupo de reproducción tiene Multi-AZ habilitadas. En algunos casos, la réplica que se desea promocionar puede ser miembro de un grupo de reproducción con Multi-AZ habilitadas. En estos casos, debe modificar el grupo de reproducción para deshabilitar la característica Multi-AZ antes de continuar. Esto no requiere que todos los clústeres estén en la misma zona de disponibilidad. Para obtener más información acerca de cómo modificar un grupo de reproducción, consulte [Modificación de un grupo de reproducción](#).

El siguiente AWS CLI comando modifica el grupo `sample-repl-group` de replicación y convierte a la réplica de lectura en `my-replica-1` la principal del grupo de replicación.

Para Linux, macOS o Unix:

```
aws elasticache modify-replication-group \  
  --replication-group-id sample-repl-group \  
  --primary-cluster-id my-replica-1
```

Para Windows:

```
aws elasticache modify-replication-group ^  
  --replication-group-id sample-repl-group ^  
  --primary-cluster-id my-replica-1
```

Para obtener más información sobre la modificación de un grupo de replicación, consulte [modify-replication-group](#) la Amazon ElastiCache Command Line Reference.

Uso de la ElastiCache API

No puede promocionar una réplica de lectura a principal si el grupo de reproducción tiene Multi-AZ habilitadas. En algunos casos, la réplica que se desea promocionar puede ser miembro de un grupo de reproducción con Multi-AZ habilitadas. En estos casos, debe modificar el grupo de reproducción para deshabilitar la característica Multi-AZ antes de continuar. Esto no requiere que todos los clústeres estén en la misma zona de disponibilidad. Para obtener más información acerca de cómo modificar un grupo de reproducción, consulte [Modificación de un grupo de reproducción](#).

La siguiente acción de la ElastiCache API modifica el grupo myRep1Group de replicación y convierte a la réplica de lectura en myReplica-1 la principal del grupo de replicación.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=ModifyReplicationGroup  
&ReplicationGroupId=myRep1Group  
&PrimaryClusterId=myReplica-1  
&Version=2014-12-01  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20141201T220302Z  
&X-Amz-Algorithm=&AWS;4-HMAC-SHA256  
&X-Amz-Date=20141201T220302Z  
&X-Amz-SignedHeaders=Host  
&X-Amz-Expires=20141201T220302Z  
&X-Amz-Credential=<credential>  
&X-Amz-Signature=<signature>
```

Para obtener más información sobre la modificación de un grupo de replicación, consulte [ModifyReplicationGroup](#) la referencia de la ElastiCache API de Amazon.

Administración del mantenimiento

Cada clúster y grupo de reproducción tienen un periodo de mantenimiento semanal durante el que se aplican los cambios del sistema. Si no especifica un período de mantenimiento preferido al crear o modificar un clúster o un grupo de replicación, ElastiCache asigne un período de mantenimiento de 60 minutos dentro del período de mantenimiento de su región en un día de la semana elegido al azar.

El periodo de mantenimiento de 60 minutos se elige al azar de un bloque de 8 horas por cada región. En la siguiente tabla, se muestran los bloques de tiempo de cada región desde los que se asignan los periodos predeterminados de mantenimiento. Puede elegir un periodo de mantenimiento preferido fuera del bloque del periodo de mantenimiento de la región.

Código de región	Nombre de la región	Periodo de mantenimiento de la región
ap-northeast-1	Región Asia-Pacífico (Tokio)	13:00 — 21:00 UTC
ap-northeast-2	Región Asia-Pacífico (Seúl)	12:00 — 20:00 UTC
ap-northeast-3	Región Asia-Pacífico (Osaka)	12:00 — 20:00 UTC
ap-southeast-3	Región Asia-Pacífico (Yakarta)	14:00 — 22:00 UTC
ap-south-1	Región Asia-Pacífico (Mumbai)	17:30 — 01:30 UTC
ap-southeast-1	Región Asia-Pacífico (Singapur)	14:00 — 22:00 UTC
cn-north-1	Región China (Pekín)	14:00 — 22:00 UTC
cn-northwest-1	Región China (Ningxia)	14:00 — 22:00 UTC
ap-east-1	Región de Asia-Pacífico (Hong Kong)	13:00 — 21:00 UTC
ap-southeast-2	Región Asia-Pacífico (Sídney)	12:00 — 20:00 UTC
eu-west-3	Región EU (París)	23:59–07:29 UTC
af-south-1	Región África (Ciudad del Cabo)	13:00 — 21:00 UTC
eu-central-1	Región de Europa (Fráncfort)	23:00 — 07:00 UTC
eu-west-1	Región de Europa (Irlanda)	22:00 — 06:00 UTC
eu-west-2	Región de Europa (Londres)	23:00 — 07:00 UTC
me-south-1	Región Medio Oriente (Baréin)	13:00 — 21:00 UTC

Código de región	Nombre de la región	Periodo de mantenimiento de la región
me-central-1	Región de Oriente Medio (EAU)	13:00 — 21:00 UTC
eu-south-1	Región Europa (Milán)	21:00 — 05:00 UTC
sa-east-1	Región de América del Sur (São Paulo)	01:00 — 09:00 UTC
us-east-1	Región del este de EE. UU. (Norte de Virginia)	03:00 — 11:00 UTC
us-east-2	Región del este de EE. UU. (Ohio)	04:00 a 12:00 h UTC
us-gov-west-1	AWS GovCloud (US) región	06:00–14:00 UTC
us-west-1	Región del oeste de EE. UU. (Norte de California)	06:00 — 17:00 UTC
us-west-2	Región del oeste de EE. UU (Oregón)	06:00–14:00 UTC

Cambio del periodo de mantenimiento del clúster o grupo de reproducción

La ventana de mantenimiento debe corresponder al momento de mínimo uso y, por tanto, podría ser preciso modificarla cada cierto tiempo. Puede modificar el clúster o grupo de reproducción de modo que especifique un intervalo de tiempo de hasta 24 horas durante las cuales deban llevarse a cabo todas las actividades de mantenimiento que solicite. Las modificaciones de clúster pendientes o aplazadas que ha solicitado tendrán lugar en este periodo.

Note

Si desea aplicar inmediatamente modificaciones en el tipo de nodo o mejoras del motor, utilice la AWS Management Console casilla Aplicar ahora. De lo contrario, estas modificaciones se aplicarán durante el siguiente periodo de mantenimiento programado. Para utilizar la API, consulte [modify-replication-groupo](#) [modify-cache-cluster](#).

Más información

Para obtener más información sobre el periodo de mantenimiento y de la sustitución de nodos, consulte lo siguiente:

- [ElastiCache Mantenimiento](#): preguntas frecuentes sobre el mantenimiento y el reemplazo de nodos
- [Sustitución de nodos](#): administración de la sustitución de nodos
- [Modificación de un grupo de reproducción](#): cambio del periodo de mantenimiento de un grupo de reproducción

Configuración de los parámetros de motor mediante los grupos de parámetros

Amazon ElastiCache utiliza parámetros para controlar las propiedades de tiempo de ejecución de sus nodos y clústeres. Por lo general, las versiones de motor más reciente incluyen parámetros adicionales para ofrecer compatibilidad con la funcionalidad más reciente. Para ver las tablas de parámetros, consulte [Parámetros específicos de Redis OSS](#).

Como cabe esperar, determinados valores de parámetros, como `maxmemory`, dependen del tipo de nodo y de motor. Para ver una tabla de estos valores de los parámetros por tipo de nodo, consulte [Parámetros específicos del tipo de nodo de Redis OSS](#).

Temas

- [Administración de parámetros](#)
- [Niveles de grupo de parámetros de caché](#)
- [Creación de un grupo de parámetros](#)
- [Enumeración de grupos de parámetros por nombre](#)
- [Enumeración de valores de un grupo de parámetros](#)
- [Modificación de un grupo de parámetros](#)
- [Eliminación de un grupo de parámetros](#)
- [Parámetros específicos de Memcached](#)
- [Parámetros específicos de Redis OSS](#)

Administración de parámetros

Los parámetros se agrupan en grupos de parámetros identificados para facilitar la administración de parámetros. Un grupo de parámetros representa una combinación de valores específicos de parámetros que se pasan al software del motor durante el startup. Estos valores determinan cómo se comportan los procesos del motor en cada nodo durante el tiempo de ejecución. Los valores de parámetros de un grupo de parámetros determinado se aplican a todos los nodos asociados al grupo, independientemente del clúster al que pertenezcan.

Para ajustar el rendimiento del clúster, puede modificar los valores de algunos parámetros o cambiar el grupo de parámetros del clúster.

- No puede modificar ni eliminar los grupos de parámetros predeterminados. Si necesita valores de parámetros personalizados, debe crear un grupo de parámetros personalizado.
- La familia del grupo parámetros y el clúster que va a asignar deben ser compatibles. Por ejemplo, si su clúster ejecuta Redis OSS versión 3.2.10, solo puede usar grupos de parámetros, predeterminados o personalizados, de la familia Redis OSS 3.2.
- Si cambia el grupo de parámetros de un clúster, los valores de los parámetros modificables condicionalmente deben ser los mismos tanto en los grupos de parámetros actuales como en los nuevos.
- Cuando se modifican los parámetros de un clúster, el cambio se aplica al clúster de inmediato o, con las excepciones indicadas a continuación, después de que se reinicien los nodos del clúster. Esto es cierto tanto si se modifica el propio grupo de parámetros del clúster como si se modifica el valor de un parámetro del grupo. Para determinar cuándo se aplica un cambio de parámetros determinado, consulte la columna Aplicación de los cambios en las tablas de [Parámetros específicos de Redis OSS](#).

Para obtener más información, consulte [Nodos de reinicio](#).

Cambios en los parámetros de Redis OSS (modo de clúster activado)

Si realiza cambios en los siguientes parámetros en un clúster de Redis OSS (modo de clúster activado), siga los pasos siguientes.

- activerehashing
- bases de datos

1. Cree una copia de seguridad manual de su clúster. Consulte [Copias de seguridad manuales](#).

2. Elimine el clúster de Redis OSS (modo de clúster activado). Consulte [Deleting clusters](#).
3. almacenar el clúster con la copia de seguridad y el grupo de parámetros modificados para propagar los datos en el nuevo clúster. Consulte [Restauración desde una copia de seguridad a una nueva caché](#).

Los cambios en otros parámetros no requieren esta acción.

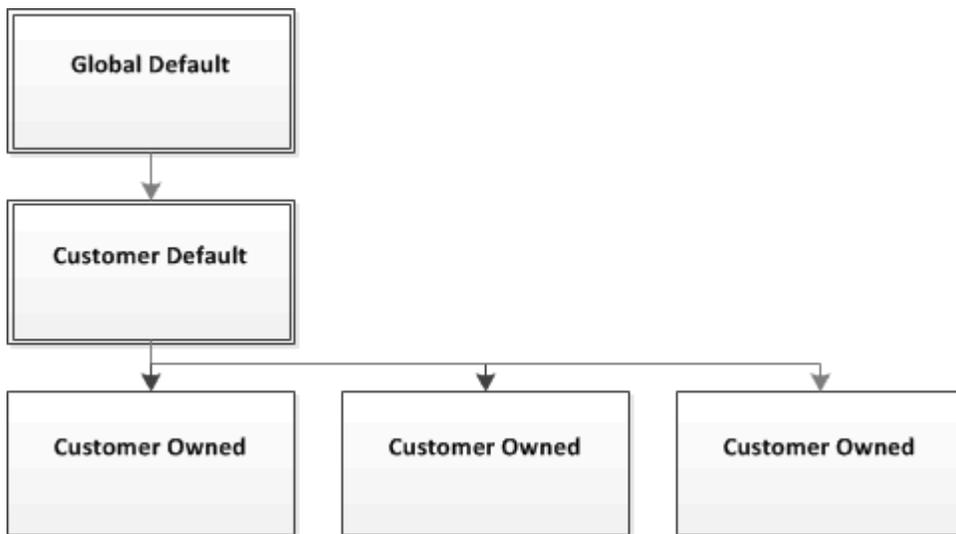
- Puede asociar grupos de parámetros a los almacenes de datos globales de Redis OSS. Los almacenes de datos globales son una colección de uno o más clústeres que abarcan regiones. AWS En este caso, el grupo de parámetros lo comparten todos los clústeres que conforman el almacén de datos global. Cualquier modificación del grupo de parámetros del clúster principal se replica en todos los clústeres restantes del almacén de datos global. Para obtener más información, consulte [Replicación entre AWS regiones mediante almacenes de datos globales](#).

Puede comprobar si un grupo de parámetros forma parte de un almacén de datos global buscando en estas ubicaciones:

- En la ElastiCache consola de la página Grupos de parámetros, el atributo global sí/no
- La `IsGlobal` propiedad sí/no de la operación de la API [CacheParameterGroup](#)

Niveles de grupo de parámetros de caché

Amazon ElastiCache tiene tres niveles de grupos de parámetros de caché como se muestra a continuación.



Niveles de grupo de parámetros de Amazon ElastiCache

Predeterminado global

Este es el grupo de parámetros raíz de nivel superior para todos los clientes de Amazon ElastiCache de la región.

Características del grupo de parámetros de caché Predeterminado global:

- Se encuentra reservado para ElastiCache y no se encuentra disponible para el cliente.

Predeterminado del cliente

Se trata de una copia del grupo de parámetros de caché Predeterminado global que se crea para el uso del cliente.

Características del grupo de parámetros de caché Predeterminado del cliente:

- Lo crea ElastiCache y es de su propiedad.
- Está disponible para el cliente para el uso como grupo de parámetros de caché para cualquier clúster que ejecute una versión de motor compatible con este grupo de parámetros de caché.
- No admite modificación del cliente.

Propiedad del cliente

Se trata de una copia del grupo de parámetros de caché predeterminado del cliente. El grupo de parámetros de caché Propiedad del cliente se crea cuando el cliente crea un grupo de parámetros de caché.

Características del grupo de parámetros de caché Propiedad del cliente:

- Lo crea el cliente y es de su propiedad.
- Puede asignarse a cualquiera de los clústeres compatibles del cliente.
- Lo puede modificar el cliente para crear un nuevo grupo de parámetros de caché personalizado.

No todos los valores de parámetros se pueden modificar. Para obtener más información, consulte [Parámetros específicos de Redis OSS](#).

Creación de un grupo de parámetros

Debe crear un nuevo grupo de parámetros si existe uno o varios parámetros que desee cambiar con respecto a los valores predeterminados. Puede crear un grupo de parámetros con la consola de ElastiCache, la AWS CLI o la API de ElastiCache.

Creación de un grupo de parámetros (consola)

En el siguiente procedimiento se muestra cómo crear un grupo de parámetros mediante la consola de ElastiCache.

Para crear un grupo de parámetros con la consola de ElastiCache

1. Inicie sesión en la AWS Management Console y abra la consola de ElastiCache en <https://console.aws.amazon.com/elasticache/>.
2. Para ver una lista de todos los grupos de parámetros disponibles, en el panel de navegación izquierdo, elija Parameter Groups.
3. Para crear un grupo de parámetros, elija Create Parameter Group.

Aparece la ventana de Create Parameter Group (Crear grupo de parámetros).

4. En la lista Family, elija la familia del grupo de parámetros que será la plantilla de su grupo de parámetros.

La familia de grupos de parámetros, como redis3.2, define los parámetros reales del grupo de parámetros, así como sus valores iniciales. La familia del grupo de parámetros debe coincidir con la versión y el motor del clúster.

5. En el cuadro Name, escriba un nombre único para este grupo de parámetros.

Al crear un clúster o modificar un grupo de parámetros de clúster, podrá elegir el grupo de parámetros por su nombre. Por lo tanto, se recomienda que el nombre sea informativo y que identifique de algún modo la familia del grupo de parámetros.

Las restricciones de nomenclatura de los grupos de parámetros son las siguientes:

- Deben comenzar por una letra ASCII.
 - Solo puede contener letras ASCII, dígitos y guiones.
 - Debe tener de 1 a 255 caracteres.
 - No pueden contener dos guiones consecutivos.
 - No pueden terminar con un guion.
6. En el cuadro Description, escriba una descripción para el grupo de parámetros.
 7. Para crear el grupo de parámetros, elija Create.

Para finalizar el proceso sin crear el grupo de parámetros, seleccione Cancel.

8. Cuando se cree el grupo de parámetros, tendrá los valores predeterminados de la familia. Para cambiar los valores predeterminados, debe modificar el grupo de parámetros. Para obtener más información, consulte [Modificación de un grupo de parámetros](#).

Creación de un grupo de parámetros (AWS CLI)

Para crear un grupo de parámetros con la AWS CLI, use el comando `create-cache-parameter-group` con los parámetros que se indican a continuación.

- `--cache-parameter-group-name`: el nombre del grupo de parámetros.

Las restricciones de nomenclatura de los grupos de parámetros son las siguientes:

- Deben comenzar por una letra ASCII.
- Solo puede contener letras ASCII, dígitos y guiones.
- Debe tener de 1 a 255 caracteres.
- No pueden contener dos guiones consecutivos.

- No pueden terminar con un guion.
- `--cache-parameter-group-family`: la familia de versión y motor del grupo de parámetros.
- `--description`: una descripción del usuario para el grupo de parámetros.

Example

En el ejemplo siguiente, se crea un grupo de parámetros denominado `myRed28` que usa la familia `redis2.8` como plantilla.

Para Linux, macOS o Unix:

```
aws elasticache create-cache-parameter-group \  
  --cache-parameter-group-name myRed28 \  
  --cache-parameter-group-family redis2.8 \  
  --description "My first parameter group"
```

Para Windows:

```
aws elasticache create-cache-parameter-group ^  
  --cache-parameter-group-name myRed28 ^  
  --cache-parameter-group-family redis2.8 ^  
  --description "My first parameter group"
```

La salida de este comando será similar a lo que se muestra a continuación.

```
{  
  "CacheParameterGroup": {  
    "CacheParameterGroupName": "myRed28",  
    "CacheParameterGroupFamily": "redis2.8",  
    "Description": "My first parameter group"  
  }  
}
```

Cuando se cree el grupo de parámetros, tendrá los valores predeterminados de la familia. Para cambiar los valores predeterminados, debe modificar el grupo de parámetros. Para obtener más información, consulte [Modificación de un grupo de parámetros](#).

Para obtener más información, consulte [create-cache-parameter-group](#).

Creación de un grupo de parámetros (API de ElastiCache)

Para crear un grupo de parámetros con la API de ElastiCache, utilice la acción `CreateCacheParameterGroup` con los parámetros que se indican a continuación.

- `ParameterGroupName`: el nombre del grupo de parámetros.

Las restricciones de nomenclatura de los grupos de parámetros son las siguientes:

- Deben comenzar por una letra ASCII.
- Solo puede contener letras ASCII, dígitos y guiones.
- Debe tener de 1 a 255 caracteres.
- No pueden contener dos guiones consecutivos.
- No pueden terminar con un guion.
- `CacheParameterGroupFamily`: la familia de versión y motor del grupo de parámetros. Por ejemplo, `redis2.8`.
- `Description`: una descripción del usuario para el grupo de parámetros.

Example

En el ejemplo siguiente, se crea un grupo de parámetros denominado `myRed28` que usa la familia `redis2.8` como plantilla.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=CreateCacheParameterGroup  
&CacheParameterGroupFamily=redis2.8  
&CacheParameterGroupName=myRed28  
&Description=My%20first%20parameter%20group  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20150202T192317Z  
&Version=2015-02-02  
&X-Amz-Credential=<credential>
```

La respuesta a esta acción será similar a lo que se muestra a continuación.

```
<CreateCacheParameterGroupResponse xmlns="http://elasticache.amazonaws.com/  
doc/2013-06-15/">  
  <CreateCacheParameterGroupResult>  
    <CacheParameterGroup>
```

```
<CacheParameterGroupName>myRed28</CacheParameterGroupName>
<CacheParameterGroupFamily>redis2.8</CacheParameterGroupFamily>
<Description>My first parameter group</Description>
</CacheParameterGroup>
</CreateCacheParameterGroupResult>
<ResponseMetadata>
  <RequestId>d8465952-af48-11e0-8d36-859edca6f4b8</RequestId>
</ResponseMetadata>
</CreateCacheParameterGroupResponse>
```

Cuando se cree el grupo de parámetros, tendrá los valores predeterminados de la familia. Para cambiar los valores predeterminados, debe modificar el grupo de parámetros. Para obtener más información, consulte [Modificación de un grupo de parámetros](#).

Para obtener más información, consulte [CreateCacheParameterGroup](#).

Enumeración de grupos de parámetros por nombre

Puede enumerar los grupos de parámetros mediante la ElastiCache consola AWS CLI, la o la ElastiCache API.

Enumeración de grupos de parámetros por nombre (consola)

El siguiente procedimiento muestra cómo ver una lista de los grupos de parámetros mediante la ElastiCache consola.

Para enumerar los grupos de parámetros mediante la ElastiCache consola

1. Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en <https://console.aws.amazon.com/elasticache/>.
2. Para ver una lista de todos los grupos de parámetros disponibles, en el panel de navegación izquierdo, elija Parameter Groups.

Enumeración de grupos de parámetros por nombre (AWS CLI)

Para generar una lista de grupos de parámetros mediante el AWS CLI, utilice el comando `describe-cache-parameter-groups`. Si proporciona un nombre de grupo de parámetros, solo se mostrará el grupo de parámetros de dicho nombre. Si no proporciona ningún nombre de grupo de parámetros, se mostrarán hasta `--max-records` grupos de parámetros. En cualquier caso, se mostrarán el nombre, la familia y la descripción del grupo de parámetros.

Example

El siguiente código de ejemplo muestra el grupo de parámetros `myRed28`.

Para Linux, macOS o Unix:

```
aws elasticache describe-cache-parameter-groups \  
  --cache-parameter-group-name myRed28
```

Para Windows:

```
aws elasticache describe-cache-parameter-groups ^  
  --cache-parameter-group-name myRed28
```

La salida de este comando tendrá un aspecto similar al siguiente y mostrará el nombre, la familia y la descripción del grupo de parámetros.

```
{
  "CacheParameterGroups": [
    {
      "CacheParameterGroupName": "myRed28",
      "CacheParameterGroupFamily": "redis2.8",
      "Description": "My first parameter group"
    }
  ]
}
```

Example

El siguiente código de ejemplo muestra el grupo de parámetros MyRed56 para los grupos de parámetros que se ejecutan en el motor OSS de Redis a partir de la versión 5.0.6. Si el grupo de parámetros forma parte de un [Replicación entre AWS regiones mediante almacenes de datos globales](#), el valor de propiedad `IsGlobal` devuelto en la salida será `Yes`.

Para Linux, macOS o Unix:

```
aws elasticache describe-cache-parameter-groups \
  --cache-parameter-group-name myRed56
```

Para Windows:

```
aws elasticache describe-cache-parameter-groups ^
  --cache-parameter-group-name myRed56
```

La salida de este comando tendrá un aspecto similar al siguiente y mostrará el nombre, la familia, `isGlobal` y la descripción del grupo de parámetros.

```
{
  "CacheParameterGroups": [
    {
      "CacheParameterGroupName": "myRed56",
      "CacheParameterGroupFamily": "redis5.0",
      "Description": "My first parameter group",
      "IsGlobal": "yes"
    }
  ]
}
```

Example

El siguiente código de ejemplo muestra hasta 10 grupos de parámetros.

```
aws elasticache describe-cache-parameter-groups --max-records 10
```

La salida JSON de este comando tendrá un aspecto similar al siguiente y mostrará el nombre, la familia, la descripción y, en el caso de redis5.6, si el grupo de parámetros forma parte de un almacén de datos global (isGlobal), para cada grupo de parámetros.

```
{
  "CacheParameterGroups": [
    {
      "CacheParameterGroupName": "custom-redis32",
      "CacheParameterGroupFamily": "redis3.2",
      "Description": "custom parameter group with reserved-memory > 0"
    },
    {
      "CacheParameterGroupName": "default.memcached1.4",
      "CacheParameterGroupFamily": "memcached1.4",
      "Description": "Default parameter group for memcached1.4"
    },
    {
      "CacheParameterGroupName": "default.redis2.6",
      "CacheParameterGroupFamily": "redis2.6",
      "Description": "Default parameter group for redis2.6"
    },
    {
      "CacheParameterGroupName": "default.redis2.8",
      "CacheParameterGroupFamily": "redis2.8",
      "Description": "Default parameter group for redis2.8"
    },
    {
      "CacheParameterGroupName": "default.redis3.2",
      "CacheParameterGroupFamily": "redis3.2",
      "Description": "Default parameter group for redis3.2"
    },
    {
      "CacheParameterGroupName": "default.redis3.2.cluster.on",
      "CacheParameterGroupFamily": "redis3.2",
      "Description": "Customized default parameter group for redis3.2 with
cluster mode on"
    },
  ],
}
```

```
{
  "CacheParameterGroupName": "default.redis5.6.cluster.on",
  "CacheParameterGroupFamily": "redis5.0",
  "Description": "Customized default parameter group for redis5.6 with
cluster mode on",
  "isGlobal": "yes"
},
]
```

Para obtener más información, consulte [describe-cache-parameter-groups](#).

Listar los grupos de parámetros por nombre (API) ElastiCache

Para generar una lista de grupos de parámetros mediante la ElastiCache API, utilice la `DescribeCacheParameterGroups` acción. Si proporciona un nombre de grupo de parámetros, solo se mostrará el grupo de parámetros de dicho nombre. Si no proporciona ningún nombre de grupo de parámetros, se mostrarán hasta `MaxRecords` grupos de parámetros. En cualquier caso, se mostrarán el nombre, la familia y la descripción del grupo de parámetros.

Example

El siguiente código de ejemplo muestra hasta 10 grupos de parámetros.

```
https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeCacheParameterGroups
&MaxRecords=10
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&Version=2015-02-02
&X-Amz-Credential=<credential>
```

La respuesta de esta acción tendrá un aspecto similar al siguiente y mostrará el nombre, la familia, la descripción y, en el caso de redis5.6, si el grupo de parámetros pertenece a un almacén de datos global (`isGlobal`), para cada grupo de parámetros.

```
<DescribeCacheParameterGroupsResponse xmlns="http://elasticache.amazonaws.com/
doc/2013-06-15/">
  <DescribeCacheParameterGroupsResult>
    <CacheParameterGroups>
```

```

<CacheParameterGroup>
  <CacheParameterGroupName>myRedis28</CacheParameterGroupName>
  <CacheParameterGroupFamily>redis2.8</CacheParameterGroupFamily>
  <Description>My custom Redis 2.8 parameter group</Description>
</CacheParameterGroup>
<CacheParameterGroup>
  <CacheParameterGroupName>myMem14</CacheParameterGroupName>
  <CacheParameterGroupFamily>memcached1.4</CacheParameterGroupFamily>
  <Description>My custom Memcached 1.4 parameter group</Description>
</CacheParameterGroup>
<CacheParameterGroup>
  <CacheParameterGroupName>myRedis56</CacheParameterGroupName>
  <CacheParameterGroupFamily>redis5.0</CacheParameterGroupFamily>
  <Description>My custom redis 5.6 parameter group</Description>
  <isGlobal>yes</isGlobal>
</CacheParameterGroup>
</CacheParameterGroups>
</DescribeCacheParameterGroupsResult>
<ResponseMetadata>
  <RequestId>3540cc3d-af48-11e0-97f9-279771c4477e</RequestId>
</ResponseMetadata>
</DescribeCacheParameterGroupsResponse>

```

Example

El siguiente código de ejemplo muestra el grupo de parámetros myRed28.

```

https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeCacheParameterGroups
&CacheParameterGroupName=myRed28
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&Version=2015-02-02
&X-Amz-Credential=<credential>

```

La respuesta de esta acción tendrá un aspecto similar al siguiente y mostrará el nombre, la familia y la descripción.

```

<DescribeCacheParameterGroupsResponse xmlns="http://elasticache.amazonaws.com/doc/2013-06-15/">
  <DescribeCacheParameterGroupsResult>
    <CacheParameterGroups>

```

```

    <CacheParameterGroup>
      <CacheParameterGroupName>myRed28</CacheParameterGroupName>
      <CacheParameterGroupFamily>redis2.8</CacheParameterGroupFamily>
      <Description>My custom Redis 2.8 parameter group</Description>
    </CacheParameterGroup>
  </CacheParameterGroups>
</DescribeCacheParameterGroupsResult>
<ResponseMetadata>
  <RequestId>3540cc3d-af48-11e0-97f9-279771c4477e</RequestId>
</ResponseMetadata>
</DescribeCacheParameterGroupsResponse>

```

Example

El siguiente código de muestra indica el grupo de parámetros myRed56.

```

https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeCacheParameterGroups
&CacheParameterGroupName=myRed56
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&Version=2015-02-02
&X-Amz-Credential=<credential>

```

La respuesta de esta acción tendrá un aspecto similar al siguiente y mostrará el nombre, la familia, la descripción y si el grupo de parámetros forma parte de un almacén de datos global (isGlobal).

```

<DescribeCacheParameterGroupsResponse xmlns="http://elasticache.amazonaws.com/doc/2013-06-15/">
  <DescribeCacheParameterGroupsResult>
    <CacheParameterGroups>
      <CacheParameterGroup>
        <CacheParameterGroupName>myRed56</CacheParameterGroupName>
        <CacheParameterGroupFamily>redis5.0</CacheParameterGroupFamily>
        <Description>My custom Redis 5.6 parameter group</Description>
        <isGlobal>yes</isGlobal>
      </CacheParameterGroup>
    </CacheParameterGroups>
  </DescribeCacheParameterGroupsResult>
  <ResponseMetadata>
    <RequestId>3540cc3d-af48-11e0-97f9-279771c4477e</RequestId>
  </ResponseMetadata>

```

```
</DescribeCacheParameterGroupsResponse>
```

Para obtener más información, consulte [DescribeCacheParameterGroups](#).

Enumeración de valores de un grupo de parámetros

Puede obtener una lista de los parámetros de un grupo de parámetros, junto con sus valores, mediante la consola de ElastiCache, la AWS CLI o la API de ElastiCache.

Enumeración de valores de un grupo de parámetros (consola)

El siguiente procedimiento muestra cómo obtener una lista de los parámetros de un grupo de parámetros, junto con sus valores, mediante la consola de ElastiCache.

Para obtener una lista de los parámetros de un grupo de parámetros, junto con sus valores, mediante la consola de ElastiCache

1. Inicie sesión en la AWS Management Console y abra la consola de ElastiCache en <https://console.aws.amazon.com/elasticache/>.
2. Para ver una lista de todos los grupos de parámetros disponibles, en el panel de navegación izquierdo, elija Parameter Groups.
3. Elija el grupo de parámetros del que desea obtener una lista de los parámetros y sus valores activando la casilla situada a la izquierda del nombre del grupo de parámetros.

Los parámetros y sus valores se mostrarán en la parte inferior de la pantalla. Debido al número de parámetros, puede que tenga que desplazarse hacia arriba y hacia abajo para encontrar el parámetro que le interesa.

Enumeración de valores de un grupo de parámetros (AWS CLI)

Para obtener una lista de los parámetros de un grupo de parámetros, junto con sus valores, mediante la AWS CLI, use el comando `describe-cache-parameters`.

Example

El siguiente código de ejemplo muestra todos los parámetros, junto con sus valores, del grupo de parámetros `myRedis28`.

Para Linux, macOS o Unix:

```
aws elasticache describe-cache-parameters \  
  --cache-parameter-group-name myRedis28
```

Para Windows:

```
aws elasticache describe-cache-parameters ^  
  --cache-parameter-group-name myRed28
```

Para obtener más información, consulte [describe-cache-parameters](#).

Enumeración de valores de un grupo de parámetros (API de ElastiCache)

Para obtener una lista de los parámetros de un grupo de parámetros, junto con sus valores, mediante la API de ElastiCache, utilice la acción DescribeCacheParameters.

Example

El siguiente código de ejemplo muestra todos los parámetros del grupo de parámetros myRed28.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=DescribeCacheParameters  
&CacheParameterGroupName=myRed28  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20150202T192317Z  
&Version=2015-02-02  
&X-Amz-Credential=<credential>
```

La respuesta a esta acción tendrá un aspecto similar al siguiente. Esta respuesta se ha truncado.

```
<DescribeCacheParametersResponse xmlns="http://elasticache.amazonaws.com/  
doc/2013-06-15/">  
  <DescribeCacheParametersResult>  
    <CacheClusterClassSpecificParameters>  
      <CacheNodeTypeSpecificParameter>  
        <DataType>integer</DataType>  
        <Source>system</Source>  
        <IsModifiable>>false</IsModifiable>  
        <Description>The maximum configurable amount of memory to use to store items,  
in megabytes.</Description>  
        <CacheNodeTypeSpecificValues>  
          <CacheNodeTypeSpecificValue>  
            <Value>1000</Value>  
            <CacheClusterClass>cache.c1.medium</CacheClusterClass>  
          </CacheNodeTypeSpecificValue>
```

```

    <CacheNodeTypeSpecificValue>
      <Value>6000</Value>
      <CacheClusterClass>cache.c1.xlarge</CacheClusterClass>
    </CacheNodeTypeSpecificValue>
    <CacheNodeTypeSpecificValue>
      <Value>7100</Value>
      <CacheClusterClass>cache.m1.large</CacheClusterClass>
    </CacheNodeTypeSpecificValue>
    <CacheNodeTypeSpecificValue>
      <Value>1300</Value>
      <CacheClusterClass>cache.m1.small</CacheClusterClass>
    </CacheNodeTypeSpecificValue>

```

...output omitted...

```

    </CacheClusterClassSpecificParameters>
  </DescribeCacheParametersResult>
  <ResponseMetadata>
    <RequestId>6d355589-af49-11e0-97f9-279771c4477e</RequestId>
  </ResponseMetadata>
</DescribeCacheParametersResponse>

```

Para obtener más información, consulte [DescribeCacheParameters](#).

Modificación de un grupo de parámetros

Important

No es posible modificar ningún grupo de parámetros predeterminado.

Es posible modificar algunos parámetros de un grupo de parámetros. Dichos valores de parámetros se aplican a los clústeres asociados al grupo de parámetros. Para obtener más información acerca de cuándo se aplica un cambio en los valores de los parámetros a un grupo de parámetros, consulte [Parámetros específicos de Redis OSS](#).

Modificación de un grupo de parámetros (consola)

El siguiente procedimiento muestra cómo cambiar el valor del `cluster-enabled` parámetro mediante la ElastiCache consola. Puede usar el mismo procedimiento para cambiar el valor de cualquier parámetro.

Para cambiar el valor de un parámetro mediante la ElastiCache consola

1. Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en <https://console.aws.amazon.com/elasticache/>.
2. Para ver una lista de todos los grupos de parámetros disponibles, en el panel de navegación izquierdo, elija Parameter Groups.
3. Elija el grupo de parámetros que desea modificar activando la casilla situada a la izquierda del nombre del grupo de parámetros.

Los parámetros del grupo de parámetros se mostrarán en la parte inferior de la pantalla. Es posible que necesite desplazarse por las páginas de la lista para ver todos los parámetros.

4. Para modificar uno o varios parámetros, elija Edit Parameters.
5. Seleccione Guardar cambios.
6. Para encontrar el nombre del parámetro que ha cambiado, consulte [Parámetros específicos de Redis OSS](#). Si tiene un clúster OSS de Redis (modo de clúster desactivado) y realiza cambios en los siguientes parámetros, debe reiniciar los nodos del clúster:

- activerehashing
- bases de datos

Para obtener más información, consulte [Nodos de reinicio](#).

Cambios en los parámetros de Redis OSS (modo de clúster activado)

Si realiza cambios en los siguientes parámetros en un clúster de Redis OSS (modo de clúster activado), siga los pasos siguientes.

- activerehashing
- bases de datos

1. Cree una copia de seguridad manual de su clúster. Consulte [Copias de seguridad manuales](#).
2. Elimine el clúster de Redis OSS (modo de clúster activado). Consulte [Deleting clusters](#).

3. Restaure el clúster con la copia de seguridad y el grupo de parámetros modificados para propagar los datos en el nuevo clúster. Consulte [Restauración desde una copia de seguridad a una nueva caché](#).

Los cambios en otros parámetros no requieren esta acción.

Modificación de un grupo de parámetros (AWS CLI)

Para cambiar el valor de un parámetro mediante el AWS CLI, utilice el comando `modify-cache-parameter-group`.

Example

Para encontrar el nombre y los valores permitidos del parámetro que desea cambiar, consulte [Parámetros específicos de Redis OSS](#).

El siguiente código de ejemplo establece el valor de dos parámetros `reserved-memory-percent` está habilitado para el clúster en el grupo de parámetros. `myredis32-on-30` Lo configuramos en `30` (30 por ciento) y habilitado `reserved-memory-percent` para clústeres para `yes` que el grupo de parámetros se pueda usar con clústeres de Redis OSS (habilitados para el modo de clúster) (grupos de replicación).

Para Linux, macOS o Unix:

```
aws elasticache modify-cache-parameter-group \
  --cache-parameter-group-name myredis32-on-30 \
  --parameter-name-values \
    ParameterName=reserved-memory-percent,ParameterValue=30 \
    ParameterName=cluster-enabled,ParameterValue=yes
```

Para Windows:

```
aws elasticache modify-cache-parameter-group ^
  --cache-parameter-group-name myredis32-on-30 ^
  --parameter-name-values ^
    ParameterName=reserved-memory-percent,ParameterValue=30 ^
    ParameterName=cluster-enabled,ParameterValue=yes
```

El resultado de este comando será algo similar a lo que se muestra a continuación.

```
{
  "CacheParameterGroupName": "my-redis32-on-30"
}
```

Para obtener más información, consulte [modify-cache-parameter-group](#).

Para encontrar el nombre del parámetro que ha cambiado, consulte [Parámetros específicos de Redis OSS](#).

Si tiene un clúster de Redis OSS (modo de clúster desactivado) y realiza cambios en los siguientes parámetros, debe reiniciar los nodos del clúster:

- activerehashing
- bases de datos

Para obtener más información, consulte [Nodos de reinicio](#).

Cambios en los parámetros de Redis OSS (modo de clúster activado)

Si realiza cambios en los siguientes parámetros en un clúster de Redis OSS (modo de clúster activado), siga los pasos siguientes.

- activerehashing
 - bases de datos
1. Cree una copia de seguridad manual de su clúster. Consulte [Copias de seguridad manuales](#).
 2. Elimine el clúster de Redis OSS (modo de clúster activado). Consulte [Deleting clusters](#).
 3. Restaure el clúster con la copia de seguridad y el grupo de parámetros modificados para propagar los datos en el nuevo clúster. Consulte [Restauración desde una copia de seguridad a una nueva caché](#).

Los cambios en otros parámetros no requieren esta acción.

Modificación de un grupo de parámetros (ElastiCache API)

Para cambiar los valores de los parámetros de un grupo de parámetros mediante la ElastiCache API, utilice la `ModifyCacheParameterGroup` acción.

Example

Para encontrar el nombre y los valores permitidos del parámetro que desea cambiar, consulte [Parámetros específicos de Redis OSS](#).

El siguiente código de ejemplo establece el valor de dos parámetros `reserved-memory-percent` está habilitado para el clúster en el grupo de parámetros. `myredis32-on-30` Lo configuramos en 30 (30 por ciento) y habilitado `reserved-memory-percent` para clústeres para `yes` que el grupo de parámetros se pueda usar con clústeres de Redis OSS (habilitados para el modo de clúster) (grupos de replicación).

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=ModifyCacheParameterGroup  
&CacheParameterGroupName=myredis32-on-30  
&ParameterNameValues.member.1.ParameterName=reserved-memory-percent  
&ParameterNameValues.member.1.ParameterValue=30  
&ParameterNameValues.member.2.ParameterName=cluster-enabled  
&ParameterNameValues.member.2.ParameterValue=yes  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20150202T192317Z  
&Version=2015-02-02  
&X-Amz-Credential=<credential>
```

Para obtener más información, consulte [ModifyCacheParameterGroup](#).

Si tiene un clúster de Redis OSS (modo de clúster desactivado) y realiza cambios en los siguientes parámetros, debe reiniciar los nodos del clúster:

- `activeresharding`
- bases de datos

Para obtener más información, consulte [Nodos de reinicio](#).

Cambios en los parámetros de Redis OSS (modo de clúster activado)

Si realiza cambios en los siguientes parámetros en un clúster de Redis OSS (modo de clúster activado), siga los pasos siguientes.

- activerehashing
 - bases de datos
1. Cree una copia de seguridad manual de su clúster. Consulte [Copias de seguridad manuales](#).
 2. Elimine el clúster de Redis OSS (modo de clúster activado). Consulte [Eliminación de un clúster](#).
 3. Restaure el clúster con la copia de seguridad y el grupo de parámetros modificados para propagar los datos en el nuevo clúster. Consulte [Restauración desde una copia de seguridad a una nueva caché](#).

Los cambios en otros parámetros no requieren esta acción.

Eliminación de un grupo de parámetros

Puede eliminar un grupo de parámetros personalizados con la consola de ElastiCache, la AWS CLI o la API de ElastiCache.

No podrá eliminar un grupo de parámetros si está asociado a un clúster. Tampoco podrá eliminar ninguno de los grupos de parámetros predeterminados.

Eliminación de un grupo de parámetros (consola)

En el siguiente procedimiento se muestra cómo eliminar un grupo de parámetros mediante la consola de ElastiCache.

Para eliminar un grupo de parámetros con la consola de ElastiCache

1. Inicie sesión en la AWS Management Console y abra la consola de ElastiCache en <https://console.aws.amazon.com/elasticache/>.
2. Para ver una lista de todos los grupos de parámetros disponibles, en el panel de navegación izquierdo, elija Parameter Groups.
3. Elija los grupos de parámetros que desea eliminar activando la casilla situada a la izquierda del nombre del grupo de parámetros.

El botón Delete se activará.

4. Elija Eliminar (Delete).

Aparecerá la pantalla de confirmación Delete Parameter Groups.

5. Para eliminar los grupos de parámetros, en la pantalla de confirmación Delete Parameter Groups, elija Delete.

Para conservar los grupos de parámetros, elija Cancel.

Eliminación de un grupo de parámetros (AWS CLI)

Para eliminar un grupo de parámetros mediante la AWS CLI, use el comando `delete-cache-parameter-group`. Para que el grupo de parámetros se elimine, el grupo de parámetros especificado mediante `--cache-parameter-group-name` no puede tener ningún clúster asociado al grupo ni puede ser un grupo de parámetros predeterminado.

El siguiente código de muestra elimina el grupo de parámetros myMem14.

Example

Para Linux, macOS o Unix:

```
aws elasticache delete-cache-parameter-group \  
  --cache-parameter-group-name myRed28
```

Para Windows:

```
aws elasticache delete-cache-parameter-group ^  
  --cache-parameter-group-name myRed28
```

Para obtener más información, consulte [delete-cache-parameter-group](#).

Eliminación de un grupo de parámetros (API de ElastiCache)

Para eliminar un grupo de parámetros mediante la API de ElastiCache, utilice la acción `DeleteCacheParameterGroup`. Para que el grupo de parámetros se elimine, el grupo de parámetros especificado mediante `CacheParameterGroupName` no puede tener ningún clúster asociado al grupo ni puede ser un grupo de parámetros predeterminado.

Example

El siguiente código de muestra elimina el grupo de parámetros `myRed28`.

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=DeleteCacheParameterGroup  
  &CacheParameterGroupName=myRed28  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Timestamp=20150202T192317Z  
  &Version=2015-02-02  
  &X-Amz-Credential=<credential>
```

Para obtener más información, consulte [DeleteCacheParameterGroup](#).

Parámetros específicos de Memcached

Si no se especifica ningún grupo de parámetros para el clúster de Memcached, se usará un grupo de parámetros predeterminado apropiado para la versión del motor. No puede cambiar los valores de los parámetros de un grupo de parámetros predeterminado. Sin embargo, puede crear un grupo de parámetros personalizados y asignarlo a su clúster en cualquier momento. Para obtener más información, consulte [Creación de un grupo de parámetros](#).

Temas

- [Cambios en Memcached 1.6.17](#)
- [Parámetros agregados a Memcached 1.6.6](#)
- [Cambios en los parámetros de Memcached 1.5.10](#)
- [Parámetros agregados a Memcached 1.4.34](#)
- [Parámetros agregados a Memcached 1.4.33](#)
- [Parámetros agregados a Memcached 1.4.24](#)
- [Parámetros agregados a Memcached 1.4.14](#)
- [Parámetros compatibles con Memcached 1.4.5](#)
- [Capacidad adicional para conexiones de Memcached](#)
- [Parámetros específicos de tipo de nodo de Memcached](#)

Cambios en Memcached 1.6.17

A partir de la versión 1.6.17 de Memcached, ya no se admiten estos comandos administrativos: `lru_crawler`, `lru` y `slabs`. Con estos cambios, no podrá habilitar o deshabilitar `lru_crawler` en el tiempo de ejecución mediante comandos. Habilite o deshabilite `lru_crawler` modificando su grupo de parámetros personalizado.

Parámetros agregados a Memcached 1.6.6

Para Memcached 1.6.6, no se admiten parámetros adicionales.

Familia del grupo de parámetros: `memcached1.6`

Cambios en los parámetros de Memcached 1.5.10

Para Memcached 1.5.10, se admiten los siguientes parámetros adicionales.

Familia del grupo de parámetros: `memcached1.5`

Nombre	Detalles	Descripción
no_modern	<p>Predeterminado: 1</p> <p>Tipo: booleano</p> <p>Modificable: sí</p> <p>Valores permitidos: 0, 1</p> <p>Aplicación de los cambios: en el momento del lanzamiento</p>	<p>Alias para deshabilitar los comandos <code>slab_reassign</code> , <code>slab_automove</code> , <code>lru_crawler</code> , <code>lru_maintainer</code> y <code>maxconns_fast</code> . No modern también establece <code>hash_algorithm</code> en <code>jenkins</code> y permite incorporar un valor ASCII.</p> <p>Aplicable a <code>memcached</code> 1.5 y versiones posteriores. Para volver a lo moderno, debe deshabilitar este parámetro y volver a iniciar, lo que habilitará automáticamente <code>slab_reassign</code> , <code>slab_automove</code> , <code>lru_crawler</code> , <code>lru_maintainer</code> , y <code>maxconns_fast</code> .</p> <div data-bbox="1008 1171 1511 1885" style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px; margin-top: 20px;"> <p> Note</p> <p>El valor de configuración predeterminado para este parámetro se ha cambiado de 0 a 1 a partir del 20 de agosto de 2021. Los nuevos usuarios de <code>elasticache</code> recogerán automáticamente el valor predeterminado actualizado para cada región después del 20 de agosto de 2021. Los usuarios de <code>ElastiCache</code> existentes en</p> </div>

Nombre	Detalles	Descripción
		<p>las regiones antes del 20 de agosto de 2021 deben modificar manualmente sus grupos de parámetros personalizados para poder realizar este nuevo cambio.</p>
<code>inline_ascii_resp</code>	<p>Predeterminado: 0</p> <p>Tipo: booleano</p> <p>Modificable: sí</p> <p>Valores permitidos: 0, 1</p> <p>Aplicación de los cambios: en el momento del lanzamiento</p>	<p>Almacena números de la respuesta de VALUE, dentro de un elemento, utilizando hasta 24 bytes. Supone una ligera ralentización para las operaciones <code>get</code> y <code>fastex</code> en ASCII.</p>

Para Memcached 1.5.10, se han eliminado los parámetros siguientes.

Nombre	Detalles	Descripción
<code>expirezero_does_no_t_evict</code>	<p>Predeterminado: 0</p> <p>Tipo: booleano</p> <p>Modificable: sí</p> <p>Valores permitidos: 0, 1</p>	<p>Ya no se admite en esta versión.</p>

Nombre	Detalles	Descripción
	Aplicación de los cambios: en el momento del lanzamiento	
<code>modern</code>	Predeterminado: 1 Tipo: booleano Modificable: sí (requiere volver a lanzar si se establece en <code>no_modern</code>) Valores permitidos: 0, 1 Aplicación de los cambios: en el momento del lanzamiento	Ya no se admite en esta versión. A partir de esta versión, <code>no-modern</code> está habilitado de forma predeterminada con cada lanzamiento o relanzamiento.

Parámetros agregados a Memcached 1.4.34

Para Memcached 1.4.34, no se admite ningún parámetro adicional.

Familia del grupo de parámetros: `memcached1.4`

Parámetros agregados a Memcached 1.4.33

Para Memcached 1.4.33, se admiten los siguientes parámetros adicionales.

Familia del grupo de parámetros: `memcached1.4`

Nombre	Detalles	Descripción
<code>modern</code>	Valor predeterminado: habilitado	

Nombre	Detalles	Descripción
	<p>Tipo: booleano</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: en el momento del lanzamiento</p>	<p>Alias para varias características. Habilitar <code>modern</code> equivale a habilitar los comandos siguientes y usar un algoritmo de hash <code>murmur3: slab_reassign</code>, <code>slab_automove</code>, <code>lru_crawler</code>, <code>lru_maintainer</code>, <code>maxconns_fast</code> y <code>hash_algorithm=murmur3</code>.</p>
watch	<p>Valor predeterminado: habilitado</p> <p>Tipo: booleano</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p> <p>Los logs pueden eliminarse si el usuario alcanza los límites <code>watcher_logbuf_size</code> y <code>worker_logbuf_size</code>.</p>	<p>Recuperaciones de logs, expulsiones o mutaciones. Cuando, por ejemplo, el usuario activa <code>watch</code>, podrá ver los logs cuando tengan lugar las operaciones <code>get</code>, <code>set</code>, <code>delete</code> o <code>update</code>.</p>

Nombre	Detalles	Descripción
idle_timeout	<p>Valor predeterminado: 0 (deshabilitado)</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: en el momento del lanzamiento</p>	<p>Número mínimo de segundos que se permitirá que un cliente permanezca inactivo antes de solicitar el cierre. Rango de valores: de 0 a 86400.</p>
track_sizes	<p>Valor predeterminado: deshabilitado</p> <p>Tipo: booleano</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: en el momento del lanzamiento</p>	<p>Muestra el tamaño que cada grupo de slab ha consumido.</p> <p>Habilitar <code>track_sizes</code> le permite ejecutar <code>stats sizes</code> sin necesidad de ejecutar <code>stats sizes_enable</code>.</p>
watcher_logbuf_size	<p>Valor predeterminado: 256 (KB)</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: en el momento del lanzamiento</p>	<p>El comando <code>watch</code> activa el registro de transmisión para Memcached. Sin embargo, <code>watch</code> puede eliminar logs si la tasa de expulsiones, mutaciones o recuperaciones es lo suficientemente alta como para llenar el búfer de registro. En estas situaciones, los usuarios pueden aumentar el tamaño del búfer para reducir la posibilidad de que se produzcan pérdidas de logs.</p>

Nombre	Detalles	Descripción
<code>worker_logbuf_size</code>	<p>Valor predeterminado: 64 (KB)</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: en el momento del lanzamiento</p>	<p>El comando <code>watch</code> activa el registro de transmisión para Memcached. Sin embargo, <code>watch</code> puede eliminar logs si la tasa de expulsiones, mutaciones o recuperaciones es lo suficientemente alta como para llenar el búfer de registro. En estas situaciones, los usuarios pueden aumentar el tamaño del búfer para reducir la posibilidad de que se produzcan pérdidas de logs.</p>
<code>slab_chunk_max</code>	<p>Valor predeterminado: 524288 (bytes)</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: en el momento del lanzamiento</p>	<p>Especifica el tamaño de slab máximo. Establecer un tamaño de slab más reducido permite usar la memoria de un modo más eficaz. Los elementos cuyo tamaño supere <code>slab_chunk_max</code>, se dividen en múltiples slabs.</p>
<code>lru_crawler metadump [all 1 2 3]</code>	<p>Valor predeterminado: deshabilitado</p> <p>Tipo: booleano</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	<p>Si <code>lru_crawler</code> está habilitado, este comando volcará todas las claves.</p> <p><code>all 1 2 3</code> : todos los slabs; o bien especifique un número de slab específico</p>

Parámetros agregados a Memcached 1.4.24

Para Memcached 1.4.24, se admiten los siguientes parámetros adicionales.

Familia del grupo de parámetros: memcached1.4

Nombre	Detalles	Descripción
<code>disable_flush_all</code>	<p>Valor predeterminado: 0 (deshabilitado)</p> <p>Tipo: booleano</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: en el momento del lanzamiento</p>	<p>Agregue el parámetro (-F) para deshabilitar <code>flush_all</code>. Esto resulta útil si no quiere que se pueda ejecutar un vaciado completo en instancias de producción.</p> <p>Valores: 0, 1 (el usuario puede usar <code>flush_all</code> cuando el valor es 0).</p>
<code>hash_algorithm</code>	<p>Valor predeterminado: jenkins</p> <p>Tipo: String</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: en el momento del lanzamiento</p>	<p>Algoritmo de hash que se va a usar. Valores permitidos: <code>murmur3</code> y <code>jenkins</code>.</p>
<code>lru_crawler</code>	<p>Valor predeterminado: 0 (deshabilitado)</p> <p>Tipo: booleano</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: tras reiniciar</p>	<p>Limpia clases de slab de elementos que han caducado. Este es un proceso de bajo impacto que se ejecuta en segundo plano. En la actualidad, es necesario iniciar un rastreo mediante un comando manual.</p>

Nombre	Detalles	Descripción
	<p> Note</p> <p>Puede habilitar temporalmente <code>lru_crawler</code> durante el tiempo de ejecución desde la línea de comandos. Para obtener más información, consulte la columna Descripción.</p>	<p>Para habilitar el proceso temporalmente, ejecute <code>lru_crawler enable</code> en la línea de comandos.</p> <p><code>lru_crawler 1,3,5</code> rastrea clases de slab 1, 3 y 5 buscando elementos caducados para agregarlos a la lista libre.</p> <p>Valores: 0, 1</p> <p> Note</p> <p>Habilitar <code>lru_crawler</code> en la línea de comandos, se habilita el rastreador hasta que se deshabilita en la línea de comandos o hasta el siguiente reinicio. Para habilitar el proceso de forma permanente, debe modificar el valor del parámetro. Para obtener más información, consulte Modificación de un grupo de parámetros.</p>

Nombre	Detalles	Descripción
<code>lru_maintainer</code>	<p>Valor predeterminado: 0 (deshabilitado)</p> <p>Tipo: booleano</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: en el momento del lanzamiento</p>	<p>Subproceso que se ejecuta en segundo plano y que cambia los elementos entre LRU a medida que se agota la capacidad.</p> <p>Valores: 0, 1</p>
<code>expirezero_does_not_evict</code>	<p>Valor predeterminado: 0 (deshabilitado)</p> <p>Tipo: booleano</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: en el momento del lanzamiento</p>	<p>Cuando se usa con <code>lru_maintainer</code>, los elementos cuyo periodo de vencimiento es 0 no se pueden expulsar.</p> <div data-bbox="1008 989 1508 1304" style="border: 1px solid #f08080; border-radius: 10px; padding: 10px; margin: 10px 0;"> <p> Warning</p> <p>Esto puede agotar la memoria disponible para otros elementos que se pueden expulsar.</p> </div> <p>Se puede establecer para ignorar <code>lru_maintainer</code>.</p>

Parámetros agregados a Memcached 1.4.14

Para Memcached 1.4.14, se admiten los siguientes parámetros adicionales.

Familia del grupo de parámetros: memcached1.4

Parámetros agregados a Memcached 1.4.14

Nombre	Descripción
<code>config_max</code>	El número máximo de entradas de configuración de ElastiCache.
<code>config_size_max</code>	Tamaño máximo de las entradas de configuración en bytes.
<code>hashpower_init</code>	El tamaño inicial de la tabla hash de ElastiCache, expresado en potencia de dos. El valor predeterminado es 16 (2^{16}) o 65536 claves.

Nombre	Descripción
maxconns_fast	<p>Cambia el modo en que se gestionan las solicitudes de nuevas conexiones cuando se alcanza el límite máximo de la conexión. Si este parámetro se establece en 0 (cero), las conexiones nuevas se agregan a la cola de tareas pendientes a la espera de que se cierren otras conexiones. Si el parámetro se establece en 1, ElastiCache envía un error al cliente y cierra la conexión de inmediato.</p>

Nombre	Descripción
s1ab_automove	<p>Ajusta el algoritmo automove de los slabs: si este parámetro se establece en 0 (cero), el algoritmo automove estará deshabilitado. Si se establece en 1, ElastiCache adoptará un enfoque lento y conservador para mover los bloques de forma automática. Si se establece en 2, ElastiCache mueve los bloques de forma agresiva siempre que se produce una expulsión. (Este modo no se recomienda, excepto con fines de prueba.)</p>

Nombre	Descripción
slab_reassign	Habilita o deshabilita la reasignación de slab. Si este parámetro se establece en 1, podrá usar el comando "slabs reassign" para reasignar manualmente la memoria.

Parámetros compatibles con Memcached 1.4.5

Familia del grupo de parámetros: memcached1.4

Para Memcached 1.4.5, se admiten los siguientes parámetros.

Parámetros agregados a Memcached 1.4.5

Nombre	Detalles	Descripción
backlog_queue_limit	<p>Predeterminado: 1024</p> <p>Tipo: número entero</p> <p>Modificable: no</p>	Límite de cola de tareas pendientes.

Nombre	Detalles	Descripción
<code>binding_protocol</code>	<p>Valor predeterminado: automático</p> <p>Tipo: String</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: tras reiniciar</p>	<p>Protocolo de enlace.</p> <p>Valores permitidos: <code>ascii</code> y <code>auto</code>.</p> <p>Para obtener información acerca de la modificación del valor de <code>binding_protocol</code>, consulte Modificación de un grupo de parámetros.</p>
<code>cas_disabled</code>	<p>Valor predeterminado: 0 (falso)</p> <p>Tipo: Booleano</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: tras reiniciar</p>	<p>Si se establece en 1 (<code>true</code>), compruebe y defina (CAS) las operaciones que se deshabilitarán. Los elementos almacenados ocuparán 8 bytes menos que con CAS habilitado.</p>
<code>chunk_size</code>	<p>Predeterminado: 48</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: tras reiniciar</p>	<p>Cantidad mínima, en bytes, de espacio para asignar a las marcas, los valores y las claves de elemento más pequeños.</p>
<code>chunk_size_growth_factor</code>	<p>Predeterminado: 1.25</p> <p>Tipo: flotante</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: tras reiniciar</p>	<p>Factor de crecimiento que controla el tamaño de cada fragmento de Memcached sucesivo; cada fragmento tendrá un tamaño <code>chunk_size_growth_factor</code> veces mayor que el anterior.</p>

Nombre	Detalles	Descripción
<code>error_on_memory_exhausted</code>	<p>Valor predeterminado: 0 (falso)</p> <p>Tipo: Booleano</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: tras reiniciar</p>	Si se establece en 1 (true), cuando no haya más memoria para almacenar elementos, Memcached devolverá un mensaje de error en lugar de expulsar elementos.
<code>large_memory_pages</code>	<p>Valor predeterminado: 0 (falso)</p> <p>Tipo: Booleano</p> <p>Modificable: no</p>	Si se establece en 1 (true [verdadero]), ElastiCache intentará utilizar páginas de gran memoria.
<code>lock_down_paged_memory</code>	<p>Valor predeterminado: 0 (falso)</p> <p>Tipo: Booleano</p> <p>Modificable: no</p>	Si se establece en 1 (true [verdadero]), ElastiCache bloqueará toda la memoria paginada.
<code>max_item_size</code>	<p>Predeterminado: 1048576</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: tras reiniciar</p>	Tamaño, en bytes, de los elementos de mayor tamaño que se pueden almacenar en el clúster.
<code>max_simultaneous_connections</code>	<p>Predeterminado: 65000</p> <p>Tipo: número entero</p> <p>Modificable: no</p>	Número máximo de conexiones simultáneas.

Nombre	Detalles	Descripción
<code>maximize_core_file_limit</code>	<p>Valor predeterminado: 0 (falso)</p> <p>Tipo: Booleano</p> <p>Modificable:</p> <p>Aplicación de los cambios: tras reiniciar</p>	Si se establece en 1 (true [verdadero]), ElastiCache maximizará el límite de archivo de núcleo.
<code>memcached_connections_overhead</code>	<p>Predeterminado: 100</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: tras reiniciar</p>	<p>Cantidad de memoria reservada para las conexiones de Memcached y capacidad adicional para otras operaciones. Para obtener información acerca de este parámetro, consulte Capacidad adicional para conexiones de Memcached.</p>
<code>requests_per_event</code>	<p>Predeterminado: 20</p> <p>Tipo: número entero</p> <p>Modificable: no</p>	Número máximo de solicitudes por evento para una conexión determinada. Este límite es necesario para evitar el agotamiento de los recursos.

Capacidad adicional para conexiones de Memcached

En cada nodo, la memoria disponible para almacenar elementos equivale a la cantidad total de memoria disponible de dicho nodo (que se almacena en el parámetro `max_cache_memory`) menos la memoria que se usa para las conexiones y la capacidad adicional para otras operaciones (que se almacena en el parámetro `memcached_connections_overhead`). Por ejemplo, un nodo del tipo `cache.m1.small` tiene una `max_cache_memory` de 1300 MB. Con el valor `memcached_connections_overhead` predeterminado de 100 MB, el proceso de Memcached tendrá 1200 MB de espacio disponible para almacenar elementos.

Los valores predeterminados para el parámetro `memcached_connections_overhead` son válidos para la mayoría de los casos de uso; sin embargo, la asignación necesaria para la capacidad

adicional de conexión puede variar dependiendo de varios factores, incluida la tasa de solicitudes, el tamaño de carga y el número de conexiones.

Puede cambiar el valor de `memcached_connections_overhead` según las necesidades de su aplicación. Por ejemplo, al aumentar el valor del parámetro `memcached_connections_overhead`, se reducirá la cantidad de memoria disponible para almacenar elementos y dispondrá de un búfer de mayor tamaño para la sobrecarga de las conexiones. La reducción del valor del parámetro `memcached_connections_overhead` le dará más memoria para almacenar elementos, pero puede aumentar el riesgo de agotar el espacio de intercambio y de que se reduzca el rendimiento. Si se agota el espacio de intercambio y observa una pérdida de rendimiento, pruebe incrementar el valor del parámetro `memcached_connections_overhead`.

Important

Para el tipo nodo `cache.t1.micro`, el valor de `memcached_connections_overhead` se determina de la manera siguiente:

- Si su clúster utiliza el grupo de parámetros predeterminado, ElastiCache establecerá el valor de `memcached_connections_overhead` en 13 MB.
- Si su clúster usa un grupo de parámetros de su creación, podrá establecer el valor de `memcached_connections_overhead` según su preferencia.

Parámetros específicos de tipo de nodo de Memcached

Aunque la mayoría de los parámetros tienen un único valor, algunos parámetros tienen distintos valores en función del tipo de nodo que se use. La tabla siguiente muestra los valores predeterminados de los parámetros `max_cache_memory` y `num_threads` para cada tipo de nodo. Los valores de estos parámetros no se pueden modificar.

Tipo de nodo	<code>max_cache_memory</code> (en megabytes)	<code>num_threads</code>
<code>cache.t1.micro</code>	213	1
<code>cache.t2.micro</code>	555	1
<code>cache.t2.small</code>	1588	1

Tipo de nodo	max_cache_memory (en megabytes)	num_threads
cache.t2.medium	3301	2
cache.t3.micro	512	2
cache.t3.small	1402	2
cache.t3.medium	3364	2
cache.t4g.micro	512	2
cache.t4g.small	1402	2
cache.t4g.medium	3164	2
cache.m1.small	1301	1
cache.m1.medium	3350	1
cache.m1.large	7100	2
cache.m1.xlarge	14600	4
cache.m2.xlarge	33800	2
cache.m2.2xlarge	30412	4
cache.m2.4xlarge	68000	16
cache.m3.medium	2850	1
cache.m3.large	6200	2
cache.m3.xlarge	13600	4
cache.m3.2xlarge	28600	8
cache.m4.large	6573	2
cache.m4.xlarge	11 496	4

Tipo de nodo	max_cache_memory (en megabytes)	num_threads
cache.m4.2xlarge	30412	8
cache.m4.4xlarge	62234	16
cache.m4.10xlarge	158355	40
cache.m5.large	6537	2
cache.m5.xlarge	13248	4
cache.m5.2xlarge	26671	8
cache.m5.4xlarge	53516	16
cache.m5.12xlarge	160900	48
cache.m5.24xlarge	321865	96
cache.m6g.large	6537	2
cache.m6g.xlarge	13248	4
cache.m6g.2xlarge	26671	8
cache.m6g.4xlarge	53516	16
cache.m6g.8xlarge	107 000	32
cache.m6g.12xlarge	160900	48
cache.m6g.16xlarge	214 577	64
cache.c1.xlarge	6600	8
cache.r3.large	13800	2
cache.r3.xlarge	29100	4
cache.r3.2xlarge	59600	8

Tipo de nodo	max_cache_memory (en megabytes)	num_threads
cache.r3.4xlarge	120600	16
cache.r3.8xlarge	120600	32
cache.r4.large	12590	2
cache.r4.xlarge	25652	4
cache.r4.2xlarge	51686	8
cache.r4.4xlarge	103815	16
cache.r4.8xlarge	208144	32
cache.r4.16xlarge	416776	64
cache.r5.large	13387	2
cache.r5.xlarge	26953	4
cache.r5.2xlarge	54084	8
cache.r5.4xlarge	108347	16
cache.r5.12xlarge	325400	48
cache.r5.24xlarge	650869	96
cache.r6g.large	13387	2
cache.r6g.xlarge	26953	4
cache.r6g.2xlarge	54084	8
cache.r6g.4xlarge	108347	16
cache.r6g.8xlarge	214 577	32
cache.r6g.12xlarge	325400	48

Tipo de nodo	max_cache_memory (en megabytes)	num_threads
cache.r6g.16xlarge	429 154	64
cache.c7gn.large	3164	2
cache.c7gn.xlarge	6537	4
cache.c7gn.2xlarge	13248	8
cache.c7gn.4xlarge	26671	16
cache.c7gn.8xlarge	53516	32
cache.c7gn.12xlarge	325400	48
cache.c7gn.16xlarge	108347	64

Note

Todas las instancias T2 se crean en una instancia de Amazon Virtual Private Cloud (Amazon VPC).

Parámetros específicos de Redis OSS

Si no especifica un grupo de parámetros para su clúster de OSS de Redis, se utilizará un grupo de parámetros predeterminado adecuado a la versión de su motor. No puede cambiar los valores de los parámetros de un grupo de parámetros predeterminado. Sin embargo, puede crear un grupo de parámetros personalizado y asignarlo a su clúster en cualquier momento, siempre y cuando los valores de los parámetros modificables condicionalmente sean iguales en ambos grupos de parámetros. Para obtener más información, consulte [Creación de un grupo de parámetros](#).

Temas

- [Cambios en los parámetros de Redis OSS 7](#)
- [Cambios en los parámetros de Redis OSS 6.x](#)
- [Cambios en los parámetros de Redis OSS 5.0.3](#)
- [Cambios en los parámetros de Redis OSS 5.0.0](#)
- [Cambios en los parámetros de Redis OSS 4.0.10](#)
- [Cambios en los parámetros de Redis OSS 3.2.10](#)
- [Cambios en los parámetros de Redis OSS 3.2.6](#)
- [Cambios en los parámetros de Redis OSS 3.2.4](#)
- [Se agregaron parámetros a Redis OSS 2.8.24 \(mejorada\)](#)
- [Se agregaron parámetros a Redis OSS 2.8.23 \(mejorado\)](#)
- [Se agregaron parámetros a Redis OSS 2.8.22 \(mejorado\)](#)
- [Redis OSS 2.8.21 agregó parámetros](#)
- [Se agregaron parámetros a Redis OSS 2.8.19](#)
- [Parámetros añadidos de Redis OSS 2.8.6](#)
- [Parámetros de Redis OSS 2.6.13](#)
- [Parámetros específicos del tipo de nodo de Redis OSS](#)

Cambios en los parámetros de Redis OSS 7

Familia del grupo de parámetros: redis7

Los grupos de parámetros predeterminados de Redis OSS 7 son los siguientes:

- `default.redis7`— Utilice este grupo de parámetros, o uno derivado de él, para los clústeres y grupos de replicación de Redis OSS (modo de clúster desactivado).

- `default.redis7.cluster.on`— Utilice este grupo de parámetros, o uno derivado del mismo, para los clústeres y grupos de replicación de Redis OSS (modo de clúster habilitado).

Los parámetros agregados en Redis OSS 7 son los siguientes.

Nombre	Detalles	Descripción
<code>cluster-allow-pubsubshard-when-down</code>	<p>Valores permitidos: <code>yes</code>, <code>no</code></p> <p>Valor predeterminado: <code>yes</code></p> <p>Tipo: cadena</p> <p>Modificable: sí</p> <p>Los cambios surten efecto: inmediatamente en todos los nodos del clúster.</p>	<p>Cuando se establece sí como valor predeterminado, se permite a los nodos atender el tráfico de particiones de pubsub mientras el clúster esté inactivo, siempre y cuando crea que es el propietario de las ranuras.</p>
<code>cluster-preferred-endpoint-type</code>	<p>Valores permitidos: <code>ip</code>, <code>tls-dynamic</code></p> <p>Valor predeterminado: <code>tls-dynamic</code></p> <p>Tipo: cadena</p> <p>Modificable: sí</p> <p>Los cambios surten efecto: inmediatamente en todos los nodos del clúster.</p>	<p>Este valor controla el punto de conexión que se devuelve para las solicitudes <code>MOVED/ASKING</code>, así como el campo de punto de conexión para <code>CLUSTER SLOTS</code> y <code>CLUSTER SHARDS</code>. Cuando el valor se establece en <code>IP</code>, el nodo comunicará su dirección IP. Si el valor se establece en <code>tls-dynamic</code>, el nodo anunciará un nombre de host cuando <code>encryption-in-transit</code> esté activado y, en caso contrario, una dirección IP.</p>
<code>latency-tracking</code>	<p>Valores permitidos: <code>yes</code>, <code>no</code></p> <p>Valor predeterminado: <code>no</code></p> <p>Tipo: cadena</p>	<p>Cuando se establece en sí, realiza un seguimiento de las latencias por comando y permite exportar la distribución de percentil es mediante el comando de estadísticas de latencia <code>INFO</code> y las distribuciones de latencia</p>

Nombre	Detalles	Descripción
	<p>Modificable: sí</p> <p>Los cambios surten efecto: inmediatamente en todos los nodos del clúster.</p>	<p>acumulada (histogramas) mediante el comando LATENCY.</p>
<p>hash-max-listpack-entries</p>	<p>Valores permitidos: 0+</p> <p>Valor predeterminado: 512</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Los cambios surten efecto: inmediatamente en todos los nodos del clúster.</p>	<p>El número máximo de entradas de hash para comprimir el conjunto de datos.</p>
<p>hash-max-listpack-value</p>	<p>Valores permitidos: 0+</p> <p>Valor predeterminado: 64</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Los cambios surten efecto: inmediatamente en todos los nodos del clúster.</p>	<p>El umbral de entradas de hash más grandes para comprimir el conjunto de datos.</p>

Nombre	Detalles	Descripción
<code>zset-max-listpack-entries</code>	<p>Valores permitidos: 0+</p> <p>Valor predeterminado: 128</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Los cambios surten efecto: inmediatamente en todos los nodos del clúster.</p>	El número máximo de entradas de conjuntos ordenados para comprimir el conjunto de datos.
<code>zset-max-listpack-value</code>	<p>Valores permitidos: 0+</p> <p>Valor predeterminado: 64</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Los cambios surten efecto: inmediatamente en todos los nodos del clúster.</p>	El umbral de entradas de conjuntos ordenados más grandes para comprimir el conjunto de datos.

Los parámetros modificados en Redis OSS 7 son los siguientes.

Nombre	Detalles	Descripción
<code>activeresharding</code>	<p>Modificable: no. En Redis OSS 7, este parámetro está oculto y activado de forma predeterminada. Para desactivarlo, debe crear un caso de soporte.</p>	Modificable era sí.

Los parámetros eliminados en Redis OSS 7 son los siguientes.

Nombre	Detalles	Descripción
hash-max-ziplist-entries	<p>Valores permitidos: 0+</p> <p>Valor predeterminado: 512</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Los cambios surten efecto: inmediatamente en todos los nodos del clúster.</p>	Use listpack en lugar de ziplist para representar una codificación de hash pequeña
hash-max-ziplist-value	<p>Valores permitidos: 0+</p> <p>Valor predeterminado: 64</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Los cambios surten efecto: inmediatamente en todos los nodos del clúster.</p>	Use listpack en lugar de ziplist para representar una codificación de hash pequeña
zset-max-ziplist-entries	<p>Valores permitidos: 0+</p> <p>Valor predeterminado: 128</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Los cambios surten efecto: inmediatamente en todos los nodos del clúster.</p>	Use listpack en lugar de ziplist para representar una codificación de hash pequeña.

Nombre	Detalles	Descripción
<code>zset-max-ziplist-value</code>	Valores permitidos: 0+ Valor predeterminado: 64 Tipo: número entero Modificable: sí Los cambios surten efecto: inmediatamente en todos los nodos del clúster.	Use <code>listpack</code> en lugar de <code>ziplist</code> para representar una codificación de hash pequeña.
<code>list-max-ziplist-size</code>	Valores permitidos: Valor predeterminado: -2 Tipo: número entero Modificable: sí Los cambios surten efecto: inmediatamente en todos los nodos del clúster.	El número de entradas permitido por nodo de lista interna.

Cambios en los parámetros de Redis OSS 6.x

Familia del grupo de parámetros: `redis6.x`

Los grupos de parámetros predeterminados de Redis OSS 6.x son los siguientes:

- `default.redis6.x`— Utilice este grupo de parámetros, o uno derivado de él, para los clústeres y grupos de replicación de Redis OSS (modo de clúster desactivado).
- `default.redis6.x.cluster.on`— Utilice este grupo de parámetros, o uno derivado del mismo, para los clústeres y grupos de replicación de Redis OSS (modo de clúster habilitado).

Note

En la versión 6.2 del motor OSS de Redis, cuando se introdujo la familia de nodos r6gd para su uso con, solo las políticas de memoria máxima noeviction [Organización de datos en niveles](#), volatile-lru y allkeys-lru eran compatibles con los tipos de nodos r6gd.

Para obtener más información, consulte [ElastiCache \(Redis OSS\) versión 6.2 \(mejorada\)](#) y [ElastiCache \(Redis OSS\) versión 6.0 \(mejorada\)](#).

Los parámetros agregados en Redis OSS 6.x son los siguientes.

Nombre	Detalles	Descripción
<code>acl-pubsub-default</code> (added in 6.2)	Valores permitidos: <code>resetchannels</code> , <code>allchannels</code> Valor predeterminado: <code>allchannels</code> Tipo: cadena Modificable: sí Los cambios surten efecto: los usuarios actuales de Redis OSS asociados al clúster seguirán teniendo los permisos existentes. Actualice los usuarios o reinicie el clúster para actualizar los usuarios de Redis OSS existentes.	Permisos de canal pubsub predeterminados para los usuarios de ACL implementados en este clúster.
<code>cluster-allow-reads-when-do</code>	Valor predeterminado: no Tipo: String	Si se establece en sí, un grupo de replicación de Redis OSS (modo de clúster activado) continúa procesando los comandos de lectura

Nombre	Detalles	Descripción
wn (added in 6.0)	<p>Modificable: sí</p> <p>Los cambios surten efecto: inmediatamente en todos los clústeres</p>	<p>incluso cuando un nodo no puede alcanzar el quórum de servidores principales.</p> <p>Cuando se establece en el valor predeterminado no, el grupo de reproducción rechaza todos los comandos. Se recomienda establecer este valor en yes (sí) si utiliza un clúster con menos de tres grupos de nodos o si la aplicación puede manejar de forma segura las lecturas obsoletas.</p>
tracking-table-max-keys (added in 6.0)	<p>Predeterminado: 1 000 000</p> <p>Tipo: número</p> <p>Modificable: sí</p> <p>Los cambios surten efecto: inmediatamente en todos los clústeres</p>	<p>Para facilitar el almacenamiento en caché del lado del cliente, Redis OSS permite rastrear qué clientes han accedido a qué claves.</p> <p>Cuando se modifica la clave rastreada, se envían mensajes de invalidación a todos los clientes para notificarles que sus valores almacenados en caché ya no son válidos. Este valor permite especificar el límite superior de esta tabla. Una vez superado este valor de parámetro, los clientes se envían invalidación aleatoriamente. Este valor debe ajustarse para limitar la utilización de memoria sin dejar de realizar un seguimiento de las claves suficientes. Las claves también se invalidan en condiciones de poca memoria.</p>

Nombre	Detalles	Descripción
<p><code>acllog-max-len</code> (added in 6.0)</p>	<p>Valor predeterminado: 128</p> <p>Tipo: número</p> <p>Modificable: sí</p> <p>Los cambios surten efecto: inmediatamente en todos los clústeres</p>	<p>Este valor corresponde al número máximo de entradas en el registro ACL.</p>
<p><code>active-expire-effort</code> (added in 6.0)</p>	<p>Valor predeterminado: 1</p> <p>Tipo: número</p> <p>Modificable: sí</p> <p>Los cambios surten efecto: inmediatamente en todos los clústeres</p>	<p>Redis OSS elimina las claves que han superado su tiempo de vida mediante dos mecanismos. En uno, se accede a una clave y se encuentra que ha vencido. En el otro, un trabajo periódico muestra claves y hace que se venzan aquellas que han excedido su periodo de vida. Este parámetro define el esfuerzo que Redis OSS emplea para hacer caducar los elementos del trabajo periódico.</p> <p>El valor predeterminado de 1 intenta evitar tener más del 10 % de las claves vencidas que todavía se encuentran en la memoria. También intenta evitar consumir más del 25 % de la memoria total y agregar latencia al sistema. Puede aumentar este valor hasta 10 para aumentar la cantidad de esfuerzo invertido en las claves vencidas. La desventaja es mayor CPU y una latencia potencialmente mayor. Recomendamos un valor de 1 a menos que vea un uso elevado de memoria y pueda tolerar un aumento en la utilización de la CPU.</p>

Nombre	Detalles	Descripción
lazyfree-lazy-user-del (added in 6.0)	Valor predeterminado: no Tipo: String Modificable: sí Los cambios surten efecto: inmediatamente en todos los clústeres	Cuando el valor se establece en yes (sí), el comando DEL actúa de la misma manera que UNLINK.

Los parámetros eliminados en Redis OSS 6.x son los siguientes.

Nombre	Detalles	Descripción
lua-replique-commands	Valores permitidos: sí/no Valor predeterminado: yes Tipo: booleano Modificable: sí Aplicación de los cambios: inmediatamente	Indica si siempre se debe habilitar o no la replicación de los efectos de Lua en los scripts de Lua.

Cambios en los parámetros de Redis OSS 5.0.3

Familia del grupo de parámetros: redis5.0

Grupos de parámetros predeterminados de Redis OSS 5.0

- `default.redis5.0`— Utilice este grupo de parámetros, o uno derivado de él, para los clústeres y grupos de replicación de Redis OSS (modo de clúster desactivado).
- `default.redis5.0.cluster.on`— Utilice este grupo de parámetros, o uno derivado del mismo, para los clústeres y grupos de replicación de Redis OSS (modo de clúster habilitado).

Parámetros agregados en Redis OSS 5.0.3

Nombre	Detalles	Descripción
rename-commands	<p>Predeterminado: none</p> <p>Tipo: String</p> <p>Modificable: sí</p> <p>Los cambios surten efecto: inmediatamente en todos los clústeres</p>	<p>Lista separada por espacios de comandos de Redis OSS renombrados. A continuación se muestra una lista restringida de comandos disponibles a lo que se puede cambiar el nombre:</p> <p>APPEND AUTH BITCOUNT BITFIELD BITOP BITPOS BLPOP BRPOP BRPOPLUSH BZPOPMIN BZPOPMAX CLIENT CLUSTER COMMAND DBSIZE DECR DECRBY DEL DISCARD DUMP ECHO EVAL EVALSHA EXEC EXISTS EXPIRE EXPIREAT FLUSHALL FLUSHDB GEOADD GEOHASH GEOPOS GEODIST GEORADIUS GEORADIUSBYMEMBER GET GETBIT GETRANGE GETSET HDEL HEXISTS HGET HGETALL HINCRBY HINCRBYFLOAT HKEYS HLEN HMGET HMSET HSET HSETNX HSTRLEN HVALS INCR INCRBY INCRBYFLOAT INFO KEYS LASTSAVE LINDEX LINSERT LLEN LPOP LPUSSH LPUSHX LRANGE LREM LSET LTRIM MEMORY MGET MONITOR MOVE MSET MSETNX MULTI OBJECT PERSIST PEXPIRE PEXPIREAT PFADD PFCOUNT PFMERGE PING PSETEX PSUBSCRIBE PUBSUB PTTL PUBLISH PUNSUBSCRIBE RANDOMKEY READONLY READWRITE RENAME RENAMENX RESTORE ROLE RPOP RPOPLUSH RPUSH RPUSHX SADD SCARD SCRIPT SDIFF SDIFFSTORE SELECT SET SETBIT SETEX SETNX SETRANGE SINTER SINTERSTORE SISMEMBER</p>

Nombre	Detalles	Descripción
		SLOWLOG SMEMBERS SMOVE SORT SPOP SRANDMEMBER SREM STRLEN SUBSCRIBE UNION UNIONSTORE SWAPDB TIME TOUCH TTL TYPE UNSUBSCRIBE UNLINK UNWATCH WAIT WATCH ZADD ZCARD ZCOUNT ZINCRBY ZINTERSTO RE ZLEXCOUNT ZPOPMAX ZPOPMIN ZRANGE ZRANGEBYLEX ZREVRANGE BYLEX ZRANGEBYSCORE ZRANK ZREM ZREMRANGEBYLEX ZREMRANGEBYRANK ZREMRANGEBYSCORE ZREVRANGE ZREVRANGEBYSCORE ZREVRANK ZSCORE ZUNIONSTORE SCAN SSCAN HSCAN ZSCAN XINFO XADD XTRIM XDEL XRA NGE XREVRANGE XLEN XREAD XGROUP XREADGROUP XACK XCLAIM XPENDING GEORADIUS_RO GEORADIUSBYMEMBER_ RO LOLWUT XSETID SUBSTR

Para obtener más información, consulte [ElastiCache \(Redis OSS\) versión 5.0.6 \(mejorada\)](#).

Cambios en los parámetros de Redis OSS 5.0.0

Familia del grupo de parámetros: redis5.0

Grupos de parámetros predeterminados de Redis OSS 5.0

- `default.redis5.0`— Utilice este grupo de parámetros, o uno derivado de él, para los clústeres y grupos de replicación de Redis OSS (modo de clúster desactivado).
- `default.redis5.0.cluster.on`— Utilice este grupo de parámetros, o uno derivado del mismo, para los clústeres y grupos de replicación de Redis OSS (modo de clúster habilitado).

Parámetros agregados en Redis OSS 5.0

Nombre	Detalles	Descripción
<code>stream-node-max-bytes</code>	<p>Valores permitidos: 0+</p> <p>Predeterminado: 4096</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	<p>La estructura de datos de secuencia es un árbol de prefijos de nodos que contiene varios elementos. Utilice esta configuración para especificar el tamaño máximo de un único nodo de un árbol de prefijos in bytes. Si se establece en 0, el tamaño del nodo del árbol es ilimitado.</p>
<code>stream-node-max-entries</code>	<p>Valores permitidos: 0+</p> <p>Predeterminado: 100</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	<p>La estructura de datos de secuencia es un árbol de prefijos de nodos que contiene varios elementos. Utilice esta configuración para especificar el número máximo de elementos que puede contener un único nodo antes de cambiar a un nodo nuevo al agregar entradas nuevas de secuencia. Si se establece en 0, el número de elementos del nodo del árbol es ilimitado.</p>
<code>active-defrag-max-scan-fields</code>	<p>Valores permitidos: de 1 a 1000000</p> <p>Predeterminado: 1000</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	<p>Número máximo de campos set/hash/zset/list que se procesarán desde el análisis del diccionario principal.</p>

Nombre	Detalles	Descripción
lua-replicate-commands	Valores permitidos: sí/no Valor predeterminado: yes Tipo: booleano Modificable: sí Aplicación de los cambios: inmediatamente	Indica si siempre se debe habilitar o no la replicación de los efectos de Lua en los scripts de Lua.
replica-ignore-maxmemory	Valor predeterminado: yes Tipo: booleano Modificable: no	Determina si la réplica no tiene en cuenta el ajuste <code>maxmemory</code> al no expulsar elementos independientes del nodo primario

Redis OSS ha cambiado el nombre de varios parámetros de la versión 5.0 del motor en respuesta a los comentarios de la comunidad. Para obtener más información, consulte [¿Qué hay de nuevo en Redis OSS 5?](#). En la tabla siguiente, se muestran los nombres nuevos y a qué comandos corresponden en las versiones anteriores.

Parámetros renombrados en Redis OSS 5.0

Nombre	Detalles	Descripción
replica-lazy-flush	Valor predeterminado: yes Tipo: booleano Modificable: no Nombre anterior: slave-lazy-flush	Realiza un volcado de base de datos asíncrono durante la sincronización de réplicas.

Nombre	Detalles	Descripción
<code>client-output-buffer-limit-replica-hard-limit</code>	<p>Valor predeterminado: para obtener información acerca de los valores, consulte Parámetros específicos del tipo de nodo de Redis OSS</p> <p>Tipo: número entero</p> <p>Modificable: no</p> <p>Nombre anterior: <code>client-output-buffer-limit - slave-hard-limit</code></p>	<p>Para réplicas de lectura de Redis OSS: si el búfer de salida de un cliente alcanza el número especificado de bytes, el cliente se desconectará.</p>
<code>client-output-buffer-limit-replica-soft-limit</code>	<p>Valor predeterminado: para obtener información acerca de los valores, consulte Parámetros específicos del tipo de nodo de Redis OSS</p> <p>Tipo: número entero</p> <p>Modificable: no</p> <p>Nombre anterior: <code>- client-output-buffer-limit slave-soft-limit</code></p>	<p>Para réplicas de lectura de Redis OSS: si el búfer de salida de un cliente alcanza el número de bytes especificado, el cliente se desconectará, pero solo si esta condición persiste durante un tiempo. <code>client-output-buffer-limit-replica-soft-seconds</code></p>
<code>client-output-buffer-limit-replica-soft-seconds</code>	<p>Predeterminado: 60</p> <p>Tipo: número entero</p> <p>Modificable: no</p> <p>Nombre anterior: <code>- client-output-buffer-limit slave-soft-seconds</code></p>	<p>Para las réplicas de lectura del OSS de Redis: si el búfer de salida de un cliente permanece en <code>client-output-buffer-limit-replica-soft-limit</code> bytes durante más de este número de segundos, el cliente se desconectará.</p>

Nombre	Detalles	Descripción
<p>replica-allow-chaining</p>	<p>Valor predeterminado: no</p> <p>Tipo: String</p> <p>Modificable: no</p> <p>Nombre anterior: slave-allow-chaining</p>	<p>Determina si una réplica de lectura en Redis OSS puede tener réplicas de lectura propias.</p>
<p>min-replicas-to-write</p>	<p>Predeterminado: 0</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Nombre anterior: min-slaves-to-write</p> <p>Aplicación de los cambios: inmediatamente</p>	<p>Número mínimo de réplicas de lectura que deben estar disponibles para el nodo principal acepte operaciones de escritura de los clientes. Si el número de réplicas disponibles cae por debajo de este número, el nodo principal dejará de aceptar solicitudes de escritura.</p> <p>Si este parámetro min-replicas-max-lag es 0, el nodo principal siempre aceptará solicitudes de escritura, incluso si no hay réplicas disponibles.</p>
<p>min-replicas-max-lag</p>	<p>Predeterminado: 10</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Nombre anterior: min-slaves-max-lag</p> <p>Aplicación de los cambios: inmediatamente</p>	<p>Número de segundos en los que el nodo principal debe recibir una solicitud de ping desde una réplica de lectura. Si este periodo de tiempo transcurre y el nodo principal no recibe ningún ping, la réplica deja de considerarse como disponible. Si el número de réplicas disponibles es inferior min-replicas-to-write, el principal dejará de aceptar escrituras en ese momento.</p> <p>Si este parámetro min-replicas-to-write es 0, el nodo principal siempre aceptará solicitudes de escritura, incluso si no hay réplicas disponibles.</p>

Nombre	Detalles	Descripción
<code>close-on-replica-write</code>	<p>Valor predeterminado: <code>yes</code></p> <p>Tipo: booleano</p> <p>Modificable: sí</p> <p>Nombre anterior: <code>close-on-slave-write</code></p> <p>Aplicación de los cambios: inmediatamente</p>	Si está habilitado, los clientes que intenten escribir en una réplica de solo lectura se desconectarán.

Parámetros eliminados en Redis OSS 5.0

Nombre	Detalles	Descripción
<code>repl-timeout</code>	<p>Predeterminado: 60</p> <p>Modificable: no</p>	El parámetro no está disponible en esta versión.

Cambios en los parámetros de Redis OSS 4.0.10

Familia del grupo de parámetros: `redis4.0`

Grupos de parámetros predeterminados de Redis OSS 4.0.x

- `default.redis4.0`— Utilice este grupo de parámetros, o uno derivado de él, para los clústeres y grupos de replicación de Redis OSS (modo de clúster desactivado).
- `default.redis4.0.cluster.on`— Utilice este grupo de parámetros, o uno derivado del mismo, para los clústeres y grupos de replicación de Redis OSS (modo de clúster habilitado).

Los parámetros cambiaron en Redis OSS 4.0.10

Nombre	Detalles	Descripción
maxmemory-policy	<p>Valores permitidos: allkeys-lru , volatile-lru , allkeys-lfu , volatile-lfu , allkeys-random , volatile-random , volatile-ttl , noeviction</p> <p>Valor predeterminado: volatile-lru</p> <p>Tipo: String</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	<p>maxmemory-policy se agregó en la versión 2.6.13. En la versión 4.0.10, se agregaron dos nuevos valores permitidos: allkeys-lfu , que expulsará claves utilizando un criterio LFU aproximado, y volatile-lfu , que realizará la expulsión utilizando un criterio de LFU aproximado entre las claves que tienen un vencimiento establecido. En la versión 6.2, cuando se introdujo la familia de nodos r6gd para utilizarla con la organización de datos en niveles, solo las políticas max-memory noeviction , volatile-lru y allkeys-lru se admiten con tipos de nodos r6gd.</p>

Parámetros añadidos en Redis OSS 4.0.10

Nombre	Detalles	Descripción
Parámetros de eliminación asíncrona		
lazyfree-lazy-eviction	<p>Valores permitidos: sí/no</p> <p>Valor predeterminado: no</p> <p>Tipo: booleano</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	Realiza una eliminación asíncrona en las expulsiones.

Nombre	Detalles	Descripción
<code>lazyfree-lazy-expire</code>	Valores permitidos: sí/no Valor predeterminado: no Tipo: booleano Modificable: sí Aplicación de los cambios: inmediatamente	Realiza una eliminación asíncrona en las claves vencidas.
<code>lazyfree-lazy-server-del</code>	Valores permitidos: sí/no Valor predeterminado: no Tipo: booleano Modificable: sí Aplicación de los cambios: inmediatamente	Realiza una eliminación asíncrona de los comandos que actualizan valores.
<code>slave-lazy-flush</code>	Valores permitidos: N/A Valor predeterminado: no Tipo: booleano Modificable: no Aplicación de los cambios: N/A	Realiza un volcado de base de datos asíncrono durante la sincronización de esclavos.

Parámetros de LFU

Nombre	Detalles	Descripción
<code>lfu-log-factor</code>	<p>Valores permitido s: cualquier número entero > 0</p> <p>Predeterminado: 10</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	Configure el factor de registro, que determina el número de aciertos de clave para saturar el contador de claves.
<code>lfu-decay-time</code>	<p>Valores permitidos: cualquier número entero</p> <p>Valor predeterminado: 1</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	El tiempo en minutos para disminuir el contador de claves.
Parámetros de desfragmentación activa		
<code>activedefrag</code>	<p>Valores permitidos: sí/no</p> <p>Valor predeterminado: no</p> <p>Tipo: booleano</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	Desfragmentación activa habilitada.

Nombre	Detalles	Descripción
<code>active-defrag-ignore-bytes</code>	<p>Valores permitidos: 10485760-104857600</p> <p>Predeterminado: 104857600</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	Cantidad mínima de restos de fragmentación para comenzar la desfragmentación activa.
<code>active-defrag-threshold-lower</code>	<p>Valores permitidos: 1-100</p> <p>Predeterminado: 10</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	Porcentaje mínimo de fragmentación para comenzar la desfragmentación activa.
<code>active-defrag-threshold-upper</code>	<p>Valores permitidos: 1-100</p> <p>Predeterminado: 100</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	Porcentaje máximo de fragmentación en el que usará el máximo esfuerzo.

Nombre	Detalles	Descripción
<code>active-defrag-cycle-min</code>	Valores permitidos: 1-75 Predeterminado: 25 Tipo: número entero Modificable: sí Aplicación de los cambios: inmediatamente	Esfuerzo mínimo para desfragmentar en porcentaje de CPU.
<code>active-defrag-cycle-max</code>	Valores permitidos: 1-75 Predeterminado: 75 Tipo: número entero Modificable: sí Aplicación de los cambios: inmediatamente	Esfuerzo máximo para desfragmentar en porcentaje de CPU.
Parámetros de búfer de salida de cliente		
<code>client-query-buffer-limit</code>	Valores permitidos: 1048576-1073741824 Predeterminado: 1073741824 Tipo: número entero Modificable: sí Aplicación de los cambios: inmediatamente	Tamaño máximo de un búfer de consulta de cliente.

Nombre	Detalles	Descripción
<code>proto-max-bulk-len</code>	<p>Valores permitidos: 1048576-536870912</p> <p>Predeterminado: 536870912</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	Tamaño máximo de una sola solicitud de elemento.

Cambios en los parámetros de Redis OSS 3.2.10

Familia del grupo de parámetros: `redis3.2`

ElastiCache (Redis OSS) 3.2.10 no se admiten parámetros adicionales.

Cambios en los parámetros de Redis OSS 3.2.6

Familia del grupo de parámetros: `redis3.2`

En el caso de Redis OSS 3.2.6, no se admiten parámetros adicionales.

Cambios en los parámetros de Redis OSS 3.2.4

Familia del grupo de parámetros: `redis3.2`

A partir de Redis OSS 3.2.4, hay dos grupos de parámetros predeterminados.

- `default.redis3.2`— Al ejecutar Redis OSS 3.2.4, especifique este grupo de parámetros o uno derivado del mismo si desea crear un grupo de replicación de Redis OSS (modo de clúster desactivado) y seguir utilizando las funciones adicionales de Redis OSS 3.2.4.
- `default.redis3.2.cluster.on`— Especifique este grupo de parámetros o uno derivado del mismo cuando desee crear un grupo de replicación de Redis OSS (modo de clúster habilitado).

Temas

- [Nuevos parámetros para Redis OSS 3.2.4](#)

- [Los parámetros se modificaron en Redis OSS 3.2.4 \(mejorada\)](#)

Nuevos parámetros para Redis OSS 3.2.4

Familia del grupo de parámetros: redis3.2

Para Redis OSS 3.2.4, se admiten los siguientes parámetros adicionales.

Nombre	Detalles	Descripción
<code>list-max-ziplist-size</code>	<p>Valor predeterminado: -2</p> <p>Tipo: número entero</p> <p>Modificable: no</p>	<p>Las listas se codifican de una manera especial para ahorrar espacio. El número de entradas permitidas por nodo de lista interna se puede especificar como de tamaño máximo fijo o como un número máximo de elementos. Para establecer un tamaño máximo fijo, use un valor de -5 a -1 teniendo en cuenta lo siguiente:</p> <ul style="list-style-type: none"> • -5: tamaño máximo de 64 Kb; no se recomienda esta opción para cargas de trabajo normales • -4: tamaño máximo de 32 Kb; no se recomienda esta opción • -3: tamaño máximo de 16 Kb; no se recomienda esta opción • -2: tamaño máximo de 8 Kb; se recomienda esta opción • -1: tamaño máximo de 4 Kb; se recomienda esta opción • El uso de un número positivo indican que se puede almacenar hasta dicho número de elementos por cada nodo de lista.

Nombre	Detalles	Descripción
list-compress-depth	<p>Predeterminado: 0</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	<p>Las listas también pueden comprimirse. La profundidad de compresión es el número de nodos de listas comprimidas de listas rápidas de ambos lados de la lista que se excluirán de la compresión. El principio y el final de la lista están siempre sin comprimir para agilizar las operaciones de inserción y extracción. Los valores son los siguientes:</p> <ul style="list-style-type: none"> • 0: deshabilitar toda la compresión. • 1: comenzar a comprimir con el primer nodo a partir del principio y el final. [principio]->nodo->nodo->...->nodo->[final] Se comprimen todos los nodos excepto los nodos [principio] y [final]. • 2: comenzar a comprimir con el segundo nodo a partir del principio y el final. [principio]->[siguiente]->nodo->nodo->...->nodo->[penúltimo]->[final] Los nodos [principio], [siguiente], [penúltimo] y [final] no se comprimen. Todos los demás nodos se comprimen. • etc.

Nombre	Detalles	Descripción
cluster-enabled	<p>Valor predeterminado: no/yes *</p> <p>Tipo: cadena</p> <p>Modificable: no</p>	<p>Indica si se trata de un grupo de replicación de Redis OSS (modo de clúster activado) en modo de clúster (sí) o de un grupo de replicación de Redis OSS (modo de clúster activado) en modo no de clúster (no). Los grupos de replicación de Redis OSS (modo de clúster activado) en modo de clúster pueden particionar sus datos en un máximo de 500 grupos de nodos.</p> <p>* Redis OSS 3.2. x tiene dos grupos de parámetros predeterminados.</p> <ul style="list-style-type: none"> • <code>default.redis3.2</code> : valor predeterminado no. • <code>default.redis3.2.cluster.on</code> : valor predeterminado yes.

Nombre	Detalles	Descripción
<code>cluster-require-full-coverage</code>	Valor predeterminado: no Tipo: booleano Modificable: sí Aplicación de los cambios: inmediatamente	<p>Cuando se establece enyes, los nodos OSS (modo de clúster activado) de Redis en modo de clúster dejan de aceptar consultas si detectan que hay al menos una ranura de hash descubierta (no hay ningún nodo disponible que la sirva). De esta forma, si el clúster está parcialmente inactivo, este deja de estar disponible. El clúster vuelve a estar disponible automáticamente en el momento en que todas las ranuras vuelven a estar cubiertas.</p> <p>Sin embargo, puede que desee que el subconjunto del clúster que funciona siga aceptando consultas de la parte del espacio de clave que sigue cubierta. Para ello, solo tiene que establecer la opción <code>cluster-require-full-coverage</code> en no.</p>

Nombre	Detalles	Descripción
hll-spars e-max-byt es	Predeterminado: 3000 Tipo: número entero Modificable: sí Aplicación de los cambios: inmediatamente	<p>HyperLogLog límite de bytes de representación dispersa. El límite incluye el encabezado de 16 bytes. Cuando el HyperLogLog uso de la representación dispersa cruza este límite, se convierte en la representación densa.</p> <p>No se recomienda usar un valor superior a 16 000, ya que en ese punto, la representación densa es más eficaz desde el punto de vista de la memoria.</p> <p>Se recomienda usar un valor próximo a 3000 con el fin de disponer de los beneficios de la codificación eficaz desde el punto de vista del espacio sin ralentizar PFADD, que está habilitado con la codificación dispersa. El valor se puede aumentar a ~10000 cuando la CPU no es un problema, pero sí el espacio, y el conjunto de datos está compuesto por muchos HyperLogLogs con una cardinalidad en el rango de 0 a 15000.</p>

Nombre	Detalles	Descripción
<code>reserved-memory-percent</code>	<p>Predeterminado: 25</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	<p>Porcentaje de memoria reservada para un nodo y no destinado a los datos. De forma predeterminada, el espacio de datos del OSS de Redis aumenta hasta que consume toda la memoria del nodo. Si esto ocurre, es posible que el desempeño del nodo se vea afectado debido a un exceso de paginación de la memoria. Al reservar memoria, puede reservar parte de la memoria disponible para fines ajenos al OSS de ReDIS, a fin de reducir la cantidad de paginación.</p> <p>Este parámetro es específico y no forma parte de la distribución estándar de Redis OSS. ElastiCache</p> <p>Para obtener más información, consulte <code>reserved-memory</code> y Administración de la memoria reservada.</p>

Los parámetros se modificaron en Redis OSS 3.2.4 (mejorada)

Familia del grupo de parámetros: redis3.2

En el caso de Redis OSS 3.2.4, se modificaron los siguientes parámetros.

Nombre	Detalles	Cambio
<code>activeresharding</code>	<p>Modificable: sí si el grupo de parámetros no está asociado con ningún clúster de caché. De lo contrario, no lo es.</p>	Modificable era No.
<code>databases</code>	<p>Modificable: sí si el grupo de parámetros no está asociado</p>	Modificable era No.

Nombre	Detalles	Cambio
	con ningún clúster de caché. De lo contrario, no lo es.	
appendonly	Valor predeterminado: desactivado Modificable: no	Si desea actualizar desde una versión anterior de Redis OSS, primero debe apagarla. appendonly
appendfsync	Valor predeterminado: desactivado Modificable: no	Si desea actualizar desde una versión anterior de Redis OSS, primero debe appendfsync apagarla.
repl-timeout	Predeterminado: 60 Modificable: no	Ahora, este parámetro no puede modificarse y tiene un valor predeterminado de 60.
tcp-keepalive	Predeterminado: 300	El valor predeterminado anterior era 0.
list-max-ziplist-entries		El parámetro ya no está disponible.
list-max-ziplist-value		El parámetro ya no está disponible.

Se agregaron parámetros a Redis OSS 2.8.24 (mejorada)

Familia del grupo de parámetros: redis2.8

En el caso de Redis OSS 2.8.24, no se admiten parámetros adicionales.

Se agregaron parámetros a Redis OSS 2.8.23 (mejorado)

Familia del grupo de parámetros: redis2.8

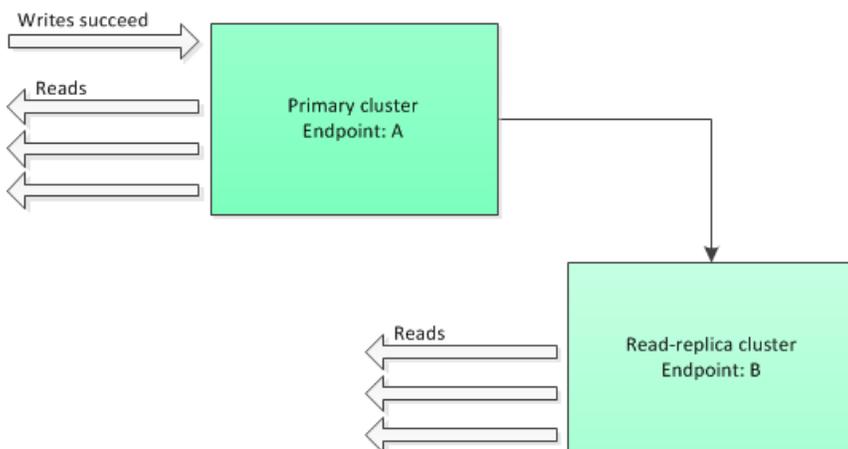
Para Redis OSS 2.8.23 se admite el siguiente parámetro adicional.

Nombre	Detalles	Descripción
<code>close-on-slave-write</code>	Valor predeterminado: <code>yes</code> Tipo: cadena (yes/no) Modificable: sí Aplicación de los cambios: inmediatamente	Si está habilitado, los clientes que intenten escribir en una réplica de solo lectura se desconectarán.

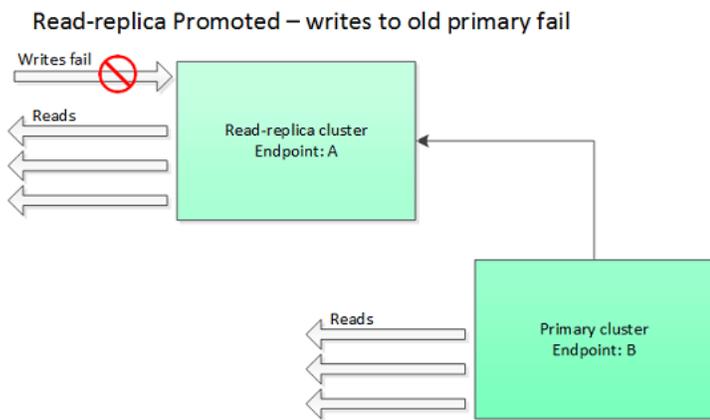
¿Cómo funciona `close-on-slave-write`

Amazon introdujo el `close-on-slave-write` parámetro ElastiCache para darte más control sobre la forma en que responde tu clúster cuando un nodo principal y un nodo de réplica de lectura intercambian funciones debido a la promoción de una réplica de lectura a principal.

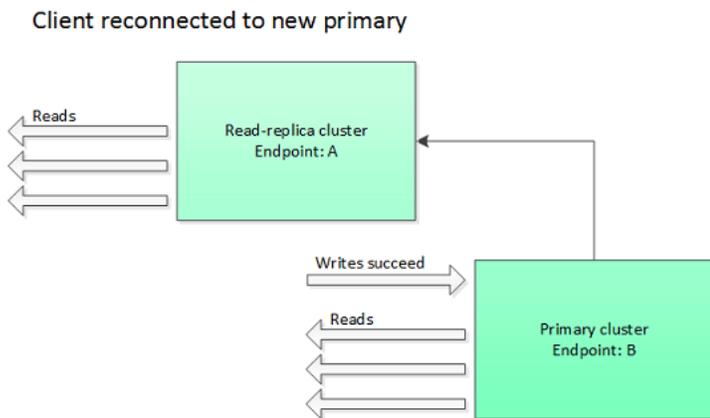
Before read-replica promotion



Si el clúster de réplica de lectura se promociona a nodo principal por cualquier motivo que no sea la conmutación por error de un grupo de reproducción con Multi-AZ habilitadas, el cliente seguirá intentando escribir en el punto de conexión A. Puesto que el punto de conexión A es ahora el punto de conexión de la réplica de lectura, se producirá un error en estas operaciones de escritura. Este es el comportamiento de Redis OSS antes de su ElastiCache introducción `close-on-replica-write` y el comportamiento si lo deshabilita `close-on-replica-write`.



Con el parámetro `close-on-replica-write` habilitado, siempre que un cliente intente escribir en una réplica de lectura, se cerrará la conexión del cliente al clúster. La lógica de su aplicación debe detectar la desconexión, comprobar la tabla de DNS y volver a conectarse al punto de conexión principal, que ahora debería ser el punto de conexión B.



Cuándo puedes deshabilitar `close-on-replica-write`

Si al deshabilitar el parámetro `close-on-replica-write` se producen errores en las operaciones de escritura en el clúster, ¿por qué deshabilitar el parámetro `close-on-replica-write`?

Tal como se ha mencionado anteriormente, con el parámetro `close-on-replica-write` habilitado, siempre que un cliente intenta escribir en una réplica de lectura, se cierra la conexión del cliente al clúster. Establecer una nueva conexión al nodo implica tiempo. Por lo tanto, la desconexión y la reconexión como resultado de una solicitud de escritura en la réplica también afecta la latencia de las solicitudes de lectura que se sirven a través de la misma conexión. Este efecto continúa hasta que se establezca una nueva conexión. Si su aplicación realiza un uso intensivo de operaciones de lectura o es muy sensible a la latencia, es posible que desee mantener sus clientes conectados para que el rendimiento de lectura no se vea afectado.

Se agregaron parámetros a Redis OSS 2.8.22 (mejorado)

Familia del grupo de parámetros: redis2.8

En el caso de Redis OSS 2.8.22, no se admiten parámetros adicionales.

Important

- A partir de la versión 2.8.22 de Redis OSS, `repl-backlog-size` se aplica tanto al clúster principal como a los clústeres de réplica.
- A partir de la versión 2.8.22 de Redis OSS, el parámetro no es compatible. `repl-timeout` Si se cambia, se ElastiCache sobrescribirá con el valor predeterminado (60 segundos), como hacemos con nosotros. `appendonly`

A continuación se describen parámetros que ya no son compatibles.

- `appendonly`
- `appendfsync`
- `repl-timeout`

Redis OSS 2.8.21 agregó parámetros

Familia del grupo de parámetros: redis2.8

Para Redis OSS 2.8.21, no se admiten parámetros adicionales.

Se agregaron parámetros a Redis OSS 2.8.19

Familia del grupo de parámetros: redis2.8

Para Redis OSS 2.8.19 no se admiten parámetros adicionales.

Parámetros añadidos de Redis OSS 2.8.6

Familia del grupo de parámetros: redis2.8

Para Redis OSS 2.8.6, se admiten los siguientes parámetros adicionales.

Nombre	Detalles	Descripción
<code>min-slaves-max-lag</code>	<p>Predeterminado: 10</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	<p>Número de segundos en los que el nodo principal debe recibir una solicitud de ping desde una réplica de lectura. Si este periodo de tiempo transcurre y el nodo principal no recibe ningún ping, la réplica deja de considerarse como disponible. Si el número de réplicas disponibles es inferior <code>min-slaves-to-write</code>, la principal dejará de aceptar escrituras en ese momento.</p> <p>Si este parámetro <code>min-slaves-to-write</code> es 0, el nodo principal siempre aceptará las solicitud es de escritura, incluso si no hay réplicas disponibles.</p>
<code>min-slaves-to-write</code>	<p>Predeterminado: 0</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	<p>Número mínimo de réplicas de lectura que deben estar disponibles para el nodo principal acepte operaciones de escritura de los clientes. Si el número de réplicas disponibles cae por debajo de este número, el nodo principal dejará de aceptar solicitudes de escritura.</p> <p>Si este parámetro <code>min-slaves-max-lag</code> es 0, el nodo principal siempre aceptará las solicitud es de escritura, incluso si no hay réplicas disponibles.</p>

Nombre	Detalles	Descripción
notify-keyspace-events	<p>Valor predeterminado: (cadena vacía)</p> <p>Tipo: String</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	<p>Los tipos de eventos de espacio de claves que Redis OSS puede notificar a los clientes. Cada tipo de evento se representa con una única letra:</p> <ul style="list-style-type: none"> • K: eventos de espacio de claves, publicados con un prefijo <code>__keyspace@<db>__</code> • E: eventos de eventos de claves, publicados con un prefijo <code>__keyevent@<db>__</code> • g: comandos genéricos no específicos, como DEL, EXPIRE, RENAME, etc. • \$: comandos de cadena • l: comandos de lista • s: comandos de establecimiento • h: comandos de hash • z: comandos de conjuntos ordenados • x: eventos caducados (eventos que se generan cada vez que una clave caduca) • e: eventos expulsados (eventos que se generan cuando

Nombre	Detalles	Descripción
		<p>se expulsa una clave para maxmemory)</p> <ul style="list-style-type: none">• A: alias para g\$!shzxe <p>Puede usar cualquier combinación de estos tipos de eventos. Por ejemplo, AKE significa que Redis OSS puede publicar notificaciones de todos los tipos de eventos.</p> <p>No use caracteres distintos de los mencionados anteriormente. Si intenta hacerlo, se mostrarán mensajes de error.</p> <p>De forma predeterminada, este parámetro está definido como una cadena vacía, lo que significa que la notificación de eventos de espacio de claves está deshabilitada.</p>

Nombre	Detalles	Descripción
repl-backlog-size	<p>Predeterminado: 1048576</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediata mente</p>	<p>Tamaño, en bytes, del búfer de tareas pendientes del nodo principal. Las tareas pendientes se usan para registrar actualizaciones de los datos en el nodo principal. Cuando una réplica de lectura se conecta al nodo principal, intenta realizar una sincronización parcial (psync), donde se aplican los datos de las tareas pendientes para ponerse al día con el nodo principal. Si se produce un error en psync, es necesario realizar una sincronización completa.</p> <p>El valor mínimo para este parámetro es 16384.</p> <div data-bbox="1008 1052 1507 1413" style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px;"><p> Note</p><p>A partir de la versión 2.8.22 de Redis OSS, este parámetro se aplica tanto al clúster principal como a las réplicas de lectura.</p></div>

Nombre	Detalles	Descripción
repl-backlog-ttl	Predeterminado: 3600 Tipo: número entero Modificable: sí Aplicación de los cambios: inmediatamente	<p>Número de segundos que el nodo principal conservará el búfer de tareas pendientes. A partir del momento en que se desconecta a el último nodo de réplica, los datos de las tareas pendientes permanecerán intactos hasta que caduque <code>repl-backlog-ttl</code>. Si la réplica no conecta al nodo principal antes de que transcurra este periodo, el nodo principal liberará el búfer de tareas pendientes. Cuando la réplica vuelva a conectarse, tendrá que realizar una sincronización completa con el nodo principal.</p> <p>Si este parámetro se establece en 0, el búfer de tareas pendientes no se liberará nunca.</p>

Nombre	Detalles	Descripción
<code>repl-timeout</code>	<p>Predeterminado: 60</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	<p>Representa el periodo de tiempo de espera, en segundos, para las operaciones siguientes:</p> <ul style="list-style-type: none"> • Transferencia de datos masiva durante la sincronización, desde la perspectiva de la réplica de lectura • Tiempo de espera del nodo principal desde la perspectiva de la réplica • Tiempo de espera de la réplica desde la perspectiva del nodo principal

Parámetros de Redis OSS 2.6.13

Familia del grupo de parámetros: redis2.6

Redis OSS 2.6.13 fue la primera versión de Redis OSS compatible con. ElastiCache En la siguiente tabla se muestran los parámetros compatibles con Redis OSS 2.6.13. ElastiCache

Nombre	Detalles	Descripción
<code>activeresharding</code>	<p>Valor predeterminado: yes</p> <p>Tipo: cadena (yes/no)</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: en el momento de la creación</p>	<p>Determina si desea habilitar la característica de recombinación activa de Redis. La tabla de hash principal se recombina diez veces por segundo; cada operación de recombinación consume 1 milisegundo de tiempo de procesamiento de la CPU.</p> <p>Este valor se establece al crear el grupo de parámetros. Cuando se asigne un nuevo grupo</p>

Nombre	Detalles	Descripción
appendonly	<p>Valor predeterminado: no</p> <p>Tipo: String</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	<p>de parámetros a un clúster, este valor debe ser el mismo tanto en el nuevo grupo de parámetros como en el anterior.</p> <p>Habilita o deshabilita la característica de archivos solo anexados (AOF) de Redis. El AOF captura cualquier comando de Redis OSS que modifique los datos de la memoria caché y se utiliza para recuperarse de determinados errores de nodos.</p> <p>El valor predeterminado es no, lo que significa AOF está desactivado. Establezca este parámetro en yes para habilitar AOF.</p> <p>Para obtener más información, consulte Mitigación de errores.</p> <div data-bbox="829 1003 1507 1367" style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px; margin: 10px 0;"> <p> Note</p> <p>La característica de archivos solo anexados (AOF) no es compatible con los nodos cache.t1.micro y cache.t2.*. Para los nodos de este tipo, se ignora el valor del parámetro appendonly .</p> </div> <div data-bbox="829 1465 1507 1686" style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px; margin: 10px 0;"> <p> Note</p> <p>AOF no se permite en grupos de reproducción con Multi-AZ.</p> </div>

Nombre	Detalles	Descripción
<code>appendfsync</code>	<p>Valor predeterminado: <code>everysec</code></p> <p>Tipo: String</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	<p>Cuando <code>appendonly</code> está establecido en "yes", controla la frecuencia con la que el búfer de salida de AOF se escribe en el disco:</p> <ul style="list-style-type: none"> • <code>no</code>: el búfer vacía en el disco en función de las necesidades. • <code>everysec</code>: el búfer se vacía una vez por segundo. Esta es la opción predeterminada. • <code>always</code>: el búfer se vacía siempre que se modifican datos en el clúster. • <code>Appendfsync</code> no es compatible con las versiones 2.8.22 y posteriores.
<code>client-output-buffer-limit-normal-hard-limit</code>	<p>Predeterminado: 0</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	<p>Si el búfer de salida de un cliente alcanza el número de bytes especificado, el cliente se desconectará. El valor predeterminado es cero (sin límite flexible).</p>
<code>client-output-buffer-limit-normal-soft-limit</code>	<p>Predeterminado: 0</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	<p>Si el búfer de salida de un cliente alcanza el número de bytes especificado, el cliente se desconectará solo si esta condición se mantiene durante <code>client-output-buffer-limit-normal-soft-seconds</code>. El valor predeterminado es cero (sin límite duro).</p>

Nombre	Detalles	Descripción
<code>client-output-buffer-limit-normal-soft-seconds</code>	<p>Predeterminado: 0</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	Si el búfer de salida de un cliente permanece en <code>client-output-buffer-limit-normal-soft-limit</code> bytes por un periodo superior a este número de segundos, el cliente se desconectará. El valor predeterminado es cero (sin límite de tiempo).
<code>client-output-buffer-limit-pubsub-hard-limit</code>	<p>Predeterminado: 33554432</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	Para los clientes de publicación/suscripción de Redis OSS: si el búfer de salida de un cliente alcanza el número de bytes especificado, el cliente se desconectará.
<code>client-output-buffer-limit-pubsub-soft-limit</code>	<p>Predeterminado: 8388608</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	Para los clientes de publicación/suscripción de Redis OSS: si el búfer de salida de un cliente alcanza el número de bytes especificado, el cliente se desconectará, pero solo si esta condición persiste durante un tiempo. <code>client-output-buffer-limit-pubsub-soft-seconds</code>
<code>client-output-buffer-limit-pubsub-soft-seconds</code>	<p>Predeterminado: 60</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	Para los clientes de publicación/suscripción de Redis OSS: si el búfer de salida de un cliente permanece en <code>client-output-buffer-limit-pubsub-soft-limit</code> bytes durante más de este número de segundos, el cliente se desconectará.

Nombre	Detalles	Descripción
<code>client-output-buffer-limit-hard-limit</code>	<p>Valor predeterminado: para obtener información acerca de los valores, consulte Parámetros específicos del tipo de nodo de Redis OSS</p> <p>Tipo: número entero</p> <p>Modificable: no</p>	<p>Para las réplicas de lectura de Redis OSS: si el búfer de salida de un cliente alcanza el número de bytes especificado, el cliente se desconectará.</p>
<code>client-output-buffer-limit-soft-limit</code>	<p>Valor predeterminado: para obtener información acerca de los valores, consulte Parámetros específicos del tipo de nodo de Redis OSS</p> <p>Tipo: número entero</p> <p>Modificable: no</p>	<p>Para las réplicas de lectura del OSS de Redis: si el búfer de salida de un cliente alcanza el número de bytes especificado, el cliente se desconectará, pero solo si esta condición persiste durante un tiempo. <code>client-output-buffer-limit-slave-soft-seconds</code></p>
<code>client-output-buffer-limit-slave-soft-seconds</code>	<p>Predeterminado: 60</p> <p>Tipo: número entero</p> <p>Modificable: no</p>	<p>Para las réplicas de lectura del OSS de Redis: si el búfer de salida de un cliente permanece en <code>client-output-buffer-limit-slave-soft-limit</code> bytes durante más de este número de segundos, el cliente se desconectará.</p>
<code>databases</code>	<p>Predeterminado: 16</p> <p>Tipo: número entero</p> <p>Modificable: no</p> <p>Aplicación de los cambios: en el momento de la creación</p>	<p>Número de particiones lógicas en que se dividen las bases de datos. Se recomienda mantener este valor bajo.</p> <p>Este valor se establece al crear el grupo de parámetros. Cuando se asigne un nuevo grupo de parámetros a un clúster, este valor debe ser el mismo tanto en el nuevo grupo de parámetros como en el anterior.</p>

Nombre	Detalles	Descripción
hash-max-ziplist-entries	Predeterminado: 512 Tipo: número entero Modificable: sí Aplicación de los cambios: inmediatamente	Determina la cantidad de memoria que usan los hash. Los hash con un número de entradas inferior al especificado se almacenan con una codificación especial que permite ahorrar espacio.
hash-max-ziplist-value	Predeterminado: 64 Tipo: número entero Modificable: sí Aplicación de los cambios: inmediatamente	Determina la cantidad de memoria que usan los hash. Los hash con entradas de tamaño inferior al número de bytes especificado se almacenan con una codificación especial que permite ahorrar espacio.
list-max-ziplist-entries	Predeterminado: 512 Tipo: número entero Modificable: sí Aplicación de los cambios: inmediatamente	Determina la cantidad de memoria que se usa para las listas. Las listas con un número de entradas inferior al especificado se almacenan con una codificación especial que permite ahorrar espacio.
list-max-ziplist-value	Predeterminado: 64 Tipo: número entero Modificable: sí Aplicación de los cambios: inmediatamente	Determina la cantidad de memoria que se usa para las listas. Las listas con entradas de tamaño inferior al número de bytes especificado se almacenan con una codificación especial que permite ahorrar espacio.

Nombre	Detalles	Descripción
<code>lua-time-limit</code>	<p>Predeterminado: 5000</p> <p>Tipo: número entero</p> <p>Modificable: no</p>	<p>El tiempo máximo de ejecución de un script de Lua, en milisegundos, antes de que ElastiCache tome medidas para detener el script.</p> <p>Si <code>lua-time-limit</code> se supera, todos los comandos OSS de Redis devolverán un error del formato <code>____-BUSY</code>. Dado que este estado puede interferir con muchas operaciones esenciales del OSS de Redis, primero ElastiCache emitirá un comando <code>SCRIPT KILL</code>. Si esto no funciona, reiniciará ElastiCache Redis OSS por la fuerza.</p>
<p><code>maxclients</code> Este valor se aplica a todos los tipos de instancias, excepto a los especificados.</p>	<p>Predeterminado: 65000</p> <p>Tipo: número entero</p> <p>Modificable: no</p> <p>t2.medium Valor predeterminado: 20000</p> <p>Tipo: número entero</p> <p>Modificable: no</p> <p>t2.small Valor predeterminado: 20000</p> <p>Tipo: número entero</p> <p>Modificable: no</p>	<p>Número máximo de clientes que pueden conectarse a la vez.</p>

Nombre	Detalles	Descripción
	<p>t2.micro Valor predeterminado: 20000</p> <p>Tipo: número entero</p> <p>Modificable: no</p>	
	<p>t4g.micro Valor predeterminado: 20000</p> <p>Tipo: número entero</p> <p>Modificable: no</p>	
	<p>t3.medium Valor predeterminado: 46000</p> <p>Tipo: número entero</p> <p>Modificable: no</p>	
	<p>t3.small Valor predeterminado: 46000</p> <p>Tipo: número entero</p> <p>Modificable: no</p>	
	<p>t3.micro Valor predeterminado: 20000</p> <p>Tipo: número entero</p> <p>Modificable: no</p>	

Nombre	Detalles	Descripción
<code>maxmemory-policy</code>	<p>Valor predeterminado: <code>volatile-lru</code></p> <p>Tipo: String</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	<p>Política de expulsión de claves cuando se alcanza el uso máximo de la memoria.</p> <p>Los valores válidos son: <code>volatile-lru</code> <code>allkeys-lru</code> <code>volatile-random</code> <code>allkeys-random</code> <code>volatile-ttl</code> <code>noeviction</code></p> <p>Para obtener más información, consulte Uso de Redis OSS como caché LRU.</p>
<code>maxmemory-samples</code>	<p>Valor predeterminado: 3</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	<p>Para los cálculos <code>least-recently-used</code> (LRU) y <code>time-to-live</code> (TTL), este parámetro representa el tamaño de la muestra de las claves que se van a comprobar. De forma predeterminada, Redis OSS elige 3 claves y usa la que se usó menos recientemente.</p>

Nombre	Detalles	Descripción
<code>reserved-memory</code>	<p>Predeterminado: 0</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	<p>Memoria total, en bytes, reservado para elementos que no son datos. De forma predeterminada, el nodo OSS de Redis crecerá hasta consumir el nodo <code>maxmemory</code> (consulte Parámetros específicos del tipo de nodo de Redis OSS). Si esto ocurre, es posible que el desempeño del nodo se vea afectado debido a un exceso de paginación de la memoria. Al reservar memoria, puede reservar parte de la memoria disponible para fines ajenos al OSS de ReDIS, a fin de reducir la cantidad de paginación.</p> <p>Este parámetro es específico y no forma parte de la distribución estándar de Redis OSS. ElastiCache</p> <p>Para obtener más información, consulte <code>reserved-memory-percent</code> y Administración de la memoria reservada.</p>
<code>set-max-intset-entries</code>	<p>Predeterminado: 512</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	<p>Determina la cantidad de memoria que se usa para determinados tipos de conjuntos (cadenas que son enteros en base 10 en el rango de enteros con signo de 64 bits). Estos conjuntos con un número de entradas inferior al especificado se almacenan con una codificación especial que permite ahorrar espacio.</p>
<code>slave-allow-chaining</code>	<p>Valor predeterminado: no</p> <p>Tipo: String</p> <p>Modificable: no</p>	<p>Determina si una réplica de lectura en Redis OSS puede tener réplicas de lectura propias.</p>

Nombre	Detalles	Descripción
slowlog-log-slower-than	Predeterminado: 10000 Tipo: número entero Modificable: sí Aplicación de los cambios: inmediatamente	El tiempo máximo de ejecución, en microsegundos, para que la función OSS Slow Log de Redis registre los comandos.
slowlog-max-len	Valor predeterminado: 128 Tipo: número entero Modificable: sí Aplicación de los cambios: inmediatamente	La longitud máxima del registro lento del OSS de Redis.
tcp-keepalive	Predeterminado: 0 Tipo: número entero Modificable: sí Aplicación de los cambios: inmediatamente	Si se establecen un valor distinto de cero (N), los clientes de nodo se sondearán cada N segundos para asegurarse de que siguen conectados. Con el valor predeterminado 0, el sondeo se desactiva. <div data-bbox="829 1230 1507 1591" style="border: 1px solid #f08080; border-radius: 10px; padding: 10px; margin-top: 10px;"> <p> Important</p> <p>Algunos aspectos de este parámetro cambiaron en la versión 3.2.4 de Redis OSS. Consulte Los parámetros se modificaron en Redis OSS 3.2.4 (mejorada).</p> </div>

Nombre	Detalles	Descripción
<code>timeout</code>	<p>Predeterminado: 0</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	<p>Número de segundos que un nodo espera antes de caducar. Valores son los siguientes:</p> <ul style="list-style-type: none"> • 0: no desconectar nunca un cliente inactivo. • 1-19: valores no válidos. • ≥ 20: número de segundos que un nodo espera antes de desconectar un cliente inactivo.
<code>zset-max-ziplist-entries</code>	<p>Valor predeterminado: 128</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	<p>Determina la cantidad de memoria que se usa para los conjuntos ordenados. Los conjuntos ordenados con un número de elementos inferior al especificado se almacenan con una codificación especial que permite ahorrar espacio.</p>
<code>zset-max-ziplist-value</code>	<p>Predeterminado: 64</p> <p>Tipo: número entero</p> <p>Modificable: sí</p> <p>Aplicación de los cambios: inmediatamente</p>	<p>Determina la cantidad de memoria que se usa para los conjuntos ordenados. Los conjuntos ordenados con entradas de tamaño inferior al número de bytes especificado se almacenan con una codificación especial que permite ahorrar espacio.</p>

Note

Si no especifica un grupo de parámetros para el clúster de Redis OSS 2.6.13, se utilizará un grupo de parámetros predeterminado (`default.redis2.6`). No puede cambiar los valores de ninguno de los parámetros del grupo de parámetros predeterminado; sin embargo, siempre podrá crear un grupo de parámetros predeterminado y asignarlo a su clúster en cualquier momento.

Parámetros específicos del tipo de nodo de Redis OSS

Aunque la mayoría de los parámetros tienen un único valor, algunos parámetros tienen distintos valores en función del tipo de nodo que se use. La tabla siguiente muestra los valores predeterminados de los parámetros `maxmemory`, `client-output-buffer-limit-slave-hard-limit` y `client-output-buffer-limit-slave-soft-limit` para cada tipo de nodo. El valor de `maxmemory` es el número máximo de bytes disponibles para el uso, los datos y otros usos en el nodo. Para obtener más información, consulte [Memoria disponible](#).

Note

El parámetro `maxmemory` no se puede modificar.

Tipo de nodo	Maxmemory	client-output-buffer-limit-slave-hard-limit	client-output-buffer-limit-slave-soft-limit
cache.t1.micro	142606336	14260633	14260633
cache.t2.micro	581959680	58195968	58195968
cache.t2.small	1665138688	166513868	166513868
cache.t2.medium	3461349376	346134937	346134937
cache.t3.micro	536870912	53687091	53687091
cache.t3.small	1471026299	147102629	147102629
cache.t3.medium	3317862236	331786223	331786223
cache.t4g.micro	536870912	53687091	53687091
cache.t4g.small	1471026299	147102629	147102629
cache.t4g.medium	3317862236	331786223	331786223
cache.m1.small	943718400	94371840	94371840

Tipo de nodo	Maxmemory	lient-output-buffer-limitC - slave-hard-limit	C lient-output-buffe r-limit - slave-soft-limit
cache.m1.medium	3093299200	309329920	309329920
cache.m1.large	7025459200	702545920	702545920
cache.m1.xlarge	14889779200	1488977920	1488977920
cache.m2.xlarge	17091788800	1709178880	1709178880
cache.m2.2xlarge	35022438400	3502243840	3502243840
cache.m2.4xlarge	70883737600	7088373760	7088373760
cache.m3.medium	2988441600	309329920	309329920
cache.m3.large	6501171200	650117120	650117120
cache.m3.xlarge	14260633600	1426063360	1426063360
cache.m3.2xlarge	29989273600	2998927360	2998927360
cache.m4.large	6892593152	689259315	689259315
cache.m4.xlarge	15328501760	1532850176	1532850176
cache.m4.2xlarge	31889126359	3188912636	3188912636
cache.m4.4xlarge	65257290629	6525729063	6525729063
cache.m4.10xlarge	166047614239	16604761424	16604761424
cache.m5.large	6854542746	685454275	685454275
cache.m5.xlarge	13891921715	1389192172	1389192172
cache.m5.2xlarge	27966669210	2796666921	2796666921
cache.m5.4xlarge	56116178125	5611617812	5611617812

Tipo de nodo	Maxmemory	lient-output-buffer-limitC - slave-hard-limit	C lient-output-buffe r-limit - slave-soft-limit
cache.m5.12xlarge	168715971994	16871597199	16871597199
cache.m5.24xlarge	337500562842	33750056284	33750056284
cache.m6g.large	6854542746	685454275	685454275
cache.m6g.xlarge	13891921715	1389192172	1389192172
cache.m6g.2xlarge	27966669210	2796666921	2796666921
cache.m6g.4xlarge	56116178125	5611617812	5611617812
cache.m6g.8xlarge	111325552312	11132555231	11132555231
cache.m6g.12xlarge	168715971994	16871597199	16871597199
cache.m6g.16xlarge	225000375228	22500037523	22500037523
cache.c1.xlarge	6501171200	650117120	650117120
cache.r3.large	14470348800	1468006400	1468006400
cache.r3.xlarge	30513561600	3040870400	3040870400
cache.r3.2xlarge	62495129600	6081740800	6081740800
cache.r3.4xlarge	126458265600	12268339200	12268339200
cache.r3.8xlarge	254384537600	24536678400	24536678400
cache.r4.large	13201781556	1320178155	1320178155
cache.r4.xlarge	26898228839	2689822883	2689822883
cache.r4.2xlarge	54197537997	5419753799	5419753799
cache.r4.4xlarge	108858546586	10885854658	10885854658

Tipo de nodo	Maxmemory	lient-output-buffer-limitC - slave-hard-limit	C lient-output-buffe r-limit - slave-soft-limit
cache.r4.8xlarge	218255432090	21825543209	21825543209
cache.r4.16xlarge	437021573120	43702157312	43702157312
cache.r5.large	14037181030	1403718103	1403718103
cache.r5.xlarge	28261849702	2826184970	2826184970
cache.r5.2xlarge	56711183565	5671118356	5671118356
cache.r5.4xlarge	113609865216	11360986522	11360986522
cache.r5.12xlarge	341206346547	34120634655	34120634655
cache.r5.24xlarge	682485973811	68248597381	68248597381
cache.r6g.large	14037181030	1403718103	1403718103
cache.r6g.xlarge	28261849702	2826184970	2826184970
cache.r6g.2xlarge	56711183565	5671118356	5671118356
cache.r6g.4xlarge	113609865216	11360986522	11360986522
cache.r6g.8xlarge	225000375228	22500037523	22500037523
cache.r6g.12xlarge	341206346547	34120634655	34120634655
cache.r6g.16xlarge	450000750456	45000075046	45000075046
cache.r6gd.xlarge	28261849702	2826184970	2826184970
cache.r6gd.2xlarge	56711183565	5671118356	5671118356
cache.r6gd.4xlarge	113609865216	11360986522	11360986522
cache.r6gd.8xlarge	225000375228	22500037523	22500037523

Tipo de nodo	Maxmemory	lient-output-buffer-limitC - slave-hard-limit	C lient-output-buffe r-limit - slave-soft-limit
cache.r6gd.12xlarge	341206346547	34120634655	34120634655
cache.r6gd.16xlarge	450000750456	45000075046	45000075046
cache.r7g.large	14037181030	1403718103	1403718103
cache.r7g.xlarge	28261849702	2826184970	2826184970
cache.r7g.2xlarge	56711183565	5671118356	5671118356
cache.r7g.4xlarge	113609865216	11360986522	11360986522
cache.r7g.8xlarge	225000375228	22500037523	22500037523
cache.r7g.12xlarge	341206346547	34120634655	34120634655
cache.r7g.16xlarge	450000750456	45000075046	45000075046
cache.m7g.large	6854542746	685454275	685454275
cache.m7g.xlarge	13891921715	1389192172	1389192172
cache.m7g.2xlarge	27966669210	2796666921	2796666921
cache.m7g.4xlarge	56116178125	5611617812	5611617812
cache.m7g.8xlarge	111325552312	11132555231	11132555231
cache.m7g.12xlarge	168715971994	16871597199	16871597199
cache.m7g.16xlarge	225000375228	22500037523	22500037523
cache.c7gn.large	3317862236	1403718103	1403718103
cache.c7gn.xlarge	6854542746	2826184970	2826184970
cache.c7gn.2xlarge	13891921715	5671118356	5671118356

Tipo de nodo	Maxmemory	lient-output-buffer-limitC - slave-hard-limit	C lient-output-buffe r-limit - slave-soft-limit
cache.c7gn.4xlarge	27966669210	11360986522	11360986522
cache.c7gn.8xlarge	56116178125	22500037523	22500037523
cache.c7gn.12xlarge	84357985997	34120634655	34120634655
cache.c7gn.16xlarge	113609865216	45000075046	45000075046

Note

Todos los tipos de instancias de la generación actual se crean en Amazon Virtual Private Cloud (VPC) de forma predeterminada.

Las instancias T1 no admiten Multi-AZ.

Las instancias T1 y T2 no admiten el AOF de Redis OSS.

Las variables de configuración de Redis OSS no `appendfsync` son compatibles con la versión 2.8.22 `appendonly` y posteriores de Redis OSS.

Escalado ElastiCache (Redis OSS)

Escalado sin servidor ElastiCache

ElastiCache Serverless acomoda automáticamente el tráfico de su carga de trabajo a medida que aumenta o disminuye. Para cada caché ElastiCache sin servidor, realiza un seguimiento continuo del uso de recursos como la CPU, la memoria y la red. Cuando alguno de estos recursos está limitado, ElastiCache Serverless amplía su capacidad añadiendo un nuevo fragmento y redistribuyendo los datos al nuevo fragmento, sin que la aplicación pierda tiempo de inactividad. Puede supervisar los recursos que consume la memoria caché supervisando la `BytesUsedForCache` métrica del almacenamiento de datos en CloudWatch la memoria caché y `ElastiCacheProcessingUnits` (ECPU) del uso informático.

Establecimiento de límites de escalado para administrar los costes

A fin de controlar el coste de la caché, puede configurar un uso máximo del almacenamiento de datos en caché y de la métrica ECPU/segundo para su caché. Si lo hace, se asegurará de que el uso de la memoria caché nunca supere el máximo configurado.

Si establece un escalado máximo, es posible que su aplicación disminuya el rendimiento de la caché cuando esta alcance el máximo. Cuando estableces un máximo de almacenamiento de datos en caché y el almacenamiento de datos en caché alcanza el máximo, ElastiCache empezará a desalojar los datos de la caché que tengan un tiempo de vida (TTL) establecido, mediante la lógica LRU. Si no hay datos que se puedan expulsar, las solicitudes para escribir datos adicionales recibirán un mensaje de error por memoria insuficiente (OOM). Cuando establezca un máximo de ECPU/segundo y la utilización informática de la carga de trabajo supere este valor, empezará a limitar las solicitudes de Redis OSS. ElastiCache

Si configura un límite máximo en `BytesUsedForCache` o `ElastiCacheProcessingUnits`, le recomendamos encarecidamente que configure una CloudWatch alarma con un valor inferior al límite máximo para que se le notifique cuando la memoria caché esté funcionando cerca de estos límites. Le recomendamos configurar una alarma al 75 % del límite máximo que haya establecido. Consulte la documentación sobre cómo configurar CloudWatch las alarmas.

Escalado previo con Serverless ElastiCache

ElastiCache Escalado previo sin servidor

Con el preescalado, también denominado precalentamiento, puede establecer los límites mínimos admitidos para la memoria caché. ElastiCache Puede establecer estos mínimos para las unidades de ElastiCache procesamiento (ECPUs) por segundo o para el almacenamiento de datos. Esto puede resultar útil para prepararse para los eventos de escalado previstos. Por ejemplo, si una empresa de videojuegos prevé multiplicar por cinco el número de inicios de sesión en el primer minuto de lanzamiento de su nuevo juego, puede preparar su caché para este importante aumento de uso.

Puede realizar el escalado previo mediante la ElastiCache consola, la CLI o la API. ElastiCache Serverless actualiza las ECPUs disponibles por segundo en la memoria caché en 60 minutos y envía una notificación de evento cuando se completa la actualización del límite mínimo.

Cómo funciona el escalado previo

Cuando el límite mínimo de ECPU/segundo o almacenamiento de datos se actualiza mediante la consola, la CLI o la API, ese nuevo límite estará disponible en 1 hora. ElastiCache Serverless admite

30 000 ECPU/segundo en una caché vacía y hasta 90 000 ECPUs/segundo cuando se utiliza la función de lectura desde réplica. ElastiCache puede duplicar las ECPU/segundo cada 10-12 minutos. Esta velocidad de escalado es suficiente para la mayoría de las cargas de trabajo. Si prevé que un próximo evento de escalado podría superar esta tasa, le recomendamos que establezca el número mínimo de ECPU/segundo en el máximo de ECPUs/segundo que espere al menos 60 minutos antes del pico. De lo contrario, es posible que la aplicación experimente una latencia elevada y se limiten las solicitudes.

Una vez completada la actualización del límite mínimo, ElastiCache Serverless empezará a calcularle las nuevas ECPUs mínimas por segundo o el nuevo almacenamiento mínimo. Esto ocurre incluso si la aplicación no ejecuta las solicitudes en la memoria caché o si el uso del almacenamiento de datos es inferior al mínimo. Al reducir el límite mínimo con respecto a su configuración actual, la actualización es inmediata, por lo que ElastiCache Serverless empezará a medir el nuevo límite mínimo de forma inmediata.

Note

- Cuando estableces un límite de uso mínimo, se te cobrará por ese límite incluso si tu uso real es inferior al límite de uso mínimo. El uso de la ECPU o del almacenamiento de datos que supere el límite de uso mínimo se cobrará la tarifa normal. Por ejemplo, si estableces un límite de uso mínimo de 100 000 ECPU/segundo, se te cobrará al menos 1,224\$ por hora (utilizando los precios de la ECPU en us-east-1), incluso si tu uso es inferior al mínimo establecido.
- ElastiCache Serverless admite la escala mínima solicitada a nivel agregado en la memoria caché. ElastiCache Serverless también admite un máximo de 30 000 ECPU/segundo por ranura (90 000 ECPU/segundo cuando se utiliza Read from Replica con conexiones READONLY). Como práctica recomendada, su aplicación debe garantizar que la distribución de claves entre las ranuras OSS de Redis y el tráfico entre las claves sean lo más uniformes posible.

Establecer límites de escalado mediante la consola y AWS CLI

Establecer límites de escalado mediante la AWS consola

1. Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en <https://console.aws.amazon.com/elasticache/>.

2. En el panel de navegación, elija el motor que se ejecuta en la caché que desea modificar.
3. Aparecerá una lista de las cachés que ejecutan el motor elegido.
4. Elija la caché que desee modificar seleccionando el botón de opción (a la izquierda del nombre de la caché).
5. Elija Actions (Acciones) y después Modify (Modificar).
6. En Límites de uso, establece los límites de memoria o cómputo adecuados.
7. Haga clic en Vista previa de los cambios y seleccione Guardar los cambios.

Definir los límites de escalado mediante el AWS CLI

Para cambiar los límites de escalado mediante la CLI, utilice la `modify-serverless-cache` API.

Linux:

```
aws elasticache modify-serverless-cache --serverless-cache-name <cache name> \  
--cache-usage-limits 'DataStorage={Minimum=10,Maximum=100,Unit=GB},  
ECPUPerSecond={Minimum=1000,Maximum=100000}'
```

Windows:

```
aws elasticache modify-serverless-cache --serverless-cache-name <cache name> ^  
--cache-usage-limits 'DataStorage={Minimum=10,Maximum=100,Unit=GB},  
ECPUPerSecond={Minimum=1000,Maximum=100000}'
```

Eliminación de los límites de escalado mediante la CLI

Para eliminar los límites de escalado mediante la CLI, establezca los parámetros de límite mínimo y máximo en 0.

Linux:

```
aws elasticache modify-serverless-cache --serverless-cache-name <cache name> \  
--cache-usage-limits 'DataStorage={Minimum=0,Maximum=0,Unit=GB},  
ECPUPerSecond={Minimum=0,Maximum=0}'
```

Windows:

```
aws elasticache modify-serverless-cache --serverless-cache-name <cache name> ^
```

```
--cache-usage-limits 'DataStorage={Minimum=0,Maximum=0,Unit=GB},
ECPUPerSecond={Minimum=0,Maximum=0}'
```

Escalar clústeres de ElastiCache diseño propio (RedisOSS)

La cantidad de datos que necesita su aplicación para procesar casi nunca es fija. Aumenta y disminuye a medida que su negocio crece o experimenta las fluctuaciones normales de la demanda. Si administra por sí mismo su caché, necesita aprovisionar suficiente hardware para los picos de demanda, lo cual puede resultar caro. Al usar Amazon ElastiCache, puedes escalar para satisfacer la demanda actual y pagar solo por lo que usas. ElastiCache le permite escalar su caché para adaptarla a la demanda.

Note

Si un OSS clúster de Redis se replica en una o más regiones, esas regiones se escalan en orden. Al escalar, las regiones secundarias se escalan primero y, después, la región principal. Al reducir la escala, la región principal aparece primero y, a continuación, las regiones secundarias.

Al actualizar la versión del motor, el orden es la región secundaria y, a continuación, la región principal.

Lo siguiente lo ayuda a encontrar el tema correcto para las acciones de escalado que desea realizar.

Escalar los clústeres de Redis OSS

Acción	Redis OSS (modo de clúster desactivado)	Redis OSS (modo de clúster activado)
Escalado descendente	Eliminación de nodos de un clúster	Escalar clústeres en Redis OSS (modo de clúster habilitado)
Escalado ascendente	Adición de nodos a un clúster	Refragmentación y reequilibrio de particiones en línea para Redis OSS (modo de clúster habilitado)
Cambios de tipos de nodos	A un tipo de nodo más grande:	

Acción	Redis OSS (modo de clúster desactivado)	Redis OSS (modo de clúster activado)
	<ul style="list-style-type: none"> • Ampliación de clústeres de un solo nodo para Redis OSS (modo de clúster desactivado) • Ampliación de los clústeres de Redis OSS con réplicas <p>A un tipo de nodo más pequeño:</p> <ul style="list-style-type: none"> • Reducir el tamaño de los clústeres Redis OSS de un solo nodo • Reducir los clústeres de Redis OSS con réplicas 	Escalado vertical en línea mediante la modificación del tipo de nodo
Cambio del número de grupos de nodos	No es compatible con los clústeres de Redis OSS (modo de clúster desactivado)	Escalar clústeres en Redis OSS (modo de clúster habilitado)

Temas

- [Escalar clústeres para Redis OSS \(modo de clúster desactivado\)](#)
- [Escalar clústeres en Redis OSS \(modo de clúster habilitado\)](#)

Escalar clústeres para Redis OSS (modo de clúster desactivado)

Los clústeres de Redis OSS (modo de clúster desactivado) pueden ser clústeres de un solo nodo con 0 fragmentos o clústeres de varios nodos con 1 fragmento. Los clústeres de un solo nodo utilizan el nodo para las lecturas y las escrituras. Los clústeres de varios nodos siempre tienen 1 nodo como nodo primario de lectura/escritura con entre 0 y 5 nodos de réplica de solo lectura.

Contenido

- [Escalado de clústeres de un solo nodo para Redis OSS \(modo de clúster desactivado\)](#)
 - [Ampliación de clústeres de un solo nodo para Redis OSS \(modo de clúster desactivado\)](#)
 - [Ampliación de clústeres de un solo nodo para Redis OSS \(modo de clúster desactivado\) \(consola\)](#)
 - [Ampliación de los clústeres de caché de Redis OSS de un solo nodo \(AWS CLI\)](#)
 - [Ampliación de los clústeres de caché \(API\) de Redis OSS de un solo nodo ElastiCache](#)
 - [Reducir el tamaño de los clústeres Redis OSS de un solo nodo](#)
 - [Reducir el tamaño de un clúster de Redis OSS de un solo nodo \(consola\)](#)
 - [Reducir el tamaño de los clústeres de caché de Redis OSS de un solo nodo \(AWS CLI\)](#)
 - [Reducir los clústeres de caché \(API\) de Redis OSS de un solo nodo ElastiCache](#)
- [Escalado de clústeres de Redis OSS \(modo de clúster desactivado\) con nodos de réplica](#)
 - [Ampliación de los clústeres de Redis OSS con réplicas](#)
 - [Reducir los clústeres de Redis OSS con réplicas](#)
 - [Aumento de la capacidad de lectura](#)
 - [Reducción de la capacidad de lectura](#)

Escalado de clústeres de un solo nodo para Redis OSS (modo de clúster desactivado)

Los nodos OSS de Redis (modo de clúster desactivado) deben ser lo suficientemente grandes como para contener todos los datos de la caché, además de la sobrecarga de Redis OSS. Para cambiar la capacidad de datos de su clúster de Redis OSS (modo de clúster desactivado), debe escalarlos verticalmente: ampliarlos a un tipo de nodo más grande para aumentar la capacidad de datos o reducirlos a un tipo de nodo más pequeño para reducir la capacidad de datos.

El proceso de ampliación ElastiCache (Redis OSS) está diseñado para hacer todo lo posible por conservar los datos existentes y requiere una replicación correcta de Redis OSS. Para los clústeres

de Redis OSS (modo de clúster desactivado), recomendamos disponer de memoria suficiente para Redis OSS.

No puede particionar los datos en varios clústeres de Redis OSS (modo de clúster desactivado). Sin embargo, si solo necesita aumentar o disminuir la capacidad de lectura del clúster, puede crear un clúster de Redis OSS (modo de clúster desactivado) con nodos de réplica y añadir o eliminar réplicas de lectura. Para crear un clúster de Redis OSS (modo de clúster desactivado) con nodos de réplica utilizando el clúster de caché de Redis OSS de un solo nodo como clúster principal, consulte [Creación de un clúster de Redis OSS \(modo de clúster desactivado\) \(consola\)](#)

Después de crear el clúster con réplicas, puede aumentar la capacidad de lectura agregando réplicas de lectura. Después, si lo necesita, puede reducir la capacidad de lectura quitando réplicas de lectura. Para obtener más información, consulte [Aumento de la capacidad de lectura](#) o [Reducción de la capacidad de lectura](#).

Además de poder escalar la capacidad de lectura, los clústeres OSS (modo de clúster desactivado) de Redis con réplicas ofrecen otras ventajas empresariales. Para obtener más información, consulte [Alta disponibilidad a través de grupos de reproducción](#).

Important

Si su grupo de parámetros suele `reserved-memory` reservar memoria para la sobrecarga de Redis OSS, antes de empezar a escalar, asegúrese de tener un grupo de parámetros personalizado que reserve la cantidad correcta de memoria para el nuevo tipo de nodo. De forma alternativa, puede modificar un grupo de parámetros personalizado para que use `reserved-memory-percent` y utilizar ese grupo de parámetros para su nuevo clúster. Si va a utilizar `reserved-memory-percent`, esto no es necesario. Para obtener más información, consulte [Administración de la memoria reservada](#).

Temas

- [Ampliación de clústeres de un solo nodo para Redis OSS \(modo de clúster desactivado\)](#)
- [Reducir el tamaño de los clústeres Redis OSS de un solo nodo](#)

Ampliación de clústeres de un solo nodo para Redis OSS (modo de clúster desactivado)

Al escalar un clúster de Redis OSS de un solo nodo, ElastiCache realiza el siguiente proceso, ya sea que utilice la ElastiCache consola, la API o la API AWS CLI. ElastiCache

1. Un nuevo clúster de caché con el nuevo tipo de nodo se activa en la misma zona de disponibilidad que el clúster de caché existente.
2. Los datos de caché del clúster de caché existente se copian en el nuevo clúster de caché. El tiempo que tarda este proceso depende de su tipo de nodo y de la cantidad de datos que haya en el clúster de caché.
3. Las operaciones de lectura y escritura se sirven ahora utilizando el nuevo clúster de caché. Puesto que los puntos de enlace del nuevo clúster de caché son los mismos que los que había en el clúster de caché anterior, no es necesario actualizar los puntos de enlace de su aplicación. Observará una breve interrupción (algunos segundos) de las operaciones de lectura y escritura del nodo primario mientras se actualiza la entrada de DNS.
4. ElastiCache elimina el clúster de caché anterior. Observará una breve interrupción (unos segundos) de las operaciones de lectura y escritura del nodo anterior porque se desconectarán las conexiones con el nodo anterior.

Note

Para los clústeres que ejecutan el tipo de nodo r6gd, solo puede escalar a tamaños de nodo dentro de la familia de nodos r6gd.

Como se muestra en la siguiente tabla, la operación de ampliación del OSS de Redis se bloquea si tiene una actualización del motor programada para el próximo período de mantenimiento. Para obtener más información acerca de los periodos de mantenimiento, consulte [Administración del mantenimiento](#).

Operaciones bloqueadas de Redis OSS

Operaciones pendientes	Operaciones bloqueadas
Escalado ascendente	Actualización del motor inmediata
Actualización del motor	Ampliación inmediata

Operaciones pendientes	Operaciones bloqueadas
Ampliación y actualización del motor	Ampliación inmediata
	Actualización del motor inmediata

Si hay una operación pendiente que esté bloqueando, tiene las siguientes opciones.

- Programe la operación de ampliación de Redis OSS para el siguiente período de mantenimiento desactivando la casilla Aplicar inmediatamente (uso de CLI: `--no-apply-immediately`, uso de API: `ApplyImmediately=false`)
- Espere hasta el próximo período de mantenimiento (o hasta después) para realizar la operación de ampliación de Redis OSS.
- Agregue la actualización del motor OSS de Redis a esta modificación del clúster de caché con la casilla Aplicar inmediatamente seleccionada (uso de CLI: `--apply-immediately`, uso de API: `ApplyImmediately=true`). Esto desbloqueará su operación de ampliación, ya que hará que la actualización del motor se realice inmediatamente.

Puede ampliar un clúster de Redis OSS (modo de clúster desactivado) de un solo nodo mediante la ElastiCache consola, la o la API AWS CLI. ElastiCache

Important

Si su grupo de parámetros solía `reserved-memory` reservar memoria para la sobrecarga de Redis OSS, antes de empezar a escalar, asegúrese de tener un grupo de parámetros personalizado que reserve la cantidad correcta de memoria para el nuevo tipo de nodo. De forma alternativa, puede modificar un grupo de parámetros personalizado para que use `reserved-memory-percent` y utilizar ese grupo de parámetros para su nuevo clúster. Si va a utilizar `reserved-memory-percent`, esto no es necesario. Para obtener más información, consulte [Administración de la memoria reservada](#).

Ampliación de clústeres de un solo nodo para Redis OSS (modo de clúster desactivado) (consola)

El siguiente procedimiento describe cómo escalar un clúster de Redis OSS de un solo nodo mediante la consola de administración. ElastiCache Durante este proceso, el clúster de Redis OSS seguirá atendiendo las solicitudes con un tiempo de inactividad mínimo.

Para ampliar un clúster de Redis OSS de un solo nodo (consola)

1. [Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en https://console.aws.amazon.com/elasticache/.](https://console.aws.amazon.com/elasticache/)
2. En el panel de navegación, elija los clústeres de Redis OSS.
3. En la lista de clústeres, elija el clúster que desee ampliar (debe ejecutar el motor OSS de Redis, no el motor OSS de Redis agrupado).
4. Elija Modificar.
5. En el asistente Modify Cluster:
 - a. Elija el tipo de nodo que desee ampliar en la lista Node type.
 - b. Si va a utilizar `reserved-memory` para administrar su memoria, en la lista Parameter Group, elija el grupo de parámetros personalizado que reserve la cantidad correcta de memoria para su nuevo tipo de nodo.
6. Si desea realizar el proceso de ampliación de forma inmediata, elija el cuadro Apply immediately. Si no selecciona el cuadro Apply immediately, el proceso de ampliación se realizará durante el siguiente periodo de mantenimiento de este clúster.
7. Elija Modify (Modificar).

Si eligió Apply immediately (Aplicar inmediatamente) en el paso anterior, el estado del clúster cambiará a `modifying` (en modificación). Cuando el estado cambie a `available` (disponible), la modificación se habrá completado y podrá empezar a utilizar el nuevo clúster.

Ampliación de los clústeres de caché de Redis OSS de un solo nodo (AWS CLI)

El siguiente procedimiento describe cómo escalar un clúster de caché Redis OSS de un solo nodo mediante AWS CLI. Durante este proceso, el clúster de OSS de Redis seguirá atendiendo las solicitudes con un tiempo de inactividad mínimo.

Para ampliar un clúster de caché de Redis OSS de un solo nodo (AWS CLI)

1. Determine los tipos de nodos a los que puede escalar ejecutando el AWS CLI `list-allowed-node-type-modifications` comando con el siguiente parámetro.
 - `--cache-cluster-id`

Para Linux, macOS o Unix:

```
aws elasticache list-allowed-node-type-modifications \  
  --cache-cluster-id my-cache-cluster-id
```

Para Windows:

```
aws elasticache list-allowed-node-type-modifications ^  
  --cache-cluster-id my-cache-cluster-id
```

La salida del comando anterior es similar a la siguiente (formato JSON).

```
{  
  "ScaleUpModifications": [  
    "cache.m3.2xlarge",  
    "cache.m3.large",  
    "cache.m3.xlarge",  
    "cache.m4.10xlarge",  
    "cache.m4.2xlarge",  
    "cache.m4.4xlarge",  
    "cache.m4.large",  
    "cache.m4.xlarge",  
    "cache.r3.2xlarge",  
    "cache.r3.4xlarge",  
    "cache.r3.8xlarge",  
    "cache.r3.large",  
    "cache.r3.xlarge"  
  ],  
  "ScaleDownModifications": [  
    "cache.t2.micro",  
    "cache.t2.small",  
    "cache.t2.medium",  
    "cache.t1.small"  
  ],  
}
```

Para obtener más información consulte [list-allowed-node-type-modifications](#) en la Referencia de la AWS CLI .

2. Modifique el clúster de caché existente especificando el clúster de caché que desea ampliarse y el nuevo tipo de nodo, de mayor tamaño, mediante el AWS CLI `modify-cache-cluster` comando y los siguientes parámetros.

- `--cache-cluster-id`: nombre del clúster de caché que está ampliando.
- `--cache-node-type`: tipo de nodo nuevo al que desea escalar el clúster de caché. Este valor debe ser uno de los tipos de nodos devueltos por el comando `list-allowed-node-type-modifications` en el paso 1.
- `--cache-parameter-group-name`: [Opcional] Utilice este parámetro si va a utilizar `reserved-memory` para administrar la memoria reservada de su clúster. Especifique un grupo de parámetros de caché personalizado que reserve la cantidad correcta de memoria para el nuevo tipo de nodo. Si va a utilizar `reserved-memory-percent`, puede omitir este parámetro.
- `--apply-immediately`: hace que el proceso de escalado vertical se aplique de inmediato. Para aplazar el proceso de ampliación al siguiente periodo de mantenimiento del clúster, utilice el parámetro `--no-apply-immediately`.

Para Linux, macOS o Unix:

```
aws elasticache modify-cache-cluster \  
  --cache-cluster-id my-redis-cache-cluster \  
  --cache-node-type cache.m3.xlarge \  
  --cache-parameter-group-name redis32-m2-x1 \  
  --apply-immediately
```

Para Windows:

```
aws elasticache modify-cache-cluster ^  
  --cache-cluster-id my-redis-cache-cluster ^  
  --cache-node-type cache.m3.xlarge ^  
  --cache-parameter-group-name redis32-m2-x1 ^  
  --apply-immediately
```

La salida del comando anterior es similar a la siguiente (formato JSON).

```
{  
  "CacheCluster": {  
    "Engine": "redis",  
    "CacheParameterGroup": {  
      "CacheNodeIdsToReboot": [],  
      "CacheParameterGroupName": "default.redis6.x",
```

```

        "ParameterApplyStatus": "in-sync"
    },
    "SnapshotRetentionLimit": 1,
    "CacheClusterId": "my-redis-cache-cluster",
    "CacheSecurityGroups": [],
    "NumCacheNodes": 1,
    "SnapshotWindow": "00:00-01:00",
    "CacheClusterCreateTime": "2017-02-21T22:34:09.645Z",
    "AutoMinorVersionUpgrade": true,
    "CacheClusterStatus": "modifying",
    "PreferredAvailabilityZone": "us-west-2a",
    "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
    "CacheSubnetGroupName": "default",
    "EngineVersion": "6.0",
    "PendingModifiedValues": {
        "CacheNodeType": "cache.m3.2xlarge"
    },
    "PreferredMaintenanceWindow": "tue:11:30-tue:12:30",
    "CacheNodeType": "cache.m3.medium",
    "DataTiering": "disabled"
}
}

```

Para obtener más información consulte [modify-cache-cluster](#) en la Referencia de la AWS CLI .

3. Si utilizó el `--apply-immediately`, compruebe el estado del nuevo clúster de caché mediante el AWS CLI `describe-cache-clusters` comando con el siguiente parámetro. Cuando el estado cambie a `available`, puede comenzar con el nuevo clúster de caché, de mayor tamaño.

- `--cache-cluster-id`— El nombre de su clúster de caché Redis OSS de un solo nodo. Use este parámetro para describir un determinado clúster de caché en lugar de todos los clústeres de caché.

```
aws elasticache describe-cache-clusters --cache-cluster-id my-redis-cache-cluster
```

Para obtener más información consulte [describe-cache-clusters](#) en la Referencia de la AWS CLI

Ampliación de los clústeres de caché (API) de Redis OSS de un solo nodo ElastiCache

El siguiente procedimiento describe cómo escalar un clúster de caché de Redis OSS de un solo nodo mediante la API. ElastiCache Durante este proceso, el clúster de Redis OSS seguirá atendiendo las solicitudes con un tiempo de inactividad mínimo.

Para ampliar un clúster de caché (API) de Redis OSS de un solo nodo ElastiCache

1. Determine los tipos de nodos a los que puede escalar ejecutando la `ListAllowedNodeTypeModifications` acción de la ElastiCache API con el siguiente parámetro.
 - `CacheClusterId`— El nombre del clúster de caché de Redis OSS de un solo nodo que desea escalar.

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=ListAllowedNodeTypeModifications  
  &CacheClusterId=MyRedisCacheCluster  
  &Version=2015-02-02  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Timestamp=20150202T192317Z  
  &X-Amz-Credential=<credential>
```

Para obtener más información, consulta [ListAllowedNodeTypeModifications](#) la referencia de la ElastiCache API de Amazon.

2. Modifique su clúster de caché existente especificando el clúster de caché que desea ampliarse y el nuevo tipo de nodo, de mayor tamaño, mediante la acción de la `ModifyCacheCluster` ElastiCache API y los siguientes parámetros.
 - `CacheClusterId`: nombre del clúster de caché que está ampliando.
 - `CacheNodeType`: tipo del nuevo nodo, de mayor tamaño, al que desea escalar verticalmente el clúster de caché. Este valor debe ser uno de los tipos de nodos devueltos por la acción `ListAllowedNodeTypeModifications` en el paso 1.
 - `CacheParameterGroupName`: [Opcional] Utilice este parámetro si va a utilizar `reserved-memory` para administrar la memoria reservada de su clúster. Especifique un grupo de parámetros de caché personalizado que reserve la cantidad correcta de memoria para

el nuevo tipo de nodo. Si va a utilizar `reserved-memory-percent`, puede omitir este parámetro.

- `ApplyImmediately`: establézcalo en `true` para que el proceso de escalado vertical se realice de inmediato. Para aplazar el proceso de ampliación al siguiente periodo de mantenimiento del clúster, utilice `ApplyImmediately=false`.

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=ModifyCacheCluster  
  &ApplyImmediately=true  
  &CacheClusterId=MyRedisCacheCluster  
  &CacheNodeType=cache.m3.xlarge  
  &CacheParameterGroupName redis32-m2-x1  
  &Version=2015-02-02  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Timestamp=20150202T192317Z  
  &X-Amz-Credential=<credential>
```

Para obtener más información, consulta [ModifyCacheCluster](#) la referencia de la ElastiCache API de Amazon.

3. Si lo utilizaste `ApplyImmediately=true`, comprueba el estado del nuevo clúster de caché mediante la `DescribeCacheClusters` acción de la ElastiCache API con el siguiente parámetro. Cuando el estado cambie a `available`, puede comenzar con el nuevo clúster de caché, de mayor tamaño.
 - `CacheClusterId`— El nombre de su clúster de caché de Redis OSS de un solo nodo. Use este parámetro para describir un determinado clúster de caché en lugar de todos los clústeres de caché.

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=DescribeCacheClusters  
  &CacheClusterId=MyRedisCacheCluster  
  &Version=2015-02-02  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Timestamp=20150202T192317Z  
  &X-Amz-Credential=<credential>
```

Para obtener más información, consulta [DescribeCacheClusters](#) la referencia de la ElastiCache API de Amazon.

Reducir el tamaño de los clústeres Redis OSS de un solo nodo

En las siguientes secciones, se explica cómo escalar un clúster de Redis OSS de un solo nodo a un tipo de nodo más pequeño. Garantizar que el nuevo tipo de nodo, más pequeño, sea lo suficientemente grande como para alojar todos los datos y la sobrecarga de Redis OSS es importante para el éxito a largo plazo del nuevo clúster de Redis OSS. Para obtener más información, consulte [Asegurarse de tener suficiente memoria para crear una instantánea de Redis OSS](#).

Note

Para los clústeres que ejecutan el tipo de nodo r6gd, solo puede escalar a tamaños de nodo dentro de la familia de nodos r6gd.

Temas

- [Reducir el tamaño de un clúster de Redis OSS de un solo nodo \(consola\)](#)
- [Reducir el tamaño de los clústeres de caché de Redis OSS de un solo nodo \(AWS CLI\)](#)
- [Reducir los clústeres de caché \(API\) de Redis OSS de un solo nodo ElastiCache](#)

Reducir el tamaño de un clúster de Redis OSS de un solo nodo (consola)

El siguiente procedimiento le explica cómo escalar su clúster Redis OSS de un solo nodo a un tipo de nodo más pequeño mediante la consola. ElastiCache

Important

Si su grupo de parámetros suele `reserved-memory` reservar memoria para la sobrecarga de Redis OSS, antes de empezar a escalar, asegúrese de tener un grupo de parámetros personalizado que reserve la cantidad correcta de memoria para el nuevo tipo de nodo. De forma alternativa, puede modificar un grupo de parámetros personalizado para que use `reserved-memory-percent` y utilizar ese grupo de parámetros para su nuevo clúster. Si va a utilizar `reserved-memory-percent`, esto no es necesario. Para obtener más información, consulte [Administración de la memoria reservada](#).

Para reducir su clúster de Redis OSS de un solo nodo (consola)

1. Asegúrese de que el tipo de nodo más pequeño es adecuado para sus necesidades de datos y de capacidad adicional.
2. Si su grupo de parámetros `reserved-memory` solía reservar memoria para la sobrecarga de Redis OSS, asegúrese de tener un grupo de parámetros personalizado para reservar la cantidad correcta de memoria para su nuevo tipo de nodo.

De forma alternativa, puede modificar su grupo de parámetros personalizado para utilizar `reserved-memory-percent`. Para obtener más información, consulte [Administración de la memoria reservada](#).

3. Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en <https://console.aws.amazon.com/elasticache/>.
4. En la lista de clústeres, seleccione el clúster que desee reducir. Este clúster debe ejecutar el motor OSS de Redis y no el motor OSS de Redis agrupado.
5. Elija Modificar.
6. En el asistente Modify Cluster:
 - a. Elija el tipo de nodo que desee reducir en la lista Node type (Tipo de nodo).
 - b. Si va a utilizar `reserved-memory` para administrar su memoria, en la lista Parameter Group, elija el grupo de parámetros personalizado que reserve la cantidad correcta de memoria para su nuevo tipo de nodo.
7. Si desea realizar el proceso de reducción de forma inmediata, elija la casilla de verificación Apply immediately (Aplicar inmediatamente). Si no se deja seleccionada la casilla de verificación Apply immediately (Aplicar inmediatamente), el proceso de reducción se realizará durante el siguiente periodo de mantenimiento de este clúster.
8. Elija Modificar.
9. Cuando el estado del clúster cambie de `modifying` a `available`, el clúster se habrá escalado al nuevo tipo de nodo. No es necesario actualizar los puntos de enlace de su aplicación.

Reducir el tamaño de los clústeres de caché de Redis OSS de un solo nodo (AWS CLI)

El siguiente procedimiento describe cómo reducir la escala de un clúster de caché Redis OSS de un solo nodo mediante AWS CLI

Para reducir la escala de un clúster de caché Redis OSS de un solo nodo (AWS CLI)

1. Determine los tipos de nodos a los que puede reducir la escala ejecutando el AWS CLI `list-allowed-node-type-modifications` comando con el siguiente parámetro.

- `--cache-cluster-id`

Para Linux, macOS o Unix:

```
aws elasticache list-allowed-node-type-modifications \  
  --cache-cluster-id my-cache-cluster-id
```

Para Windows:

```
aws elasticache list-allowed-node-type-modifications ^  
  --cache-cluster-id my-cache-cluster-id
```

La salida del comando anterior es similar a la siguiente (formato JSON).

```
{  
  "ScaleUpModifications": [  
    "cache.m3.2xlarge",  
    "cache.m3.large",  
    "cache.m3.xlarge",  
    "cache.m4.10xlarge",  
    "cache.m4.2xlarge",  
    "cache.m4.4xlarge",  
    "cache.m4.large",  
    "cache.m4.xlarge",  
    "cache.r3.2xlarge",  
    "cache.r3.4xlarge",  
    "cache.r3.8xlarge",  
    "cache.r3.large",  
    "cache.r3.xlarge"  
  ],  
  "ScaleDownModifications": [  
    "cache.t2.micro",  
    "cache.t2.small",  
    "cache.t2.medium",  
    "cache.t1.small"  
  ],  
}
```

```
}
```

Para obtener más información consulte [list-allowed-node-type-modifications](#) en la Referencia de la AWS CLI .

2. Modifique el clúster de caché existente especificando el clúster de caché que desea reducir y el nuevo tipo de nodo más pequeño mediante el AWS CLI `modify-cache-cluster` comando y los siguientes parámetros.
 - `--cache-cluster-id`: nombre del clúster de caché que se reduce verticalmente.
 - `--cache-node-type`: tipo de nodo nuevo al que desea escalar el clúster de caché. Este valor debe ser uno de los tipos de nodos devueltos por el comando `list-allowed-node-type-modifications` en el paso 1.
 - `--cache-parameter-group-name`: [Opcional] Utilice este parámetro si va a utilizar `reserved-memory` para administrar la memoria reservada de su clúster. Especifique un grupo de parámetros de caché personalizado que reserve la cantidad correcta de memoria para el nuevo tipo de nodo. Si va a utilizar `reserved-memory-percent`, puede omitir este parámetro.
 - `--apply-immediately`: hace que el proceso de reducción vertical se aplique de inmediato. Para aplazar el proceso de ampliación al siguiente periodo de mantenimiento del clúster, utilice el parámetro `--no-apply-immediately`.

Para Linux, macOS o Unix:

```
aws elasticache modify-cache-cluster \  
  --cache-cluster-id my-redis-cache-cluster \  
  --cache-node-type cache.m3.xlarge \  
  --cache-parameter-group-name redis32-m2-xl \  
  --apply-immediately
```

Para Windows:

```
aws elasticache modify-cache-cluster ^  
  --cache-cluster-id my-redis-cache-cluster ^  
  --cache-node-type cache.m3.xlarge ^  
  --cache-parameter-group-name redis32-m2-xl ^  
  --apply-immediately
```

La salida del comando anterior es similar a la siguiente (formato JSON).

```
{
  "CacheCluster": {
    "Engine": "redis",
    "CacheParameterGroup": {
      "CacheNodeIdsToReboot": [],
      "CacheParameterGroupName": "default.redis6.x",
      "ParameterApplyStatus": "in-sync"
    },
    "SnapshotRetentionLimit": 1,
    "CacheClusterId": "my-redis-cache-cluster",
    "CacheSecurityGroups": [],
    "NumCacheNodes": 1,
    "SnapshotWindow": "00:00-01:00",
    "CacheClusterCreateTime": "2017-02-21T22:34:09.645Z",
    "AutoMinorVersionUpgrade": true,
    "CacheClusterStatus": "modifying",
    "PreferredAvailabilityZone": "us-west-2a",
    "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/home#client-download:",
    "CacheSubnetGroupName": "default",
    "EngineVersion": "6.0",
    "PendingModifiedValues": {
      "CacheNodeType": "cache.m3.2xlarge"
    },
    "PreferredMaintenanceWindow": "tue:11:30-tue:12:30",
    "CacheNodeType": "cache.m3.medium",
    "DataTiering": "disabled"
  }
}
```

Para obtener más información consulte [modify-cache-cluster](#) en la Referencia de la AWS CLI .

3. Si utilizó el `--apply-immediately`, compruebe el estado del nuevo clúster de caché mediante el AWS CLI `describe-cache-clusters` comando con el siguiente parámetro. Cuando el estado cambie a `available`, puede comenzar con el nuevo clúster de caché, de mayor tamaño.
 - `--cache-cache cluster-id`— El nombre de su clúster de caché Redis OSS de un solo nodo. Use este parámetro para describir un determinado clúster de caché en lugar de todos los clústeres de caché.

```
aws elasticache describe-cache-clusters --cache-cluster-id my-redis-cache-cluster
```

Para obtener más información consulte [describe-cache-clusters](#) en la Referencia de la AWS CLI

Reducir los clústeres de caché (API) de Redis OSS de un solo nodo ElastiCache

El siguiente procedimiento describe cómo escalar hacia arriba un clúster de caché de Redis OSS de un solo nodo mediante la API. ElastiCache

Para reducir la escala de un clúster de caché (API) de Redis OSS de un solo nodo ElastiCache

1. Determine los tipos de nodos a los que puede reducir la escala ejecutando la `ListAllowedNodeTypeModifications` acción de la ElastiCache API con el siguiente parámetro.
 - `CacheClusterId`— El nombre del clúster de caché de Redis OSS de un solo nodo al que desea reducir la escala.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=ListAllowedNodeTypeModifications  
&CacheClusterId=MyRedisCacheCluster  
&Version=2015-02-02  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20150202T192317Z  
&X-Amz-Credential=<credential>
```

Para obtener más información, consulta [ListAllowedNodeTypeModifications](#) la referencia de la ElastiCache API de Amazon.

2. Modifique su clúster de caché existente especificando el clúster de caché que desea ampliarse y el nuevo tipo de nodo, de mayor tamaño, mediante la acción de la `ModifyCacheCluster` ElastiCache API y los siguientes parámetros.
 - `CacheClusterId`: nombre del clúster de caché que se reduce verticalmente.

- `CacheNodeType`: el tipo de nodo nuevo y más pequeño al que desea reducir verticalmente el clúster de caché. Este valor debe ser uno de los tipos de nodos devueltos por la acción `ListAllowedNodeTypeModifications` en el paso 1.
- `CacheParameterGroupName`: [Opcional] Utilice este parámetro si va a utilizar `reserved-memory` para administrar la memoria reservada de su clúster. Especifique un grupo de parámetros de caché personalizado que reserve la cantidad correcta de memoria para el nuevo tipo de nodo. Si va a utilizar `reserved-memory-percent`, puede omitir este parámetro.
- `ApplyImmediately`: establézcalo en `true` para que el proceso de reducción vertical se realice de inmediato. Para aplazar el proceso de ampliación al siguiente periodo de mantenimiento del clúster, utilice `ApplyImmediately=false`.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=ModifyCacheCluster  
&ApplyImmediately=true  
&CacheClusterId=MyRedisCacheCluster  
&CacheNodeType=cache.m3.xlarge  
&CacheParameterGroupName=redis32-m2-x1  
&Version=2015-02-02  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20150202T192317Z  
&X-Amz-Credential=<credential>
```

Para obtener más información, consulta [ModifyCacheCluster](#) la referencia de la ElastiCache API de Amazon.

3. Si lo utilizaste `ApplyImmediately=true`, comprueba el estado del nuevo clúster de caché mediante la `DescribeCacheClusters` acción de la ElastiCache API con el siguiente parámetro. Cuando el estado cambie a `available` (disponible), puede comenzar con el nuevo clúster de caché de menor tamaño.
 - `CacheClusterId`— El nombre del clúster de caché de Redis OSS de un solo nodo. Use este parámetro para describir un determinado clúster de caché en lugar de todos los clústeres de caché.

```
https://elasticache.us-west-2.amazonaws.com/
```

```
?Action=DescribeCacheClusters  
&CacheClusterId=MyRedisCacheCluster  
&Version=2015-02-02  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20150202T192317Z  
&X-Amz-Credential=<credential>
```

Para obtener más información, consulta [DescribeCacheClusters](#) la referencia de la ElastiCache API de Amazon.

Escalado de clústeres de Redis OSS (modo de clúster desactivado) con nodos de réplica

Un clúster OSS de Redis con nodos de réplica (denominado grupo de replicación en la API/CLI) proporciona alta disponibilidad mediante una replicación que tiene Multi-AZ con conmutación por error automática habilitada. Un clúster con nodos de réplica es un conjunto lógico de hasta seis nodos OSS de Redis, donde un nodo, el principal, puede atender solicitudes de lectura y escritura. Todos los demás nodos del clúster son réplicas de solo lectura del principal. Los datos que se escriben en el principal se replican de forma asíncrona en todas las réplicas de lectura del clúster. Como el OSS de Redis (modo de clúster desactivado) no admite la partición de los datos en varios clústeres, cada nodo de un grupo de replicación de Redis OSS (modo de clúster desactivado) contiene todo el conjunto de datos de caché. Los clústeres OSS de Redis (modo de clúster activado) permiten particionar los datos en hasta 500 fragmentos.

Para cambiar la capacidad de datos de su clúster, debe ampliarlo a un tipo de nodo más grande, o reducirlo a un tipo de nodo más pequeño.

Para cambiar la capacidad de lectura del clúster, agregue más réplicas de lectura, hasta un máximo de 5, o bien elimine réplicas de lectura.

El proceso de ElastiCache ampliación está diseñado para hacer todo lo posible por conservar los datos existentes y requiere una replicación correcta de Redis OSS. Para los clústeres de Redis OSS con réplicas, recomendamos que Redis OSS disponga de memoria suficiente.

Temas relacionados

- [Alta disponibilidad a través de grupos de reproducción](#)
- [Replicación: Redis OSS \(modo de clúster desactivado\) frente a Redis OSS \(modo de clúster activado\)](#)
- [Minimizar el tiempo de inactividad en ElastiCache \(Redis OSS\) con Multi-AZ](#)
- [Asegurarse de tener suficiente memoria para crear una instantánea de Redis OSS](#)

Temas

- [Ampliación de los clústeres de Redis OSS con réplicas](#)
- [Reducir los clústeres de Redis OSS con réplicas](#)
- [Aumento de la capacidad de lectura](#)
- [Reducción de la capacidad de lectura](#)

Ampliación de los clústeres de Redis OSS con réplicas

Amazon ElastiCache ofrece soporte de consola, CLI y API para escalar su grupo de replicación de Redis OSS (modo de clúster deshabilitado) hacia arriba.

Cuando se inicia el proceso de escalado, ElastiCache hace lo siguiente:

1. Lanza un grupo de reproducción con el nuevo tipo de nodo.
2. Copia todos los datos del nodo principal actual al nuevo nodo principal.
3. Sincroniza las nuevas réplicas de lectura con el nuevo nodo principal.
4. Actualiza los entradas de DNS para que apunten a los nuevos nodos. Por ello, no tiene que actualizar los puntos de enlace de su aplicación. Para Redis OSS 5.0.5 y versiones posteriores, puede escalar los clústeres habilitados para la conmutación automática por error mientras el clúster permanece en línea y atiende las solicitudes entrantes. En la versión 4.0.10 y anteriores, puede observar una breve interrupción de las operaciones de lectura y escritura del nodo principal mientras se actualiza la entrada de DNS.
5. Elimina los nodos antiguos (CLI/API: grupo de reproducción). Observará una breve interrupción (unos segundos) de las operaciones de lectura y escritura de los nodos anteriores porque se desconectarán las conexiones con los nodos anteriores.

El tiempo que tarda este proceso depende de su tipo de nodo y de la cantidad de datos que haya en su clúster.

Como se muestra en la siguiente tabla, la operación de ampliación de Redis OSS se bloquea si tiene programada una actualización del motor para el próximo período de mantenimiento del clúster.

Operaciones bloqueadas de Redis OSS

Operaciones pendientes	Operaciones bloqueadas
Escalado ascendente	Actualización del motor inmediata
Actualización del motor	Ampliación inmediata
Ampliación y actualización del motor	Ampliación inmediata
	Actualización del motor inmediata

Si hay una operación pendiente que esté bloqueando, tiene las siguientes opciones.

- Programe la operación de ampliación de Redis OSS para el siguiente período de mantenimiento desactivando la casilla Aplicar inmediatamente (uso de CLI: `--no-apply-immediately`, uso de API: `ApplyImmediately=false`)
- Espere hasta el próximo período de mantenimiento (o hasta después) para realizar la operación de ampliación de Redis OSS.
- Agregue la actualización del motor OSS de Redis a esta modificación del clúster de caché con la casilla Aplicar inmediatamente seleccionada (uso de CLI: `--apply-immediately`, uso de API: `ApplyImmediately=true`). Esto desbloqueará su operación de ampliación, ya que hará que la actualización del motor se realice inmediatamente.

En las siguientes secciones, se describe cómo escalar el clúster de Redis OSS con réplicas superiores mediante la ElastiCache consola, la API y la AWS CLI API. ElastiCache

Important

Si su grupo de parámetros suele `reserved-memory` reservar memoria para la sobrecarga de Redis OSS, antes de empezar a escalar, asegúrese de tener un grupo de parámetros personalizado que reserve la cantidad correcta de memoria para el nuevo tipo de nodo. De forma alternativa, puede modificar un grupo de parámetros personalizado para que use `reserved-memory-percent` y utilizar ese grupo de parámetros para su nuevo clúster. Si va a utilizar `reserved-memory-percent`, esto no es necesario. Para obtener más información, consulte [Administración de la memoria reservada](#).

Ampliación de un clúster de Redis OSS con réplicas (consola)

El tiempo que se tarda en la ampliación a un tipo de nodo más grande varía en función del tipo de nodo y de la cantidad de datos de su clúster actual.

El siguiente proceso escala el clúster con réplicas desde su tipo de nodo actual a un tipo de nodo nuevo y más grande mediante la consola. ElastiCache Durante este proceso, puede haber una breve interrupción en las operaciones de lectura y escritura de otras versiones en el nodo principal mientras se actualiza la entrada de DNS. Es posible que observe un tiempo de inactividad inferior a 1 segundo en los nodos que ejecutan la versión 5.0.6 y posteriores, y de varios segundos en versiones anteriores.

Para ampliar el clúster OSS de Redis con réplicas (consola)

1. [Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en https://console.aws.amazon.com/elasticache/.](https://console.aws.amazon.com/elasticache/)
2. En el panel de navegación, elija los clústeres de Redis OSS
3. En la lista de clústeres, seleccione el clúster que desea ampliar. Este clúster debe ejecutar el motor OSS de Redis y no el motor OSS de Redis agrupado.
4. Elija Modificar.
5. En el asistente Modify Cluster:
 - a. Elija el tipo de nodo que desee ampliar en la lista Node type. Tenga en cuenta que no todos los tipos de nodos están disponibles para el proceso de reducción.
 - b. Si va a utilizar `reserved-memory` para administrar su memoria, en la lista Parameter Group, elija el grupo de parámetros personalizado que reserve la cantidad correcta de memoria para su nuevo tipo de nodo.
6. Si desea realizar el proceso de ampliación de forma inmediata, elija la casilla de verificación Apply immediately. Si no se deja seleccionada la casilla de verificación Apply immediately, el proceso de ampliación se realizará durante el siguiente periodo de mantenimiento de este clúster.
7. Elija Modificar.
8. Cuando el estado del clúster cambie de `modifying` a `available`, el clúster se habrá escalado al nuevo tipo de nodo. No es necesario actualizar los puntos de enlace de su aplicación.

Ampliación de un grupo de replicación de Redis OSS (AWS CLI)

El siguiente proceso amplía su grupo de reproducción de su tipo de nodo actual a un nuevo tipo de nodo más grande utilizando AWS CLI. Durante este proceso, ElastiCache (Redis OSS) actualiza las entradas de DNS para que apunten a los nuevos nodos. Por ello, no tiene que actualizar los puntos de enlace de su aplicación. Para Redis OSS 5.0.5 y versiones posteriores, puede escalar los clústeres habilitados para la conmutación automática por error mientras el clúster permanece en línea y atiende las solicitudes entrantes. En la versión 4.0.10 y anteriores, puede observar una breve interrupción de las versiones de lectura y escritura del nodo principal mientras se actualiza la entrada de DNS.

El tiempo que se tarda en la ampliación a un tipo de nodo más grande varía en función de su tipo de nodo y de la cantidad de datos de su clúster de caché actual.

Para ampliar un grupo de replicación de Redis OSS ()AWS CLI

1. Determine los tipos de nodos a los que puede escalar ejecutando el AWS CLI `list-allowed-node-type-modifications` comando con el siguiente parámetro.
 - `--replication-group-id`: nombre del grupo de replicación. Use este parámetro para describir un determinado grupo de replicación en lugar de todos los grupos de replicación.

Para Linux, macOS o Unix:

```
aws elasticache list-allowed-node-type-modifications \  
  --replication-group-id my-repl-group
```

Para Windows:

```
aws elasticache list-allowed-node-type-modifications ^  
  --replication-group-id my-repl-group
```

La salida de esta operación tiene un aspecto similar al siguiente (formato JSON).

```
{  
  "ScaleUpModifications": [  
    "cache.m3.2xlarge",  
    "cache.m3.large",  
    "cache.m3.xlarge",  
    "cache.m4.10xlarge",  
    "cache.m4.2xlarge",  
    "cache.m4.4xlarge",  
    "cache.m4.large",  
    "cache.m4.xlarge",  
    "cache.r3.2xlarge",  
    "cache.r3.4xlarge",  
    "cache.r3.8xlarge",  
    "cache.r3.large",  
    "cache.r3.xlarge"  
  ]  
}
```

Para obtener más información consulte [list-allowed-node-type-modifications](#) en la Referencia de la AWS CLI .

2. Amplíe su grupo de replicación actual hasta el nuevo tipo de nodo mediante el AWS CLI `modify-replication-group` comando con los siguientes parámetros.
 - `--replication-group-id`: nombre del grupo de replicación.
 - `--cache-node-type`: el nuevo tipo de nodo más grande de clústeres de caché en este grupo de reproducción. Este valor debe ser uno de los tipos de instancia devueltos por el comando `list-allowed-node-type-modifications` en el paso 1.
 - `--cache-parameter-group-name`: [Opcional] Utilice este parámetro si va a utilizar `reserved-memory` para administrar la memoria reservada de su clúster. Especifique un grupo de parámetros de caché personalizado que reserve la cantidad correcta de memoria para el nuevo tipo de nodo. Si va a utilizar `reserved-memory-percent`, puede omitir este parámetro.
 - `--apply-immediately`: hace que el proceso de escalado vertical se aplique de inmediato. Para aplazar la operación de ampliación al siguiente periodo de mantenimiento, utilice `--no-apply-immediately`.

Para Linux, macOS o Unix:

```
aws elasticache modify-replication-group \  
  --replication-group-id my-repl-group \  
  --cache-node-type cache.m3.xlarge \  
  --cache-parameter-group-name redis32-m3-2x1 \  
  --apply-immediately
```

Para Windows:

```
aws elasticache modify-replication-group ^  
  --replication-group-id my-repl-group ^  
  --cache-node-type cache.m3.xlarge ^  
  --cache-parameter-group-name redis32-m3-2x1 \  
  --apply-immediately
```

La salida de este comando tiene un aspecto similar al siguiente (formato JSON).

```
{  
  "ReplicationGroup": {  
    "Status": "available",
```

```

"Description": "Some description",
"NodeGroups": [{
  "Status": "available",
  "NodeGroupMembers": [{
    "CurrentRole": "primary",
    "PreferredAvailabilityZone": "us-west-2b",
    "CacheNodeId": "0001",
    "ReadEndpoint": {
      "Port": 6379,
      "Address": "my-repl-group-001.8fdx4s.0001.usw2.cache.amazonaws.com"
    },
    "CacheClusterId": "my-repl-group-001"
  },
  {
    "CurrentRole": "replica",
    "PreferredAvailabilityZone": "us-west-2c",
    "CacheNodeId": "0001",
    "ReadEndpoint": {
      "Port": 6379,
      "Address": "my-repl-group-002.8fdx4s.0001.usw2.cache.amazonaws.com"
    },
    "CacheClusterId": "my-repl-group-002"
  }
  ],
  "NodeGroupId": "0001",
  "PrimaryEndpoint": {
    "Port": 6379,
    "Address": "my-repl-group.8fdx4s.ng.0001.usw2.cache.amazonaws.com"
  }
}],
"ReplicationGroupId": "my-repl-group",
"SnapshotRetentionLimit": 1,
"AutomaticFailover": "disabled",
"SnapshotWindow": "12:00-13:00",
"SnapshottingClusterId": "my-repl-group-002",
"MemberClusters": [
  "my-repl-group-001",
  "my-repl-group-002"
],
"PendingModifiedValues": {}
}
}

```

Para obtener más información consulte [modify-replication-group](#) en la Referencia de la AWS CLI

3. Si utilizó el `--apply-immediately` parámetro, supervise el estado del grupo de replicación mediante el AWS CLI `describe-replication-group` comando con el siguiente parámetro. Mientras el estado sea en modificación, es posible que observe un tiempo de inactividad de un segundo en los nodos que ejecutan la versión 5.0.6 y posteriores y una breve interrupción de las operaciones de lectura y escritura con las versiones anteriores en el nodo principal mientras se actualiza la entrada de DNS.
- `--replication-group-id`: nombre del grupo de replicación. Use este parámetro para describir un determinado grupo de replicación en lugar de todos los grupos de replicación.

Para Linux, macOS o Unix:

```
aws elasticache describe-replication-groups \  
  --replication-group-id my-replication-group
```

Para Windows:

```
aws elasticache describe-replication-groups ^  
  --replication-group-id my-replication-group
```

Para obtener más información, consulte [describe-replication-groups](#) la AWS CLI Referencia.

Ampliación de un grupo de replicación (ElastiCache API) de Redis OSS

El siguiente proceso escala el grupo de replicación desde su tipo de nodo actual a un tipo de nodo nuevo y más grande mediante la ElastiCache API. Para Redis OSS 5.0.5 y versiones posteriores, puede escalar los clústeres habilitados para la conmutación automática por error mientras el clúster permanece en línea y atiende las solicitudes entrantes. En la versión 4.0.10 y anteriores, puede observar una breve interrupción de las operaciones de lectura y escritura del nodo principal mientras se actualiza la entrada de DNS.

El tiempo que se tarda en la ampliación a un tipo de nodo más grande varía en función de su tipo de nodo y de la cantidad de datos de su clúster de caché actual.

Para ampliar un grupo de replicación (API) de Redis OSS ElastiCache

1. Determine los tipos de nodos a los que puede escalar mediante la `ListAllowedNodeTypeModifications` acción de la ElastiCache API con el siguiente parámetro.

- `ReplicationGroupId`: nombre del grupo de replicación. Use este parámetro para describir un grupo de replicación específico en lugar de todos los grupos de replicación.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=ListAllowedNodeTypeModifications  
&ReplicationGroupId=MyReplGroup  
&Version=2015-02-02  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20150202T192317Z  
&X-Amz-Credential=<credential>
```

Para obtener más información, consulta [ListAllowedNodeTypeModifications](#) la referencia de la ElastiCache API de Amazon.

2. Amplíe su grupo de replicación actual hasta el nuevo tipo de nodo mediante la acción de la `ModifyRedplicationGroup` ElastiCache API y con los siguientes parámetros.

- `ReplicationGroupId`: nombre del grupo de replicación.
- `CacheNodeType`: el nuevo tipo de nodo más grande de clústeres de caché en este grupo de reproducción. Este valor debe ser uno de los tipos de instancia devueltos por la acción `ListAllowedNodeTypeModifications` en el paso 1.
- `CacheParameterGroupName`: [Opcional] Utilice este parámetro si va a utilizar `reserved-memory` para administrar la memoria reservada de su clúster. Especifique un grupo de parámetros de caché personalizado que reserve la cantidad correcta de memoria para el nuevo tipo de nodo. Si va a utilizar `reserved-memory-percent`, puede omitir este parámetro.
- `ApplyImmediately`: establézcalo en `true` para que el proceso de escalado vertical se aplique de inmediato. Para aplazar el proceso de ampliación al siguiente periodo de mantenimiento, utilice `ApplyImmediately=false`.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=ModifyReplicationGroup  
&ApplyImmediately=true  
&CacheNodeType=cache.m3.2xlarge  
&CacheParameterGroupName=redis32-m3-2x1  
&ReplicationGroupId=myReplGroup  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20141201T220302Z  
&Version=2014-12-01  
&X-Amz-Algorithm=&AWS;4-HMAC-SHA256  
&X-Amz-Date=20141201T220302Z  
&X-Amz-SignedHeaders=Host  
&X-Amz-Expires=20141201T220302Z  
&X-Amz-Credential=<credential>  
&X-Amz-Signature=<signature>
```

Para obtener más información, consulta [ModifyReplicationGroup](#) la referencia de la ElastiCache API de Amazon.

3. Si lo utilizó `ApplyImmediately=true`, supervise el estado del grupo de replicación mediante la `DescribeReplicationGroups` acción de la ElastiCache API con los siguientes parámetros. Cuando el estado cambie de `modifying` a `available`, podrá empezar a escribir en su nuevo grupo de replicación ampliado.
 - `ReplicationGroupId`: nombre del grupo de replicación. Use este parámetro para describir un determinado grupo de replicación en lugar de todos los grupos de replicación.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=DescribeReplicationGroups  
&ReplicationGroupId=MyReplGroup  
&Version=2015-02-02  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20150202T192317Z  
&X-Amz-Credential=<credential>
```

Para obtener más información, consulta [DescribeReplicationGroups](#) la referencia de la ElastiCache API de Amazon.

Reducir los clústeres de Redis OSS con réplicas

En las siguientes secciones, se explica cómo escalar un clúster de caché de Redis OSS (modo de clúster desactivado) con nodos de réplica a un tipo de nodo más pequeño. Para que el proceso se realice correctamente, es muy importante asegurarse de que el nuevo tipo de nodo más pequeño sea lo suficientemente grande para alojar todos los datos, más la capacidad adicional. Para obtener más información, consulte [Asegurarse de tener suficiente memoria para crear una instantánea de Redis OSS](#).

Note

Para los clústeres que ejecutan el tipo de nodo r6gd, solo puede escalar a tamaños de nodo dentro de la familia de nodos r6gd.

Important

Si su grupo de parámetros `reserved-memory` solía reservar memoria para la sobrecarga de Redis OSS, antes de empezar a escalar, asegúrese de tener un grupo de parámetros personalizado que reserve la cantidad correcta de memoria para el nuevo tipo de nodo. De forma alternativa, puede modificar un grupo de parámetros personalizado para que use `reserved-memory-percent` y utilizar ese grupo de parámetros para su nuevo clúster. Si va a utilizar `reserved-memory-percent`, esto no es necesario. Para obtener más información, consulte [Administración de la memoria reservada](#).

Reducir el tamaño de un clúster de Redis OSS con réplicas (consola)

El siguiente proceso escala el clúster OSS de Redis con nodos de réplica a un tipo de nodo más pequeño mediante la consola. ElastiCache

Para reducir la escala de un clúster de OSS de Redis con nodos de réplica (consola)

1. Asegúrese de que el tipo de nodo más pequeño es adecuado para sus necesidades de datos y de capacidad adicional.
2. Si su grupo de parámetros `reserved-memory` solía reservar memoria para la sobrecarga de Redis OSS, asegúrese de tener un grupo de parámetros personalizado para reservar la cantidad de memoria correcta para el nuevo tipo de nodo.

De forma alternativa, puede modificar su grupo de parámetros personalizado para utilizar `reserved-memory-percent`. Para obtener más información, consulte [Administración de la memoria reservada](#).

3. Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en <https://console.aws.amazon.com/elasticache/>.
4. En la lista de clústeres, seleccione el clúster que desee reducir. Este clúster debe ejecutar el motor OSS de Redis y no el motor OSS de Redis agrupado.
5. Elija Modificar.
6. En el asistente Modify Cluster:
 - a. Elija el tipo de nodo que desee reducir en la lista Node type (Tipo de nodo).
 - b. Si va a utilizar `reserved-memory` para administrar su memoria, en la lista Parameter Group, elija el grupo de parámetros personalizado que reserve la cantidad correcta de memoria para su nuevo tipo de nodo.
7. Si desea realizar el proceso de reducción de forma inmediata, elija la casilla de verificación Apply immediately (Aplicar inmediatamente). Si no se deja seleccionada la casilla de verificación Apply immediately (Aplicar inmediatamente), el proceso de reducción se realizará durante el siguiente periodo de mantenimiento de este clúster.
8. Elija Modificar.
9. Cuando el estado del clúster cambie de `modifying` a `available`, el clúster se habrá escalado al nuevo tipo de nodo. No es necesario actualizar los puntos de enlace de su aplicación.

Reducir el tamaño de un grupo de replicación de Redis OSS ()AWS CLI

El siguiente proceso escala su grupo de reproducción de su tipo de nodo actual a un nuevo tipo de nodo más pequeño utilizando AWS CLI. Durante este proceso, ElastiCache (Redis OSS) actualiza las entradas de DNS para que apunten a los nuevos nodos. Por ello, no tiene que actualizar los puntos de enlace de su aplicación. Para Redis OSS 5.0.5 y versiones posteriores, puede escalar los clústeres habilitados para la conmutación automática por error mientras el clúster permanece en línea y atiende las solicitudes entrantes. En la versión 4.0.10 y anteriores, puede observar una breve interrupción de las versiones de lectura y escritura del nodo principal mientras se actualiza la entrada de DNS.

Sin embargo, las lecturas de los clústeres de caché de la réplica de lectura continuarán de forma ininterrumpida.

El tiempo que se tarda en la realización del escalado descendente a un tipo de nodo más pequeño varía en función de su tipo de nodo y de la cantidad de datos de su clúster de caché actual.

Para reducir un grupo de replicación de Redis OSS ()AWS CLI

1. Determine a qué tipos de nodos puede reducir la escala ejecutando el AWS CLI `list-allowed-node-type-modifications` comando con el siguiente parámetro.
 - `--replication-group-id`: nombre del grupo de replicación. Use este parámetro para describir un determinado grupo de replicación en lugar de todos los grupos de replicación.

Para Linux, macOS o Unix:

```
aws elasticache list-allowed-node-type-modifications \  
  --replication-group-id my-repl-group
```

Para Windows:

```
aws elasticache list-allowed-node-type-modifications ^  
  --replication-group-id my-repl-group
```

La salida de esta operación tiene un aspecto similar al siguiente (formato JSON).

```
{  
  "ScaleDownModifications": [  
    "cache.m3.2xlarge",  
    "cache.m3.large",  
    "cache.m3.xlarge",  
    "cache.m4.10xlarge",  
    "cache.m4.2xlarge",  
    "cache.m4.4xlarge",  
    "cache.m4.large",  
    "cache.m4.xlarge",  
    "cache.r3.2xlarge",  
    "cache.r3.4xlarge",  
    "cache.r3.8xlarge",  
    "cache.r3.large",  
    "cache.r3.xlarge"  
  ]  
}
```

Para obtener más información consulte [list-allowed-node-type-modifications](#) en la Referencia de la AWS CLI .

2. Amplíe su grupo de replicación actual hasta el nuevo tipo de nodo mediante el AWS CLI `modify-replication-group` comando con los siguientes parámetros.

- `--replication-group-id`: nombre del grupo de replicación.
- `--cache-node-type`: el nuevo tipo de nodo más pequeño de clústeres de caché en este grupo de reproducción. Este valor debe ser uno de los tipos de instancia devueltos por el comando `list-allowed-node-type-modifications` en el paso 1.
- `--cache-parameter-group-name`: [Opcional] Utilice este parámetro si va a utilizar `reserved-memory` para administrar la memoria reservada de su clúster. Especifique un grupo de parámetros de caché personalizado que reserve la cantidad correcta de memoria para el nuevo tipo de nodo. Si va a utilizar `reserved-memory-percent`, puede omitir este parámetro.
- `--apply-immediately`: hace que el proceso de escalado vertical se aplique de inmediato. Para aplazar la operación de ampliación al siguiente periodo de mantenimiento, utilice `--no-apply-immediately`.

Para Linux, macOS o Unix:

```
aws elasticache modify-replication-group \  
  --replication-group-id my-repl-group \  
  --cache-node-type cache.t2.small \  
  --cache-parameter-group-name redis32-m3-2x1 \  
  --apply-immediately
```

Para Windows:

```
aws elasticache modify-replication-group ^  
  --replication-group-id my-repl-group ^  
  --cache-node-type cache.t2.small ^  
  --cache-parameter-group-name redis32-m3-2x1 \  
  --apply-immediately
```

La salida de este comando tiene un aspecto similar al siguiente (formato JSON).

```

{"ReplicationGroup": {
  "Status": "available",
  "Description": "Some description",
  "NodeGroups": [
    {
      "Status": "available",
      "NodeGroupMembers": [
        {
          "CurrentRole": "primary",
          "PreferredAvailabilityZone": "us-west-2b",
          "CacheNodeId": "0001",
          "ReadEndpoint": {
            "Port": 6379,
            "Address": "my-repl-
group-001.8fdx4s.0001.usw2.cache.amazonaws.com"
          },
          "CacheClusterId": "my-repl-group-001"
        },
        {
          "CurrentRole": "replica",
          "PreferredAvailabilityZone": "us-west-2c",
          "CacheNodeId": "0001",
          "ReadEndpoint": {
            "Port": 6379,
            "Address": "my-repl-
group-002.8fdx4s.0001.usw2.cache.amazonaws.com"
          },
          "CacheClusterId": "my-repl-group-002"
        }
      ],
      "NodeGroupId": "0001",
      "PrimaryEndpoint": {
        "Port": 6379,
        "Address": "my-repl-
group.8fdx4s.ng.0001.usw2.cache.amazonaws.com"
      }
    }
  ],
  "ReplicationGroupId": "my-repl-group",
  "SnapshotRetentionLimit": 1,
  "AutomaticFailover": "disabled",
  "SnapshotWindow": "12:00-13:00",

```

```

    "SnapshottingClusterId": "my-repl-group-002",
    "MemberClusters": [
      "my-repl-group-001",
      "my-repl-group-002",
    ],
    "PendingModifiedValues": {}
  }
}

```

Para obtener más información consulte [modify-replication-group](#) en la Referencia de la AWS CLI

3. Si utilizó el `--apply-immediately` parámetro, supervise el estado del grupo de replicación mediante el AWS CLI `describe-replication-group` comando con el siguiente parámetro. Cuando el estado cambie de `modifying` a `available`, podrá empezar a escribir en su nuevo grupo de replicación reducido.
 - `--replication-group-id`: nombre del grupo de replicación. Use este parámetro para describir un determinado grupo de replicación en lugar de todos los grupos de replicación.

Para Linux, macOS o Unix:

```

aws elasticache describe-replication-group \
  --replication-group-id my-replication-group

```

Para Windows:

```

aws elasticache describe-replication-groups ^
  --replication-group-id my-replication-group

```

Para obtener más información, consulte [describe-replication-groups](#) la AWS CLI Referencia.

Reducir el tamaño de un grupo de replicación (ElastiCache API) de Redis OSS

El siguiente proceso escala el grupo de replicación desde su tipo de nodo actual a un tipo de nodo nuevo y más pequeño mediante la ElastiCache API. Durante este proceso, ElastiCache (Redis OSS) actualiza las entradas de DNS para que apunten a los nuevos nodos. Por ello, no tiene que actualizar los puntos de enlace de su aplicación. Para Redis OSS 5.0.5 y versiones posteriores, puede escalar los clústeres habilitados para la conmutación automática por error mientras el clúster permanece en

línea y atiende las solicitudes entrantes. En la versión 4.0.10 y anteriores, puede observar una breve interrupción de las versiones de lectura y escritura del nodo principal mientras se actualiza la entrada de DNS. Sin embargo, las lecturas de los clústeres de caché de la réplica de lectura continuarán de forma ininterrumpida.

El tiempo que se tarda en la realización del escalado descendente a un tipo de nodo más pequeño varía en función de su tipo de nodo y de la cantidad de datos de su clúster de caché actual.

Para reducir un grupo de replicación (API) de Redis OSS ElastiCache

1. Determine los tipos de nodos a los que puede reducir la escala mediante la `ListAllowedNodeTypeModifications` acción de la ElastiCache API con el siguiente parámetro.
 - `ReplicationGroupId`: nombre del grupo de replicación. Use este parámetro para describir un grupo de replicación específico en lugar de todos los grupos de replicación.

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=ListAllowedNodeTypeModifications  
  &ReplicationGroupId=MyReplGroup  
  &Version=2015-02-02  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Timestamp=20150202T192317Z  
  &X-Amz-Credential=<credential>
```

Para obtener más información, consulta [ListAllowedNodeTypeModifications](#) la referencia de la ElastiCache API de Amazon.

2. Amplíe su grupo de replicación actual hasta el nuevo tipo de nodo mediante la acción de la `ModifyRedplicationGroup` ElastiCache API y con los siguientes parámetros.
 - `ReplicationGroupId`: nombre del grupo de replicación.
 - `CacheNodeType`: el nuevo tipo de nodo más pequeño de clústeres de caché en este grupo de reproducción. Este valor debe ser uno de los tipos de instancia devueltos por la acción `ListAllowedNodeTypeModifications` en el paso 1.
 - `CacheParameterGroupName`: [Opcional] Utilice este parámetro si va a utilizar `reserved-memory` para administrar la memoria reservada de su clúster. Especifique un grupo de parámetros de caché personalizado que reserve la cantidad correcta de memoria para

el nuevo tipo de nodo. Si va a utilizar `reserved-memory-percent`, puede omitir este parámetro.

- `ApplyImmediately`: establézcalo en `true` para que el proceso de escalado vertical se aplique de inmediato. Para aplazar el proceso de escalado descendente al siguiente periodo de mantenimiento, utilice `ApplyImmediately=false`.

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=ModifyReplicationGroup  
  &ApplyImmediately=true  
  &CacheNodeType=cache.m3.2xlarge  
  &CacheParameterGroupName=redis32-m3-2x1  
  &ReplicationGroupId=myReplGroup  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Timestamp=20141201T220302Z  
  &Version=2014-12-01  
  &X-Amz-Algorithm=&AWS;4-HMAC-SHA256  
  &X-Amz-Date=20141201T220302Z  
  &X-Amz-SignedHeaders=Host  
  &X-Amz-Expires=20141201T220302Z  
  &X-Amz-Credential=<credential>  
  &X-Amz-Signature=<signature>
```

Para obtener más información, consulta [ModifyReplicationGroup](#) la referencia de la ElastiCache API de Amazon.

3. Si lo utilizó `ApplyImmediately=true`, supervise el estado del grupo de replicación mediante la `DescribeReplicationGroups` acción de la ElastiCache API con los siguientes parámetros. Cuando el estado cambie de `modifying` a `available`, podrá empezar a escribir en su nuevo grupo de replicación reducido.
- `ReplicationGroupId`: nombre del grupo de replicación. Use este parámetro para describir un determinado grupo de replicación en lugar de todos los grupos de replicación.

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=DescribeReplicationGroups  
  &ReplicationGroupId=MyReplGroup  
  &Version=2015-02-02  
  &SignatureVersion=4
```

```
&SignatureMethod=HmacSHA256  
&Timestamp=20150202T192317Z  
&X-Amz-Credential=<credential>
```

Para obtener más información, consulta [DescribeReplicationGroups](#) la referencia de la ElastiCache API de Amazon.

Aumento de la capacidad de lectura

Para aumentar la capacidad de lectura, añada réplicas de lectura (hasta un máximo de cinco) a su grupo de replicación de Redis OSS.

Puede escalar la capacidad de lectura del clúster de Redis OSS mediante la ElastiCache consola AWS CLI, la o la API. ElastiCache Para obtener más información, consulte [Agregar una réplica de lectura, para grupos de replicación de Redis OSS \(modo de clúster deshabilitado\)](#).

Reducción de la capacidad de lectura

Para reducir la capacidad de lectura, elimine una o más réplicas de lectura del clúster de Redis OSS con réplicas (denominado grupo de replicación en la API/CLI). Si el clúster tiene habilitado Multi-AZ con conmutación por error automática, no puede eliminar la última réplica de lectura sin deshabilitar primero Multi-AZ. Para obtener más información, consulte [Modificación de un grupo de reproducción](#).

Para obtener más información, consulte [Eliminar una réplica de lectura, para grupos de replicación de Redis OSS \(modo de clúster deshabilitado\)](#).

Escalar clústeres en Redis OSS (modo de clúster habilitado)

A medida que cambie la demanda de sus clústeres, puede decidir mejorar el rendimiento o reducir los costes cambiando la cantidad de particiones de su clúster de Redis OSS (habilitado para el modo de clúster). Recomendamos que utilice el escalado horizontal online, ya que permite que el clúster continúe sirviendo las solicitudes durante el proceso de escalado.

Entre las condiciones en las que puede decidir cambiar el escalado de su clúster se incluyen las siguientes:

- Presión de memoria:

Si los nodos del clúster tienen presión de la memoria, puede decidir realizar un escalado ascendente para tener más recursos con el fin de almacenar los datos y servir las solicitudes mejor.

Puede determinar si sus nodos están bajo presión de memoria supervisando las siguientes métricas: `FreeableMemorySwapUsage`, y `BytesUseForCache`

- CPUo un cuello de botella en la red:

Si se producen muchos problemas de latencia o rendimiento en su clúster, tal vez deba hacer un escalado ascendente para resolverlos.

Puede supervisar sus niveles de latencia y rendimiento supervisando las siguientes métricas: `CPUUtilization`, `NetworkBytesInNetworkBytesOut`, `CurrConnections` y `NewConnections`

- El escalado del clúster es excesivo:

La demanda actual en su clúster es tal que el escalado descendente no afecta al rendimiento y reduce los costos.

Puede supervisar el uso del clúster para determinar si puede ampliarlo de forma segura utilizando las siguientes métricas: `FreeableMemory`, `SwapUsage`, `BytesUseForCache`, `CPUUtilization`, `NetworkBytesIn`, `NetworkBytesOut`, `CurrConnections`, y `NewConnections`.

Impacto de rendimiento del escalado

Cuando escala utilizando el proceso sin conexión, el clúster no está en línea durante una parte importante del proceso y, por tanto, no puede atender las solicitudes. Cuando escala utilizando el método online, como el escalado es una operación que realiza un uso intensivo de computación, se

deteriora algo el rendimiento, aunque el clúster sigue atendiendo las solicitudes en toda la operación de escalado. El grado de degradación que experimente depende de su CPU uso normal y de sus datos.

Hay dos formas de escalar su clúster de Redis OSS (habilitado para el modo de clúster): escalando horizontal y verticalmente.

- El escalado horizontal le permite cambiar el número de grupos de nodo (fragmentos) en el grupo de reproducción agregando o eliminando grupos de nodos (fragmentos). El proceso de partición en línea le permite escalar verticalmente/horizontalmente mientras el clúster sigue ofreciendo solicitudes entrantes.

Configurar las ranuras en el nuevo clúster de forma diferente que en el clúster anterior. Solo método sin conexión.

- Escalado vertical: cambie el tipo de nodo para cambiar el tamaño del clúster. El proceso de escalado vertical online le permite el escalado ascendente y descendente mientras el clúster sigue ofreciendo solicitudes entrantes.

Si va a reducir el tamaño y la capacidad de memoria del clúster, ampliándolo o reduciéndolo, asegúrese de que la nueva configuración tenga memoria suficiente para sus datos y la sobrecarga de RedisOSS.

Para obtener más información, consulte [Select cache node size](#).

Contenido

- [Refragmentación y reequilibrio de particiones sin conexión para Redis OSS \(modo de clúster habilitado\)](#)
- [Refragmentación y reequilibrio de particiones en línea para Redis OSS \(modo de clúster habilitado\)](#)
 - [Adición de particiones con los cambios de particiones en línea](#)
 - [Eliminación de particiones con los cambios de particiones en línea](#)
 - [Eliminación de particiones \(consola\)](#)
 - [Eliminación de particiones \(AWS CLI\)](#)
 - [Eliminar fragmentos \(API\) ElastiCache](#)
 - [Reequilibrado de particiones en línea](#)
 - [Reequilibrado de fragmentos online \(consola\)](#)
 - [Reequilibrado de particiones en línea \(AWS CLI\)](#)

- [Reequilibrio compartido \(API\) en línea ElastiCache](#)
- [Escalado vertical en línea mediante la modificación del tipo de nodo](#)
- [Escalado vertical en línea](#)
 - [Ampliación de los clústeres de caché de Redis OSS \(consola\)](#)
 - [Ampliación de los clústeres de caché de Redis OSS \(AWS CLI\)](#)
 - [Ampliar los clústeres de caché \(ElastiCache API\) de Redis OSS](#)
- [Reducción vertical en línea](#)
 - [Reducir los clústeres de caché de Redis OSS \(consola\)](#)
 - [Reducir los clústeres de caché de Redis OSS \(AWS CLI\)](#)
 - [Reducir los clústeres de caché \(ElastiCache API\) de Redis OSS](#)

Refragmentación y reequilibrio de particiones sin conexión para Redis OSS (modo de clúster habilitado)

La ventaja principal que obtiene de la reconfiguración de fragmentos sin conexión es que puede hacer algo más que agregar o eliminar fragmentos de su grupo de reproducción. Al hacer cambios de los fragmentos sin conexión, además de cambiar el número de fragmentos en el grupo de reproducción, puede hacer lo siguiente:

 Note

La refragmentación sin conexión no se admite en los clústeres de Redis OSS con la organización de datos en niveles habilitada. Para obtener más información, consulte [Organización de datos en niveles](#).

- Cambiar el tipo de nodo de su grupo de reproducción.
- Especificar la zona de disponibilidad de cada nodo del grupo de reproducción.
- Actualizar a una nueva versión del motor.
- Especificar el número de nodos de réplica de cada fragmento independientemente.
- Especificar el espacio de claves de cada fragmento.

La desventaja principal de la reconfiguración de particiones sin conexión es que el clúster está sin conexión al comentar la parte de restauración del proceso y así continuará hasta que actualice

los puntos de conexión de la aplicación. El tiempo que el clúster está sin conexión varía según la cantidad de datos del clúster.

Para volver a configurar sus fragmentos, Redis OSS (modo de clúster activado) agrupe el clúster sin conexión

1. Cree una copia de seguridad manual de su clúster de Redis OSS existente. Para obtener más información, consulte [Copias de seguridad manuales](#).
2. Cree un nuevo clúster restaurándolo a partir de esta copia de seguridad. Para obtener más información, consulte [Restauración desde una copia de seguridad a una nueva caché](#).
3. Actualice los puntos de conexión de la aplicación a los puntos de conexión del nuevo clúster. Para obtener más información, consulte [Búsqueda de puntos de conexión](#).

Refragmentación y reequilibrio de particiones en línea para Redis OSS (modo de clúster habilitado)

Al utilizar la refragmentación y el reequilibrio de fragmentos en línea con ElastiCache Amazon (Redis OSS) versión 3.2.10 o posterior, puede escalar su clúster de Redis OSS (modo de clúster habilitado) de forma dinámica sin tiempo de inactividad. Este enfoque significa que el clúster puede seguir atendiendo las solicitudes, incluso mientras esté en curso el escalado o el reequilibrado.

Puede hacer lo siguiente:

- Ampliación horizontal: aumente la capacidad de lectura y escritura añadiendo fragmentos (grupos de nodos) a su clúster de Redis OSS (modo de clúster activado) (grupo de replicación).

Si agrega uno o varios fragmentos a su grupo de reproducción, el número de nodos de cada nuevo fragmento es el mismo que el número de nodos en el menor de los fragmentos existentes.

- Amplíe: reduzca la capacidad de lectura y escritura y, por lo tanto, los costes al eliminar los fragmentos de su clúster de Redis OSS (habilitado para el modo de clúster).
- Reequilibre: mueva los espacios clave entre los fragmentos de su clúster de Redis OSS (habilitado para el modo de clúster) para que se distribuyan lo más equitativamente posible entre los fragmentos.

No puede hacer lo siguiente:

- Configurar los fragmentos de forma independiente:

No puede especificar el espacio de claves para fragmentos de manera independiente. Para ello, debe utilizar el proceso sin conexión.

Actualmente, se aplican las siguientes limitaciones a la refragmentación y el reequilibrio en ElastiCache línea (Redis OSS):

- Estos procesos requieren la versión 3.2.10 o posterior del motor OSS de Redis. Para obtener información acerca de la actualización de la versión del motor, consulte [Versiones del motor y actualizaciones](#).
- Existen limitaciones con ranuras o espacios de claves y grandes elementos:

Si alguna de las claves de una partición contiene un elemento grande, la clave no se puede migrar a una partición nueva en el escalado ascendente o el reequilibrado. Esta funcionalidad puede provocar particiones desequilibradas.

Si alguna de las claves de una partición contiene un elemento grande (elementos mayores que 256 MB después de la serialización), esa partición no se elimina en la reducción horizontal. Esta funcionalidad puede provocar que algunas particiones no se eliminen.

- Al realizar el escalado ascendente, el número de nodos de cualquier fragmento nuevo es igual al número de nodo del fragmento existente más pequeño.
- Al realizar el escalado ascendente, las etiquetas que son comunes a todos los fragmentos existentes se copian en los fragmentos nuevos.
- Al ampliar un clúster de almacenamiento de datos global, no ElastiCache replicará automáticamente las funciones de uno de los nodos existentes a los nuevos nodos. Recomendamos cargar las funciones en las particiones nuevas después de escalar horizontalmente el clúster para que todas las particiones tengan las mismas funciones.

Note

En la versión 7 ElastiCache (Redis OSS) o superior: al escalar el clúster, ElastiCache replicará automáticamente las funciones cargadas en uno de los nodos existentes (seleccionadas al azar) en los nuevos nodos. Si su aplicación utiliza [las funciones de Redis OSS](#), le recomendamos que cargue todas las funciones en todos los fragmentos antes de ampliarlos para que su clúster ElastiCache (de Redis OSS) no acabe con definiciones de funciones diferentes en los distintos fragmentos.

Para obtener más información, consulte [Cambio de tamaño de clústeres online](#).

Puede escalar o reequilibrar horizontalmente sus clústeres de Redis OSS (habilitado para el modo de clúster) mediante la AWS Management Console, la y la API. AWS CLI ElastiCache

Adición de particiones con los cambios de particiones en línea

Puede añadir fragmentos a su clúster de Redis OSS (habilitado para el modo de clúster) mediante la AWS Management Console, AWS CLI o la API. ElastiCache Al añadir fragmentos a un clúster de Redis OSS (modo de clúster activado), todas las etiquetas de los fragmentos existentes se copian en los nuevos fragmentos.

Adición de particiones (consola)

Puede utilizar el AWS Management Console para añadir uno o más fragmentos a su clúster de Redis OSS (modo de clúster activado). El siguiente procedimiento describe el proceso.

Para añadir fragmentos a su clúster de Redis OSS (modo de clúster activado)

1. [Abra la ElastiCache consola en https://console.aws.amazon.com/elasticache/](https://console.aws.amazon.com/elasticache/).
2. En el panel de navegación, elija los clústeres de Redis OSS.
3. Busque y elija el nombre del clúster OSS (modo de clúster habilitado) al que desee añadir fragmentos y no el cuadro situado a la izquierda del nombre del clúster.

 Tip

El OSS de Redis (modo de clúster activado) muestra el OSS de Redis agrupado en la columna Modo

4. Elija Add shards (Agregar fragmentos).
 - a. Para Number of shards to be added, elija el número de fragmentos que desee agregar a este clúster.
 - b. En Availability zone(s), elija No preference o Specify availability zones.
 - c. Si ha elegido Specify availability zones, por cada nodo de cada fragmento, seleccione la zona de disponibilidad del nodo en la lista de zonas de disponibilidad.
 - d. Elija Añadir.

Adición de particiones (AWS CLI)

El siguiente proceso describe cómo reconfigurar los fragmentos de su clúster de Redis OSS (modo de clúster habilitado) añadiendo fragmentos mediante el AWS CLI

Use los siguientes parámetros con `modify-replication-group-shard-configuration`.

Parámetros

- `--apply-immediately`: obligatorio. Especifica que la operación de reconfiguración de fragmentos comience inmediatamente.
- `--replication-group-id`: obligatorio. Especifica en qué grupo de replicación (clúster) se debe realizar la operación de reconfiguración de fragmentos.
- `--node-group-count`: obligatorio. Especifica el número de fragmentos (grupos de nodos) que deben existir cuando se complete la operación. Al añadir fragmentos, el valor de `--node-group-count` debe ser mayor que el número actual de fragmentos.

Si lo desea, puede especificar la zona de disponibilidad de cada nodo en el grupo de replicación con `--resharding-configuration`.

- `--resharding-configuration`: opcional. Una lista de zonas de disponibilidad preferidas para cada nodo en cada fragmento en el grupo de replicación. Utilice este parámetro solo si el valor de `--node-group-count` es mayor que el número actual de fragmentos. Si se omite este parámetro al añadir fragmentos, Amazon ElastiCache selecciona las zonas de disponibilidad para los nuevos nodos.

En el siguiente ejemplo, se reconfiguran los espacios de claves de cuatro fragmentos del clúster OSS (modo de clúster habilitado) de Redis. `my-cluster` El ejemplo también especifica la zona de disponibilidad de cada nodo en cada fragmento. La operación comienza inmediatamente.

Example - Adición de fragmentos

Para Linux, macOS o Unix:

```
aws elasticache modify-replication-group-shard-configuration \
  --replication-group-id my-cluster \
  --node-group-count 4 \
  --resharding-configuration \
    "PreferredAvailabilityZones=us-east-2a,us-east-2c" \
    "PreferredAvailabilityZones=us-east-2b,us-east-2a" \
    "PreferredAvailabilityZones=us-east-2c,us-east-2d" \
```

```
"PreferredAvailabilityZones=us-east-2d,us-east-2c" \  
--apply-immediately
```

Para Windows:

```
aws elasticache modify-replication-group-shard-configuration ^  
--replication-group-id my-cluster ^  
--node-group-count 4 ^  
--resharding-configuration ^  
  "PreferredAvailabilityZones=us-east-2a,us-east-2c" ^  
  "PreferredAvailabilityZones=us-east-2b,us-east-2a" ^  
  "PreferredAvailabilityZones=us-east-2c,us-east-2d" ^  
  "PreferredAvailabilityZones=us-east-2d,us-east-2c" ^  
--apply-immediately
```

[Para obtener más información, consulte `-configuration` en la documentación `modify-replication-group-shard`. AWS CLI](#)

Añadir fragmentos (API) ElastiCache

Puede usar la ElastiCache API para reconfigurar los fragmentos de su clúster de Redis OSS (modo de clúster habilitado) en línea mediante la operación.

ModifyReplicationGroupShardConfiguration

Use los siguientes parámetros con `ModifyReplicationGroupShardConfiguration`.

Parámetros

- `ApplyImmediately=true`: obligatorio. Especifica que la operación de reconfiguración de fragmentos comience inmediatamente.
- `ReplicationGroupId`: obligatorio. Especifica en qué grupo de replicación (clúster) se debe realizar la operación de reconfiguración de fragmentos.
- `NodeGroupCount`: obligatorio. Especifica el número de fragmentos (grupos de nodos) que deben existir cuando se complete la operación. Al añadir fragmentos, el valor de `NodeGroupCount` debe ser mayor que el número actual de fragmentos.

Si lo desea, puede especificar la zona de disponibilidad de cada nodo en el grupo de replicación con `ReshardingConfiguration`.

- `ReshardingConfiguration`: opcional. Una lista de zonas de disponibilidad preferidas para cada nodo en cada fragmento en el grupo de replicación. Utilice este parámetro solo si el valor de

NodeGroupCount es mayor que el número actual de fragmentos. Si se omite este parámetro al añadir fragmentos, Amazon ElastiCache selecciona las zonas de disponibilidad para los nuevos nodos.

El siguiente proceso describe cómo reconfigurar los fragmentos de su clúster de Redis OSS (modo de clúster habilitado) añadiendo fragmentos mediante la API. ElastiCache

Example - Adición de fragmentos

En el siguiente ejemplo, se agregan grupos de nodos al clúster OSS de Redis (modo de clúster habilitado) `my-cluster`, de modo que hay un total de cuatro grupos de nodos cuando se completa la operación. El ejemplo también especifica la zona de disponibilidad de cada nodo en cada fragmento. La operación comienza inmediatamente.

```
https://elasticache.us-east-2.amazonaws.com/
  ?Action=ModifyReplicationGroupShardConfiguration
  &ApplyImmediately=true
  &NodeGroupCount=4
  &ReplicationGroupId=my-cluster

  &ReshardingConfiguration.ReshardingConfiguration.1.PreferredAvailabilityZones.AvailabilityZone
east-2a

  &ReshardingConfiguration.ReshardingConfiguration.1.PreferredAvailabilityZones.AvailabilityZone
east-2c

  &ReshardingConfiguration.ReshardingConfiguration.2.PreferredAvailabilityZones.AvailabilityZone
east-2b

  &ReshardingConfiguration.ReshardingConfiguration.2.PreferredAvailabilityZones.AvailabilityZone
east-2a

  &ReshardingConfiguration.ReshardingConfiguration.3.PreferredAvailabilityZones.AvailabilityZone
east-2c

  &ReshardingConfiguration.ReshardingConfiguration.3.PreferredAvailabilityZones.AvailabilityZone
east-2d

  &ReshardingConfiguration.ReshardingConfiguration.4.PreferredAvailabilityZones.AvailabilityZone
east-2d
```

```
&ReshardingConfiguration.ReshardingConfiguration.4.PreferredAvailabilityZones.AvailabilityZone
east-2c
  &Version=2015-02-02
  &SignatureVersion=4
  &SignatureMethod=HmacSHA256
  &Timestamp=20171002T192317Z
  &X-Amz-Credential=<credential>
```

Para obtener más información, consulte [ModifyReplicationGroupShardConfiguration](#) la referencia de la ElastiCache API.

Eliminación de particiones con los cambios de particiones en línea

Puede eliminar fragmentos de su clúster de Redis OSS (modo de clúster habilitado) mediante la API AWS Management Console AWS CLI, oElastiCache .

Temas

- [Eliminación de particiones \(consola\)](#)
- [Eliminación de particiones \(AWS CLI\)](#)
- [Eliminar fragmentos \(API\) ElastiCache](#)

Eliminación de particiones (consola)

El siguiente proceso describe cómo reconfigurar los fragmentos de su clúster de Redis OSS (modo de clúster habilitado) mediante la eliminación de los fragmentos mediante el. AWS Management Console

Antes de eliminar grupos de nodos (fragmentos) del grupo de replicación, ElastiCache asegúrese de que todos los datos quepan en los fragmentos restantes. Si los datos caben, los fragmentos especificados se eliminan del grupo de reproducción según lo solicitado. Si los datos no van a caber en los grupos de nodos restantes, el proceso se termina y el grupo de reproducción se deja con la misma configuración de nodo de antes de que se hiciera la solicitud.

Puede usarlo AWS Management Console para eliminar uno o más fragmentos de su clúster de Redis OSS (habilitado para el modo de clúster). No puede eliminar todos los fragmentos de un grupo de replicación. En su lugar, debe eliminar el grupo de reproducción. Para obtener más información, consulte [Eliminación de un grupo de reproducción](#). El siguiente procedimiento describe el proceso para eliminar uno o varios fragmentos.

Para eliminar fragmentos de su clúster de Redis OSS (modo de clúster activado)

1. [Abra la ElastiCache consola en https://console.aws.amazon.com/elasticache/](https://console.aws.amazon.com/elasticache/).
2. En el panel de navegación, elija los clústeres de Redis OSS.
3. Busque y elija el nombre del clúster de Redis OSS (modo de clúster habilitado) del que desea eliminar los fragmentos y no la casilla situada a la izquierda del nombre del clúster.

 Tip

Los clústeres OSS de Redis (modo de clúster activado) tienen un valor igual o superior a 1 en la columna Fragmentos.

4. En la lista de fragmentos, elija la casilla situada a la izquierda del nombre de cada fragmento que desee eliminar.
5. Elija Delete shard (Eliminar fragmentos).

Eliminación de particiones (AWS CLI)

El siguiente proceso describe cómo reconfigurar los fragmentos de su clúster de Redis OSS (modo de clúster habilitado) mediante la eliminación de los fragmentos mediante el AWS CLI

 Important

Antes de eliminar grupos de nodos (fragmentos) del grupo de replicación, ElastiCache asegúrese de que todos los datos quepan en los fragmentos restantes. Si los datos caben, los fragmentos especificados (`--node-groups-to-remove`) se eliminan del grupo de reproducción según lo solicitado y sus espacios de clave se asignan a los fragmentos restantes. Si los datos no van a caber en los grupos de nodos restantes, el proceso se termina y el grupo de reproducción se deja con la misma configuración de nodo de antes de que se hiciera la solicitud.

Puede usarlo AWS CLI para eliminar uno o más fragmentos de su clúster de Redis OSS (habilitado para el modo de clúster). No puede eliminar todos los fragmentos de un grupo de replicación. En su lugar, debe eliminar el grupo de reproducción. Para obtener más información, consulte [Eliminación de un grupo de reproducción](#).

Use los siguientes parámetros con `modify-replication-group-shard-configuration`.

Parámetros

- `--apply-immediately`: obligatorio. Especifica que la operación de reconfiguración de fragmentos comience inmediatamente.
- `--replication-group-id`: obligatorio. Especifica en qué grupo de replicación (clúster) se debe realizar la operación de reconfiguración de fragmentos.
- `--node-group-count`: obligatorio. Especifica el número de fragmentos (grupos de nodos) que deben existir cuando se complete la operación. Al eliminar fragmentos, el valor de `--node-group-count` debe ser menor que el número actual de fragmentos.
- `--node-groups-to-remove`: obligatorio cuando `--node-group-count` es menor que el número actual de grupos de nodos (particiones). Una lista de ID de fragmento (grupo de nodos) para eliminar del grupo de replicación.

El siguiente procedimiento describe el proceso para eliminar uno o varios fragmentos.

Example - Eliminación de fragmentos

En el siguiente ejemplo, se eliminan dos grupos de nodos del clúster OSS de Redis (modo de clúster activado) `my-cluster`, por lo que hay un total de dos grupos de nodos cuando se completa la operación. Los espacios de claves de los fragmentos eliminados se distribuyen de manera uniforme en los fragmentos restantes.

Para Linux, macOS o Unix:

```
aws elasticache modify-replication-group-shard-configuration \  
  --replication-group-id my-cluster \  
  --node-group-count 2 \  
  --node-groups-to-remove "0002" "0003" \  
  --apply-immediately
```

Para Windows:

```
aws elasticache modify-replication-group-shard-configuration ^  
  --replication-group-id my-cluster ^  
  --node-group-count 2 ^  
  --node-groups-to-remove "0002" "0003" ^  
  --apply-immediately
```

Eliminar fragmentos (API) ElastiCache

Puede usar la ElastiCache API para reconfigurar los fragmentos de su clúster de Redis OSS (modo de clúster habilitado) en línea mediante la operación.

`ModifyReplicationGroupShardConfiguration`

El siguiente proceso describe cómo reconfigurar los fragmentos de su clúster de OSS (modo de clúster habilitado) de Redis mediante la eliminación de los fragmentos mediante la API. ElastiCache

Important

Antes de eliminar grupos de nodos (fragmentos) de su grupo de replicación, ElastiCache asegúrese de que todos los datos quepan en los fragmentos restantes. Si los datos caben, los fragmentos especificados (`NodeGroupsToRemove`) se eliminan del grupo de reproducción según lo solicitado y sus espacios de clave se asignan a los fragmentos restantes. Si los datos no van a caber en los grupos de nodos restantes, el proceso se termina y el grupo de reproducción se deja con la misma configuración de nodo de antes de que se hiciera la solicitud.

Puede usar la ElastiCache API para eliminar uno o más fragmentos de su clúster de Redis OSS (modo de clúster habilitado). No puede eliminar todos los fragmentos de un grupo de replicación. En su lugar, debe eliminar el grupo de reproducción. Para obtener más información, consulte [Eliminación de un grupo de reproducción](#).

Use los siguientes parámetros con `ModifyReplicationGroupShardConfiguration`.

Parámetros

- `ApplyImmediately=true`: obligatorio. Especifica que la operación de reconfiguración de fragmentos comience inmediatamente.
- `ReplicationGroupId`: obligatorio. Especifica en qué grupo de replicación (clúster) se debe realizar la operación de reconfiguración de fragmentos.
- `NodeGroupCount`: obligatorio. Especifica el número de fragmentos (grupos de nodos) que deben existir cuando se complete la operación. Al eliminar fragmentos, el valor de `NodeGroupCount` debe ser menor que el número actual de fragmentos.
- `NodeGroupsToRemove`: obligatorio cuando `--node-group-count` es menor que el número actual de grupos de nodos (particiones). Una lista de ID de fragmento (grupo de nodos) para eliminar del grupo de replicación.

El siguiente procedimiento describe el proceso para eliminar uno o varios fragmentos.

Example - Eliminación de fragmentos

En el siguiente ejemplo, se eliminan dos grupos de nodos del clúster OSS de Redis (modo de clúster habilitado) `my-cluster`, por lo que hay un total de dos grupos de nodos cuando se completa la operación. Los espacios de claves de los fragmentos eliminados se distribuyen de manera uniforme en los fragmentos restantes.

```
https://elasticache.us-east-2.amazonaws.com/  
  ?Action=ModifyReplicationGroupShardConfiguration  
  &ApplyImmediately=true  
  &NodeGroupCount=2  
  &ReplicationGroupId=my-cluster  
  &NodeGroupsToRemove.member.1=0002  
  &NodeGroupsToRemove.member.2=0003  
  &Version=2015-02-02  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Timestamp=20171002T192317Z  
  &X-Amz-Credential=<credential>
```

Reequilibrado de particiones en línea

Puede reequilibrar las particiones de su clúster de Redis OSS (modo de clúster habilitado) mediante la AWS Management Console, AWS CLI o la API. ElastiCache

Temas

- [Reequilibrado de fragmentos online \(consola\)](#)
- [Reequilibrado de particiones en línea \(AWS CLI\)](#)
- [Reequilibrio compartido \(API\) en línea ElastiCache](#)

Reequilibrado de fragmentos online (consola)

El siguiente proceso describe cómo reconfigurar los fragmentos de su clúster de Redis OSS (modo de clúster habilitado) reequilibrando los fragmentos mediante el. AWS Management Console

Para reequilibrar los espacios clave entre los fragmentos de su clúster de Redis OSS (modo de clúster habilitado)

1. [Abra la consola en https://console.aws.amazon.com/elasticache/ ElastiCache](https://console.aws.amazon.com/elasticache/) .
2. En el panel de navegación, elija los clústeres de Redis OSS.
3. Elija el nombre, no la casilla situada a la izquierda del nombre, del clúster OSS (modo de clúster habilitado) de Redis que desee reequilibrar.

 Tip

Los clústeres OSS de Redis (modo de clúster activado) tienen un valor igual o superior a 1 en la columna Fragmentos.

4. Elija Rebalance.
5. Cuando se solicite, elija Rebalance. Puede ver un mensaje similar a este: *Slots in the replication group are uniformly distributed (Las ranuras del grupo de reproducción se distribuyen de forma uniforme). Nothing to do. (Servicio: código de estado: 400AmazonElastiCache; código de error;; ID de solicitud: InvalidReplicationGroupState 2246cebd-9721-11e7-8d5b-e1b0f086c8cf)*. Si lo ve, elija Cancel.

Reequilibrado de particiones en línea (AWS CLI)

Use los siguientes parámetros con `modify-replication-group-shard-configuration`.

Parámetros

- `-apply-immediately`: obligatorio. Especifica que la operación de reconfiguración de fragmentos comience inmediatamente.
- `--replication-group-id`: obligatorio. Especifica en qué grupo de replicación (clúster) se debe realizar la operación de reconfiguración de fragmentos.
- `--node-group-count`: obligatorio. Para reequilibrar los espacios de claves en todos los fragmentos del clúster, este valor debe ser igual que el número de fragmentos.

El siguiente proceso describe cómo reconfigurar los fragmentos de su clúster de Redis OSS (modo de clúster habilitado) reequilibrando los fragmentos mediante el AWS CLI

Example - Reequilibrado de los fragmentos en un clúster

En el siguiente ejemplo, se reequilibran las ranuras del clúster OSS de Redis (modo de clúster habilitado) `my-cluster` para que las ranuras se distribuyan de la forma más equitativa posible. El valor de `--node-group-count` (4) es el número de fragmentos que hay actualmente en el clúster.

Para Linux, macOS o Unix:

```
aws elasticache modify-replication-group-shard-configuration \  
  --replication-group-id my-cluster \  
  --node-group-count 4 \  
  --apply-immediately
```

Para Windows:

```
aws elasticache modify-replication-group-shard-configuration ^  
  --replication-group-id my-cluster ^  
  --node-group-count 4 ^  
  --apply-immediately
```

Reequilibrio compartido (API) en línea ElastiCache

Puede usar la ElastiCache API para reconfigurar los fragmentos de su clúster de Redis OSS (modo de clúster habilitado) en línea mediante la operación.

ModifyReplicationGroupShardConfiguration

Use los siguientes parámetros con `ModifyReplicationGroupShardConfiguration`.

Parámetros

- `ApplyImmediately=true`: obligatorio. Especifica que la operación de reconfiguración de fragmentos comience inmediatamente.
- `ReplicationGroupId`: obligatorio. Especifica en qué grupo de replicación (clúster) se debe realizar la operación de reconfiguración de fragmentos.
- `NodeGroupCount`: obligatorio. Para reequilibrar los espacios de claves en todos los fragmentos del clúster, este valor debe ser igual que el número de fragmentos.

El siguiente proceso describe cómo reconfigurar los fragmentos de su clúster de Redis OSS (modo de clúster habilitado) mediante el reequilibrio de los fragmentos mediante la API. ElastiCache

Example - Reequilibrado de un clúster

En el siguiente ejemplo, se reequilibran las ranuras del clúster OSS de Redis (modo de clúster habilitado) `my-cluster` para que las ranuras se distribuyan de la forma más equitativa posible. El valor de `NodeGroupCount` (4) es el número de fragmentos que hay actualmente en el clúster.

```
https://elasticache.us-east-2.amazonaws.com/  
?Action=ModifyReplicationGroupShardConfiguration  
&ApplyImmediately=true  
&NodeGroupCount=4  
&ReplicationGroupId=my-cluster  
&Version=2015-02-02  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20171002T192317Z  
&X-Amz-Credential=<credential>
```

Escalado vertical en línea mediante la modificación del tipo de nodo

Al utilizar el escalado vertical online con Amazon ElastiCache (RedisOSS) versión 3.2.10 o posterior, puede escalar sus OSS clústeres de Redis de forma dinámica con un tiempo de inactividad mínimo. Esto permite que su OSS clúster de Redis atienda las solicitudes incluso mientras se escala.

Note

No se admite el escalado entre un clúster de organización de datos en niveles (por ejemplo, un clúster que utiliza un tipo de nodo `r6gd`) y un clúster que no utiliza la organización de datos en niveles (por ejemplo, un clúster que utiliza un tipo de nodo `r6g`). Para obtener más información, consulte [Organización de datos en niveles](#).

Puede hacer lo siguiente:

- **Ampliar:** aumente la capacidad de lectura y escritura ajustando el tipo de nodo de su OSS clúster de Redis para utilizar un tipo de nodo más grande.

ElastiCache cambia el tamaño de su clúster de forma dinámica mientras permanece en línea y atiende las solicitudes.

- **Reducción vertical:** reduzca verticalmente la capacidad de lectura y escritura al ajustar el tipo de nodo para utilizar un nodo más pequeño. De nuevo, cambia el tamaño del clúster de ElastiCache

forma dinámica mientras permanece en línea y atiende las solicitudes. En este caso, reduzca los costos reduciendo el tamaño del nodo.

Note

Los procesos de escalado ascendente y descendente dependen de la creación de clústeres con tipos de nodo seleccionados recientemente y la sincronización de los nuevos nodos con los anteriores. Para garantizar un flujo de escalado ascendente/descendente uniforme, realice el siguiente procedimiento:

- Asegúrese de tener suficiente capacidad ENI (interfaz de red elástica). Si se aplica el escalado descendente, asegúrese de que el nodo de tamaño más reducido tenga suficiente memoria para absorber el tráfico esperado.

Consulte [Administración de la memoria reservada](#) para conocer las prácticas recomendadas sobre la administración de memoria.

- Aunque el proceso de escalado vertical está diseñado para que sea completamente online, se basa en la sincronización de datos entre el nodo antiguo y el nuevo. Recomendamos que inicie el escalado ascendente/descendente durante las horas en las que espera que el tráfico de datos sea mínimo.
- Pruebe el comportamiento de la aplicación durante el escalado en un entorno de ensayo, si es posible.

Contenido

- [Escalado vertical en línea](#)
 - [Ampliación de los clústeres de caché de Redis OSS \(consola\)](#)
 - [Ampliación de los clústeres de caché de Redis OSS \(AWS CLI\)](#)
 - [Ampliar los clústeres de caché \(ElastiCache API\) de Redis OSS](#)
- [Reducción vertical en línea](#)
 - [Reducir los clústeres de caché de Redis OSS \(consola\)](#)
 - [Reducir los clústeres de caché de Redis OSS \(AWS CLI\)](#)
 - [Reducir los clústeres de caché \(ElastiCache API\) de Redis OSS](#)

Escalado vertical en línea

Temas

- [Ampliación de los clústeres de caché de Redis OSS \(consola\)](#)
- [Ampliación de los clústeres de caché de Redis OSS \(AWS CLI\)](#)
- [Ampliar los clústeres de caché \(ElastiCache API\) de Redis OSS](#)

Ampliación de los clústeres de caché de Redis OSS (consola)

El siguiente procedimiento describe cómo escalar un clúster de Redis OSS mediante la consola ElastiCache de administración. Durante este proceso, el clúster de Redis OSS seguirá atendiendo las solicitudes con un tiempo de inactividad mínimo.

Para ampliar un clúster OSS de Redis (consola)

1. Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación, elija los clústeres de Redis OSS.
3. En la lista de clústeres, elija el clúster.
4. Elija Modificar.
5. En el asistente Modify Cluster:
 - Elija el tipo de nodo que desee ampliar en la lista Node type. Para aplicar el escalado ascendente, seleccione un tipo de nodo superior a su nodo existente.
6. Si desea realizar el proceso de escalado ascendente de forma inmediata, elija el cuadro Apply immediately (Aplicar inmediatamente). Si no selecciona el cuadro Apply immediately, el proceso de ampliación se realizará durante el siguiente periodo de mantenimiento de este clúster.
7. Elija Modify (Modificar).

Si eligió Apply immediately (Aplicar inmediatamente) en el paso anterior, el estado del clúster cambiará a modifying (en modificación). Cuando el estado cambie a available (disponible), la modificación se habrá completado y podrá empezar a utilizar el nuevo clúster.

Ampliación de los clústeres de caché de Redis OSS (AWS CLI)

El siguiente procedimiento describe cómo escalar un clúster de caché OSS de Redis mediante el AWS CLI. Durante este proceso, el clúster de OSS de Redis seguirá atendiendo las solicitudes con un tiempo de inactividad mínimo.

Para ampliar un clúster de caché de Redis OSS (AWS CLI)

1. Determine los tipos de nodos a los que puede escalar ejecutando el AWS CLI `list-allowed-node-type-modifications` comando con el siguiente parámetro.

Para Linux, macOS o Unix:

```
aws elasticache list-allowed-node-type-modifications \  
  --replication-group-id my-replication-group-id
```

Para Windows:

```
aws elasticache list-allowed-node-type-modifications ^  
  --replication-group-id my-replication-group-id
```

La salida del comando anterior es similar a la siguiente (formato JSON).

```
{  
  "ScaleUpModifications": [  
    "cache.m3.2xlarge",  
    "cache.m3.large",  
    "cache.m3.xlarge",  
    "cache.m4.10xlarge",  
    "cache.m4.2xlarge",  
    "cache.m4.4xlarge",  
    "cache.m4.large",  
    "cache.m4.xlarge",  
    "cache.r3.2xlarge",  
    "cache.r3.4xlarge",  
    "cache.r3.8xlarge",  
    "cache.r3.large",  
    "cache.r3.xlarge"  
  ],  
  "ScaleDownModifications": [  
    "cache.t2.micro",
```

```
        "cache.t2.small ",
        "cache.t2.medium",
        "cache.t1.small "
    ],
}
```

Para obtener más información consulte [list-allowed-node-type-modifications](#) en la Referencia de la AWS CLI .

2. Modifique el grupo de replicación para ampliarlo hasta el nuevo tipo de nodo más grande mediante el AWS CLI `modify-replication-group` comando y los siguientes parámetros.
 - `--replication-group-id`: el nombre del grupo de reproducción al que va a escalar verticalmente.
 - `--cache-node-type`: tipo de nodo nuevo al que desea escalar el clúster de caché. Este valor debe ser uno de los tipos de nodos devueltos por el comando `list-allowed-node-type-modifications` en el paso 1.
 - `--cache-parameter-group-name`: [Opcional] Utilice este parámetro si va a utilizar `reserved-memory` para administrar la memoria reservada de su clúster. Especifique un grupo de parámetros de caché personalizado que reserve la cantidad correcta de memoria para el nuevo tipo de nodo. Si va a utilizar `reserved-memory-percent`, puede omitir este parámetro.
 - `--apply-immediately`: hace que el proceso de escalado vertical se aplique de inmediato. Para aplazar el proceso de ampliación al siguiente periodo de mantenimiento del clúster, utilice el parámetro `--no-apply-immediately`.

Para Linux, macOS o Unix:

```
aws elasticache modify-replication-group \
  --replication-group-id my-redis-cluster \
  --cache-node-type cache.m3.xlarge \
  --apply-immediately
```

Para Windows:

```
aws elasticache modify-replication-group ^
  --replication-group-id my-redis-cluster ^
  --cache-node-type cache.m3.xlarge ^
```

```
--apply-immediately
```

La salida del comando anterior es similar a la siguiente (formato JSON).

```
{
  "ReplicationGroup": {
    "Status": "modifying",
    "Description": "my-redis-cluster",
    "NodeGroups": [
      {
        "Status": "modifying",
        "Slots": "0-16383",
        "NodeGroupId": "0001",
        "NodeGroupMembers": [
          {
            "PreferredAvailabilityZone": "us-east-1f",
            "CacheNodeId": "0001",
            "CacheClusterId": "my-redis-cluster-0001-001"
          },
          {
            "PreferredAvailabilityZone": "us-east-1d",
            "CacheNodeId": "0001",
            "CacheClusterId": "my-redis-cluster-0001-002"
          }
        ]
      }
    ],
    "ConfigurationEndpoint": {
      "Port": 6379,
      "Address": "my-redis-cluster.r7gdfi.clustercfg.use1.cache.amazonaws.com"
    },
    "ClusterEnabled": true,
    "ReplicationGroupId": "my-redis-cluster",
    "SnapshotRetentionLimit": 1,
    "AutomaticFailover": "enabled",
    "SnapshotWindow": "07:30-08:30",
    "MemberClusters": [
      "my-redis-cluster-0001-001",
      "my-redis-cluster-0001-002"
    ],
    "CacheNodeType": "cache.m3.xlarge",
  }
}
```

```
    "DataTiering": "disabled"  
    "PendingModifiedValues": {}  
  }  
}
```

Para obtener más información consulte [modify-replication-group](#) en la Referencia de la AWS CLI

3. Si utilizó el `--apply-immediately`, compruebe el estado del clúster de caché mediante el `AWS CLI describe-cache-clusters` comando con el siguiente parámetro. Cuando el estado cambie a `available` (disponible), podrá comenzar a usar el nuevo nodo de clúster de la caché de mayor tamaño.

Ampliar los clústeres de caché (ElastiCache API) de Redis OSS

El siguiente proceso escala el clúster de caché desde su tipo de nodo actual a un tipo de nodo nuevo y más grande mediante la ElastiCache API. Durante este proceso, ElastiCache (Redis OSS) actualiza las entradas de DNS para que apunten a los nuevos nodos. Por ello, no tiene que actualizar los puntos de enlace de su aplicación. Para Redis OSS 5.0.5 y versiones posteriores, puede escalar los clústeres habilitados para la conmutación automática por error mientras el clúster permanece en línea y atiende las solicitudes entrantes. En la versión 4.0.10 y anteriores, puede observar una breve interrupción de las versiones de lectura y escritura del nodo principal mientras se actualiza la entrada de DNS.

El tiempo que se tarda en la ampliación a un tipo de nodo más grande varía en función de su tipo de nodo y de la cantidad de datos de su clúster de caché actual.

Para ampliar un clúster de caché (API) de Redis OSS ElastiCache

1. Determine los tipos de nodos a los que puede escalar mediante la `ListAllowedNodeTypeModifications` acción de la ElastiCache API con el siguiente parámetro.
 - `ReplicationGroupId`: nombre del grupo de replicación. Use este parámetro para describir un grupo de replicación específico en lugar de todos los grupos de replicación.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=ListAllowedNodeTypeModifications  
&ReplicationGroupId=MyReplGroup
```

```
&Version=2015-02-02
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>
```

Para obtener más información, consulta [ListAllowedNodeTypeModifications](#) la referencia de la ElastiCache API de Amazon.

- Amplíe su grupo de replicación actual hasta el nuevo tipo de nodo mediante la acción de la `ModifyReplicationGroup` ElastiCache API y con los siguientes parámetros.
 - `ReplicationGroupId`: nombre del grupo de replicación.
 - `CacheNodeType`: el nuevo tipo de nodo más grande de clústeres de caché en este grupo de reproducción. Este valor debe ser uno de los tipos de instancia devueltos por la acción `ListAllowedNodeTypeModifications` en el paso 1.
 - `CacheParameterGroupName`: [Opcional] Utilice este parámetro si va a utilizar `reserved-memory` para administrar la memoria reservada de su clúster. Especifique un grupo de parámetros de caché personalizado que reserve la cantidad correcta de memoria para el nuevo tipo de nodo. Si va a utilizar `reserved-memory-percent`, puede omitir este parámetro.
 - `ApplyImmediately`: establézcalo en `true` para que el proceso de escalado vertical se aplique de inmediato. Para aplazar el proceso de ampliación al siguiente periodo de mantenimiento, utilice `ApplyImmediately=false`.

```
https://elasticache.us-west-2.amazonaws.com/
?Action=ModifyReplicationGroup
&ApplyImmediately=true
&CacheNodeType=cache.m3.2xlarge
&CacheParameterGroupName=redis32-m3-2x1
&ReplicationGroupId=myReplGroup
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20141201T220302Z
&Version=2014-12-01
&X-Amz-Algorithm=&AWS;4-HMAC-SHA256
&X-Amz-Date=20141201T220302Z
&X-Amz-SignedHeaders=Host
&X-Amz-Expires=20141201T220302Z
```

```
&X-Amz-Credential=<credential>  
&X-Amz-Signature=<signature>
```

Para obtener más información, consulta [ModifyReplicationGroup](#) la referencia de la ElastiCache API de Amazon.

3. Si lo utilizó `ApplyImmediately=true`, supervise el estado del grupo de replicación mediante la `DescribeReplicationGroups` acción de la ElastiCache API con los siguientes parámetros. Cuando el estado cambie de `modifying` a `available`, podrá empezar a escribir en su nuevo grupo de replicación ampliado.
 - `ReplicationGroupId`: nombre del grupo de replicación. Use este parámetro para describir un determinado grupo de replicación en lugar de todos los grupos de replicación.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=DescribeReplicationGroups  
&ReplicationGroupId=MyReplGroup  
&Version=2015-02-02  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20150202T192317Z  
&X-Amz-Credential=<credential>
```

Para obtener más información, consulta [DescribeReplicationGroups](#) la referencia de la ElastiCache API de Amazon.

Reducción vertical en línea

Temas

- [Reducir los clústeres de caché de Redis OSS \(consola\)](#)
- [Reducir los clústeres de caché de Redis OSS \(AWS CLI\)](#)
- [Reducir los clústeres de caché \(ElastiCache API\) de Redis OSS](#)

Reducir los clústeres de caché de Redis OSS (consola)

El siguiente procedimiento describe cómo reducir la escala de un clúster OSS de Redis mediante la consola ElastiCache de administración. Durante este proceso, el clúster de Redis OSS seguirá atendiendo las solicitudes con un tiempo de inactividad mínimo.

Para reducir la escala de un clúster OSS de Redis (consola)

1. Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación, elija los clústeres de Redis OSS.
3. En la lista de clústeres, elija el clúster preferido.
4. Elija Modificar.
5. En el asistente Modify Cluster:
 - Elija el tipo de nodo que desee ampliar en la lista Node type. Para aplicar el escalado descendente, seleccione un tipo de nodo inferior a su nodo existente. Tenga en cuenta que no todos los tipos de nodos están disponibles para el proceso de reducción.
6. Si desea realizar el proceso de escalado descendente de forma inmediata, elija el cuadro Apply immediately (Aplicar inmediatamente). Si no selecciona el cuadro Apply immediately (Aplicar inmediatamente), el proceso de escalado descendente se realizará durante el siguiente periodo de mantenimiento de este clúster.
7. Elija Modify (Modificar).

Si eligió Apply immediately (Aplicar inmediatamente) en el paso anterior, el estado del clúster cambiará a modifying (en modificación). Cuando el estado cambie a available (disponible), la modificación se habrá completado y podrá empezar a utilizar el nuevo clúster.

Reducir los clústeres de caché de Redis OSS (AWS CLI)

El siguiente procedimiento describe cómo reducir la escala de un clúster de caché OSS de Redis mediante AWS CLI. Durante este proceso, el clúster de OSS de Redis seguirá atendiendo las solicitudes con un tiempo de inactividad mínimo.

Para reducir la escala de un clúster de caché de Redis OSS (AWS CLI)

1. Determine los tipos de nodos a los que puede reducir la escala ejecutando el AWS CLI `list-allowed-node-type-modifications` comando con el siguiente parámetro.

Para Linux, macOS o Unix:

```
aws elasticache list-allowed-node-type-modifications \  
  --replication-group-id my-replication-group-id
```

Para Windows:

```
aws elasticache list-allowed-node-type-modifications ^  
  --replication-group-id my-replication-group-id
```

La salida del comando anterior es similar a la siguiente (formato JSON).

```
{  
  "ScaleUpModifications": [  
    "cache.m3.2xlarge",  
    "cache.m3.large",  
    "cache.m3.xlarge",  
    "cache.m4.10xlarge",  
    "cache.m4.2xlarge",  
    "cache.m4.4xlarge",  
    "cache.m4.large",  
    "cache.m4.xlarge",  
    "cache.r3.2xlarge",  
    "cache.r3.4xlarge",  
    "cache.r3.8xlarge",  
    "cache.r3.large",  
    "cache.r3.xlarge"  
  ]  
  
  "ScaleDownModifications": [  
    "cache.t2.micro",  
    "cache.t2.small",  
    "cache.t2.medium",  
    "cache.t1.small"  
  ]  
}
```

Para obtener más información consulte [list-allowed-node-type-modifications](#) en la Referencia de la AWS CLI .

2. Modifique el grupo de replicación para reducirlo al nuevo tipo de nodo más pequeño mediante el AWS CLI `modify-replication-group` comando y los siguientes parámetros.
 - `--replication-group-id`: el nombre del grupo de reproducción al que va a reducir verticalmente.

- `--cache-node-type`: tipo de nodo nuevo al que desea escalar el clúster de caché. Este valor debe ser uno de los tipos de nodos devueltos por el comando `list-allowed-node-type-modifications` en el paso 1.
- `--cache-parameter-group-name`: [Opcional] Utilice este parámetro si va a utilizar `reserved-memory` para administrar la memoria reservada de su clúster. Especifique un grupo de parámetros de caché personalizado que reserve la cantidad correcta de memoria para el nuevo tipo de nodo. Si va a utilizar `reserved-memory-percent`, puede omitir este parámetro.
- `--apply-immediately`: hace que el proceso de escalado vertical se aplique de inmediato. Para aplazar el proceso de reducción al siguiente periodo de mantenimiento del clúster, utilice el parámetro `--no-apply-immediately`.

Para Linux, macOS o Unix:

```
aws elasticache modify-replication-group \  
  --replication-group-id my-redis-cluster \  
  --cache-node-type cache.t2.micro \  
  --apply-immediately
```

Para Windows:

```
aws elasticache modify-replication-group ^  
  --replication-group-id my-redis-cluster ^  
  --cache-node-type cache.t2.micro ^  
  --apply-immediately
```

La salida del comando anterior es similar a la siguiente (formato JSON).

```
{  
  "ReplicationGroup": {  
    "Status": "modifying",  
    "Description": "my-redis-cluster",  
    "NodeGroups": [  
      {  
        "Status": "modifying",  
        "Slots": "0-16383",  
        "NodeGroupId": "0001",
```

```

        "NodeGroupMembers": [
            {
                "PreferredAvailabilityZone": "us-east-1f",
                "CacheNodeId": "0001",
                "CacheClusterId": "my-redis-cluster-0001-001"
            },
            {
                "PreferredAvailabilityZone": "us-east-1d",
                "CacheNodeId": "0001",
                "CacheClusterId": "my-redis-cluster-0001-002"
            }
        ]
    },
    "ConfigurationEndpoint": {
        "Port": 6379,
        "Address": "my-redis-
cluster.r7gdfi.clustercfg.use1.cache.amazonaws.com"
    },
    "ClusterEnabled": true,
    "ReplicationGroupId": "my-redis-cluster",
    "SnapshotRetentionLimit": 1,
    "AutomaticFailover": "enabled",
    "SnapshotWindow": "07:30-08:30",
    "MemberClusters": [
        "my-redis-cluster-0001-001",
        "my-redis-cluster-0001-002"
    ],
    "CacheNodeType": "cache.t2.micro",
    "DataTiering": "disabled"
    "PendingModifiedValues": {}
}
}

```

Para obtener más información consulte [modify-replication-group](#) en la Referencia de la AWS CLI

3. Si utilizó el `--apply-immediately`, compruebe el estado del clúster de caché mediante el AWS CLI `describe-cache-clusters` comando con el siguiente parámetro. Cuando el estado cambie a `available` (disponible), puede comenzar con el nuevo clúster de caché de menor tamaño.

Reducir los clústeres de caché (ElastiCache API) de Redis OSS

El siguiente proceso escala el grupo de replicación desde su tipo de nodo actual a un tipo de nodo nuevo y más pequeño mediante la ElastiCache API. Durante este proceso, su clúster de OSS de Redis seguirá atendiendo las solicitudes con un tiempo de inactividad mínimo.

El tiempo que se tarda en la realización del escalado descendente a un tipo de nodo más pequeño varía en función de su tipo de nodo y de la cantidad de datos de su clúster de caché actual.

Reducción de escala (ElastiCache API)

1. Determina los tipos de nodos a los que puedes reducir la escala mediante la `ListAllowedNodeTypeModifications` acción de la ElastiCache API con el siguiente parámetro.
 - `ReplicationGroupId`: nombre del grupo de replicación. Use este parámetro para describir un grupo de replicación específico en lugar de todos los grupos de replicación.

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=ListAllowedNodeTypeModifications  
  &ReplicationGroupId=MyReplGroup  
  &Version=2015-02-02  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Timestamp=20150202T192317Z  
  &X-Amz-Credential=<credential>
```

Para obtener más información, consulta [ListAllowedNodeTypeModifications](#) la referencia de la ElastiCache API de Amazon.

2. Reduzca su grupo de replicación actual al nuevo tipo de nodo mediante la acción de la `ModifyReplicationGroup` ElastiCache API y con los siguientes parámetros.
 - `ReplicationGroupId`: nombre del grupo de replicación.
 - `CacheNodeType`: el nuevo tipo de nodo más pequeño de clústeres de caché en este grupo de reproducción. Este valor debe ser uno de los tipos de instancia devueltos por la acción `ListAllowedNodeTypeModifications` en el paso 1.
 - `CacheParameterGroupName`: [Opcional] Utilice este parámetro si va a utilizar `reserved-memory` para administrar la memoria reservada de su clúster. Especifique un grupo de

parámetros de caché personalizado que reserve la cantidad correcta de memoria para el nuevo tipo de nodo. Si va a utilizar `reserved-memory-percent`, puede omitir este parámetro.

- `ApplyImmediately`: establézcalo en `true` para que el proceso de reducción vertical se aplique de inmediato. Para aplazar el proceso de escalado descendente al siguiente periodo de mantenimiento, utilice `ApplyImmediately=false`.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=ModifyReplicationGroup  
&ApplyImmediately=true  
&CacheNodeType=cache.t2.micro  
&CacheParameterGroupName=redis32-m3-2x1  
&ReplicationGroupId=myReplGroup  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20141201T220302Z  
&Version=2014-12-01  
&X-Amz-Algorithm=&AWS;4-HMAC-SHA256  
&X-Amz-Date=20141201T220302Z  
&X-Amz-SignedHeaders=Host  
&X-Amz-Expires=20141201T220302Z  
&X-Amz-Credential=<credential>  
&X-Amz-Signature=<signature>
```

Para obtener más información, consulta [ModifyReplicationGroup](#) la referencia de la ElastiCache API de Amazon.

Introducción a JSON en ElastiCache (Redis OSS)

ElastiCache (Redis OSS) admite el formato nativo de notación de JavaScript objetos (JSON), que es una forma sencilla y sin esquemas de codificar conjuntos de datos complejos dentro de los clústeres de Redis OSS. Puede almacenar y acceder a los datos de forma nativa mediante el formato de notación de JavaScript objetos (JSON) dentro de los clústeres OSS de Redis, y actualizar los datos JSON almacenados en esos clústeres, sin necesidad de gestionar un código personalizado para serializarlos y deserializarlos.

Además de utilizar las operaciones de la API OSS de Redis para las aplicaciones que funcionan con JSON, ahora puede recuperar y actualizar de forma eficiente partes específicas de un documento

JSON sin necesidad de manipular todo el objeto. Esto puede mejorar el rendimiento y reducir los costos. También puede buscar en el contenido de su documento JSON mediante la consulta [Estilo GoessnerJSONPath](#).

Después de crear un clúster con una versión de motor compatible, el tipo de datos JSON y los comandos asociados están disponibles automáticamente. Es compatible con la API y la RDB con la versión 2 del módulo RedisJSON, por lo que puede migrar fácilmente las aplicaciones Redis OSS existentes basadas en JSON a (Redis OSS). ElastiCache Para obtener más información sobre los comandos de Redis OSS compatibles, consulte. [Comandos JSON de Redis OSS compatibles](#)

Las métricas `JsonBasedCmds` relacionadas con `JsonBasedCmdsLatency` JSON se incorporan CloudWatch para supervisar el uso de este tipo de datos. Para obtener más información, consulte [Métricas para Redis OSS](#).

Note

Para usar JSON, debe ejecutar la versión 6.2.6 o posterior del motor OSS de Redis.

Temas

- [Descripción general del tipo de datos JSON de Redis OSS](#)
- [Comandos JSON de Redis OSS compatibles](#)

Descripción general del tipo de datos JSON de Redis OSS

ElastiCache (Redis OSS) admite varios comandos de Redis OSS para trabajar con el tipo de datos JSON. A continuación, se ofrece una descripción general del tipo de datos JSON y una lista detallada de los comandos de Redis OSS compatibles.

Terminología

Plazo	Descripción
Documento JSON	Hace referencia al valor de una clave JSON de Redis OSS.
Valor JSON	Hace referencia a un subconjunto de un JSON, incluida la raíz que representa a todo el

Plazo	Descripción
	documento. Un valor podría ser un contenedor o una entrada dentro de un contenedor.
Elemento JSON	Equivalente al valor JSON.

Estándares JSON admitidos

El formato JSON es compatible con el estándar de intercambio de datos JSON [RFC 7159](#) y [ECMA-404](#). Se admite UTF-8 [Unicode](#) en texto JSON.

Elemento raíz

El elemento raíz puede ser de cualquier tipos de datos de JSON. Tenga en cuenta que en la RFC 4627 anterior, solo se permitían objetos o matrices como valores raíz. Desde la actualización a RFC 7159, la raíz de un documento JSON puede ser de cualquier tipo de datos JSON.

Límite de tamaño del documento

Los documentos JSON se almacenan de manera interna en un formato que se optimiza para lograr un acceso y modificación rápidos. Este formato suele consumir algo más de memoria que la representación serializada equivalente del mismo documento.

El consumo de memoria de un solo documento JSON está limitado a 64 MB, que es el tamaño de la estructura de datos en memoria, no la cadena JSON. Puede comprobar la cantidad de memoria que consume un documento JSON mediante el uso del comando `JSON.DEBUG MEMORY`.

JSON ACL

- Similar a las categorías existentes por tipo de datos (`@string`, `@hash`, etc.), se agrega una nueva categoría `@json` para simplificar la administración del acceso a los comandos y datos JSON. Ningún otro comando OSS de Redis existente es miembro de la categoría `@json`. Todos los comandos JSON aplican cualquier restricción y permiso de espacio de teclas o comandos.
- Existen cinco categorías de ACL de Redis OSS existentes que se han actualizado para incluir los nuevos comandos de JSON: `@read`, `@write`, `@fast`, `@slow` y `@admin`. La siguiente tabla indica la asignación de los comandos JSON a las categorías apropiadas.

ACL

Comando JSON	@read	@write	@fast	@slow	@admin
JSON.ARRAPPEND		y	y		
JSON.ARRINDEX	y		y		
JSON.ARRINSERT		y	y		
JSON.ARRLEN	y		y		
JSON.ARRPOP		y	y		
JSON.ARRTRIM		y	y		
JSON.CLEAR		y	y		
JSON.DEBUG	y			y	y
JSON.DEL		y	y		
JSON.FORGET		y	y		
JSON.GET	y		y		
JSON.MGET	y		y		
JSON.NUMINCRBY		y	y		

Comando JSON	@read	@write	@fast	@slow	@admin
JSON.NUMMULTIBY		y	y		
JSON.OBJECTEYS	y		y		
JSON.OBJECTEN	y		y		
JSON.RESP	y		y		
JSON.SET		y		y	
JSON.STRINGAPPEND		y	y		
JSON.STRINGEN	y		y		
JSON.STRINGEN	y		y		
JSON.TOGGLE		y	y		
JSON.TYPE	y		y		
JSON.NUMMULTIBY		y	y		

Límite de profundidad de anidado

Cuando un objeto o matriz JSON tiene un elemento que es otro objeto o matriz JSON, se dice que ese objeto o matriz interior se “anida” dentro del objeto o matriz exterior. El límite máximo de profundidad de anidamiento es 128. Cualquier intento de crear un documento que contenga una profundidad de anidamiento superior a 128 se rechazará con un error.

Sintaxis de comandos

La mayoría de los comandos requieren un nombre clave de Redis OSS como primer argumento. Algunos comandos también tienen un argumento ruta. El argumento ruta se establece por defecto en la raíz si es opcional y no proporcionado.

Notación:

- Los argumentos obligatorios se incluyen entre corchetes angulares. Por ejemplo: <key>
- Los argumentos opcionales deben ir entre corchetes. Por ejemplo: [path]
- Los argumentos opcionales adicionales se indican mediante puntos suspensivos (“...”). Por ejemplo: [json ...]

Sintaxis de ruta

Redis JSON admite dos tipos de sintaxis de rutas:

- Sintaxis mejorada: sigue la sintaxis JSONPath descrita por [Goessner](#), como se muestra en la siguiente tabla. Hemos reordenado y modificado las descripciones de la tabla para mayor claridad.
- Sintaxis restringida: tiene capacidades de consulta limitadas.

Note

Los resultados de algunos comandos son sensibles al tipo de sintaxis de ruta que se utiliza.

Si una ruta de consulta comienza por '\$', utiliza la sintaxis mejorada. De lo contrario, se utiliza la sintaxis restringida.

Sintaxis mejorada

Símbolo o expresión	Descripción
\$	El elemento raíz.
. o bien []	Operador secundario.
..	Descenso recursivo.

Símbolo o expresión	Descripción
*	Comodín. Todos los elementos de un objeto o matriz.
[]	Operador de subíndice de matriz. El índice se basa en 0.
[,]	Operador de unión.
[start:end:step]	Operador de Slice de la matriz.
?()	Aplica una expresión de filtro (script) a la matriz u objeto actual.
()	Expresión de filtro.
@	Se usa en expresiones de filtro que hacen referencia al nodo actual que se está procesando.
==	Igual a, se utiliza en las expresiones de filtro.
!=	No es igual a, se utiliza en las expresiones de filtro.
>	Mayor que, se utiliza en las expresiones de filtro.
>=	Mayor o igual que, se utiliza en las expresiones de filtro.
<	Menor que, se utiliza en expresiones de filtro.
<=	Menor o igual que, se utiliza en las expresiones de filtro.
&&	Lógico Y, que se utiliza para combinar múltiples expresiones de filtro.

Símbolo o expresión	Descripción
	Lógico O, se utiliza para combinar múltiples expresiones de filtro.

Ejemplos

Los siguientes ejemplos se basan en los datos XML del ejemplo de [Goessner](#), que hemos modificado agregando matrices adicionales.

```
{ "store": {
  "book": [
    { "category": "reference",
      "author": "Nigel Rees",
      "title": "Sayings of the Century",
      "price": 8.95,
      "in-stock": true,
      "sold": true
    },
    { "category": "fiction",
      "author": "Evelyn Waugh",
      "title": "Sword of Honour",
      "price": 12.99,
      "in-stock": false,
      "sold": true
    },
    { "category": "fiction",
      "author": "Herman Melville",
      "title": "Moby Dick",
      "isbn": "0-553-21311-3",
      "price": 8.99,
      "in-stock": true,
      "sold": false
    },
    { "category": "fiction",
      "author": "J. R. R. Tolkien",
      "title": "The Lord of the Rings",
      "isbn": "0-395-19395-8",
      "price": 22.99,
      "in-stock": false,
      "sold": false
    }
  ]
}
```

```

    ],
    "bicycle": {
      "color": "red",
      "price": 19.95,
      "in-stock": true,
      "sold": false
    }
  }
}

```

Ruta	Descripción
<code>\$.store.book[*].author</code>	Los autores de todos los libros de la tienda.
<code>\$.author</code>	Todos los autores.
<code>\$.store.*</code>	Todos los miembros de la tienda.
<code>\$.store.*</code>	Todos los miembros de la tienda.
<code>\$.store..price</code>	El precio de todo lo que hay en la tienda.
<code>\$.*</code>	Todos los miembros recursivos de la estructura JSON.
<code>\$.book[*]</code>	Todos los libros.
<code>\$.book[0]</code>	El primer libro.
<code>\$.book[-1]</code>	El último libro.
<code>\$.book[0:2]</code>	Los dos primeros libros.
<code>\$.book[0,1]</code>	Los dos primeros libros.
<code>\$.book[0:4]</code>	Los libros del índice 0 al 3 (el índice final no está incluido).
<code>\$.book[0:4:2]</code>	Los libros en el índice 0, 2.
<code>\$.book[?(@.isbn)]</code>	Todos los libros con un número de ISBN.

Ruta	Descripción
<code>\$.book[?(@.price<10)]</code>	Todos los libros que cuestan menos de 10 dólares.
<code>'\$.book[?(@.price < 10)]'</code>	Todos los libros que cuestan menos de 10 dólares. (La ruta debe estar entre comillas si contiene espacios en blanco).
<code>'\$.book[?(@["price"] < 10)]'</code>	Todos los libros que cuestan menos de 10 dólares.
<code>'\$.book[?(@["price"] < 10)]'</code>	Todos los libros que cuestan menos de 10 dólares.
<code>\$.book[?(@.price>=10&&@.price<=100)]</code>	Todos los libros en el rango de precios de 10 a 100 dólares, incluidos.
<code>'\$.book[?(@.price>=10 && @.price<=100)]'</code>	Todos los libros en el rango de precios de 10 a 100 dólares, incluidos. (La ruta debe estar entre comillas si contiene espacios en blanco).
<code>\$.book[?(@.sold==true @.in-stock==false)]</code>	Todos los libros vendidos o agotados.
<code>'\$.book[?(@.sold == true @.in-stock == false)]'</code>	Todos los libros vendidos o agotados. (La ruta debe estar entre comillas si contiene espacios en blanco).
<code>'\$.store.book[?(@["category"] == "fiction")]</code>	Todos los libros de la categoría Ficción.
<code>'\$.store.book[?(@["category"] != "fiction")]</code>	Todos los libros de las categorías que no sean ficción.

Ejemplos de expresiones de filtro adicionales:

```
127.0.0.1:6379> JSON.SET k1 . '{"books": [{"price":5,"sold":true,"in-stock":true,"title":"foo"}, {"price":15,"sold":false,"title":"abc"}]}'
OK
127.0.0.1:6379> JSON.GET k1 $.books[?(@.price>1&&@.price<20&&@.in-stock)]
```

```

"[{"price":5,"sold":true,"in-stock":true,"title":"foo"}]"
127.0.0.1:6379> JSON.GET k1 '$.books[?(@.price>1 && @.price<20 && @.in-stock)]'
"[{"price":5,"sold":true,"in-stock":true,"title":"foo"}]"
127.0.0.1:6379> JSON.GET k1 '$.books[?((@.price>1 && @.price<20) && (@.sold==false))]'
"[{"price":15,"sold":false,"title":"abc"}]"
127.0.0.1:6379> JSON.GET k1 '$.books[?(@.title == "abc")]'
[{"price":15,"sold":false,"title":"abc"}]

127.0.0.1:6379> JSON.SET k2 . '[1,2,3,4,5]'
127.0.0.1:6379> JSON.GET k2 $.*.[?(@>2)]
"[3,4,5]"
127.0.0.1:6379> JSON.GET k2 '$.*.[?(@ > 2)]'
"[3,4,5]"

127.0.0.1:6379> JSON.SET k3 . '[true,false,true,false,null,1,2,3,4]'
OK
127.0.0.1:6379> JSON.GET k3 $.*.[?(@==true)]
"[true,true]"
127.0.0.1:6379> JSON.GET k3 '$.*.[?(@ == true)]'
"[true,true]"
127.0.0.1:6379> JSON.GET k3 $.*.[?(@>1)]
"[2,3,4]"
127.0.0.1:6379> JSON.GET k3 '$.*.[?(@ > 1)]'
"[2,3,4]"

```

Sintaxis restringida

Símbolo o expresión	Descripción
. o bien []	Operador secundario.
[]	Operador de subíndice de matriz. El índice se basa en 0.

Ejemplos

Ruta	Descripción
.store.book[0].author	El autor del primer libro.

Ruta	Descripción
.store.book[-1].author	El autor del último libro.
.address.city	Nombre de la ciudad.
["store"]["book"][0]["title"]	El título del primer libro.
["store"]["book"][-1]["title"]	El título del último libro.

Note

Todo el contenido de [Goessner](#) citado en esta documentación está sujeto a la [Licencia de Creative Commons](#).

Prefijos comunes de errores

Cada mensaje de error tiene un prefijo. A continuación se muestra una lista de prefijos comunes de errores.

Prefix	Descripción
ERR	Un error general.
LIMIT	Un error que se produce cuando se excede el límite de tamaño. Por ejemplo, cuando se excede el límite de tamaño del documento o el límite de profundidad de anidación.
INEXISTENTE	Una clave o ruta no existe.
FUERA DE LOS LÍMITES	Un índice de matrices fuera de los límites.
SYNTAXERR	Error de sintaxis.
WRONGTYPE	Tipo de valor incorrecto.

Métricas relacionadas con JSON

Se proporcionan las siguientes métricas de información JSON:

Información	Descripción
json_total_memory_bytes	Memoria total asignada a objetos JSON.
json_num_documents	Número total de documentos en Redis OSS.

Para consultar las métricas principales, ejecute el siguiente comando de Redis OSS:

```
info json_core_metrics
```

Cómo ElastiCache interactúa (Redis OSS) con JSON

En la siguiente sección se describe cómo ElastiCache (Redis OSS) interactúa con el tipo de datos JSON.

Jerarquía de los operadores

Al evaluar las expresiones condicionales para el filtrado, las `&&`s tienen prioridad y, a continuación, se evalúan las `||`s, como es común en la mayoría de los idiomas. Las operaciones entre paréntesis se ejecutan primero.

Comportamiento del límite máximo de anidación

El límite máximo de anidación de rutas en ElastiCache (Redis OSS) es 128. Así que un valor como `$.a.b.c.d...` solo puede alcanzar 128 niveles.

Administración de valores numéricos

JSON no tiene tipos de datos separados para números enteros y de coma flotante. Todos se llaman números.

Representaciones numéricas:

Cuando se recibe un número JSON en la entrada, se convierte en una de las dos representaciones binarias internas: un número entero firmado de 64 bits o un punto flotante de doble precisión IEEE de 64 bit. No se retiene la cadena original ni nada de su formato. Por lo tanto, cuando se genera un número como parte de una respuesta JSON, se convierte de la representación binaria interna a una

cadena imprimible que utiliza reglas de formato genérico. Estas reglas podrían dar como resultado que se genere una cadena diferente de la que se recibió.

Comandos aritméticos NUMINCRBY y NUMMULTBY:

- Si ambos números son números enteros y el resultado está fuera del rango de `int64`, automáticamente se convierte en un número IEEE de punto flotante de doble precisión de 64 bits.
- Si al menos uno de los números es un punto flotante, el resultado es un número IEEE de punto flotante de doble precisión de 64 bits.
- Si el resultado supera el rango de doble IEEE de 64 bits, el comando regresa un error `OVERFLOW`.

Para obtener una lista de los comandos disponibles, consulte el [Comandos JSON de Redis OSS compatibles](#).

Filtrado de matrices directas

ElastiCache (Redis OSS) filtra directamente los objetos de la matriz.

Para datos como `[0, 1, 2, 3, 4, 5, 6]` una consulta de ruta o datos `{"my_key": [0, 1, 2, 3, 4, 5, 6]}` y una consulta de ruta como `$.my_key[?(@<4)]`, ElastiCache (Redis OSS) devolvería `[1,2,3]` en ambas circunstancias. `$.my_key[?(@<4)]`

Comportamiento de indexación de matrices

ElastiCache (Redis OSS) permite índices positivos y negativos para las matrices. Para una matriz de longitud cinco, 0 consultaría el primer elemento, 1 el segundo y, así, sucesivamente. Los números negativos comienzan al final de la matriz, por lo que -1 consultaría el quinto elemento, -2 el cuarto elemento, y así sucesivamente.

Para garantizar un comportamiento predecible para los clientes, ElastiCache (Redis OSS) no redondea los índices de las matrices hacia abajo o hacia arriba, por lo que si tiene una matriz con una longitud de 5, llamar al índice 5 o superior, o a -6 o inferior, no generará ningún resultado.

Evaluación de sintaxis estricta

MemoryDB no permite rutas JSON con sintaxis no válida, incluso si un subconjunto de la ruta contiene una ruta válida. Esto es para mantener un comportamiento correcto para nuestros clientes.

Comandos JSON de Redis OSS compatibles

ElastiCache (Redis OSS) admite los siguientes comandos JSON de Redis OSS:

Temas

- [JSON.ARRAPPEND](#)
- [JSON.ARRINDEX](#)
- [JSON.ARRINSERT](#)
- [JSON.ARRLEN](#)
- [JSON.ARRPOP](#)
- [JSON.ARRTRIM](#)
- [JSON.CLEAR](#)
- [JSON.DEBUG](#)
- [JSON.DEL](#)
- [JSON.FORGET](#)
- [JSON.GET](#)
- [JSON.MGET](#)
- [JSON.NUMINCRBY](#)
- [JSON.NUMMULTBY](#)
- [JSON.OBJLEN](#)
- [JSON.OBJKEYS](#)
- [JSON.RESP](#)
- [JSON.SET](#)
- [JSON.STRAPPEND](#)
- [JSON.STRLEN](#)
- [JSON.TOGGLE](#)
- [JSON.TYPE](#)

JSON.ARRAPPEND

Adjunta uno o más valores a los valores de la matriz en la ruta.

Sintaxis

```
JSON.ARRAPPEND <key> <path> <json> [json ...]
```

- **clave (obligatoria):** una clave OSS de Redis del tipo de documento JSON.
- **ruta (obligatorio):** una ruta JSON.
- **json (obligatorio):** el valor JSON que se agregará a la matriz.

Devolución

Si la ruta es de sintaxis mejorada:

- Matriz de números enteros que representan la nueva longitud de la matriz en cada ruta.
- Si un valor no es una matriz, su valor devuelto correspondiente es nulo.
- Error SYNTAXERR si uno de los argumentos json de entrada no es una cadena JSON válida.
- Error NONEXISTENT si la ruta no existe.

Si la ruta es de sintaxis restringida:

- Entero, la nueva longitud de la matriz.
- Si se seleccionan varios valores de matriz, el comando devuelve la nueva longitud de la última matriz actualizada.
- Error WRONGTYPE si el valor de la ruta no es una matriz.
- Error SYNTAXERR si uno de los argumentos json de entrada no es una cadena JSON válida.
- Error NONEXISTENT si la ruta no existe.

Ejemplos

Sintaxis de la ruta mejorada:

```
127.0.0.1:6379> JSON.SET k1 . '[[[], ["a"], ["a", "b"]]'
OK
127.0.0.1:6379> JSON.ARRAPPEND k1 $[*] '"c"'
1) (integer) 1
2) (integer) 2
3) (integer) 3
127.0.0.1:6379> JSON.GET k1
"[["c"],["a","\c"],["a","\b","\c"]]"
```

Sintaxis de la ruta restringida:

```
127.0.0.1:6379> JSON.SET k1 . '[[[], ["a"], ["a", "b"]]'
OK
127.0.0.1:6379> JSON.ARRAPPEND k1 [-1] '"c"'
(integer) 3
127.0.0.1:6379> JSON.GET k1
"[[[],[\\"a\\"],[\\"a\\","\\"b\\","\\"c\\"]]"
```

JSON.ARRINDEX

Busca la primera aparición de un valor JSON escalar en las matrices de la ruta.

- Los errores fuera de rango se tratan redondeando el índice al principio y al final de la matriz.
- Si inicio > fin, devuelve -1 (no encontrado).

Sintaxis

```
JSON.ARRINDEX <key> <path> <json-scalar> [start [end]]
```

- **clave (obligatoria):** una clave OSS de Redis del tipo de documento JSON.
- **ruta (obligatorio):** una ruta JSON.
- **json-scalar (obligatorio):** el valor escalar para buscar . El escalar JSON se refiere a valores que no son objetos ni matrices. Es decir, cadena, número, Booleano y nulo son valores escalares.
- **inicio (opcional):** el índice de inicio, inclusivo. Toma 0 como valor predeterminado si no se proporciona.
- **final (opcional):** el índice final, exclusivo. Toma 0 como valor predeterminado si no se proporciona, lo que significa que se incluye el último elemento. 0 o -1 significa que se incluye el último elemento.

Devolución

Si la ruta es de sintaxis mejorada:

- **Matriz de números enteros.** Cada valor es el índice del elemento coincidente de la matriz en la ruta. El valor es -1 si no se encuentra.
- Si un valor no es una matriz, su valor devuelto correspondiente es nulo.

Si la ruta es de sintaxis restringida:

- Entero, el índice del elemento coincidente o -1 si no se encuentra.
- Error WRONGTYPE si el valor de la ruta no es una matriz.

Ejemplos

Sintaxis de la ruta mejorada:

```
127.0.0.1:6379> JSON.SET k1 . '[[[], ["a"], ["a", "b"], ["a", "b", "c"]]'
OK
127.0.0.1:6379> JSON.ARRINDEX k1 $[*] '"b"'
1) (integer) -1
2) (integer) -1
3) (integer) 1
4) (integer) 1
```

Sintaxis de la ruta restringida:

```
127.0.0.1:6379> JSON.SET k1 . '{"children": ["John", "Jack", "Tom", "Bob", "Mike"]}'
OK
127.0.0.1:6379> JSON.ARRINDEX k1 .children '"Tom"'
(integer) 2
```

JSON.ARRINSERT

Inserta uno o varios valores en los valores de la matriz en la ruta antes del índice.

Sintaxis

```
JSON.ARRINSERT <key> <path> <index> <json> [json ...]
```

- **clave (obligatoria):** una clave OSS de Redis del tipo de documento JSON.
- **ruta (obligatorio):** una ruta JSON.
- **índice (obligatorio):** un índice de matriz antes del cual se insertan los valores.
- **json (obligatorio):** el valor JSON que se agregará a la matriz.

Devolución

Si la ruta es de sintaxis mejorada:

- Matriz de números enteros que representan la nueva longitud de la matriz en cada ruta.
- Si un valor es una matriz vacía, su valor devuelto correspondiente es nulo.
- Si un valor no es una matriz, su valor devuelto correspondiente es nulo.
- Error OUTFBOUNDARIES si el argumento índice está fuera de los límites.

Si la ruta es de sintaxis restringida:

- Entero, la nueva longitud de la matriz.
- Error WRONGTYPE si el valor de la ruta no es una matriz.
- Error OUTFBOUNDARIES si el argumento índice está fuera de los límites.

Ejemplos

Sintaxis de la ruta mejorada:

```
127.0.0.1:6379> JSON.SET k1 . '[[[], ["a"], ["a", "b"]]'  
OK  
127.0.0.1:6379> JSON.ARRINSERT k1 $[*] 0 '"c"'  
1) (integer) 1  
2) (integer) 2  
3) (integer) 3  
127.0.0.1:6379> JSON.GET k1  
"[[\"c\"],[\"c\", \"a\"],[\"c\", \"a\", \"b\"]]"
```

Sintaxis de la ruta restringida:

```
127.0.0.1:6379> JSON.SET k1 . '[[[], ["a"], ["a", "b"]]'  
OK  
127.0.0.1:6379> JSON.ARRINSERT k1 . 0 '"c"'  
(integer) 4  
127.0.0.1:6379> JSON.GET k1  
"[\\"c\", [], \\"a\"],[\"a\", \"b\"]]"
```

JSON.ARRLEN

Consigue la longitud de los valores de la matriz en la ruta.

Sintaxis

```
JSON.ARRLEN <key> [path]
```

- **clave (obligatoria):** una clave OSS de Redis del tipo de documento JSON.
- **ruta (opcional):** una ruta JSON. Toma el valor predeterminado raíz si no se proporciona.

Devolución

Si la ruta es de sintaxis mejorada:

- Matriz de números enteros que representa la longitud de la matriz en cada ruta.
- Si un valor no es una matriz, su valor devuelto correspondiente es nulo.
- Es nulo si la clave del documento no existe.

Si la ruta es de sintaxis restringida:

- Matriz de cadenas a granel. Cada elemento es un nombre clave del objeto.
- Entero, longitud de matriz.
- Si hay varios objetos seleccionados, el comando devuelve la longitud de la primera matriz.
- Error `WRONGTYPE` si el valor de la ruta no es una matriz.
- Error `WRONGTYPE` si la ruta no existe.
- Es nulo si la clave del documento no existe.

Ejemplos

Sintaxis de la ruta mejorada:

```
127.0.0.1:6379> JSON.SET k1 . '[[[], [\"a\"], [\"a\", \"b\"], [\"a\", \"b\", \"c\"]]]'  
(error) SYNTAXERR Failed to parse JSON string due to syntax error  
127.0.0.1:6379> JSON.SET k1 . '[[[], [\"a\"], [\"a\", \"b\"], [\"a\", \"b\", \"c\"]]]'  
OK
```

```
127.0.0.1:6379> JSON.ARRLEN k1 $[*]
1) (integer) 0
2) (integer) 1
3) (integer) 2
4) (integer) 3

127.0.0.1:6379> JSON.SET k2 . '[[[], "a", ["a", "b"], ["a", "b", "c"], 4]'
OK
127.0.0.1:6379> JSON.ARRLEN k2 $[*]
1) (integer) 0
2) (nil)
3) (integer) 2
4) (integer) 3
5) (nil)
```

Sintaxis de la ruta restringida:

```
127.0.0.1:6379> JSON.SET k1 . '[[[], ["a"], ["a", "b"], ["a", "b", "c"]]'
OK
127.0.0.1:6379> JSON.ARRLEN k1 [*]
(integer) 0
127.0.0.1:6379> JSON.ARRLEN k1 $[3]
1) (integer) 3

127.0.0.1:6379> JSON.SET k2 . '[[[], "a", ["a", "b"], ["a", "b", "c"], 4]'
OK
127.0.0.1:6379> JSON.ARRLEN k2 [*]
(integer) 0
127.0.0.1:6379> JSON.ARRLEN k2 $[1]
1) (nil)
127.0.0.1:6379> JSON.ARRLEN k2 $[2]
1) (integer) 2
```

JSON.ARRPOP

Elimina y devuelve elemento en el índice de la matriz. Al emerger una matriz vacía, se devuelve nulo.

Sintaxis

```
JSON.ARRPOP <key> [path [index]]
```

- **clave (obligatoria):** una clave OSS de Redis del tipo de documento JSON.
- **ruta (opcional):** una ruta JSON. Toma el valor predeterminado raíz si no se proporciona.
- **índice (opcional):** la posición en la matriz desde la que empezar a salir.
 - El valor predeterminado es -1 si no se proporciona, lo que significa el último elemento.
 - Un valor negativo significa la posición desde el último elemento.
 - Los índices fuera de los límites se redondean a sus respectivos límites de matriz.

Devolución

Si la ruta es de sintaxis mejorada:

- Matriz de cadenas masivas que representan valores emergentes en cada ruta.
- Si un valor es una matriz vacía, su valor devuelto correspondiente es nulo.
- Si un valor no es una matriz, su valor devuelto correspondiente es nulo.

Si la ruta es de sintaxis restringida:

- Cadena por lotes, que representa el valor JSON emergente.
- Es nulo si la matriz está vacía.
- Error WRONGTYPE si el valor de la ruta no es una matriz.

Ejemplos

Sintaxis de la ruta mejorada:

```
127.0.0.1:6379> JSON.SET k1 . '[[[], ["a"], ["a", "b"]]'
OK
127.0.0.1:6379> JSON.ARRPOP k1 $[*]
1) (nil)
2) "\"a\""
3) "\"b\""
127.0.0.1:6379> JSON.GET k1
"[[[], [], [\"a\"]]"
```

Sintaxis de la ruta restringida:

```
127.0.0.1:6379> JSON.SET k1 . '[[[], ["a"], ["a", "b"]]'
OK
127.0.0.1:6379> JSON.ARRPOP k1
"[\\"a\\",\\"b\\"]"
127.0.0.1:6379> JSON.GET k1
"[[[],\\"a\\"]"

127.0.0.1:6379> JSON.SET k2 . '[[[], ["a"], ["a", "b"]]'
OK
127.0.0.1:6379> JSON.ARRPOP k2 . 0
"[]"
127.0.0.1:6379> JSON.GET k2
"[[\\"a\\"],[\\"a\\",\\"b\\"]"
```

JSON.ARRTRIM

Recorta una matriz en la ruta para que se convierta en un subcampo [inicio, fin], ambos inclusivos.

- Si la matriz está vacía, no se hace nada y se devuelve 0.
- Si el valor inicio es <0, trátelo como 0.
- Si el tamaño del valor final es >= (tamaño de la matriz), trátelo como tamaño-1.
- Si el tamaño del valor inicio >= o inicio > final, vacíe la matriz y devuelva 0.

Sintaxis

```
JSON.ARRINSERT <key> <path> <start> <end>
```

- **clave (obligatoria):** una clave OSS de Redis del tipo de documento JSON.
- **ruta (obligatorio):** una ruta JSON.
- **inicio (obligatorio):** el índice de inicio, inclusivo.
- **final (obligatorio):** el índice de final, inclusivo.

Devolución

Si la ruta es de sintaxis mejorada:

- Matriz de números enteros que representan la nueva longitud de la matriz en cada ruta.

- Si un valor es una matriz vacía, su valor devuelto correspondiente es nulo.
- Si un valor no es una matriz, su valor devuelto correspondiente es nulo.
- Error `OUTOFBOUNDARIES` si un argumento de índice está fuera de los límites.

Si la ruta es de sintaxis restringida:

- Entero, la nueva longitud de la matriz.
- Es nulo si la matriz está vacía.
- Error `WRONGTYPE` si el valor de la ruta no es una matriz.
- Error `OUTOFBOUNDARIES` si un argumento de índice está fuera de los límites.

Ejemplos

Sintaxis de la ruta mejorada:

```
127.0.0.1:6379> JSON.SET k1 . '[[[], ["a"], ["a", "b"], ["a", "b", "c"]]'
OK
127.0.0.1:6379> JSON.ARRTRIM k1 $[*] 0 1
1) (integer) 0
2) (integer) 1
3) (integer) 2
4) (integer) 2
127.0.0.1:6379> JSON.GET k1
"[[],[\\"a\\"],[\\"a\\","\\"b\\"],[\\"a\\","\\"b\\""]]"
```

Sintaxis de la ruta restringida:

```
127.0.0.1:6379> JSON.SET k1 . '{"children": ["John", "Jack", "Tom", "Bob", "Mike"]}'
OK
127.0.0.1:6379> JSON.ARRTRIM k1 .children 0 1
(integer) 2
127.0.0.1:6379> JSON.GET k1 .children
"[\"John\\","\\"Jack\\"]]"
```

JSON.CLEAR

Borra las matrices o un objeto en la ruta.

Sintaxis

```
JSON.CLEAR <key> [path]
```

- **clave (obligatoria):** una clave OSS de Redis del tipo de documento JSON.
- **ruta (opcional):** una ruta JSON. Toma el valor predeterminado raíz si no se proporciona.

Devolución

- Entero, el número de contenedores borrados.
- La eliminación de una matriz u objeto vacío representa 1 contenedor borrado.
- Al borrar un valor no contenedor, se devuelve 0.

Ejemplos

```
127.0.0.1:6379> JSON.SET k1 . '[[[], [0], [0,1], [0,1,2], 1, true, null, "d"]]'
OK
127.0.0.1:6379> JSON.CLEAR k1 $[*]
(integer) 7
127.0.0.1:6379> JSON.CLEAR k1 $[*]
(integer) 4
127.0.0.1:6379> JSON.SET k2 . '{"children": ["John", "Jack", "Tom", "Bob", "Mike"]}'
OK
127.0.0.1:6379> JSON.CLEAR k2 .children
(integer) 1
127.0.0.1:6379> JSON.GET k2 .children
"[]"
```

JSON.DEBUG

Reporta información. Los subcomandos admitidos son:

- **MEMORIA <key> [path]:** Informa el uso de memoria en bytes de un valor JSON. La ruta se establece de forma predeterminada en la raíz si no se proporciona.
- **CAMPOS <key> [path]:** Informa el número de campos de la ruta del documento especificada. La ruta se establece de forma predeterminada en la raíz si no se proporciona. Cada valor JSON que no es de contenedor cuenta como un campo. Los objetos y las matrices cuentan de forma

recursiva un campo para cada uno de los valores JSON que contienen. Cada valor de contenedor, excepto el contenedor raíz, cuenta como un campo adicional.

- AYUDA: imprime mensajes de ayuda del comando.

Sintaxis

```
JSON.DEBUG <subcommand & arguments>
```

Depende del subcomando:

MEMORIA

- Si la ruta es de sintaxis mejorada:
 - Devuelve una matriz de números enteros que representan tamaño de memoria (en bytes) del valor de JSON en cada ruta.
 - Devuelve una matriz vacía si la clave OSS de Redis no existe.
- Si la ruta es de sintaxis restringida:
 - Devuelve un número entero, tamaño de memoria y el valor JSON en bytes.
 - Devuelve un valor nulo si la clave OSS de Redis no existe.

FIELDS

- Si la ruta es de sintaxis mejorada:
 - Devuelve una matriz de números enteros que representan el número de campos de valor JSON en cada ruta.
 - Devuelve una matriz vacía si la clave OSS de Redis no existe.
- Si la ruta es de sintaxis restringida:
 - Devuelve un número entero, el número de campos del valor JSON.
 - Devuelve un valor nulo si la clave OSS de Redis no existe.

AYUDA: devuelve una serie de mensajes de ayuda.

Ejemplos

Sintaxis de la ruta mejorada:

```

127.0.0.1:6379> JSON.SET k1 . '[1, 2.3, "foo", true, null, {}, [], {"a":1, "b":2},
  [1,2,3]]'
OK
127.0.0.1:6379> JSON.DEBUG MEMORY k1 $[*]
1) (integer) 16
2) (integer) 16
3) (integer) 19
4) (integer) 16
5) (integer) 16
6) (integer) 16
7) (integer) 16
8) (integer) 50
9) (integer) 64
127.0.0.1:6379> JSON.DEBUG FIELDS k1 $[*]
1) (integer) 1
2) (integer) 1
3) (integer) 1
4) (integer) 1
5) (integer) 1
6) (integer) 0
7) (integer) 0
8) (integer) 2
9) (integer) 3

```

Sintaxis de la ruta restringida:

```

127.0.0.1:6379> JSON.SET k1 .
  '{"firstName":"John","lastName":"Smith","age":27,"weight":135.25,"isAlive":true,"address":
{"street":"21 2nd Street","city":"New
York","state":"NY","zipcode":"10021-3100"},"phoneNumbers":
[{"type":"home","number":"212 555-1234"}, {"type":"office","number":"646
555-4567"}],"children":[],"spouse":null}'
OK
127.0.0.1:6379> JSON.DEBUG MEMORY k1
(integer) 632
127.0.0.1:6379> JSON.DEBUG MEMORY k1 .phoneNumbers
(integer) 166

127.0.0.1:6379> JSON.DEBUG FIELDS k1
(integer) 19
127.0.0.1:6379> JSON.DEBUG FIELDS k1 .address
(integer) 4

```

```
127.0.0.1:6379> JSON.DEBUG HELP
1) JSON.DEBUG MEMORY <key> [path] - report memory size (bytes) of the JSON element.
   Path defaults to root if not provided.
2) JSON.DEBUG FIELDS <key> [path] - report number of fields in the JSON element. Path
   defaults to root if not provided.
3) JSON.DEBUG HELP - print help message.
```

JSON.DEL

Borra los valores JSON de la ruta de acceso de una clave de documento. Si la ruta es la raíz, equivale a eliminar la clave de Redis OSS.

Sintaxis

```
JSON.DEL <key> [path]
```

- clave (obligatoria): una clave de Redis OSS del tipo de documento JSON.
- ruta (opcional): una ruta JSON. Toma el valor predeterminado raíz si no se proporciona.

Devolución

- Número de elementos eliminados.
- 0 si la clave OSS de Redis no existe.
- 0 si la ruta JSON no es válida o no existe.

Ejemplos

Sintaxis de la ruta mejorada:

```
127.0.0.1:6379> JSON.SET k1 . '{"a":{ }, "b":{"a":1}, "c":{"a":1, "b":2}, "d":{"a":1,
"b":2, "c":3}, "e": [1,2,3,4,5]}'
OK
127.0.0.1:6379> JSON.DEL k1 $.d.*
(integer) 3
127.0.0.1:6379> JSOn.GET k1
"{\"a\":{ },\"b\":{\"a\":1},\"c\":{\"a\":1,\"b\":2},\"d\":{\"a\":1,\"b\":2},\"e\":[1,2,3,4,5]}"
127.0.0.1:6379> JSON.DEL k1 $.e[*]
```

```
(integer) 5
127.0.0.1:6379> JSON.GET k1
"{\"a\":{},\"b\":{\"a\":1},\"c\":{\"a\":1,\"b\":2},\"d\":{},\"e\":[]}"
```

Sintaxis de la ruta restringida:

```
127.0.0.1:6379> JSON.SET k1 . '{"a":{}, "b":{"a":1}, "c":{"a":1, "b":2}, "d":{"a":1,
"b":2, "c":3}, "e": [1,2,3,4,5]}'
OK
127.0.0.1:6379> JSON.DEL k1 .d.*
(integer) 3
127.0.0.1:6379> JSON.GET k1
"{\"a\":{},\"b\":{\"a\":1},\"c\":{\"a\":1,\"b\":2},\"d\":{},\"e\":[1,2,3,4,5]}"
127.0.0.1:6379> JSON.DEL k1 .e[*]
(integer) 5
127.0.0.1:6379> JSON.GET k1
"{\"a\":{},\"b\":{\"a\":1},\"c\":{\"a\":1,\"b\":2},\"d\":{},\"e\":[]}"
```

JSON.FORGET

Un alias de [JSON.DEL](#).

JSON.GET

Devuelve el formato JSON serializado en una o varias rutas.

Sintaxis

```
JSON.GET <key>
[INDENT indentation-string]
[NEWLINE newline-string]
[SPACE space-string]
[NOESCAPE]
[path ...]
```

- **clave (obligatoria):** una clave OSS de Redis del tipo de documento JSON.
- **SANGRÍA/NUEVA LÍNEA/ESPACIO (opcional):** controla el formato de la cadena de JSON devuelta, es decir, “impresión bonita”. El valor predeterminado de cada una es una cadena vacía. Se puede anular en cualquier combinación. Estos se pueden especificar en cualquier orden.

- SIN ESCAPE: opcional, puede estar presente para la compatibilidad con versiones anteriores y no tiene ningún otro efecto.
- ruta (opcional): cero o más rutas JSON, el valor predeterminado es la raíz si no se proporciona ninguna. Los argumentos de la ruta deben colocarse al final.

Devolución

Sintaxis de la ruta mejorada:

Si se da una ruta:

- Devuelve una cadena serializada de una matriz de valores.
- Si no selecciona ningún valor, el comando devuelve una matriz vacía.

Si se proporcionan varias rutas:

- Devuelve un objeto JSON con cadenas, en el que cada ruta es una clave.
- Si hay una sintaxis de ruta restringida y mejorada mixta, el resultado se ajusta a la sintaxis mejorada.
- Si no existe una ruta, su valor correspondiente es una matriz vacía.

Ejemplos

Sintaxis de la ruta mejorada:

```
127.0.0.1:6379> JSON.SET k1 .
'{"firstName":"John","lastName":"Smith","age":27,"weight":135.25,"isAlive":true,"address":
{"street":"21 2nd Street","city":"New
York","state":"NY","zipcode":"10021-3100"},"phoneNumbers":
[{"type":"home","number":"212 555-1234"}, {"type":"office","number":"646
555-4567"}],"children":[],"spouse":null}'
OK
127.0.0.1:6379> JSON.GET k1 $.address.*
["21 2nd Street","New York","NY","10021-3100"]
127.0.0.1:6379> JSON.GET k1 indent "\t" space " " NEWLINE "\n" $.address.*
["\n\t"21 2nd Street","\n\t"New York","\n\t"NY","\n\t"10021-3100"\n"]
127.0.0.1:6379> JSON.GET k1 $.firstName $.lastName $.age
{"$.firstName":["John"],("$.lastName":["Smith"],("$.age":["27"])
127.0.0.1:6379> JSON.SET k2 . '{"a":{},"b":{"a":1},"c":{"a":1,"b":2}}'
```

```
OK
127.0.0.1:6379> json.get k2 $.*
"[{} ,{\\"a\\":1},{\\"a\\":1,\\\"b\\":2},1,1,2]"
```

Sintaxis de la ruta restringida:

```
127.0.0.1:6379> JSON.SET k1 .
'{"firstName":"John","lastName":"Smith","age":27,"weight":135.25,"isAlive":true,"address":
{"street":"21 2nd Street","city":"New
York","state":"NY","zipcode":"10021-3100"},"phoneNumbers":
[{"type":"home","number":"212 555-1234"}, {"type":"office","number":"646
555-4567"}],"children":[],"spouse":null}'
OK
127.0.0.1:6379> JSON.GET k1 .address
"{\\"street\\":\\"21 2nd Street\\",\\"city\\":\\"New York\\",\\"state\\":\\"NY\\",\\"zipcode\\":
\\"10021-3100\\"}"
127.0.0.1:6379> JSON.GET k1 indent "\t" space " " NEWLINE "\n" .address
"{\n\t\\"street\\": \\"21 2nd Street\\",\n\t\\"city\\": \\"New York\\",\n\t\\"state\\": \\"NY\\",\n
\t\\"zipcode\\": \\"10021-3100\\"}\n}"
127.0.0.1:6379> JSON.GET k1 .firstName .lastName .age
"{\\".firstName\\":\\"John\\",\\".lastName\\":\\"Smith\\",\\".age\\":27}"
```

JSON.MGET

Consigue los comandos JSON serializados en la ruta de varias claves de documentos. Devuelve un valor nulo para una clave o ruta JSON no existente.

Sintaxis

```
JSON.MGET <key> [key ...] <path>
```

- clave (obligatoria): una o más claves OSS de Redis de tipo documento.
- ruta (obligatorio): una ruta JSON.

Devolución

- Matriz de cadenas a granel. El tamaño de la matriz es igual al número de teclas del comando. Cada elemento de la matriz se rellena con (a) el comando JSON serializado tal como se encuentra

en la ruta o (b) nulo si la clave no existe, la ruta no existe en el documento, o la ruta no es válida (error de sintaxis).

- Si alguna de las claves especificadas existe y no es una clave JSON, el comando devuelve el error WRONGTYPE.

Ejemplos

Sintaxis de la ruta mejorada:

```
127.0.0.1:6379> JSON.SET k1 . '{"address":{"street":"21 2nd Street","city":"New
York","state":"NY","zipcode":"10021"}}'
OK
127.0.0.1:6379> JSON.SET k2 . '{"address":{"street":"5 main
Street","city":"Boston","state":"MA","zipcode":"02101"}}'
OK
127.0.0.1:6379> JSON.SET k3 . '{"address":{"street":"100 Park
Ave","city":"Seattle","state":"WA","zipcode":"98102"}}'
OK
127.0.0.1:6379> JSON.MGET k1 k2 k3 $.address.city
1) "[\ "New York\"]"
2) "[\ "Boston\"]"
3) "[\ "Seattle\"]"
```

Sintaxis de la ruta restringida:

```
127.0.0.1:6379> JSON.SET k1 . '{"address":{"street":"21 2nd Street","city":"New
York","state":"NY","zipcode":"10021"}}'
OK
127.0.0.1:6379> JSON.SET k2 . '{"address":{"street":"5 main
Street","city":"Boston","state":"MA","zipcode":"02101"}}'
OK
127.0.0.1:6379> JSON.SET k3 . '{"address":{"street":"100 Park
Ave","city":"Seattle","state":"WA","zipcode":"98102"}}'
OK

127.0.0.1:6379> JSON.MGET k1 k2 k3 .address.city
1) "\"New York\""
2) "\"Seattle\""
3) "\"Seattle\""
```

JSON.NUMINCRBY

Aumenta los valores numéricos de la ruta por un número dado.

Sintaxis

```
JSON.NUMINCRBY <key> <path> <number>
```

- clave (obligatoria): una clave OSS de Redis del tipo de documento JSON.
- ruta (obligatorio): una ruta JSON.
- número (obligatorio): un número.

Devolución

Si la ruta es de sintaxis mejorada:

- Matriz de cadena a granel que representa el valor resultante en cada ruta.
- Si un valor no es un número, su valor devuelto correspondiente es nulo.
- El error `WRONGTYPE` si el número no se puede analizar.
- El error `OVERFLOW` si el resultado está fuera del rango del doble IEEE de 64 bits.
- `NONEXISTENT` si la clave del documento no existe.

Si la ruta es de sintaxis restringida:

- Cadena a granel que representa el valor resultante.
- Si se seleccionan varios valores, el comando devuelve el resultado del último valor actualizado.
- El error `WRONGTYPE` si el valor de la ruta no es un número.
- El error `WRONGTYPE` si el número no se puede analizar.
- El error `OVERFLOW` si el resultado está fuera del rango del doble IEEE de 64 bits.
- `NONEXISTENT` si la clave del documento no existe.

Ejemplos

Sintaxis de la ruta mejorada:

```
127.0.0.1:6379> JSON.SET k1 . '{"a":[], "b":[1], "c":[1,2], "d":[1,2,3]}'
```

```

OK
127.0.0.1:6379> JSON.NUMINCRBY k1 $.d[*] 10
"[11,12,13]"
127.0.0.1:6379> JSON.GET k1
"{\"a\":[],\"b\":[1],\"c\":[1,2],\"d\":[11,12,13]}"

127.0.0.1:6379> JSON.SET k1 $ '{"a":[], "b":[1], "c":[1,2], "d":[1,2,3]}'
OK
127.0.0.1:6379> JSON.NUMINCRBY k1 $.a[*] 1
"[]"
127.0.0.1:6379> JSON.NUMINCRBY k1 $.b[*] 1
"[2]"
127.0.0.1:6379> JSON.NUMINCRBY k1 $.c[*] 1
"[2,3]"
127.0.0.1:6379> JSON.NUMINCRBY k1 $.d[*] 1
"[2,3,4]"
127.0.0.1:6379> JSON.GET k1
"{\"a\":[],\"b\":[2],\"c\":[2,3],\"d\":[2,3,4]}"

127.0.0.1:6379> JSON.SET k2 $ '{"a":{}, "b":{"a":1}, "c":{"a":1, "b":2}, "d":{"a":1, "b":2, "c":3}}'
OK
127.0.0.1:6379> JSON.NUMINCRBY k2 $.a.* 1
"[]"
127.0.0.1:6379> JSON.NUMINCRBY k2 $.b.* 1
"[2]"
127.0.0.1:6379> JSON.NUMINCRBY k2 $.c.* 1
"[2,3]"
127.0.0.1:6379> JSON.NUMINCRBY k2 $.d.* 1
"[2,3,4]"
127.0.0.1:6379> JSON.GET k2
"{\"a\":[],\"b\":{\"a\":2},\"c\":{\"a\":2,\"b\":3},\"d\":{\"a\":2,\"b\":3,\"c\":4}}"

127.0.0.1:6379> JSON.SET k3 $ '{"a":{"a":"a"}, "b":{"a":"a", "b":1}, "c":{"a":"a", "b":"b"}, "d":{"a":1, "b":"b", "c":3}}'
OK
127.0.0.1:6379> JSON.NUMINCRBY k3 $.a.* 1
"[null]"
127.0.0.1:6379> JSON.NUMINCRBY k3 $.b.* 1
"[null,2]"
127.0.0.1:6379> JSON.NUMINCRBY k3 $.c.* 1
"[null,null]"
127.0.0.1:6379> JSON.NUMINCRBY k3 $.d.* 1
"[2,null,4]"

```

```
127.0.0.1:6379> JSON.GET k3
"{\"a\":{\"a\":\"a\"},\"b\":{\"a\":\"a\",\"b\":2},\"c\":{\"a\":\"a\",\"b\":\"b\"},\"d\":{\"a\":2,\"b\":\"b\",\"c\":4}}"
```

Sintaxis de la ruta restringida:

```
127.0.0.1:6379> JSON.SET k1 . '{"a":[], "b":[1], "c":[1,2], "d":[1,2,3]}'
OK
127.0.0.1:6379> JSON.NUMINCRBY k1 .d[1] 10
"12"
127.0.0.1:6379> JSON.GET k1
"{\"a\":[],\"b\":[1],\"c\":[1,2],\"d\":[1,12,3]}"

127.0.0.1:6379> JSON.SET k1 . '{"a":[], "b":[1], "c":[1,2], "d":[1,2,3]}'
OK
127.0.0.1:6379> JSON.NUMINCRBY k1 .a[*] 1
(error) NONEXISTENT JSON path does not exist
127.0.0.1:6379> JSON.NUMINCRBY k1 .b[*] 1
"2"
127.0.0.1:6379> JSON.GET k1
"{\"a\":[],\"b\":[2],\"c\":[1,2],\"d\":[1,2,3]}"
127.0.0.1:6379> JSON.NUMINCRBY k1 .c[*] 1
"3"
127.0.0.1:6379> JSON.GET k1
"{\"a\":[],\"b\":[2],\"c\":[2,3],\"d\":[1,2,3]}"
127.0.0.1:6379> JSON.NUMINCRBY k1 .d[*] 1
"4"
127.0.0.1:6379> JSON.GET k1
"{\"a\":[],\"b\":[2],\"c\":[2,3],\"d\":[2,3,4]}"

127.0.0.1:6379> JSON.SET k2 . '{"a:{}, "b":{"a":1}, "c":{"a":1, "b":2}, "d":{"a":1, "b":2, "c":3}}'
OK
127.0.0.1:6379> JSON.NUMINCRBY k2 .a.* 1
(error) NONEXISTENT JSON path does not exist
127.0.0.1:6379> JSON.NUMINCRBY k2 .b.* 1
"2"
127.0.0.1:6379> JSON.GET k2
"{\"a\":{\"a\":2},\"b\":{\"a\":1,\"b\":2},\"c\":{\"a\":1,\"b\":2,\"c\":3}}"
```

```

"{\"a\":{},\"b\":{\"a\":2},\"c\":{\"a\":2,\"b\":3},\"d\":{\"a\":1,\"b\":2,\"c\":3}}"
127.0.0.1:6379> JSON.NUMINCRBY k2 .d.* 1
"4"
127.0.0.1:6379> JSON.GET k2
"{\"a\":{\"a\":2},\"b\":{\"a\":2,\"b\":3},\"d\":{\"a\":2,\"b\":3,\"c\":4}}"

127.0.0.1:6379> JSON.SET k3 . '{"a":{"a":"a"}, "b":{"a":"a", "b":1}, "c":{"a":"a",
  "b":"b"}, "d":{"a":1, "b":"b", "c":3}}'
OK
127.0.0.1:6379> JSON.NUMINCRBY k3 .a.* 1
(error) WRONGTYPE JSON element is not a number
127.0.0.1:6379> JSON.NUMINCRBY k3 .b.* 1
"2"
127.0.0.1:6379> JSON.NUMINCRBY k3 .c.* 1
(error) WRONGTYPE JSON element is not a number
127.0.0.1:6379> JSON.NUMINCRBY k3 .d.* 1
"4"

```

JSON.NUMMULTBY

Multipla los valores numéricos de la ruta por un número dado.

Sintaxis

```
JSON.NUMMULTBY <key> <path> <number>
```

- clave (obligatoria): una clave OSS de Redis del tipo de documento JSON.
- ruta (obligatorio): una ruta JSON.
- número (obligatorio): un número.

Devolución

Si la ruta es de sintaxis mejorada:

- Matriz de cadenas a granel que representa el valor resultante en cada ruta.
- Si un valor no es un número, su valor devuelto correspondiente es nulo.
- El error `WRONGTYPE` si el número no se puede analizar.
- El error `OVERFLOW` si el resultado está fuera del rango de un número de punto flotante de doble precisión de 64 bits IEEE.

- NONEXISTENT si la clave del documento no existe.

Si la ruta es de sintaxis restringida:

- Cadena a granel que representa el valor resultante.
- Si se seleccionan varios valores, el comando devuelve el resultado del último valor actualizado.
- El error WRONGTYPE si el valor de la ruta no es un número.
- El error WRONGTYPE si el número no se puede analizar.
- El error OVERFLOW si el resultado está fuera del rango del doble IEEE de 64 bits.
- NONEXISTENT si la clave del documento no existe.

Ejemplos

Sintaxis de la ruta mejorada:

```
127.0.0.1:6379> JSON.SET k1 . '{"a":[], "b":[1], "c":[1,2], "d":[1,2,3]}'
OK
127.0.0.1:6379> JSON.NUMMULTBY k1 $.d[*] 2
"[2,4,6]"
127.0.0.1:6379> JSON.GET k1
"{\"a\":[],\"b\":[1],\"c\":[1,2],\"d\":[2,4,6]}"

127.0.0.1:6379> JSON.SET k1 $ '{"a":[], "b":[1], "c":[1,2], "d":[1,2,3]}'
OK
127.0.0.1:6379> JSON.NUMMULTBY k1 $.a[*] 2
"[]"
127.0.0.1:6379> JSON.NUMMULTBY k1 $.b[*] 2
"[2]"
127.0.0.1:6379> JSON.NUMMULTBY k1 $.c[*] 2
"[2,4]"
127.0.0.1:6379> JSON.NUMMULTBY k1 $.d[*] 2
"[2,4,6]"

127.0.0.1:6379> JSON.SET k2 $ '{"a":{}, "b":{"a":1}, "c":{"a":1, "b":2}, "d":{"a":1, "b":2, "c":3}}'
OK
127.0.0.1:6379> JSON.NUMMULTBY k2 $.a.* 2
"[]"
127.0.0.1:6379> JSON.NUMMULTBY k2 $.b.* 2
"[2]"
```

```

127.0.0.1:6379> JSON.NUMMULTBY k2 $.c.* 2
"[2,4]"
127.0.0.1:6379> JSON.NUMMULTBY k2 $.d.* 2
"[2,4,6]"

127.0.0.1:6379> JSON.SET k3 $ '{"a":{"a":"a"}, "b":{"a":"a", "b":1}, "c":{"a":"a",
  "b":"b"}, "d":{"a":1, "b":"b", "c":3}}'
OK
127.0.0.1:6379> JSON.NUMMULTBY k3 $.a.* 2
"[null]"
127.0.0.1:6379> JSON.NUMMULTBY k3 $.b.* 2
"[null,2]"
127.0.0.1:6379> JSON.NUMMULTBY k3 $.c.* 2
"[null,null]"
127.0.0.1:6379> JSON.NUMMULTBY k3 $.d.* 2
"[2,null,6]"

```

Sintaxis de la ruta restringida:

```

127.0.0.1:6379> JSON.SET k1 . '{"a":[], "b":[1], "c":[1,2], "d":[1,2,3]}'
OK
127.0.0.1:6379> JSON.NUMMULTBY k1 .d[1] 2
"4"
127.0.0.1:6379> JSON.GET k1
"{\"a\":[],\"b\":[1],\"c\":[1,2],\"d\":[1,4,3]}"

127.0.0.1:6379> JSON.SET k1 . '{"a":[], "b":[1], "c":[1,2], "d":[1,2,3]}'
OK
127.0.0.1:6379> JSON.NUMMULTBY k1 .a[*] 2
(error) NONEXISTENT JSON path does not exist
127.0.0.1:6379> JSON.NUMMULTBY k1 .b[*] 2
"2"
127.0.0.1:6379> JSON.GET k1
"{\"a\":[],\"b\":[2],\"c\":[1,2],\"d\":[1,2,3]}"
127.0.0.1:6379> JSON.NUMMULTBY k1 .c[*] 2
"4"
127.0.0.1:6379> JSON.GET k1
"{\"a\":[],\"b\":[2],\"c\":[2,4],\"d\":[1,2,3]}"
127.0.0.1:6379> JSON.NUMMULTBY k1 .d[*] 2
"6"
127.0.0.1:6379> JSON.GET k1
"{\"a\":[],\"b\":[2],\"c\":[2,4],\"d\":[2,4,6]}"

```

```

127.0.0.1:6379> JSON.SET k2 . '{"a":{}, "b":{"a":1}, "c":{"a":1, "b":2}, "d":{"a":1,
"b":2, "c":3}}'
OK
127.0.0.1:6379> JSON.NUMMULTBY k2 .a.* 2
(error) NONEXISTENT JSON path does not exist
127.0.0.1:6379> JSON.NUMMULTBY k2 .b.* 2
"2"
127.0.0.1:6379> JSON.GET k2
"{\"a\":{},\"b\":{\"a\":2},\"c\":{\"a\":1,\"b\":2},\"d\":{\"a\":1,\"b\":2,\"c\":3}}"
127.0.0.1:6379> JSON.NUMMULTBY k2 .c.* 2
"4"
127.0.0.1:6379> JSON.GET k2
"{\"a\":{},\"b\":{\"a\":2},\"c\":{\"a\":2,\"b\":4},\"d\":{\"a\":1,\"b\":2,\"c\":3}}"
127.0.0.1:6379> JSON.NUMMULTBY k2 .d.* 2
"6"
127.0.0.1:6379> JSON.GET k2
"{\"a\":{},\"b\":{\"a\":2},\"c\":{\"a\":2,\"b\":4},\"d\":{\"a\":2,\"b\":4,\"c\":6}}"

127.0.0.1:6379> JSON.SET k3 . '{"a":{"a":"a"}, "b":{"a":"a", "b":1}, "c":{"a":"a",
"b":"b"}, "d":{"a":1, "b":"b", "c":3}}'
OK
127.0.0.1:6379> JSON.NUMMULTBY k3 .a.* 2
(error) WRONGTYPE JSON element is not a number
127.0.0.1:6379> JSON.NUMMULTBY k3 .b.* 2
"2"
127.0.0.1:6379> JSON.GET k3
"{\"a\":{\"a\":\"a\"},\"b\":{\"a\":\"a\", \"b\":2},\"c\":{\"a\":\"a\", \"b\":\"b\"},\"d
\":{\"a\":1,\"b\":\"b\", \"c\":3}}"
127.0.0.1:6379> JSON.NUMMULTBY k3 .c.* 2
(error) WRONGTYPE JSON element is not a number
127.0.0.1:6379> JSON.NUMMULTBY k3 .d.* 2
"6"
127.0.0.1:6379> JSON.GET k3
"{\"a\":{\"a\":\"a\"},\"b\":{\"a\":\"a\", \"b\":2},\"c\":{\"a\":\"a\", \"b\":\"b\"},\"d
\":{\"a\":2,\"b\":\"b\", \"c\":6}}"

```

JSON.OBJLEN

Consigue el número de claves en los valores del objeto en la ruta.

Sintaxis

```
JSON.OBJLEN <key> [path]
```

- clave (obligatoria): una clave OSS de Redis del tipo de documento JSON.
- ruta (opcional): una ruta JSON. Toma el valor predeterminado raíz si no se proporciona.

Devolución

Si la ruta es de sintaxis mejorada:

- Matriz de números enteros que representa la longitud del objeto en cada ruta.
- Si un valor no es un objeto, su valor devuelto correspondiente es nulo.
- Es nulo si la clave del documento no existe.

Si la ruta es de sintaxis restringida:

- Entero, número de claves del objeto.
- Si hay varios objetos seleccionados, el comando devuelve la longitud del primer objeto.
- El error `WRONGTYPE` si el valor de la ruta no es un objeto.
- Error `WRONGTYPE` si la ruta no existe.
- Es nulo si la clave del documento no existe.

Ejemplos

Sintaxis de la ruta mejorada:

```
127.0.0.1:6379> JSON.SET k1 $ '{"a":{}, "b":{"a":"a"}, "c":{"a":"a", "b":"bb"}, "d":
{"a":1, "b":"b", "c":{"a":3, "b":4}}, "e":1}'
OK
127.0.0.1:6379> JSON.OBJLEN k1 $.a
1) (integer) 0
127.0.0.1:6379> JSON.OBJLEN k1 $.a.*
(empty array)
127.0.0.1:6379> JSON.OBJLEN k1 $.b
1) (integer) 1
127.0.0.1:6379> JSON.OBJLEN k1 $.b.*
1) (nil)
```

```

127.0.0.1:6379> JSON.OBJLEN k1 $.c
1) (integer) 2
127.0.0.1:6379> JSON.OBJLEN k1 $.c.*
1) (nil)
2) (nil)
127.0.0.1:6379> JSON.OBJLEN k1 $.d
1) (integer) 3
127.0.0.1:6379> JSON.OBJLEN k1 $.d.*
1) (nil)
2) (nil)
3) (integer) 2
127.0.0.1:6379> JSON.OBJLEN k1 $.*
1) (integer) 0
2) (integer) 1
3) (integer) 2
4) (integer) 3
5) (nil)

```

Sintaxis de la ruta restringida:

```

127.0.0.1:6379> JSON.SET k1 . '{"a":{}, "b":{"a":"a"}, "c":{"a":"a", "b":"bb"}, "d":
{"a":1, "b":"b", "c":{"a":3,"b":4}}, "e":1}'
OK
127.0.0.1:6379> JSON.OBJLEN k1 .a
(integer) 0
127.0.0.1:6379> JSON.OBJLEN k1 .a.*
(error) NONEXISTENT JSON path does not exist
127.0.0.1:6379> JSON.OBJLEN k1 .b
(integer) 1
127.0.0.1:6379> JSON.OBJLEN k1 .b.*
(error) WRONGTYPE JSON element is not an object
127.0.0.1:6379> JSON.OBJLEN k1 .c
(integer) 2
127.0.0.1:6379> JSON.OBJLEN k1 .c.*
(error) WRONGTYPE JSON element is not an object
127.0.0.1:6379> JSON.OBJLEN k1 .d
(integer) 3
127.0.0.1:6379> JSON.OBJLEN k1 .d.*
(integer) 2
127.0.0.1:6379> JSON.OBJLEN k1 .*
(integer) 0

```

JSON.OBJKEYS

Consigue nombres clave en los valores de objeto de la ruta.

Sintaxis

```
JSON.OBJKEYS <key> [path]
```

- clave (obligatoria): una clave OSS de Redis del tipo de documento JSON.
- ruta (opcional): una ruta JSON. Toma el valor predeterminado raíz si no se proporciona.

Devolución

Si la ruta es de sintaxis mejorada:

- Matriz de matriz de cadenas masivas. Cada elemento es una matriz de claves de un objeto coincidente.
- Si un valor no es un objeto, su valor devuelto correspondiente es un valor vacío.
- Es nulo si la clave del documento no existe.

Si la ruta es de sintaxis restringida:

- Matriz de cadenas a granel. Cada elemento es un nombre clave del objeto.
- Si hay varios objetos seleccionados, el comando devuelve las claves del primer objeto.
- El error `WRONGTYPE` si el valor de la ruta no es un objeto.
- Error `WRONGTYPE` si la ruta no existe.
- Es nulo si la clave del documento no existe.

Ejemplos

Sintaxis de la ruta mejorada:

```
127.0.0.1:6379> JSON.SET k1 $ '{"a":{}, "b":{"a":"a"}, "c":{"a":"a", "b":"bb"}, "d":
{"a":1, "b":"b", "c":{"a":3, "b":4}}, "e":1}'
OK
127.0.0.1:6379> JSON.OBJKEYS k1 $.*
1) (empty array)
```

```

2) 1) "a"
3) 1) "a"
   2) "b"
4) 1) "a"
   2) "b"
   3) "c"
5) (empty array)
127.0.0.1:6379> JSON.OBJKEYS k1 $.d
1) 1) "a"
   2) "b"
   3) "c"

```

Sintaxis de la ruta restringida:

```

127.0.0.1:6379> JSON.SET k1 $ '{"a":{}, "b":{"a":"a"}, "c":{"a":"a", "b":"bb"}, "d":
{"a":1, "b":"b", "c":{"a":3,"b":4}}, "e":1}'
OK
127.0.0.1:6379> JSON.OBJKEYS k1 .*
1) "a"
127.0.0.1:6379> JSON.OBJKEYS k1 .d
1) "a"
2) "b"
3) "c"

```

JSON.RESP

Devuelve el valor JSON en la ruta dada en el Protocolo de serialización OSS de Redis (RESP). Si el valor es contenedor, la respuesta es una matriz RESP o matriz anidada.

- El valor nulo de JSON se asigna a la cadena masiva nula de RESP.
- Los valores booleanos JSON se asignan a las cadenas simples de RESP respectivas.
- Los números enteros se asignan a números enteros RESP.
- Los números de coma flotante doble IEEE de 64 bits se asignan a cadenas masivas RESP.
- Las cadenas JSON se asignan a RESP Bulk Strings.
- Las matrices JSON se representan como matrices RESP, donde el primer elemento es la cadena simple [, seguida de los elementos de la matriz.
- Los objetos JSON se representan como matrices RESP, donde el primer elemento es la cadena simple {, seguida de los pares clave-valor, cada uno de los cuales es una cadena masiva RESP.

Sintaxis

```
JSON.RESP <key> [path]
```

- clave (obligatoria): una clave OSS de Redis del tipo de documento JSON.
- ruta (opcional): una ruta JSON. Toma el valor predeterminado raíz si no se proporciona.

Devolución

Si la ruta es de sintaxis mejorada:

- Matriz de matrices. Cada elemento de la matriz representa la forma RESP del valor en una ruta.
- Matriz vacía si la clave del documento no existe.

Si la ruta es de sintaxis restringida:

- Matriz que representa la forma RESP del valor en la ruta.
- Es nulo si la clave del documento no existe.

Ejemplos

Sintaxis de la ruta mejorada:

```
127.0.0.1:6379> JSON.SET k1 .
'{"firstName":"John","lastName":"Smith","age":27,"weight":135.25,"isAlive":true,"address":
{"street":"21 2nd Street","city":"New
York","state":"NY","zipcode":"10021-3100"},"phoneNumbers":
[{"type":"home","number":"212 555-1234"}, {"type":"office","number":"646
555-4567"}],"children":[],"spouse":null}'
OK
```

```
127.0.0.1:6379> JSON.RESP k1 $.address
```

```
1) 1) {
  2) 1) "street"
     2) "21 2nd Street"
  3) 1) "city"
     2) "New York"
  4) 1) "state"
     2) "NY"
```

```

5) 1) "zipcode"
    2) "10021-3100"

127.0.0.1:6379> JSON.RESP k1 $.address.*
1) "21 2nd Street"
2) "New York"
3) "NY"
4) "10021-3100"

127.0.0.1:6379> JSON.RESP k1 $.phoneNumbers
1) 1) [
    2) 1) {
        2) 1) "type"
           2) "home"
        3) 1) "number"
           2) "555 555-1234"
    3) 1) {
        2) 1) "type"
           2) "office"
        3) 1) "number"
           2) "555 555-4567"

127.0.0.1:6379> JSON.RESP k1 $.phoneNumbers[*]
1) 1) {
    2) 1) "type"
       2) "home"
    3) 1) "number"
       2) "212 555-1234"
2) 1) {
    2) 1) "type"
       2) "office"
    3) 1) "number"
       2) "555 555-4567"

```

Sintaxis de la ruta restringida:

```

127.0.0.1:6379> JSON.SET k1 .
'{"firstName":"John","lastName":"Smith","age":27,"weight":135.25,"isAlive":true,"address":
{"street":"21 2nd Street","city":"New
York","state":"NY","zipcode":"10021-3100"},"phoneNumbers":
[{"type":"home","number":"212 555-1234"}, {"type":"office","number":"646
555-4567"}],"children":[],"spouse":null}'

```

OK

```
127.0.0.1:6379> JSON.RESP k1 .address
```

```
1) {  
2) 1) "street"  
   2) "21 2nd Street"  
3) 1) "city"  
   2) "New York"  
4) 1) "state"  
   2) "NY"  
5) 1) "zipcode"  
   2) "10021-3100"
```

```
127.0.0.1:6379> JSON.RESP k1
```

```
1) {  
2) 1) "firstName"  
   2) "John"  
3) 1) "lastName"  
   2) "Smith"  
4) 1) "age"  
   2) (integer) 27  
5) 1) "weight"  
   2) "135.25"  
6) 1) "isAlive"  
   2) true  
7) 1) "address"  
   2) 1) {  
      2) 1) "street"  
         2) "21 2nd Street"  
      3) 1) "city"  
         2) "New York"  
      4) 1) "state"  
         2) "NY"  
      5) 1) "zipcode"  
         2) "10021-3100"  
8) 1) "phoneNumbers"  
   2) 1) [  
      2) 1) {  
         2) 1) "type"  
            2) "home"  
         3) 1) "number"  
            2) "212 555-1234"  
      3) 1) {  
         2) 1) "type"
```

```
    2) "office"
    3) 1) "number"
      2) "555 555-4567"
9) 1) "children"
   2) 1) [
10) 1) "spouse"
     2) (nil)
```

JSON.SET

Establece valores JSON en la ruta.

Si la ruta de acceso llama a un miembro de objeto:

- Si el elemento principal no existe, el comando regresa un error INEXISTENTE.
- Si el elemento principal existe pero no es un objeto, el comando regresa ERROR.
- Si el elemento principal existe y es un objeto:
 - Si el miembro no existe, se anexará un miembro nuevo al objeto principal si y solo si el objeto principal es el último objeto secundario de la ruta. De lo contrario, el comando regresa un error INEXISTENTE.
 - Si el miembro existe, su valor se reemplazará por el valor JSON.

Si la ruta requiere un índice de matriz:

- Si el elemento principal no existe, el comando regresa un error INEXISTENTE.
- Si el elemento principal existe pero no es una matriz, el comando regresa ERROR.
- Si el elemento principal existe pero el índice está fuera de los límites, el comando regresa un error FUERA DE LOS LÍMITES.
- Si el elemento principal existe y el índice es válido, el elemento se reemplazará por el nuevo valor JSON.

Si la ruta llama a un objeto o matriz, el valor (objeto o matriz) se reemplazará por el nuevo valor JSON.

Sintaxis

```
JSON.SET <key> <path> <json> [NX | XX]
```

[NX | XX] Donde puede tener 0 o 1 de [NX | XX] identificadores.

- **clave (obligatoria):** una clave OSS de Redis del tipo de documento JSON.
- **ruta (obligatoria):** una ruta JSON. Para una nueva clave OSS de Redis, la ruta JSON debe ser la raíz «.».
- **NX (opcional):** si la ruta es la raíz, establezca el valor solo si la clave OSS de Redis no existe. Es decir, inserte un nuevo documento. Si la ruta no es la raíz, establezca el valor solo si la ruta no existe. Es decir, inserte un valor en el documento.
- **XX (opcional):** si la ruta es la raíz, defina el valor solo si existe la clave OSS de Redis. Es decir, reemplace el documento existente. Si la ruta no es la raíz, establezca el valor solo si la ruta existe. Es decir, actualice el valor existente.

Devolución

- Cadena simple 'OK' en caso de éxito.
- Es nulo si no se cumple la condición NX o XX.

Ejemplos

Sintaxis de la ruta mejorada:

```
127.0.0.1:6379> JSON.SET k1 . '{"a":{"a":1, "b":2, "c":3}}'  
OK  
127.0.0.1:6379> JSON.SET k1 $.a.* '0'  
OK  
127.0.0.1:6379> JSON.GET k1  
"{\"a\":{\"a\":0,\"b\":0,\"c\":0}}"  
  
127.0.0.1:6379> JSON.SET k2 . '{"a": [1,2,3,4,5]}'  
OK  
127.0.0.1:6379> JSON.SET k2 $.a[*] '0'  
OK  
127.0.0.1:6379> JSON.GET k2  
"{\"a\":[0,0,0,0,0]}"
```

Sintaxis de la ruta restringida:

```
127.0.0.1:6379> JSON.SET k1 . '{"c":{"a":1, "b":2}, "e": [1,2,3,4,5]}'
OK
127.0.0.1:6379> JSON.SET k1 .c.a '0'
OK
127.0.0.1:6379> JSON.GET k1
"{\"c\":{\"a\":0,\"b\":2},\"e\":[1,2,3,4,5]}"
127.0.0.1:6379> JSON.SET k1 .e[-1] '0'
OK
127.0.0.1:6379> JSON.GET k1
"{\"c\":{\"a\":0,\"b\":2},\"e\":[1,2,3,4,0]}"
127.0.0.1:6379> JSON.SET k1 .e[5] '0'
(error) OUTOFBOUNDARIES Array index is out of bounds
```

JSON.STRAPPEND

Adjunta una cadena a las cadenas JSON de la ruta.

Sintaxis

```
JSON.STRAPPEND <key> [path] <json_string>
```

- **clave (obligatoria):** una clave OSS de Redis del tipo de documento JSON.
- **ruta (opcional):** una ruta JSON. Toma el valor predeterminado raíz si no se proporciona.
- **json_string (obligatorio):** la representación JSON de una cadena. Tenga en cuenta que se debe citar una cadena JSON. Por ejemplo: “ejemplo de cadena”.

Devolución

Si la ruta es de sintaxis mejorada:

- Matriz de números enteros que representa la nueva longitud de la cadena en cada ruta.
- Si un valor en la ruta no es una cadena, su valor devuelto correspondiente es nulo.
- Error SYNTAXERR si el argumento json de entrada no es una cadena JSON válida.
- Error NONEXISTENT si la ruta no existe.

Si la ruta es de sintaxis restringida:

- Entero, la nueva longitud de la cadena.
- Si se seleccionan varios valores de cadena, el comando devuelve la nueva longitud de la última cadena actualizada.
- Error WRONGTYPE si el valor de la ruta no es una cadena.
- Error WRONGTYPE si el argumento json de entrada no es una cadena JSON válida.
- Error NONEXISTENT si la ruta no existe.

Ejemplos

Sintaxis de la ruta mejorada:

```
127.0.0.1:6379> JSON.SET k1 $ '{"a":{"a":"a"}, "b":{"a":"a", "b":1}, "c":{"a":"a",
"b":"bb"}, "d":{"a":1, "b":"b", "c":3}}'
OK
127.0.0.1:6379> JSON.STRAPPEND k1 $.a.a "a"
1) (integer) 2
127.0.0.1:6379> JSON.STRAPPEND k1 $.a.* "a"
1) (integer) 3
127.0.0.1:6379> JSON.STRAPPEND k1 $.b.* "a"
1) (integer) 2
2) (nil)
127.0.0.1:6379> JSON.STRAPPEND k1 $.c.* "a"
1) (integer) 2
2) (integer) 3
127.0.0.1:6379> JSON.STRAPPEND k1 $.c.b "a"
1) (integer) 4
127.0.0.1:6379> JSON.STRAPPEND k1 $.d.* "a"
1) (nil)
2) (integer) 2
3) (nil)
```

Sintaxis de la ruta restringida:

```
127.0.0.1:6379> JSON.SET k1 . '{"a":{"a":"a"}, "b":{"a":"a", "b":1}, "c":{"a":"a",
"b":"bb"}, "d":{"a":1, "b":"b", "c":3}}'
OK
127.0.0.1:6379> JSON.STRAPPEND k1 .a.a "a"
(integer) 2
127.0.0.1:6379> JSON.STRAPPEND k1 .a.* "a"
```

```
(integer) 3
127.0.0.1:6379> JSON.STRAPPEND k1 .b.* '"a"'
(integer) 2
127.0.0.1:6379> JSON.STRAPPEND k1 .c.* '"a"'
(integer) 3
127.0.0.1:6379> JSON.STRAPPEND k1 .c.b '"a"'
(integer) 4
127.0.0.1:6379> JSON.STRAPPEND k1 .d.* '"a"'
(integer) 2
```

JSON.STRLLEN

Consigue las longitudes de los valores de cadena JSON en la ruta.

Sintaxis

```
JSON.STRLLEN <key> [path]
```

- **clave (obligatoria):** una clave OSS de Redis del tipo de documento JSON.
- **ruta (opcional):** una ruta JSON. Toma el valor predeterminado raíz si no se proporciona.

Devolución

Si la ruta es de sintaxis mejorada:

- Matriz de números enteros que representa la longitud del valor de cadena en cada ruta.
- Si un valor no es una cadena, su valor devuelto correspondiente es nulo.
- Es nulo si la clave del documento no existe.

Si la ruta es de sintaxis restringida:

- Entero, la longitud de la cadena.
- Si se seleccionan varios valores de cadena, el comando devuelve la longitud de la primera cadena.
- Error `WRONGTYPE` si el valor de la ruta no es una cadena.
- Error `NONEXISTENT` si la ruta no existe.
- Es nulo si la clave del documento no existe.

Ejemplos

Sintaxis de la ruta mejorada:

```
127.0.0.1:6379> JSON.SET k1 $ '{"a":{"a":"a"}, "b":{"a":"a", "b":1}, "c":{"a":"a", "b":"bb"}, "d":{"a":1, "b":"b", "c":3}}'
OK
127.0.0.1:6379> JSON.STRLEN k1 $.a.a
1) (integer) 1
127.0.0.1:6379> JSON.STRLEN k1 $.a.*
1) (integer) 1
127.0.0.1:6379> JSON.STRLEN k1 $.c.*
1) (integer) 1
2) (integer) 2
127.0.0.1:6379> JSON.STRLEN k1 $.c.b
1) (integer) 2
127.0.0.1:6379> JSON.STRLEN k1 $.d.*
1) (nil)
2) (integer) 1
3) (nil)
```

Sintaxis de la ruta restringida:

```
127.0.0.1:6379> JSON.SET k1 $ '{"a":{"a":"a"}, "b":{"a":"a", "b":1}, "c":{"a":"a", "b":"bb"}, "d":{"a":1, "b":"b", "c":3}}'
OK
127.0.0.1:6379> JSON.STRLEN k1 .a.a
(integer) 1
127.0.0.1:6379> JSON.STRLEN k1 .a.*
(integer) 1
127.0.0.1:6379> JSON.STRLEN k1 .c.*
(integer) 1
127.0.0.1:6379> JSON.STRLEN k1 .c.b
(integer) 2
127.0.0.1:6379> JSON.STRLEN k1 .d.*
(integer) 1
```

JSON.TOGGLE

Alterna los valores booleanos entre verdadero y falso en el camino.

Sintaxis

```
JSON.TOGGLE <key> [path]
```

- **clave (obligatoria):** una clave OSS de Redis del tipo de documento JSON.
- **ruta (opcional):** una ruta JSON. Toma el valor predeterminado raíz si no se proporciona.

Devolución

Si la ruta es de sintaxis mejorada:

- Matriz de números enteros (0 - falso, 1 - verdadero) que representa el valor booleano resultante en cada ruta.
- Si un valor no es un valor booleano, su valor devuelto correspondiente es nulo.
- NONEXISTENT si la clave del documento no existe.

Si la ruta es de sintaxis restringida:

- Cadena (“verdadero”/“falso”) que representa el resultantevalor booleano.
- NONEXISTENT si la clave del documento no existe.
- Error WRONGTYPE si el valor de la ruta no es un valor booleano.

Ejemplos

Sintaxis de la ruta mejorada:

```
127.0.0.1:6379> JSON.SET k1 . '{"a":true, "b":false, "c":1, "d":null, "e":"foo", "f":
[], "g":{}}'
OK
127.0.0.1:6379> JSON.TOGGLE k1 $.*
1) (integer) 0
2) (integer) 1
3) (nil)
4) (nil)
5) (nil)
6) (nil)
7) (nil)
127.0.0.1:6379> JSON.TOGGLE k1 $.*
```

- 1) (integer) 1
- 2) (integer) 0
- 3) (nil)
- 4) (nil)
- 5) (nil)
- 6) (nil)
- 7) (nil)

Sintaxis de la ruta restringida:

```
127.0.0.1:6379> JSON.SET k1 . true
OK
127.0.0.1:6379> JSON.TOGGLE k1
"false"
127.0.0.1:6379> JSON.TOGGLE k1
"true"

127.0.0.1:6379> JSON.SET k2 . '{"isAvailable": false}'
OK
127.0.0.1:6379> JSON.TOGGLE k2 .isAvailable
"true"
127.0.0.1:6379> JSON.TOGGLE k2 .isAvailable
"false"
```

JSON.TYPE

Informa el tipo de valores en la ruta dada.

Sintaxis

```
JSON.TYPE <key> [path]
```

- clave (obligatoria): una clave OSS de Redis del tipo de documento JSON.
- ruta (opcional): una ruta JSON. Toma el valor predeterminado raíz si no se proporciona.

Devolución

Si la ruta es de sintaxis mejorada:

- Matriz de cadenas que representa el tipo de valor en cada ruta. El tipo es uno de {"nulo", "booleano", "cadena", "número", "entero", "objeto" y "matriz"}.
- Si no existe una ruta, su valor de retorno correspondiente es nulo.
- Matriz vacía si la clave del documento no existe.

Si la ruta es de sintaxis restringida:

- Cadena, tipo del valor
- Es nulo si la clave del documento no existe.
- Es nulo si la ruta JSON no es válida o no existe.

Ejemplos

Sintaxis de la ruta mejorada:

```
127.0.0.1:6379> JSON.SET k1 . '[1, 2.3, "foo", true, null, {}, []]'
OK
127.0.0.1:6379> JSON.TYPE k1 $[*]
1) integer
2) number
3) string
4) boolean
5) null
6) object
7) array
```

Sintaxis de la ruta restringida:

```
127.0.0.1:6379> JSON.SET k1 .
'{"firstName":"John","lastName":"Smith","age":27,"weight":135.25,"isAlive":true,"address":
{"street":"21 2nd Street","city":"New
York","state":"NY","zipcode":"10021-3100"},"phoneNumbers":
[{"type":"home","number":"212 555-1234"}, {"type":"office","number":"646
555-4567"}],"children":[],"spouse":null}'
OK
127.0.0.1:6379> JSON.TYPE k1
object
127.0.0.1:6379> JSON.TYPE k1 .children
```

```
array
127.0.0.1:6379> JSON.TYPE k1 .firstName
string
127.0.0.1:6379> JSON.TYPE k1 .age
integer
127.0.0.1:6379> JSON.TYPE k1 .weight
number
127.0.0.1:6379> JSON.TYPE k1 .isAlive
boolean
127.0.0.1:6379> JSON.TYPE k1 .spouse
null
```

Etiquetado de los recursos de ElastiCache

Para ayudarlo a administrar sus clústeres y otros recursos de ElastiCache, puede asignar sus propios metadatos a cada recurso en forma de etiquetas. Las etiquetas le permiten clasificar los recursos de AWS de diversas maneras, por ejemplo, según su finalidad, propietario o entorno. Esto es útil cuando tiene muchos recursos del mismo tipo: puede identificar rápidamente un recurso específico en función de las etiquetas que le haya asignado. En este tema se describe qué son las etiquetas y cómo crearlas.

Warning

Como práctica recomendada, no debe incluir datos confidenciales en las etiquetas.

Conceptos básicos de etiquetas

Una etiqueta es una marca que se asigna a un recurso de AWS. Cada etiqueta está formada por una clave y un valor opcional, ambos definidos por el usuario. Las etiquetas permiten clasificar los recursos de AWS de diversas maneras, por ejemplo, según finalidad o propietario. Por ejemplo, podría definir un conjunto de etiquetas para los clústeres de ElastiCache de su cuenta que lo ayude a realizar un seguimiento del propietario y el grupo de usuario de cada instancia.

Recomendamos que idee un conjunto de claves de etiqueta que cumpla sus necesidades para cada tipo de recurso. Mediante el uso de un conjunto coherente de claves de etiquetas, podrá administrar los recursos de más fácilmente. Puede buscar y filtrar los recursos en función de las etiquetas que agregue. Para obtener más información acerca de cómo implementar una estrategia eficaz de

etiquetado de recursos, consulte el [documento técnico de Prácticas recomendadas de etiquetado de AWS](#).

Las etiquetas no tienen significado semántico para ElastiCache, por lo que se interpretan estrictamente como cadenas de caracteres. Además, las etiquetas no se asignan a los recursos automáticamente. Puede editar las claves y los valores de las etiquetas y también puede eliminar etiquetas de un recurso en cualquier momento. Puede establecer el valor de una etiqueta en `null`. Si añade una etiqueta con la misma clave que una etiqueta existente en ese recurso, el nuevo valor sobrescribirá al antiguo. Si elimina un recurso, también se eliminará cualquier etiqueta asignada a dicho recurso. Además, si agrega o elimina etiquetas en un grupo de reproducción, todos los nodos de ese grupo de reproducción también se agregarán o eliminarán sus etiquetas.

Puede trabajar con etiquetas mediante la AWS Management Console, la AWS CLI y la API de ElastiCache.

Si utiliza IAM, puede controlar qué usuarios de su cuenta de AWS tienen permiso para crear, editar o eliminar etiquetas. Para obtener más información, consulte [Permisos de nivel de recursos](#).

Recursos que se pueden etiquetar

Puede etiquetar la mayoría de los recursos de ElastiCache que ya existen en la cuenta. La siguiente tabla enumera los recursos que admiten etiquetas. Si utiliza la AWS Management Console, puede aplicar etiquetas a recursos a través del [Editor de etiquetas](#). Algunas pantallas de recursos permiten especificar etiquetas para un recurso al crear dicho recurso; por ejemplo, una etiqueta con una clave de Name (Nombre) y un valor que especifique. En la mayoría de los casos, la consola aplica las etiquetas inmediatamente después de crear el recurso (y no durante la creación del mismo). La consola puede organizar los recursos según la etiqueta de Name (Nombre), si bien dicha etiqueta no tiene significado semántico para el servicio de ElastiCache.

Además, algunas acciones de creación de recursos le permiten especificar etiquetas para un recurso al crear dicho recurso. Si no se pueden aplicar etiquetas durante la creación del recurso, el proceso de creación del recurso se revierte. Esto garantiza que los recursos se creen con etiquetas o, de lo contrario, no se creen y que ningún recurso se quede jamás sin etiquetar. Al etiquetar los recursos en el momento de su creación, se elimina la necesidad de ejecutar scripts de etiquetado personalizados tras la creación del recurso.

Si utiliza la API de Amazon ElastiCache, la AWS CLI o un SDK de AWS, puede aplicar etiquetas mediante el parámetro `Tags` en la acción de la API de ElastiCache pertinente. Son los siguientes:

- `CreateServerlessCache`

- `CreateCacheCluster`
- `CreateReplicationGroup`
- `CopyServerlessCacheSnapshot`
- `CopySnapshot`
- `CreateCacheParameterGroup`
- `CreateCacheSecurityGroup`
- `CreateCacheSubnetGroup`
- `CreateServerlessCacheSnapshot`
- `CreateSnapshot`
- `CreateUserGroup`
- `CreateUser`
- `PurchaseReservedCacheNodesOffering`

En la siguiente tabla se describen los recursos de ElastiCache que se pueden etiquetar y aquellos que se pueden etiquetar en el momento de su creación con la API de ElastiCache, la AWS CLI o un SDK de AWS.

Compatibilidad con el etiquetado de recursos de ElastiCache

Admite etiquetas	Admite el etiquetado o durante la creación
Sí	Sí

Admite etiquetas	Admite el etiquetado o durante la creación
Sí	Sí

 Note

No puede etiquetar almacenes de datos globales.

En las políticas de IAM, puede aplicar permisos de nivel de recursos basados en etiquetas a las acciones de la API de ElastiCache que admitan el etiquetado durante la creación para implementar un control pormenorizado de los usuarios y los grupos que pueden etiquetar recursos durante la creación. Sus recursos se encuentran debidamente protegidos de las etiquetas de creación que se aplican de inmediato a los recursos. Por lo tanto, cualquier permiso de nivel de recursos basado en etiquetas que controle la utilización de recursos es efectivo de inmediato. Se puede realizar un seguimiento y un registro más precisos de los recursos. Puede establecer el etiquetado obligatorio de los nuevos recursos y controlar qué claves y valores de etiquetas se usan en ellos.

Para obtener más información, consulte [Ejemplos de etiquetado de recursos](#).

A fin de obtener más información sobre el etiquetado de recursos para facturación, consulte [Monitoreo de costos con etiquetas de asignación de costos](#).

Etiquetado de cachés e instantáneas

Las siguientes reglas se aplican al etiquetado como parte de las operaciones de solicitud:

- `CreateReplicationGroup`:
 - Si los parámetros `--primary-cluster-id` y `--tags` se incluyen en la solicitud, las etiquetas de solicitud se agregarán al grupo de reproducción y se propagarán a todos los clústeres de caché del grupo de reproducción. Si el clúster de caché principal tiene etiquetas existentes, estas se sobrescribirán con las etiquetas de la solicitud para tener etiquetas coherentes en todos los nodos.

Si no hay etiquetas de la solicitud, las etiquetas de clúster de caché principales se agregarán al grupo de reproducción y se propagarán a todos los clústeres de caché.

- Si se suministra el `--snapshot-name` o el `--serverless-cache-snapshot-name`:

Si se incluyen etiquetas en la solicitud, el grupo de réplica solo se etiquetará con esas etiquetas. Si no se incluyen etiquetas en la solicitud, las etiquetas de la instantánea se agregarán al grupo de réplica.

- Si se proporciona el `--global-replication-group-id`:

Si se incluyen etiquetas en la solicitud, las etiquetas de solicitud se agregarán al grupo de réplica y se propagarán a todos los clústeres de caché.

- `CreateCacheCluster`

- Si se proporciona el `--replication-group-id`:

Si se incluyen etiquetas en la solicitud, solo se le aplicarán esas etiquetas al clúster de caché. Si no se incluyen etiquetas en la solicitud, el clúster de caché heredará las etiquetas del grupo de reproducción en lugar de las etiquetas del clúster de caché principal.

- Si se proporciona el `--snapshot-name`:

Si se incluyen etiquetas en la solicitud, solo se le aplicarán esas etiquetas al clúster de caché. Si no se incluyen etiquetas en la solicitud, las etiquetas de la instantánea se agregarán al clúster de **caché**.

- **CreateServerlessCache:**
 - Si se incluyen etiquetas en la solicitud, solo las etiquetas de solicitud se agregarán al caché sin servidor.
- **CreateSnapshot**
 - Si se proporciona el `--replication-group-id`:

Si no se incluyen etiquetas en la solicitud, las etiquetas de solicitud se agregarán a la instantánea. Si no se incluyen etiquetas en la solicitud, las etiquetas del grupo de reproducción se agregarán a la instantánea.
 - Si se proporciona el `--cache-cluster-id`:

Si no se incluyen etiquetas en la solicitud, las etiquetas de solicitud se agregarán a la instantánea. Si no se incluyen etiquetas en la solicitud, las etiquetas del clúster de caché se agregarán a la instantánea.
 - Para las instantáneas automáticas:

Las etiquetas se propagarán desde las etiquetas de grupo de reproducción.
- **CreateServerlessCacheSnapshot:**
 - Si se incluyen etiquetas en la solicitud, solo las etiquetas de solicitud se agregarán a la instantánea de la caché sin servidor.
- **CopySnapshot**
 - Si no se incluyen etiquetas en la solicitud, las etiquetas de solicitud se agregarán a la instantánea. Si no se incluyen etiquetas en la solicitud, las etiquetas de la instantánea fuente se agregarán a la instantánea copiada.
- **CopyServerlessCacheSnapshot:**
 - Si se incluyen etiquetas en la solicitud, solo las etiquetas de solicitud se agregarán a la instantánea de la caché sin servidor.
- **AddTagsToResource** y **RemoveTagsFromResource:**
 - Las etiquetas se agregarán o quitarán del grupo de reproducción y la acción se propagará a todos los clústeres del grupo de reproducción.

 Note

`AddTagsToResource` y `RemoveTagsFromResource` no se pueden utilizar para los grupos de seguridad y parámetros predeterminados.

- `IncreaseReplicaCount` y `ModifyReplicationGroupShardConfiguration`:
 - Todos los clústeres nuevos agregados al grupo de réplica tendrán las mismas etiquetas aplicadas que el grupo de réplica.

Restricciones de las etiquetas

Se aplican las siguientes restricciones básicas a las etiquetas:

- Número máximo de etiquetas por recurso: 50
- Para cada recurso, cada clave de etiqueta debe ser única y solo puede tener un valor.
- Longitud máxima de la clave: 128 caracteres Unicode en UTF-8.
- Longitud máxima del valor: 256 caracteres Unicode en UTF-8.
- Si bien ElastiCache admite la utilización de cualquier carácter en sus etiquetas, otros servicios pueden ser restrictivos. Los caracteres permitidos en los servicios son: letras, números y espacios representables en UTF-8, además de los siguientes caracteres: `+ - = . _ : / @`
- Las claves y los valores de las etiquetas distinguen entre mayúsculas y minúsculas.
- El prefijo `aws :` se reserva para uso de AWS. Si la etiqueta tiene una clave de etiqueta con este prefijo, no puede editar ni eliminar la clave o el valor de la etiqueta. Las etiquetas que tengan el prefijo `aws :` no cuentan para el límite de etiquetas por recurso.

No puede terminar, detener ni eliminar un recurso basado únicamente en sus etiquetas; debe especificar el identificador del recurso. Por ejemplo, para eliminar instantáneas que etiquetó con una clave de etiqueta llamada `DeleteMe`, debe utilizar la acción `DeleteSnapshot` con los identificadores del recurso de las instantáneas, como `snap-1234567890abcdef0`.

Para obtener más información sobre los recursos de ElastiCache que puede etiquetar, consulte [Recursos que se pueden etiquetar](#).

Ejemplos de etiquetado de recursos

- Creación de una caché sin servidor mediante etiquetas

```
aws elasticache create-serverless-cache \  
  --serverless-cache-name CacheName \  
  --engine redis \  
  --tags Key="Cost Center", Value="11110001" Key="project",Value="XYZ"
```

- Adición de etiquetas a una caché sin servidor

```
aws elasticache add-tags-to-resource \  
--resource-name arn:aws:elasticache:us-east-1:111111222233:serverlesscache:my-cache \  
--tags Key="project",Value="XYZ" Key="Elasticache",Value="Service"
```

- Adición de etiquetas a un grupo de reproducción.

```
aws elasticache add-tags-to-resource \  
--resource-name arn:aws:elasticache:us-east-1:111111222233:replicationgroup:my-rg \  
--tags Key="project",Value="XYZ" Key="Elasticache",Value="Service"
```

- Creación de un clúster de caché mediante etiquetas.

```
aws elasticache create-cache-cluster \  
--cluster-id testing-tags \  
--cluster-description cluster-test \  
--cache-subnet-group-name test \  
--cache-node-type cache.t2.micro \  
--engine redis \  
--tags Key="project",Value="XYZ" Key="Elasticache",Value="Service"
```

- Creación de una instantánea sin servidor con etiquetas.

```
aws elasticache create-serverless-cache-snapshot \  
--serverless-cache-name testing-tags \  
--serverless-cache-snapshot-name bkp-testing-tags-scs \  
--tags Key="work",Value="foo"
```

- Creación de una instantánea con etiquetas.

En este caso, si agrega etiquetas a la solicitud, incluso si el grupo de reproducción contiene etiquetas, la instantánea solo recibirá las etiquetas de la solicitud.

```
aws elasticache create-snapshot \  
--replication-group-id testing-tags \  
--snapshot-name bkp-testing-tags-rg \  
--tags Key="work",Value="foo"
```

Ejemplos de políticas de control de acceso basadas en etiquetas

1. Permitir la acción `AddTagsToResource` a un clúster solo si el clúster tiene la etiqueta `Project=XYZ`.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "elasticache:AddTagsToResource",
      "Resource": [
        "arn:aws:elasticache:*:*:cluster:*"
      ],
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/Project": "XYZ"
        }
      }
    }
  ]
}
```

2. Permitir la acción `RemoveTagsFromResource` de un grupo de reproducción si contiene las etiquetas `Project` y `Service` y las claves son diferentes de `Project` y `Service`.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "elasticache:RemoveTagsFromResource",
      "Resource": [
        "arn:aws:elasticache:*:*:replicationgroup:*"
      ],
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/Service": "Elasticache",
          "aws:ResourceTag/Project": "XYZ"
        },
        "ForAnyValue:StringNotEqualsIgnoreCase": {
          "aws:TagKeys": [
            "Project",

```

```

        "Service"
      ]
    }
  }
]
}

```

3. Permitir AddTagsToResource a cualquier recurso solo si las etiquetas son diferentes de Project y Service.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "elasticache:AddTagsToResource",
      "Resource": [
        "arn:aws:elasticache:*:*:*:*"
      ],
      "Condition": {
        "ForAnyValue:StringNotEqualsIgnoreCase": {
          "aws:TagKeys": [
            "Service",
            "Project"
          ]
        }
      }
    }
  ]
}

```

4. Denegar la acción CreateReplicationGroup si la solicitud tiene Tag Project=Foo.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": "elasticache:CreateReplicationGroup",
      "Resource": [
        "arn:aws:elasticache:*:*:replicationgroup:*"
      ],

```

```

        "Condition": {
            "StringEquals": {
                "aws:RequestTag/Project": "Foo"
            }
        }
    ]
}

```

5. Denegar la acción CopySnapshot si la instantánea fuente tiene la etiqueta Project=XYZ y la etiqueta de la solicitud es Service=Elasticache.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": "elasticache:CopySnapshot",
      "Resource": [
        "arn:aws:elasticache:*:*:snapshot:*"
      ],
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/Project": "XYZ",
          "aws:RequestTag/Service": "Elasticache"
        }
      }
    }
  ]
}

```

6. Denegar la acción CreateCacheCluster si la etiqueta de solicitud Project falta o no es igual a Dev, QA o Prod.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateCacheCluster"
      ],
      "Resource": [

```

```

        "arn:aws:elasticache:*:*:parametergroup:*",
        "arn:aws:elasticache:*:*:subnetgroup:*",
        "arn:aws:elasticache:*:*:securitygroup:*",
        "arn:aws:elasticache:*:*:replicationgroup:*"
    ]
},
{
    "Effect": "Deny",
    "Action": [
        "elasticache:CreateCacheCluster"
    ],
    "Resource": [
        "arn:aws:elasticache:*:*:cluster:*"
    ],
    "Condition": {
        "Null": {
            "aws:RequestTag/Project": "true"
        }
    }
},
{
    "Effect": "Allow",
    "Action": [
        "elasticache:CreateCacheCluster",
        "elasticache:AddTagsToResource"
    ],
    "Resource": "arn:aws:elasticache:*:*:cluster:*",
    "Condition": {
        "StringEquals": {
            "aws:RequestTag/Project": [
                "Dev",
                "Prod",
                "QA"
            ]
        }
    }
}
]
}

```

Para obtener información relacionada acerca de las claves de condición, consulte [Uso de claves de condición](#).

Monitoreo de costos con etiquetas de asignación de costos

Cuando añades etiquetas de asignación de costes a tus recursos en Amazon ElastiCache, puedes realizar un seguimiento de los costes agrupando los gastos de tus facturas por valores de etiquetas de recursos.

Una etiqueta de asignación de ElastiCache costes es un par clave-valor que se define y se asocia a un recurso. ElastiCache Las claves y los valores distinguen entre mayúsculas y minúsculas. Puede utilizar una clave de etiqueta para definir una categoría y el valor de la etiqueta puede ser un elemento dentro de esa categoría. Por ejemplo, puede definir una clave de etiqueta `CostCenter` y un valor de etiqueta `10010` para indicar que el recurso va asignado al centro de costos 10010. También puede usar etiquetas para designar recursos para pruebas o para producción a través de una clave como `Environment` y valores como `test` o `production`. Se recomienda utilizar un conjunto coherente de claves de etiqueta que facilite el seguimiento de los costos asociados a los recursos.

Utilice las etiquetas de asignación de costes para organizar la AWS factura y reflejar su propia estructura de costes. Para ello, regístrese para recibir la factura de su AWS cuenta con los valores clave de las etiquetas incluidos. A continuación, para ver los costos de los recursos combinados, organice la información de facturación de acuerdo con los recursos con los mismos valores de clave de etiquetas. Por ejemplo, puede etiquetar varios recursos con un nombre de aplicación específico y luego organizar su información de facturación para ver los costos totales de la aplicación en distintos servicios.

También puede combinar etiquetas para realizar un seguimiento de los costos con un mayor nivel de detalle. Por ejemplo, para realizar un seguimiento de los costos de su servicio por región, puede utilizar las claves de etiqueta `Service` y `Region`. En un recurso podría tener los valores `ElastiCache` y `Asia Pacific (Singapore)` y en otro recurso, los valores `ElastiCache` y `Europe (Frankfurt)`. A continuación, podrás ver ElastiCache los costes totales desglosados por región. Para obtener más información, consulte [Uso de etiquetas de asignación de costos](#) en la Guía del usuario de AWS Billing .

Puede añadir etiquetas de asignación de ElastiCache costes a los nodos OSS de Redis. Al agregar, enumerar, modificar, copiar o quitar una etiqueta, la operación se aplica únicamente al nodo especificado.

Características de las etiquetas de asignación de ElastiCache costes

- Las etiquetas de asignación de costos se aplican a ElastiCache los recursos que se especifican en las operaciones de CLI y API como un ARN. El tipo de recurso será "clúster".

Ejemplo de ARN: `arn:aws:elasticache:<region>:<customer-id>:<resource-type>:<resource-name>`

Ejemplo de ARN: `arn:aws:elasticache:us-west-2:1234567890:cluster:my-cluster`

- La clave de la etiqueta es el nombre obligatorio de la etiqueta. El valor de cadena de la clave puede tener una longitud de entre 1 y 128 caracteres Unicode y no puede llevar el prefijo `aws:`. La cadena solo puede contener un conjunto Unicode de letras, dígitos, espacios en blanco, guiones bajos (`_`), puntos (`.`), dos puntos (`:`), barras oblicuas (`\`), signos de igual (`=`), signos de suma (`+`), guiones (`-`) o signos de arroba (`@`).
- El valor de etiqueta es la parte opcional de la etiqueta. El valor de cadena del valor puede tener una longitud de entre 1 y 256 caracteres Unicode y no puede llevar el prefijo `aws:`. La cadena solo puede contener un conjunto Unicode de letras, dígitos, espacios en blanco, guiones bajos (`_`), puntos (`.`), dos puntos (`:`), barras oblicuas (`\`), signos de igual (`=`), signos de suma (`+`), guiones (`-`) o signos de arroba (`@`).
- Un ElastiCache recurso puede tener un máximo de 50 etiquetas.
- Los valores no deben ser únicos dentro de un conjunto de etiquetas. Por ejemplo, puede disponer de un conjunto de etiquetas donde las claves `Service` y `Application` tienen el valor `ElastiCache`.

AWS no aplica ningún significado semántico a las etiquetas. Las etiquetas se interpretan estrictamente como cadenas de caracteres. AWS no establece automáticamente ninguna etiqueta en ningún ElastiCache recurso.

Administrar las etiquetas de asignación de costes mediante el AWS CLI

Puede utilizarlas AWS CLI para añadir, modificar o eliminar etiquetas de asignación de costes.

Ejemplo de ARN: `arn:aws:elasticache:us-west-2:1234567890:cluster:my-cluster`

Las etiquetas de asignación de costes se aplican a los nodos ElastiCache (Redis OSS). El nodo que se va a etiquetar se especifica mediante un ARN (nombre de recurso de Amazon).

Ejemplo de ARN: `arn:aws:elasticache:us-west-2:1234567890:cluster:my-cluster`

Temas

- [Listar las etiquetas mediante el AWS CLI](#)
- [Añadir etiquetas mediante el AWS CLI](#)
- [Modificación de etiquetas mediante el AWS CLI](#)
- [Eliminar etiquetas mediante el AWS CLI](#)

Listar las etiquetas mediante el AWS CLI

Puede utilizarla AWS CLI para enumerar las etiquetas de un ElastiCache recurso existente mediante la [list-tags-for-resource](#) operación.

El código siguiente lo utiliza AWS CLI para enumerar las etiquetas del nodo OSS de Redis del `my-cluster` clúster `my-cluster-001` de la región `us-west-2`.

Para Linux, macOS o Unix:

```
aws elasticache list-tags-for-resource \  
  --resource-name arn:aws:elasticache:us-west-2:0123456789:cluster:my-cluster-001
```

Para Windows:

```
aws elasticache list-tags-for-resource ^  
  --resource-name arn:aws:elasticache:us-west-2:0123456789:cluster:my-cluster-001
```

La salida de esta operación se parecerá a lo siguiente, una lista de todas las etiquetas en el recurso.

```
{  
  "TagList": [  
    {  
      "Value": "10110",  
      "Key": "CostCenter"    }  
  ]  
}
```

```
    },
    {
      "Value": "EC2",
      "Key": "Service"
    }
  ]
}
```

Si no hay etiquetas en el recurso, la salida estará vacía. TagList

```
{
  "TagList": []
}
```

Para obtener más información, consulte AWS CLI el formulario ElastiCache [list-tags-for-resource](#).

Añadir etiquetas mediante el AWS CLI

Puede usar el AWS CLI para agregar etiquetas a un ElastiCache recurso existente mediante la operación [add-tags-to-resource](#) CLI. Si la clave de etiqueta no existe en el recurso, la clave y el valor se añadirán a los recursos. Si la clave ya existe en el recurso, el valor asociado a dicha clave se actualizará al nuevo valor.

El siguiente código utiliza el AWS CLI para añadir las claves Service y Region con los valores elasticache yus-west-2, respectivamente, al nodo my-cluster-001 del clúster de la my-cluster región us-west-2.

Para Linux, macOS o Unix:

```
aws elasticache add-tags-to-resource \
  --resource-name arn:aws:elasticache:us-west-2:0123456789:cluster:my-cluster-001 \
  --tags Key=Service,Value=elasticache \
  Key=Region,Value=us-west-2
```

Para Windows:

```
aws elasticache add-tags-to-resource ^
  --resource-name arn:aws:elasticache:us-west-2:0123456789:cluster:my-cluster-001 ^
  --tags Key=Service,Value=elasticache ^
  Key=Region,Value=us-west-2
```

Tras la operación, la salida de esta operación se parecerá a lo siguiente, una lista de todas las etiquetas en el recurso.

```
{
  "TagList": [
    {
      "Value": "elasticache",
      "Key": "Service"
    },
    {
      "Value": "us-west-2",
      "Key": "Region"
    }
  ]
}
```

Para obtener más información, consulte el AWS CLI formulario. ElastiCache [add-tags-to-resource](#)

También puede utilizar el AWS CLI para añadir etiquetas a un clúster al crear un clúster nuevo mediante la operación [create-cache-cluster](#). No puede añadir etiquetas al crear un clúster mediante la consola ElastiCache de administración. Una vez creado el clúster, puede utilizar la consola para agregar etiquetas al clúster.

Modificación de etiquetas mediante el AWS CLI

Puede usarlo AWS CLI para modificar las etiquetas de un nodo de un clúster ElastiCache (Redis OSS).

Para modificar las etiquetas:

- Use [add-tags-to-resource](#) para agregar una etiqueta y un valor nuevos o para cambiar el valor asociado a una etiqueta existente.
- Use [remove-tags-from-resource](#) para quitar etiquetas especificadas del recurso.

La salida de cualquier operación será una lista de las etiquetas y sus valores en el clúster especificado.

Eliminar etiquetas mediante el AWS CLI

Puede utilizarla AWS CLI para eliminar etiquetas de un nodo existente en un clúster ElastiCache (Redis OSS) mediante la operación. [remove-tags-from-resource](#)

El código siguiente utiliza el AWS CLI para eliminar las etiquetas con las claves `Service` y `Region` del nodo del clúster `my-cluster-001` en la región `us-west-2`.

Para Linux, macOS o Unix:

```
aws elasticache remove-tags-from-resource \  
  --resource-name arn:aws:elasticache:us-west-2:0123456789:cluster:my-cluster-001 \  
  --tag-keys PM Service
```

Para Windows:

```
aws elasticache remove-tags-from-resource ^  
  --resource-name arn:aws:elasticache:us-west-2:0123456789:cluster:my-cluster-001 ^  
  --tag-keys PM Service
```

Tras la operación, la salida de esta operación se parecerá a lo siguiente, una lista de todas las etiquetas en el recurso.

```
{  
  "TagList": []  
}
```

Para obtener más información, consulte el AWS CLI formulario. ElastiCache [remove-tags-from-resource](#)

Gestión de las etiquetas de asignación de costes mediante la ElastiCache API

Puede utilizar la ElastiCache API para añadir, modificar o eliminar etiquetas de asignación de costes.

Las etiquetas de asignación de costes se aplican a los ElastiCache clústeres de Memcached. El clúster que se va a etiquetar se especifica mediante un ARN (nombre de recurso de Amazon).

Ejemplo de ARN: `arn:aws:elasticache:us-west-2:1234567890:cluster:my-cluster`

Temas

- [Listar las etiquetas mediante la API ElastiCache](#)
- [Añadir etiquetas mediante la API ElastiCache](#)

- [Modificación de etiquetas mediante la ElastiCache API](#)
- [Eliminar etiquetas mediante la ElastiCache API](#)

Listar las etiquetas mediante la API ElastiCache

Puedes usar la ElastiCache API para enumerar las etiquetas de un recurso existente mediante la [ListTagsForResource](#) operación.

El código siguiente usa la ElastiCache API para enumerar las etiquetas del recurso `my-cluster-001` en la región `us-west-2`.

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=ListTagsForResource  
  &ResourceName=arn:aws:elasticache:us-west-2:0123456789:cluster:my-cluster-001  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Version=2015-02-02  
  &Timestamp=20150202T192317Z  
  &X-Amz-Credential=<credential>
```

Añadir etiquetas mediante la API ElastiCache

Puedes usar la ElastiCache API para añadir etiquetas a un ElastiCache clúster existente mediante la [AddTagsToResource](#) operación. Si la clave de etiqueta no existe en el recurso, la clave y el valor se añadirán a los recursos. Si la clave ya existe en el recurso, el valor asociado a dicha clave se actualizará al nuevo valor.

El siguiente código usa la ElastiCache API para agregar las claves `Service` y `Region` con los valores `elasticache` y `us-west-2`, respectivamente, al recurso `my-cluster-001` en la región `us-west-2`.

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=AddTagsToResource  
  &ResourceName=arn:aws:elasticache:us-west-2:0123456789:cluster:my-cluster-001  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Tags.member.1.Key=Service  
  &Tags.member.1.Value=elasticache  
  &Tags.member.2.Key=Region  
  &Tags.member.2.Value=us-west-2
```

```
&Version=2015-02-02
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>
```

Para obtener más información, consulta [AddTagsToResource](#) la referencia de la ElastiCache API de Amazon.

Modificación de etiquetas mediante la ElastiCache API

Puedes usar la ElastiCache API para modificar las etiquetas de un ElastiCache clúster.

Para modificar el valor de una etiqueta:

- Use la operación [AddTagsToResource](#) para agregar una etiqueta y un valor nuevos o para cambiar el valor de una etiqueta existente.
- Use [RemoveTagsFromResource](#) para quitar etiquetas del recurso.

La salida de cualquier operación será una lista de las etiquetas y sus valores en el recurso especificado.

Use [RemoveTagsFromResource](#) para quitar etiquetas del recurso.

Eliminar etiquetas mediante la ElastiCache API

Puede usar la ElastiCache API para eliminar etiquetas de un nodo existente ElastiCache (Redis OSS) mediante la operación. [RemoveTagsFromResource](#)

El código siguiente usa la ElastiCache API para eliminar las etiquetas con las claves Service y Region del nodo del clúster `my-cluster-001` en la región `us-west-2`.

```
https://elasticache.us-west-2.amazonaws.com/
?Action=RemoveTagsFromResource
&ResourceName=arn:aws:elasticache:us-west-2:0123456789:cluster:my-cluster-001
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&TagKeys.member.1=Service
&TagKeys.member.2=Region
&Version=2015-02-02
&Timestamp=20150202T192317Z
&X-Amz-Credential=<credential>
```

Uso del enfoque Well-Architected de Amazon ElastiCache

En esta sección se describe el enfoque Well-Architected de Amazon ElastiCache, una colección de principios de diseño y directrices para diseñar cargas de trabajo de ElastiCache con una buena arquitectura.

- El enfoque de ElastiCache se suma al [Marco de AWS Well-Architected](#).
- Cada pilar tiene un conjunto de preguntas para ayudar a iniciar el debate sobre la arquitectura de ElastiCache.
 - Cada pregunta tiene una serie de prácticas principales junto con sus puntuaciones para la presentación de informes.
 - Obligatorio: necesario antes de pasar a producción (su ausencia representa un alto riesgo)
 - Lo mejor: el mejor estado posible en el que podría estar un cliente
 - Bueno: lo que recomendamos a los clientes que hagan (si no hay un riesgo medio)
- Terminología de Well-Architected
 - [Componente](#): código, configuración y recursos de AWS que, en conjunto, cumplen con un requisito. Los componentes interactúan con otros componentes y, a menudo, equivalen a un servicio en las arquitecturas de microservicios.
 - [Carga de trabajo](#): conjunto de componentes que, en conjunto, ofrecen valor empresarial. Algunos ejemplos de cargas de trabajo son sitios web de marketing, sitios web de comercio electrónico, el backend de una aplicación móvil, plataformas de análisis, etc.

Temas

- [El pilar de la excelencia ElastiCache operativa de Amazon Well-Architected Lens](#)
- [El pilar de seguridad de Amazon ElastiCache Well-Architected Lens](#)
- [El pilar de confiabilidad de Amazon ElastiCache Well-Architected Lens](#)
- [Pilar de eficiencia en el ElastiCache rendimiento de Amazon Well-Architected Lens](#)
- [El pilar de optimización de ElastiCache costes de Amazon Well-Architected Lens](#)

El pilar de la excelencia ElastiCache operativa de Amazon Well-Architected Lens

El pilar de la excelencia operativa se centra en el funcionamiento y la supervisión de los sistemas para ofrecer valor empresarial, y en la mejora continua de los procesos y los procedimientos. Los temas clave incluyen la automatización de los cambios, la respuesta a los eventos y la definición de estándares para administrar las operaciones diarias.

Temas

- [OE 1: ¿Cómo entiende y responde a las alertas y los eventos activados por su ElastiCache clúster?](#)
- [OE 2: ¿Cuándo y cómo se escalan los clústeres existentes? ElastiCache](#)
- [OE 3: ¿Cómo administra los recursos ElastiCache del clúster y cómo lo mantiene up-to-date?](#)
- [OE 4: ¿Cómo gestiona las conexiones de los clientes con sus ElastiCache clústeres?](#)
- [OE 5: ¿Cómo se implementan ElastiCache los componentes para una carga de trabajo?](#)
- [OE 6: ¿Cómo se planifican y mitigan los errores?](#)
- [OE 7: ¿Cómo se solucionan los problemas de los eventos OSS del motor Redis?](#)

OE 1: ¿Cómo entiende y responde a las alertas y los eventos activados por su ElastiCache clúster?

Introducción a nivel de preguntas: cuando opera ElastiCache clústeres, puede recibir opcionalmente notificaciones y alertas cuando se produzcan eventos específicos. ElastiCache, de forma predeterminada, registra los [eventos](#) relacionados con sus recursos, como una conmutación por error, la sustitución de nodos, una operación de escalado, un mantenimiento programado, etc. Cada evento incluye la fecha y la hora, el nombre y el tipo de origen y una descripción.

Ventaja a nivel de pregunta: La capacidad de comprender y gestionar los motivos subyacentes de los eventos que activan las alertas generadas por su clúster le permite operar de manera más eficaz y responder a los eventos de manera adecuada.

- [Obligatorio] Revise los eventos generados ElastiCache en la ElastiCache consola (después de seleccionar su región) o mediante el [comando describe-events de Amazon Command Line Interface](#) (AWS CLI) y el [ElastiCache API](#) Configure ElastiCache para enviar notificaciones de eventos importantes del clúster mediante Amazon Simple Notification Service (AmazonSNS). El

uso de Amazon SNS con sus clústeres le permite realizar acciones en función ElastiCache de los eventos mediante programación.

- Hay dos grandes categorías de eventos: eventos actuales y programados. La lista de eventos actuales incluye: la creación y eliminación de recursos, las operaciones de escalado, la conmutación por error, el reinicio de nodos, la creación de una instantánea, la modificación de los parámetros del clúster, la renovación del certificado de CA, los eventos de error (error de aprovisionamiento del clúster (VPCo ENI -), errores de escalado (ENI-) y errores de instantáneas). La lista de eventos programados incluye: reemplazo de nodos programado durante el período de mantenimiento y reemplazo de nodos reprogramado.
- Si bien es posible que no necesite reaccionar de inmediato ante algunos de estos eventos, es fundamental analizar primero todos los eventos de error:
 - ElastiCache:AddCacheNodeFailed
 - ElastiCache:CacheClusterProvisioningFailed
 - ElastiCache:CacheClusterScalingFailed
 - ElastiCache:CacheNodesRebooted
 - ElastiCache: SnapshotFailed (solo en RedisOSS)
- [Recursos]:
 - [Gestión de SNS las notificaciones de ElastiCache Amazon](#)
 - [Notificaciones de eventos y Amazon SNS](#)
- [Lo mejor] Para automatizar las respuestas a los eventos, aproveche las capacidades de los AWS productos y servicios, como SNS las funciones Lambda. Siga las prácticas recomendadas y haga cambios pequeños, frecuentes y reversibles, como código para que sus operaciones evolucionen a lo largo del tiempo. Deberías usar CloudWatch las métricas de Amazon para monitorear tus clústeres.

[Recursos]: [Supervise ElastiCache \(RedisOSS\) \(modo de clúster desactivado\) y lea los puntos de enlace de réplica mediante AWS Lambda, Amazon Route 53 y Amazon SNS](#) para un caso de uso que utilice Lambda y. SNS

OE 2: ¿Cuándo y cómo se escalan los clústeres existentes? ElastiCache

Introducción a nivel de preguntas: Ajustar el tamaño del ElastiCache clúster es un acto de equilibrio que debe evaluarse cada vez que se produzcan cambios en los tipos de carga de trabajo subyacentes. El objetivo es operar con el entorno del tamaño adecuado para la carga de trabajo.

Ventaja a nivel de pregunta: La sobreutilización de los recursos puede provocar una latencia elevada y una disminución general del rendimiento. Por otro lado, la infrautilización puede provocar un aprovisionamiento excesivo de recursos con una optimización de costos deficiente. El dimensionamiento correcto de los entornos, puede lograr un equilibrio entre la eficiencia del rendimiento y la optimización de costos. Para corregir la sobreutilización o la infrautilización de sus recursos, ElastiCache puede escalarlos en dos dimensiones. Puede escalar verticalmente aumentando o disminuyendo la capacidad de los nodos. Y puede escalar horizontalmente añadiendo y eliminando nodos.

- [Obligatorio] CPU y la sobreutilización de la red en los nodos principales deben abordarse descargando y redirigiendo las operaciones de lectura a los nodos de réplica. Utilice nodos de réplica para las operaciones de lectura a fin de reducir la utilización del nodo principal. Esto se puede configurar en la biblioteca de OSS clientes de Redis conectándose al punto final del ElastiCache lector si el modo de clúster está desactivado o mediante el comando de Redis OSS READONLY si el modo de clúster está activado.

[Recursos]:

- [Búsqueda de puntos de conexión](#)
 - [Cluster Right-Sizing](#)
 - [Comando Redis OSS READONLY](#)
- [Obligatorio] Supervise la utilización de los recursos críticos del clúster CPU, como la memoria y la red. Es necesario realizar un seguimiento del uso de estos recursos específicos del clúster para tomar una decisión de escalado y determinar el tipo de operación de escalado. Si el modo de clúster ElastiCache (RedisOSS) está desactivado, los nodos principal y de réplica pueden escalarse verticalmente. Los nodos de réplica también pueden escalar horizontalmente de 0 a 5 nodos. Si el modo de clúster está habilitado, se aplica esto mismo a cada partición del clúster. Además, puede aumentar o reducir el número de particiones.

[Recursos]:

- [Supervisión de las mejores prácticas con ElastiCache \(RedisOSS\) mediante Amazon CloudWatch](#)
 - [Escalado de ElastiCache clústeres \(RedisOSS\)](#)
 - [Escalado ElastiCache para clústeres de Memcached](#)
- [Lo mejor] Supervisar el rendimiento a lo largo del tiempo puede ayudar a detectar cambios en la carga de trabajo que pasarían desapercibidos si se hiciera en un momento determinado. Para detectar tendencias a más largo plazo, utilice CloudWatch métricas para buscar intervalos de

tiempo más largos. Lo aprendido al observar períodos prolongados de CloudWatch métricas debería servir de base para su pronóstico sobre la utilización de los recursos del clúster. CloudWatch los puntos de datos y las métricas están disponibles durante un máximo de 455 días.

[Recursos]:

- [Monitorización ElastiCache \(RedisOSS\) con métricas CloudWatch](#)
 - [Supervisión de Memcached con métricas CloudWatch](#)
 - [Supervisión de las mejores prácticas con ElastiCache \(RedisOSS\) mediante Amazon CloudWatch](#)
- [Lo mejor] Si sus ElastiCache recursos se crean con CloudFormation ella, lo mejor es realizar los cambios mediante CloudFormation plantillas para preservar la coherencia operativa y evitar cambios de configuración no gestionados y desviaciones en la pila.

[Recursos]:

- [ElastiCache referencia de tipo de recurso para CloudFormation](#)
- [Lo mejor] Automatice sus operaciones de escalado utilizando datos operativos del clúster y defina umbrales CloudWatch para configurar las alarmas. Utilice CloudWatch Events y Simple Notification Service (SNS) para activar las funciones de Lambda y ejecutar y escalar ElastiCache API los clústeres automáticamente. Un ejemplo sería añadir una partición al clúster cuando la métrica EngineCPUUtilization alcance el 80 % durante un período de tiempo prolongado. Otra opción sería utilizar DatabaseMemoryUsedPercentages como umbral basado en la memoria.

[Recursos]:

- [Uso de Amazon CloudWatch Alarms](#)
- [¿Qué son los CloudWatch eventos de Amazon?](#)
- [Uso AWS Lambda con Amazon Simple Notification Service](#)
- [ElastiCacheAPIReferencia](#)

OE 3: ¿Cómo administra los recursos ElastiCache del clúster y cómo lo mantiene up-to-date?

Introducción a nivel de preguntas: Al operar a gran escala, es esencial que pueda localizar e identificar todos sus recursos. ElastiCache Al implementar nuevas funciones de la aplicación, debe crear una simetría entre las versiones de los clústeres en todos los tipos de ElastiCache entornos: desarrollo, pruebas y producción. Los atributos de los recursos permiten separar los entornos para

diferentes objetivos operativos, por ejemplo, cuando se implementan nuevas características y habilitan nuevos mecanismos de seguridad.

Ventaja a nivel de pregunta: La separación de los entornos de desarrollo, pruebas y producción es la mejor práctica operativa. También se recomienda aplicar los parches de software más recientes a los clústeres y nodos de todos los entornos mediante procesos bien conocidos y documentados. Aprovechar las ElastiCache funciones nativas permite a su equipo de ingeniería centrarse en cumplir los objetivos empresariales y no en el ElastiCache mantenimiento.

- [Lo mejor] Utilice la última versión del motor disponible y aplique las actualizaciones de autoservicio tan pronto como estén disponibles. ElastiCache actualiza automáticamente su infraestructura subyacente durante el período de mantenimiento especificado del clúster. Sin embargo, los nodos que se ejecutan en sus clústeres se actualizan mediante actualizaciones de autoservicio. Estas actualizaciones pueden ser de dos tipos: parches de seguridad o actualizaciones de software menores. Asegúrese de entender la diferencia entre los tipos de parches y cuándo se aplican.

[Recursos]:

- [Actualizaciones de autoservicio en Amazon ElastiCache](#)
- [Página de ayuda de Amazon ElastiCache Managed Maintenance and Service Updates](#)
- [Lo mejor] Organice sus ElastiCache recursos mediante etiquetas. Utilice las etiquetas en los grupos de replicación y no en los nodos individuales. Puede configurar las etiquetas de modo que se muestren cuando consulte los recursos y puede utilizarlas para realizar búsquedas y aplicar filtros. Debe utilizar los grupos de recursos para crear y mantener fácilmente colecciones de recursos que compartan conjuntos comunes de etiquetas.

[Recursos]:

- [Tagging Best Practices](#)
- [ElastiCache referencia de tipo de recurso para CloudFormation](#)
- [Grupos de parámetros](#)

OE 4: ¿Cómo gestiona las conexiones de los clientes con sus ElastiCache clústeres?

Introducción a nivel de preguntas: Cuando opere a gran escala, debe comprender cómo se conectan sus clientes con el ElastiCache clúster para gestionar los aspectos operativos de las aplicaciones (como los tiempos de respuesta).

Ventaja a nivel de pregunta: Al elegir el mecanismo de conexión más adecuado, se garantiza que la aplicación no se desconecte debido a errores de conectividad, como los tiempos de espera.

- [Obligatorio] Separe las operaciones de lectura de las de escritura y conéctese a los nodos de réplica para ejecutar las operaciones de lectura. Sin embargo, tenga en cuenta que si separa las escrituras de las lecturas, perderá la capacidad de leer una clave inmediatamente después de escribirla debido a la naturaleza asíncrona de la replicación de Redis. OSS El WAIT comando se puede utilizar para mejorar la seguridad de los datos en el mundo real y obligar a las réplicas a reconocer las escrituras antes de responder a los clientes, con un coste global de rendimiento. El uso de nodos de réplica para las operaciones de lectura se puede configurar en su biblioteca cliente ElastiCache (RedisOSS) utilizando el terminal del ElastiCache lector para el modo de clúster desactivado. Para activar el modo de clúster, utilice el comando ElastiCache (RedisOSS). READONLY En muchas de las bibliotecas cliente ElastiCache (RedisOSS), ElastiCache (RedisOSS) READONLY se implementa de forma predeterminada o mediante un ajuste de configuración.

[Recursos]:

- [Búsqueda de puntos de conexión](#)
- [READONLY](#)
- [Obligatorio] Utilice la agrupación de conexiones. El establecimiento de una TCP conexión tiene un coste de CPU tiempo tanto para el cliente como para el servidor, y la agrupación permite reutilizar la conexión. TCP

Para reducir la sobrecarga de conexiones, debe utilizar la agrupación de conexiones. Con un conjunto de conexiones, la aplicación puede reutilizar y liberar conexiones “a voluntad”, sin el costo de establecer la conexión. Puede implementar la agrupación de conexiones a través de su biblioteca de clientes ElastiCache (RedisOSS) (si es compatible), con un marco disponible para su entorno de aplicaciones, o crearlo desde cero.

- [Lo mejor] Asegúrese de que el tiempo de espera del socket del cliente esté establecido en, al menos, un segundo (en lugar del típico valor predeterminado de “ninguno” en numerosos clientes).
 - Si se establece un valor de tiempo de espera demasiado bajo, es posible que se produzcan tiempos de espera cuando la carga del servidor es alta. Si se establece demasiado alto, la aplicación puede tardar mucho en detectar problemas de conexión.
 - Controle el volumen de nuevas conexiones a través de la implementación de la agrupación de conexiones en su aplicación cliente. Esto reduce la latencia y el CPU uso necesarios para abrir y cerrar las conexiones y realizar un TLS apretón de manos si TLS está activado en el clúster.

[Recursos]: [Configure ElastiCache \(RedisOSS\) para una mayor disponibilidad](#)

- [Bueno] El uso de la canalización (cuando los casos de uso lo permitan) puede aumentar significativamente el rendimiento.
 - Con la canalización, se reduce el tiempo de ida y vuelta (RTT) entre los clientes de la aplicación y el clúster, y se pueden procesar nuevas solicitudes incluso si el cliente aún no ha leído las respuestas anteriores.
 - Con la canalización, puede enviar varios comandos al servidor sin esperar a recibir respuestas o confirmaciones. La desventaja de la canalización es que, cuando finalmente se obtienen todas las respuestas de forma masiva, es posible que se haya producido un error que no se detectará hasta el final.
 - Implemente métodos para reintentar las solicitudes cuando se devuelva un error que omita la solicitud incorrecta.

[Recursos]: [Pipelining](#)

OE 5: ¿Cómo se implementan ElastiCache los componentes para una carga de trabajo?

Introducción a nivel de preguntas: ElastiCache los entornos se pueden implementar manualmente a través de la AWS consola o mediante programación mediante kits de herramientas APIs/CLI, etc. Las prácticas recomendadas de excelencia operativa sugieren automatizar las implementaciones mediante código siempre que sea posible. Además, ElastiCache los clústeres pueden aislarse por carga de trabajo o combinarse para optimizar los costes.

Ventaja a nivel de pregunta: elegir el mecanismo de implementación más adecuado para sus ElastiCache entornos puede mejorar la excelencia operativa con el tiempo. Se recomienda realizar operaciones como código siempre que sea posible para minimizar los errores humanos y aumentar la repetibilidad, la flexibilidad y el tiempo de respuesta a los eventos.

Al comprender los requisitos de aislamiento de la carga de trabajo, puede optar por tener ElastiCache entornos dedicados por carga de trabajo o combinar varias cargas de trabajo en clústeres únicos o combinaciones de ambos. Comprender las soluciones de compromiso puede ayudar a lograr un equilibrio entre la excelencia operativa y la optimización de costos.

- [Obligatorio] Conozca las opciones de implementación disponibles y automatice estos procedimientos siempre que sea posible. ElastiCache Las posibles vías de automatización incluyen CloudFormationSDK, AWS CLI/yAPIs.

[Recursos]:

- [Referencia de tipo ElastiCache de recurso de Amazon](#)
- [elasticache](#)
- [ElastiCache APIReferencia de Amazon](#)
- [Obligatorio] Para todas las cargas de trabajo, determine el nivel de aislamiento del clúster necesario.
 - [Lo mejor]: Alto aislamiento: asignación 1:1 de la carga de trabajo al clúster. Permite un control detallado del acceso, el tamaño, el escalado y la administración de ElastiCache los recursos en función de la carga de trabajo.
 - [Mejor] Aislamiento medio: M:1 aislado por propósito, pero quizás compartido entre varias cargas de trabajo (por ejemplo, un clúster dedicado al almacenamiento en caché de las cargas de trabajo y otro dedicado a la mensajería).
 - [Bueno]: Bajo aislamiento: M:1 multiusos, totalmente compartido. Recomendado para cargas de trabajo en las que es aceptable el acceso compartido.

OE 6: ¿Cómo se planifican y mitigan los errores?

Introducción a nivel de preguntas: La excelencia operativa incluye anticipar los fallos mediante la realización de ejercicios periódicos «previos a la autopsia» para identificar las posibles fuentes de fallo y poder eliminarlos o mitigarlos. ElastiCache ofrece una conmutación por error API que permite simular eventos de fallo de nodos con fines de prueba.

Ventaja a nivel de pregunta: Al probar los escenarios de error con antelación, puede averiguar cómo afectan a su carga de trabajo. Esto permite probar de forma segura los procedimientos de respuesta y su eficacia. Además su equipo se familiarizará con su ejecución.

[Obligatorio] Realice pruebas de conmutación por error con regularidad en las cuentas de desarrollo y pruebas. [TestFailover](#)

OE 7: ¿Cómo se solucionan los problemas de los eventos OSS del motor Redis?

Introducción a nivel de preguntas: La excelencia operativa requiere la capacidad de investigar la información tanto a nivel de servicio como a nivel de motor para analizar el estado y el estado de los

clústeres. ElastiCache (RedisOSS) puede emitir registros OSS del motor de Redis tanto a Amazon como a Amazon CloudWatch Kinesis Data Firehose.

Ventaja a nivel de pregunta: la activación de los registros OSS del motor de Redis en los clústeres ElastiCache (RedisOSS) proporciona información sobre los eventos que afectan al estado y al rendimiento de los clústeres. Los registros OSS del motor de Redis proporcionan datos directamente del motor de Redis que no están disponibles a OSS través del mecanismo de eventos. ElastiCache Al observar detenidamente tanto los ElastiCache eventos (consulte la anterior OE-1) como los registros OSS del motor de Redis, es posible determinar el orden de los eventos al solucionar problemas tanto desde la perspectiva del ElastiCache servicio como desde la perspectiva del motor de Redis. OSS

- [Obligatorio] Asegúrese de que la funcionalidad de registro OSS del motor de Redis esté habilitada, que estará disponible a partir de la versión 6.2 ElastiCache (OSSRedis) y versiones posteriores. Esto se puede hacer durante la creación del clúster o modificándolo después de la creación.
 - Determine si Amazon CloudWatch Logs o Amazon Kinesis Data Firehose son el destino adecuado OSS para los registros del motor de Redis.
 - Seleccione un registro de destino adecuado en Kinesis Data Firehose CloudWatch o en Kinesis Data Firehose para conservar los registros. Si tiene varios clústeres, considere la posibilidad de utilizar un registro de destino diferente para cada clúster, ya que esto ayudará a aislar los datos a la hora de solucionar problemas.

[Recursos]:

- Entrega de registros: [Entrega de registro](#)
- Destinos de registro: [Amazon CloudWatch Logs](#)
- Introducción a Amazon CloudWatch Logs: [¿Qué es Amazon CloudWatch Logs?](#)
- Introducción a Amazon Kinesis Data Firehose: [¿Qué es Amazon Kinesis Data Firehose?](#)
- [Lo mejor] Si utiliza Amazon CloudWatch Logs, considere la posibilidad de utilizar Amazon CloudWatch Logs Insights para consultar el registro OSS del motor de Redis para obtener información importante.

Por ejemplo, cree una consulta en el grupo de CloudWatch registros que contiene los registros del OSS motor de Redis y que devolverá eventos con un valor LogLevel de 'WARNING', como:

```
fields @timestamp, LogLevel, Message
| sort @timestamp desc
```

```
| filter LogLevel = "WARNING"
```

[Recursos]: [analizar los datos de registro con CloudWatch Logs Insights](#)

El pilar de seguridad de Amazon ElastiCache Well-Architected Lens

El pilar de seguridad se centra en proteger la información y los sistemas. Los temas clave incluyen la confidencialidad e integridad de los datos, la identificación y administración de quién puede hacer qué con la administración de privilegios, la protección de los sistemas y el establecimiento de controles para detectar eventos de seguridad.

Temas

- [SEC 1: ¿Qué medidas está tomando para controlar el acceso autorizado a los datos? ElastiCache](#)
- [SEC 2: ¿Requieren sus aplicaciones una autorización adicional para ElastiCache superar los controles basados en la red?](#)
- [SEC 3: ¿Existe el riesgo de que los comandos se ejecuten de forma inadvertida y provoquen la pérdida de datos o errores?](#)
- [SEC 4: ¿Cómo se garantiza el cifrado de datos en reposo con ElastiCache](#)
- [SEC 5: ¿Cómo se cifran los datos en tránsito? ElastiCache](#)
- [SEC 6: ¿Cómo se restringe el acceso a los recursos del plano de control?](#)
- [SEC 7: ¿Cómo se detectan los eventos de seguridad y cómo se responde a ellos?](#)

SEC 1: ¿Qué medidas está tomando para controlar el acceso autorizado a los datos? ElastiCache

Introducción a nivel de preguntas: todos los ElastiCache clústeres están diseñados para que se pueda acceder a ellos desde instancias de Amazon Elastic Compute Cloud en una VPC, funciones sin servidor (AWS Lambda) o contenedores (Amazon Elastic Container Service). El escenario más frecuente es acceder a un ElastiCache clúster desde una instancia de Amazon Elastic Compute Cloud dentro de la misma Amazon Virtual Private Cloud (Amazon Virtual Private Cloud). Antes de poder conectarse a un clúster desde una instancia de Amazon EC2, debe autorizar a la instancia de Amazon EC2 el acceso al clúster. Para acceder a un ElastiCache clúster que se ejecuta en una VPC, es necesario conceder la entrada de red al clúster.

Ventaja a nivel de pregunta: El acceso a la red en el clúster se controla mediante grupos de seguridad de VPC. Un grupo de seguridad funciona como un firewall virtual para las instancias de

Amazon EC2 para controlar el tráfico entrante y saliente. Las reglas de entrada controlan el tráfico entrante a la instancia y las reglas de salida controlan el tráfico saliente desde la instancia. En el caso de ElastiCache lanzar un clúster, es necesario asociar un grupo de seguridad. Esto garantiza que haya reglas de tráfico entrante y saliente vigentes para todos los nodos que componen el clúster. Además, ElastiCache está configurado para implementarse exclusivamente en subredes privadas, de modo que solo se puede acceder a ellas a través de la red privada de la VPC.

- [Obligatorio] El grupo de seguridad asociado al clúster controla el acceso a la red y al clúster. De forma predeterminada, un grupo de seguridad no tendrá ninguna regla de entrada definida y, por lo tanto, no tendrá una ruta de entrada a ella. ElastiCache Para habilitarlo, configure una regla de entrada en el grupo de seguridad especificando la dirección o el rango IP de origen, el tráfico de tipo TCP y el puerto del ElastiCache clúster (el puerto predeterminado 6379 para ElastiCache (Redis OSS), por ejemplo). Si bien es posible permitir un conjunto muy amplio de fuentes de entrada, como todos los recursos de una VPC (0.0.0.0/0), se recomienda ser lo más detallado posible a la hora de definir las reglas de entrada, por ejemplo, autorizar únicamente el acceso entrante a los clientes OSS de Redis que se ejecuten en instancias Amazon EC2 de Amazon asociadas a un grupo de seguridad específico.

[Recursos]:

- [Subredes y grupos de subredes](#)
- [Acceso al clúster o al grupo de reproducción](#)
- [Controlar el tráfico hacia los recursos mediante grupos de seguridad](#)
- [Grupos de seguridad de Amazon Elastic Compute Cloud para instancias de Linux](#)
- Se pueden asignar AWS Identity and Access Management políticas [obligatorias] a las funciones que les permiten acceder a los datos. AWS Lambda ElastiCache Para habilitar esta función, cree una función de ejecución de IAM con el `AWSLambdaVPCLambdaAccessExecutionRole` permiso y, a continuación, asigne la función a la AWS Lambda función.

[Recursos]: Configuración de una función de Lambda para acceder a Amazon ElastiCache en una Amazon VPC: [Tutorial: Configuración de una función de Lambda para acceder a Amazon en una Amazon VPC ElastiCache](#)

SEC 2: ¿Requieren sus aplicaciones una autorización adicional para ElastiCache superar los controles basados en la red?

Introducción a nivel de preguntas: en situaciones en las que sea necesario restringir o controlar el acceso a los clústeres ElastiCache (Redis OSS) a nivel de cliente individual, se recomienda autenticarse mediante el comando AUTH ElastiCache (Redis OSS). ElastiCache Los tokens de autenticación (Redis OSS), con la administración opcional de usuarios y grupos de usuarios, permiten que ElastiCache (Redis OSS) requiera una contraseña antes de permitir a los clientes ejecutar comandos y acceder a las claves, lo que mejora la seguridad del plano de datos.

Beneficio a nivel de pregunta: para ayudar a mantener sus datos seguros, ElastiCache (Redis OSS) proporciona mecanismos para protegerlos contra el acceso no autorizado a los datos. Esto incluye aplicar el AUTH o el token de AUTH (contraseña) del control de acceso basado en roles (RBAC) al que deben conectarse los clientes antes de ejecutar comandos autorizados. ElastiCache

- [Lo mejor] Para ElastiCache (Redis OSS) 6.x y versiones posteriores, defina los controles de autenticación y autorización definiendo los grupos de usuarios, los usuarios y las cadenas de acceso. Asigne usuarios a grupos de usuarios y, a continuación, asigne grupos de usuarios a clústeres. Para utilizar el RBAC, este debe seleccionarse al crear el clúster y debe estar habilitado el cifrado en tránsito. Asegúrese de utilizar un cliente OSS de Redis que sea compatible con TLS para poder aprovechar el RBAC.

[Recursos]:

- [Aplicación del RBAC a un grupo de replicación para \(Redis OSS\) ElastiCache](#)
- [Especificación de permisos mediante una cadena de acceso](#)
- [ACL](#)
- [Versiones compatibles ElastiCache \(Redis OSS\)](#)
- [Lo mejor] En el caso de las versiones anteriores a la 6.x ElastiCache (Redis OSS), además de establecer un token o contraseña seguro y mantener una política de contraseñas estricta para la autenticación ElastiCache (Redis OSS), se recomienda rotar la contraseña/token. ElastiCache puede gestionar hasta dos (2) tokens de autenticación en un momento dado. También puede modificar el clúster para que requiera explícitamente el uso de tokens de autenticación.

[Recursos]: [modificar el token AUTH en un clúster existente ElastiCache \(Redis OSS\)](#)

SEC 3: ¿Existe el riesgo de que los comandos se ejecuten de forma inadvertida y provoquen la pérdida de datos o errores?

Introducción a nivel de preguntas: hay varios comandos de Redis OSS que pueden tener un impacto adverso en las operaciones si se ejecutan por error o por parte de actores malintencionados. Estos comandos pueden tener consecuencias no deseadas desde el punto de vista del rendimiento y la seguridad de los datos. Por ejemplo, un desarrollador puede llamar de forma rutinaria al comando FLUSHALL en un entorno de desarrollo y, debido a un error, puede intentar ejecutar este comando sin darse cuenta en un sistema de producción, lo que provoca la pérdida accidental de datos.

Ventaja a modo de pregunta: a partir de ElastiCache (Redis OSS) 5.0.3, podrá cambiar el nombre de ciertos comandos que podrían afectar su carga de trabajo. El cambio del nombre de los comandos puede ayudar a evitar que se ejecuten inadvertidamente en el clúster.

- [Obligatorio]

[Recursos]:

- [ElastiCache \(Redis OSS\) versión 5.0.3 \(obsoleta, utilice la versión 5.0.6\)](#)
- [Cambios en los parámetros de Redis OSS 5.0.3](#)
- [Seguridad de Redis OSS](#)

SEC 4: ¿Cómo se garantiza el cifrado de datos en reposo con ElastiCache

Introducción a nivel de preguntas: Si bien ElastiCache (Redis OSS) es un almacén de datos en memoria, es posible cifrar cualquier dato que pueda persistir (en el almacenamiento) como parte de las operaciones estándar del clúster. Esto incluye las copias de seguridad programadas y manuales escritas en Amazon S3, así como los datos guardados en el almacenamiento en disco como resultado de las operaciones de sincronización e intercambio. Los tipos de instancias de las familias M6g y R6g también cuentan con un cifrado en memoria siempre activo.

Ventaja a nivel de pregunta: ElastiCache (Redis OSS) ofrece un cifrado en reposo opcional para aumentar la seguridad de los datos.

- [Obligatorio] El cifrado en reposo solo se puede habilitar en un ElastiCache clúster (grupo de replicación) cuando se crea. No se puede modificar un clúster existente para empezar a cifrar los datos en reposo. De forma predeterminada, ElastiCache proporcionará y administrará las claves utilizadas en el cifrado en reposo.

[Recursos]:

- [Condiciones del cifrado en reposo](#)
- [Activación del cifrado en reposo](#)
- [Lo mejor] Aproveche los tipos de instancias de Amazon EC2 que cifran los datos mientras están en la memoria (como M6g o R6g). Siempre que sea posible, considere la posibilidad de administrar sus propias claves para el cifrado en reposo. Para entornos de seguridad de datos más estrictos, se puede utilizar AWS Key Management Service (KMS) para gestionar automáticamente las claves maestras del cliente (CMK). Mediante ElastiCache la integración con AWS Key Management Service, puede crear, poseer y administrar las claves utilizadas para el cifrado de los datos en reposo de su clúster ElastiCache (Redis OSS).

[Recursos]:

- [Uso de claves gestionadas por el cliente desde AWS Key Management Service](#)
- [AWS Servicio de administración de claves](#)
- [AWS KMS concepts](#)

SEC 5: ¿Cómo se cifran los datos en tránsito? ElastiCache

Introducción a nivel de pregunta: Es un requisito habitual evitar que los datos corran peligro mientras están en tránsito. Esto representa los datos dentro de los componentes de un sistema distribuido, así como entre los clientes de aplicaciones y los nodos del clúster. ElastiCache (Redis OSS) cumple con este requisito al permitir cifrar los datos en tránsito entre los clientes y el clúster, y entre los propios nodos del clúster. Los tipos de instancias de las familias M6g y R6g también cuentan con un cifrado en memoria siempre activo.

Ventaja a nivel de pregunta: el cifrado ElastiCache en tránsito de Amazon es una función opcional que te permite aumentar la seguridad de tus datos en los puntos más vulnerables, cuando están en tránsito de un lugar a otro.

- [Obligatorio] El cifrado en tránsito solo se puede habilitar en un clúster (grupo de replicación) ElastiCache (Redis OSS) tras su creación. Tenga en cuenta que, debido al procesamiento adicional necesario para cifrar o descifrar datos, la implementación del cifrado en tránsito afectará al rendimiento en cierta medida. Para comprender el impacto, se recomienda comparar la carga de trabajo antes y después de la activación. encryption-in-transit

[Recursos]:

- [Información general sobre el cifrado en tránsito](#)

SEC 6: ¿Cómo se restringe el acceso a los recursos del plano de control?

Introducción a nivel de preguntas: las políticas de IAM y el ARN permiten controles de acceso detallados para ElastiCache (Redis OSS), lo que permite un control más estricto para gestionar la creación, modificación y eliminación de clústeres (Redis OSS). ElastiCache

Beneficio a nivel de pregunta: la administración de ElastiCache los recursos de Amazon, como grupos de replicación, nodos, etc., puede restringirse a AWS cuentas que tengan permisos específicos basados en las políticas de IAM, lo que mejora la seguridad y la confiabilidad de los recursos.

- [Obligatorio] Gestiona el acceso a ElastiCache los recursos de Amazon mediante la asignación de AWS Identity and Access Management políticas específicas a AWS los usuarios, lo que permite controlar mejor qué cuentas pueden realizar determinadas acciones en los clústeres.

[Recursos]:

- [Descripción general de la gestión de los permisos de acceso a sus recursos ElastiCache](#)
- [Uso de políticas basadas en la identidad \(políticas de IAM\) para Amazon ElastiCache](#)

SEC 7: ¿Cómo se detectan los eventos de seguridad y cómo se responde a ellos?

Introducción a nivel de preguntas: cuando se implementa con el RBAC activado ElastiCache, exporta CloudWatch métricas para notificar a los usuarios sobre los eventos de seguridad. Estas métricas ayudan a identificar los intentos fallidos de autenticación, acceso a claves o ejecución de comandos para los que los usuarios con RBAC que se conectan no están autorizados.

Además, AWS los recursos de productos y servicios ayudan a proteger su carga de trabajo general al automatizar las implementaciones y registrar todas las acciones y modificaciones para su posterior revisión o auditoría.

Ventaja a nivel de pregunta: Con la supervisión de los eventos, usted permite a su organización responder de acuerdo con sus requisitos, políticas y procedimientos. La automatización de la supervisión y las respuestas a estos eventos de seguridad refuerza su postura de seguridad general.

- [Obligatorio] Familiarícese con las CloudWatch métricas publicadas relacionadas con los errores de autenticación y autorización del RBAC.

- `AuthenticationFailures` = Intentos fallidos de autenticarse en Redis OSS
- `KeyAuthorizationFailures` = Intentos fallidos de los usuarios de acceder a las claves sin permiso
- `CommandAuthorizationFailures` = Intentos fallidos de los usuarios de ejecutar comandos sin permiso

[Recursos]:

- [Métricas de Redis OSS](#)
- [Lo mejor] Se recomienda configurar alertas y notificaciones para estas métricas y responder según sea necesario.

[Recursos]:

- [Uso de CloudWatch alarmas de Amazon](#)
- [Mejor] Utilice el comando `ACL LOG OSS` de Redis para recopilar más detalles

[Recursos]:

- [ACL LOG](#)
- [Lo mejor] Familiarícese con las capacidades de AWS los productos y servicios en lo que respecta a la supervisión, el registro y el análisis de ElastiCache las implementaciones y los eventos

[Recursos]:

- [Registrar llamadas a ElastiCache la API de Amazon con AWS CloudTrail](#)
- [elasticache-redis-cluster-automatic-verificación de respaldo](#)
- [Monitorear el uso con Metrics CloudWatch](#)

El pilar de confiabilidad de Amazon ElastiCache Well-Architected Lens

Temas

- [REL1: ¿Cómo apoya las implementaciones de arquitecturas de alta disponibilidad \(HA\)?](#)
- [REL2: ¿Cómo está cumpliendo sus objetivos de puntos de recuperación \(RPOs\)? ElastiCache](#)
- [REL3: ¿Cómo se cumplen los requisitos de recuperación ante desastres \(DR\)?](#)
- [REL4: ¿Cómo se planifican eficazmente las conmutaciones por error?](#)
- [REL5: ¿Sus ElastiCache componentes están diseñados para ser escalables?](#)

REL1: ¿Cómo apoya las implementaciones de arquitecturas de alta disponibilidad (HA)?

Introducción a nivel de preguntas: comprender la arquitectura de alta disponibilidad de Amazon ElastiCache le permitirá operar en un estado resiliente durante los eventos de disponibilidad.

Ventaja a nivel de pregunta: Diseñar la arquitectura de sus ElastiCache clústeres para que sean resistentes a los fallos garantiza una mayor disponibilidad para sus despliegues. ElastiCache

- [Obligatorio] Determine el nivel de confiabilidad que necesita para su clúster. ElastiCache Las diferentes cargas de trabajo tienen diferentes estándares de resiliencia, desde cargas de trabajo totalmente efímeras hasta cargas de trabajo cruciales para la misión. Defina las necesidades de cada tipo de entorno en el que opere, como desarrollo, pruebas y producción.

Motor de almacenamiento en caché: ElastiCache (Memcached) frente ElastiCache a (Redis) OSS

1. ElastiCache (Memcached) no proporciona ningún mecanismo de replicación y se utiliza principalmente para cargas de trabajo efímeras.
 2. ElastiCache (RedisOSS) ofrece las funciones de alta disponibilidad que se describen a continuación
- [Lo mejor] Para las cargas de trabajo que requieren alta disponibilidad, utilice ElastiCache (RedisOSS) en modo clúster con un mínimo de dos réplicas por partición, incluso para cargas de trabajo pequeñas que requieran un rendimiento reducido y que solo requieran una partición.
 1. Si el modo de clúster está activado, multi-AZ se habilita automáticamente.

Multi-AZ minimiza el tiempo de inactividad, ya que realiza conmutaciones por error automáticas desde el nodo principal a las réplicas, en caso de mantenimiento planificado o no planificado, y mitiga los errores de AZ.

2. Para las cargas de trabajo fragmentadas, un mínimo de tres particiones proporciona una recuperación más rápida durante los eventos de conmutación por error, ya que el protocolo de OSS clústeres de Redis requiere que la mayoría de los nodos principales estén disponibles para alcanzar el quórum.
3. Configure dos o más réplicas para ofrecer disponibilidad.

Tener dos réplicas proporciona escalabilidad de lectura mejorada así como disponibilidad de lectura en situaciones en las que una réplica está en mantenimiento.

4. Utilice tipos de nodos basados en Graviton2 (nodos predeterminados en la mayoría de las regiones).

ElastiCache (RedisOSS) ha agregado un rendimiento optimizado en estos nodos. Como resultado, obtiene un mejor rendimiento de replicación y sincronización, lo que se traduce en una mejora general de la disponibilidad.

- Supervise y ajuste el tamaño adecuado para hacer frente a los picos de tráfico previstos: cuando hay mucha carga, el motor ElastiCache (RedisOSS) puede dejar de responder, lo que afecta a la disponibilidad. BytesUsedForCache y DatabaseMemoryUsagePercentage son buenos indicadores del uso de la memoria, mientras que ReplicationLag son un indicador del estado de la replicación en función de la velocidad de escritura. Puede usar estas métricas para activar el escalado de clústeres.
- Garantice la resiliencia del lado del cliente realizando pruebas con la conmutación por error [APIantes de que se produzca una conmutación por error de producción](#).

[Recursos]:

- [Configure ElastiCache \(Redis\) para una mayor disponibilidad OSS](#)
- [Alta disponibilidad a través de grupos de reproducción](#)

REL2: ¿Cómo está cumpliendo sus objetivos de puntos de recuperación (RPOs)? ElastiCache

Introducción a nivel de preguntas: comprenda la carga de trabajo RPO para fundamentar las decisiones sobre las estrategias ElastiCache de respaldo y recuperación.

Ventaja a nivel de pregunta: contar con una RPO estrategia implementada puede mejorar la continuidad del negocio en caso de que se produzcan escenarios de recuperación ante desastres. Diseñar sus políticas de respaldo y restauración puede ayudarlo a cumplir sus objetivos de puntos de recuperación (RPO) para sus datos. ElastiCache ElastiCache (RedisOSS) ofrece capacidades de instantáneas que se almacenan en Amazon S3, junto con una política de retención configurable. Estas instantáneas se toman durante un período de copia de seguridad definido y el servicio las gestiona automáticamente. Si la carga de trabajo requiere un grado de detalle de respaldo adicional, tiene la opción de crear hasta 20 copias de seguridad manuales por día. Las copias de seguridad creadas manualmente no tienen una política de retención de servicios y se pueden conservar indefinidamente.

- [Obligatorio] Comprenda y RPO documente sus ElastiCache implementaciones.
 - Tenga en cuenta que Memcached no ofrece ningún proceso de copia de seguridad.
 - Revise las capacidades de las funciones de ElastiCache Backup and Restore.

- [Lo mejor] Disponga de un proceso que se haya comunicado bien para hacer copias de seguridad de su clúster.
 - Inicie copias de seguridad manuales según sea necesario.
 - Revise las políticas de retención de las copias de seguridad automáticas.
 - Tenga en cuenta que las copias de seguridad manuales se conservarán indefinidamente.
 - Programe las copias de seguridad automáticas durante los períodos de bajo uso.
 - Realice operaciones de respaldo contra réplicas de lectura para asegurarse de minimizar el impacto en el rendimiento del clúster.
- [Bueno] Aproveche la función de copia de seguridad programada ElastiCache para realizar copias de seguridad periódicas de sus datos durante un período definido.
 - Haga pruebas periódicas de las restauraciones a partir de las copias de seguridad.
- [Recursos]:
 - [Redis OSS](#)
 - [Backup y restauración para ElastiCache \(RedisOSS\)](#)
 - [Realización de copias de seguridad manuales](#)
 - [Programación de copias de seguridad automáticas](#)
 - [Clústeres de Backup and Restore ElastiCache \(RedisOSS\)](#)

REL3: ¿Cómo se cumplen los requisitos de recuperación ante desastres (DR)?

Introducción a nivel de preguntas: La recuperación ante desastres es un aspecto importante de cualquier planificación de la carga de trabajo. ElastiCache (RedisOSS) ofrece varias opciones para implementar la recuperación ante desastres en función de los requisitos de resiliencia de la carga de trabajo. Con Amazon ElastiCache Global Datastore, puede escribir en su clúster ElastiCache (RedisOSS) de una región y tener los datos disponibles para leerlos desde otros dos clústeres de réplicas entre regiones, lo que permite lecturas de baja latencia y recuperación ante desastres en todas las regiones.

Ventaja a nivel de pregunta: Comprender y planificar una variedad de escenarios de desastres contribuye a garantizar la continuidad del negocio. Las estrategias de DR deben equilibrarse en relación al costo, el impacto en el rendimiento y la posible pérdida de datos.

- [Obligatorio] Desarrolle y documente estrategias de recuperación ante desastres para todos sus ElastiCache componentes en función de los requisitos de carga de trabajo. ElastiCache tiene la particularidad de que algunos casos de uso son totalmente efímeros y no requieren

ninguna estrategia de DR, mientras que otros se encuentran en el extremo opuesto del espectro y requieren una estrategia de DR extremadamente sólida. Todas las opciones deben sopesarse teniendo en cuenta la optimización de costos: una mayor resiliencia requiere más infraestructura.

Conozca las opciones de DR disponibles a nivel regional y multirregional.

- Se recomiendan implementaciones multi-AZ para evitar errores de AZ. Asegúrese de realizar la implementación con el modo clúster habilitado en arquitecturas Multi-AZ, con un mínimo de 3 disponibles. AZs
- Se recomienda utilizar Global Datastore para protegerse contra los errores regionales.
- [Lo mejor] Habilite Global Datastore para cargas de trabajo que requieran resiliencia a nivel de región.
 - Tenga un plan para realizar una conmutación por error a la región secundaria en caso de degradación de la primaria.
 - Pruebe el proceso de conmutación por error multirregional antes de llevar a cabo una conmutación por error en producción.
 - Supervise la métrica `ReplicationLag` para conocer el posible impacto de la pérdida de datos durante los eventos de conmutación por error.
- [Recursos]:
 - [Mitigación de errores](#)
 - [Replicación entre regiones mediante almacenes de datos globales AWS](#)
 - [Restauración desde una copia de seguridad con redimensionamiento opcional del clúster](#)
 - [Minimizar el tiempo de inactividad en ElastiCache \(RedisOSS\) con Multi-AZ](#)

REL4: ¿Cómo se planifican eficazmente las conmutaciones por error?

Introducción a nivel de preguntas: La mejor práctica es habilitar Multi-AZ con conmutaciones por error automáticas. ElastiCache En algunos casos, ElastiCache (RedisOSS) reemplaza los nodos principales como parte de las operaciones de servicio. Algunos ejemplos son eventos de mantenimiento planificado y el improbable caso de que haya un problema en el nodo o en la zona de disponibilidad. El éxito de las conmutaciones por error depende tanto ElastiCache de la configuración de la biblioteca cliente como de la configuración de la biblioteca.

Ventaja a nivel de pregunta: seguir las prácticas recomendadas para las ElastiCache conmutaciones por error junto con su biblioteca cliente específica ElastiCache (RedisOSS) le ayuda a minimizar el posible tiempo de inactividad durante las conmutaciones por error.

- [Obligatorio] Si el modo de clúster está deshabilitado, utilice los tiempos de espera para que sus clientes detecten si es necesario desconectarse del nodo principal anterior y volver a conectarse al nuevo nodo principal, mediante la dirección IP del punto de conexión principal actualizada. Si el modo de clúster está habilitado, la biblioteca cliente es responsable de detectar los cambios en la topología del clúster subyacente. La mayoría de las veces, esto se logra mediante los ajustes de configuración de la biblioteca cliente ElastiCache (RedisOSS), que también permiten configurar la frecuencia y el método de actualización. Cada biblioteca cliente ofrece su propia configuración. Encontrará más información en la documentación correspondiente.

[Recursos]:

- [Minimizar el tiempo de inactividad en ElastiCache \(RedisOSS\) con Multi-AZ](#)
- Revise las prácticas recomendadas de su biblioteca de clientes ElastiCache (RedisOSS).
- [Obligatorio] Las conmutaciones por error correctas dependen de un entorno de replicación en buen estado entre los nodos principal y de réplica. Revise y comprenda la naturaleza asíncrona de la OSS replicación de Redis, así como las CloudWatch métricas disponibles para informar sobre el retraso en la replicación entre los nodos principal y de réplica. Para los casos de uso que requieren una mayor seguridad de los datos, utilice el OSS WAIT comando de Redis para obligar a las réplicas a reconocer las escrituras antes de responder a los clientes conectados.

[Recursos]:

- [Métricas de Redis OSS](#)
- [Supervisión de las mejores prácticas con ElastiCache \(RedisOSS\) mediante Amazon CloudWatch](#)
- [Lo mejor] Valide periódicamente la capacidad de respuesta de su aplicación durante la conmutación por error mediante la ElastiCache conmutación por error de prueba. API

[Recursos]:

- [Prueba de la conmutación por error automática a una réplica de lectura en Amazon ElastiCache \(OSSRedis\)](#)
- [Prueba de la conmutación por error automática](#)

REL5: ¿Sus ElastiCache componentes están diseñados para ser escalables?

Introducción a nivel de preguntas: Al comprender las capacidades de escalado y las topologías de implementación disponibles, sus ElastiCache componentes se pueden ajustar con el tiempo para

adaptarse a los cambiantes requisitos de carga de trabajo. ElastiCache ofrece un escalado en cuatro direcciones: entrada/salida (horizontal) y arriba/abajo (vertical).

Ventaja a nivel de preguntas: seguir las mejores prácticas para las ElastiCache implementaciones proporciona la mayor flexibilidad de escalado, además de cumplir con el principio bien diseñado de escalar horizontalmente para minimizar el impacto de los fallos.

- [Obligatorio] Comprenda la diferencia entre las topologías de modo clúster habilitado y modo clúster deshabilitado. En casi todos los casos, se recomienda realizar la implementación con el modo de clúster habilitado, ya que permite una mayor escalabilidad a lo largo del tiempo. Los componentes en modo clúster deshabilitado tienen una capacidad limitada para escalar horizontalmente mediante la adición de réplicas de lectura.
- [Obligatorio] Comprenda cuándo y cómo escalar.
 - Para obtener más información: añada réplicas READIOPS
 - Para obtener más información WRITEOPS: añadir fragmentos (escalarlos)
 - Para más E/S de red: utilice instancias optimizadas para la red y escale verticalmente
- [Lo mejor] Implemente sus ElastiCache componentes con el modo clúster activado, optando por más nodos más pequeños en lugar de tener menos nodos más grandes. Esto limita de manera efectiva el radio de acción de un error de nodo.
- [Lo mejor] Incluya réplicas en los clústeres para mejorar la capacidad de respuesta durante los eventos de escalado.
- [Bueno] Si el modo de clúster está desactivado, aproveche las réplicas de lectura para aumentar la capacidad de lectura general. ElastiCache tiene deshabilitada la compatibilidad con hasta 5 réplicas de lectura en el modo de clúster, así como el escalado vertical.
- [Recursos]:
 - [Escalado de clústeres ElastiCache \(Redis\) OSS](#)
 - [Escalado vertical en línea](#)
 - [Escalado ElastiCache para clústeres de Memcached](#)

Pilar de eficiencia en el ElastiCache rendimiento de Amazon Well-Architected Lens

El pilar de eficiencia del rendimiento se centra en el uso eficiente de recursos informáticos y de TI. Los temas clave incluyen la selección de los tipos y tamaños de recursos correctos en función de los

requisitos de la carga de trabajo, la supervisión del rendimiento y la toma de decisiones informadas para mantener la eficiencia a medida que evolucionan las necesidades empresariales.

Temas

- [PE 1: ¿Cómo monitoriza el rendimiento de su ElastiCache clúster de Amazon?](#)
- [PE 2: ¿Cómo distribuye el trabajo entre los nodos de su ElastiCache clúster?](#)
- [PE 3: En el caso del almacenamiento en caché de las cargas de trabajo, ¿cómo se realiza el seguimiento de la eficacia y el rendimiento de la memoria caché y se informa al respecto?](#)
- [PE 4: ¿Cómo optimiza la carga de trabajo el uso de los recursos y las conexiones de red?](#)
- [PE 5: ¿Cómo se gestiona la eliminación o la expulsión de claves?](#)
- [PE 6: ¿Cómo se modelan los datos e interactúa con ellos ElastiCache?](#)
- [PE 7: ¿Cómo se registran los comandos de ejecución lenta en el ElastiCache clúster de Amazon?](#)
- [PE8: ¿Cómo ayuda Auto Scaling a aumentar el rendimiento del ElastiCache clúster?](#)

PE 1: ¿Cómo monitoriza el rendimiento de su ElastiCache clúster de Amazon?

Introducción a nivel de pregunta: Al comprender las métricas de supervisión existentes, puede identificar la utilización actual. La supervisión adecuada puede ayudar a identificar posibles obstáculos que afecten al rendimiento de su clúster.

Ventaja a nivel de pregunta: Comprender las métricas asociadas a su clúster puede ayudar a guiar las técnicas de optimización que pueden reducir la latencia y aumentar el rendimiento.

- [Obligatorio] Pruebas de rendimiento de referencia con un subconjunto de la carga de trabajo.
 - Debe supervisar el rendimiento de la carga de trabajo real mediante mecanismos como las pruebas de carga.
 - Supervisa las CloudWatch métricas mientras realizas estas pruebas para comprender mejor las métricas disponibles y establecer una línea base de rendimiento.
- [Lo mejor] En el caso de las cargas de trabajo ElastiCache (de RedisOSS), cambie el nombre de los comandos que son costosos desde el punto de vista computacional, por ejemplo KEYS, para limitar la capacidad de los usuarios de ejecutar comandos de bloqueo en los clústeres de producción.
 - ElastiCache Las cargas de trabajo (RedisOSS) que ejecutan el motor 6.x pueden aprovechar el control de acceso basado en roles para restringir determinados comandos. El acceso a los

comandos se puede controlar creando usuarios y grupos de usuarios con la AWS consola o CLI asociando los grupos de usuarios a un clúster (Redis). ElastiCache OSS En Redis OSS 6, cuando RBAC está activado, podemos usar «- @dangerous" y no permitirá el uso de comandos caros como KEYS,, MONITORSORT, etc. para ese usuario.

- Para la versión 5.x del motor, cambie el nombre de los comandos mediante el `rename-commands` parámetro del grupo de parámetros del clúster ElastiCache (RedisOSS).
- [Mejor] Analice las consultas lentas y busque técnicas de optimización.
 - En el ElastiCache caso de las cargas de trabajo (de RedisOSS), analiza el registro lento para obtener más información sobre tus consultas. Por ejemplo, puede utilizar el siguiente comando, `redis-cli slowlog get 10`, para mostrar los últimos 10 comandos que superaron los umbrales de latencia (10 segundos de forma predeterminada).
 - Algunas consultas se pueden realizar de manera más eficiente utilizando estructuras de datos complejas ElastiCache (RedisOSS). Por ejemplo, en el caso de las búsquedas de rangos de estilos numéricos, una aplicación puede implementar índices numéricos simples con conjuntos ordenados. La administración de estos índices puede reducir los escaneos realizados en el conjunto de datos y devolver los datos con una mayor eficiencia de rendimiento.
 - Para las cargas de trabajo ElastiCache (RedisOSS), `redis-benchmark` proporciona una interfaz sencilla para probar el rendimiento de diferentes comandos mediante entradas definidas por el usuario, como el número de clientes y el tamaño de los datos.
 - Dado que Memcached solo admite comandos simples a nivel de clave, considere la posibilidad de crear claves adicionales como índices para evitar iterar a través del espacio de claves para atender las consultas de los clientes.
- [Recursos]:
 - [Supervise el uso con métricas CloudWatch](#)
 - [Monitorear el uso con CloudWatch métricas](#)
 - [Uso de CloudWatch alarmas de Amazon](#)
 - [Parámetros específicos de Redis](#)
 - [SLOWLOG](#)
 - [Punto de referencia de Redis OSS](#)

PE 2: ¿Cómo distribuye el trabajo entre los nodos de su ElastiCache clúster?

Introducción a nivel de preguntas: La forma en que su aplicación se conecta a ElastiCache los nodos de Amazon puede afectar al rendimiento y la escalabilidad del clúster.

Ventaja a nivel de pregunta: Al utilizar de forma adecuado los nodos disponibles en el clúster garantizará que el trabajo se distribuya entre los recursos disponibles. Las siguientes técnicas también ayudan a evitar que haya recursos inactivos.

- [Obligatorio] Haga que los clientes se conecten al punto final adecuado. ElastiCache
 - ElastiCache (RedisOSS) implementa diferentes puntos finales según el modo de clúster que se utilice. Si el modo de clúster está habilitado, ElastiCache proporcionará un punto final de configuración. Para el modo de clúster desactivado, ElastiCache proporciona un punto final principal, que normalmente se utiliza para las escrituras, y un punto final de lectura para equilibrar las lecturas entre las réplicas. La implementación correcta de estos puntos de conexión resultará en un mejor rendimiento y en operaciones de escalado más sencillas. Evite conectarse a puntos de conexión de nodos individuales a menos que haya un requisito específico para hacerlo.
 - Para los clústeres de Memcached de varios nodos, ElastiCache proporciona un punto final de configuración que permite la detección automática. Se recomienda utilizar un algoritmo de hash para distribuir el trabajo de manera uniforme entre los nodos de caché. Muchas bibliotecas cliente de Memcached implementan un hash coherente. Consulte la documentación de la biblioteca que va a utilizar para ver si admite el uso consistente de hash y saber cómo implementarlo. Encontrará más información sobre la implementación de estas funciones [aquí](#).
- [Mejor] Aproveche el modo de clúster ElastiCache (RedisOSS) habilitado para mejorar la escalabilidad.
 - ElastiCache Los clústeres (RedisOSS) (habilitados para el modo clúster) admiten [operaciones de escalado en línea](#) (saliente/entrada y arriba/abajo) para ayudar a distribuir los datos de forma dinámica entre los fragmentos. El uso del punto de conexión de configuración garantizará que los clientes compatibles con clústeres puedan adaptarse a los cambios en la topología del clúster.
 - También puede reequilibrar el clúster moviendo las ranuras hash entre los fragmentos disponibles en su clúster (Redis ElastiCache) (habilitado para el modo de clúster). OSS Esto ayuda a distribuir el trabajo de manera más eficiente entre las particiones disponibles.
- [Mejor] Implemente una estrategia para identificar y corregir las teclas de acceso rápido de su carga de trabajo.
 - Tenga en cuenta el impacto de las estructuras de OSS datos multidimensionales de Redis, como listas, flujos, conjuntos, etc. Estas estructuras de datos se almacenan en OSS claves de Redis únicas, que residen en un solo nodo. Una clave multidimensional muy grande tiene el potencial de utilizar más capacidad de red y memoria que otros tipos de datos y puede provocar un uso

desproporcionado de ese nodo. Si es posible, diseñe la carga de trabajo de modo que distribuya el acceso a los datos entre varias claves discretas.

- Las teclas de acceso rápido de la carga de trabajo pueden influir en el rendimiento del nodo en uso. Para las cargas de trabajo ElastiCache (de RedisOSS), puede detectar las teclas de acceso rápido `redis-cli --hotkeys` si existe una política de LFU memoria máxima.
- Considere la posibilidad de replicar las teclas de acceso rápido en varios nodos para distribuir el acceso a ellas de manera más uniforme. Este enfoque requiere que el cliente escriba en varios nodos principales (el OSS nodo de Redis en sí no proporcionará esta funcionalidad) y que mantenga una lista de nombres clave para leer, además del nombre de clave original.
- El ElastiCache [La versión 6 de \(RedisOSS\) admite el almacenamiento en caché del lado del cliente asistido por el servidor](#). Esto permite a las aplicaciones esperar a que se modifique una clave antes de volver a realizar llamadas de red a ella. ElastiCache
- [Recursos]:
 - [Configure ElastiCache \(RedisOSS\) para una mayor disponibilidad](#)
 - [Búsqueda de puntos de conexión](#)
 - [Procedimientos recomendados para el equilibrio de cargas](#)
 - [Refragmentación y reequilibrio de particiones en línea para Redis OSS \(modo de clúster habilitado\)](#)
 - [Almacenamiento en caché del lado del cliente en Redis OSS](#)

PE 3: En el caso del almacenamiento en caché de las cargas de trabajo, ¿cómo se realiza el seguimiento de la eficacia y el rendimiento de la memoria caché y se informa al respecto?

Introducción a nivel de preguntas: El almacenamiento en caché es una carga de trabajo habitual ElastiCache y es importante que comprenda cómo gestionar la eficacia y el rendimiento de la memoria caché.

Ventaja a nivel de pregunta: Es posible que su aplicación muestre signos de un rendimiento lento. La capacidad de utilizar métricas específicas de la memoria caché para tomar una decisión sobre cómo aumentar el rendimiento de la aplicación es fundamental para la carga de trabajo de la memoria caché.

- [Obligatorio] Mida y realice un seguimiento a lo largo del tiempo de la proporción de aciertos de la caché. La eficiencia de la memoria caché está determinada por su proporción de aciertos de la

caché. La proporción de aciertos de la caché es el total de aciertos de caché dividido por el total de aciertos y errores. Cuanto más se acerque a 1 esté la proporción, más eficaz será la memoria caché. Una baja proporción de aciertos de caché se debe al volumen de errores de caché. Los errores de caché se producen cuando la clave solicitada no se encuentra en la memoria caché. Una clave no está en la memoria caché porque ha sido expulsada o eliminada, ha caducado o no ha existido nunca. Comprenda por qué las claves no están en la memoria caché y desarrolle las estrategias adecuadas para tenerlas en la memoria caché.

[Recursos]:

- [Métricas para Redis OSS](#)
- [Obligatorio] Mida y recopile el rendimiento de la caché de su aplicación junto con los valores de latencia y CPU utilización para saber si necesita realizar ajustes en sus componentes time-to-live o en otros componentes de la aplicación. ElastiCache proporciona un conjunto de CloudWatch métricas para las latencias agregadas de cada estructura de datos. Estas métricas de latencia se calculan mediante la estadística commandstats del INFO comando ElastiCache (RedisOSS) y no incluyen la red ni el tiempo de E/S. Este es solo el tiempo que tarda ElastiCache (RedisOSS) en procesar las operaciones.

[Recursos]:

- [Métricas para Redis OSS](#)
- [Supervisión de las mejores prácticas con ElastiCache \(RedisOSS\) mediante Amazon CloudWatch](#)
- [Lo mejor] Elija la estrategia de almacenamiento en caché adecuada para sus necesidades. Una baja proporción de aciertos de caché se debe al volumen de errores de caché. Si su carga de trabajo está diseñada para tener un bajo volumen de errores de caché (como comunicación en tiempo real), es mejor revisar las estrategias de almacenamiento en caché y aplicar las resoluciones más adecuadas para la carga de trabajo, como la instrumentación de consultas para medir la memoria y el rendimiento. Las estrategias reales que implemente para completar y mantener su caché dependen del tipo de datos que sus clientes necesiten almacenar en la caché, así como de los patrones de acceso a dichos datos. Por ejemplo, es poco probable que utilice la misma estrategia tanto para las recomendaciones personalizadas en una aplicación de streaming como para las noticias más populares.

[Recursos]:

- [Estrategias de almacenamiento en caché](#)
- [Caching Best Practices](#)

- [Rendimiento a gran escala con Amazon ElastiCache Whitepaper](#)

PE 4: ¿Cómo optimiza la carga de trabajo el uso de los recursos y las conexiones de red?

Introducción a nivel de preguntas: Muchos clientes de aplicaciones admiten ElastiCache (RedisOSS) y ElastiCache (Memcached) y las implementaciones pueden variar. Debe comprender la administración de redes y conexiones implementada para analizar el posible impacto en el rendimiento.

Ventaja a nivel de pregunta: El uso eficiente de los recursos de red puede mejorar la eficiencia del rendimiento de su clúster. Las siguientes recomendaciones pueden reducir las demandas de red y mejorar la latencia y el rendimiento del clúster.

- [Obligatorio] Administre de forma proactiva las conexiones a su clúster. ElastiCache
 - La agrupación de conexiones en la aplicación reduce la sobrecarga del clúster que se crea al abrir y cerrar conexiones. Supervise el comportamiento de la conexión en Amazon CloudWatch mediante `CurConnections` y `NewConnections`.
 - Cierre correctamente las conexiones del cliente cuando corresponda para evitar fugas de conexiones. Las estrategias de administración de conexiones incluyen cerrar correctamente las conexiones que no están en uso y establecer tiempos de espera de conexión.
 - Para las cargas de trabajo de Memcached, hay una cantidad configurable de memoria reservada para gestionar las conexiones denominada `memcached_connections_overhead`.
- [Mejor] Comprima objetos grandes para reducir la memoria y mejorar el rendimiento de la red.
 - La compresión de datos puede reducir la cantidad de rendimiento de red requerida (Gbps), pero aumenta la cantidad de trabajo de la aplicación para comprimir y descomprimir datos.
 - La compresión también reduce la cantidad de memoria que consumen las claves
 - En función de las necesidades de su aplicación, considere las ventajas y desventajas entre la relación y la velocidad de compresión.
- [Recursos]:
 - [ElastiCache \(RedisOSS\): almacén de datos global](#)
 - [Parámetros específicos de Memcached](#)
 - [ElastiCache \(RedisOSS\) 5.0.3 mejora el manejo de E/S para aumentar el rendimiento](#)
 - [Métricas para Redis OSS](#)

- [Configure ElastiCache \(RedisOSS\) para una mayor disponibilidad](#)

PE 5: ¿Cómo se gestiona la eliminación o la expulsión de claves?

Introducción a nivel de preguntas: las cargas de trabajo tienen requisitos y comportamientos esperados diferentes cuando un nodo del clúster se acerca a los límites de consumo de memoria. ElastiCache (RedisOSS) tiene diferentes políticas para gestionar estas situaciones.

Ventaja a nivel de pregunta: La gestión adecuada de la memoria disponible y la comprensión de las políticas de expulsión ayudarán a garantizar el conocimiento del comportamiento del clúster cuando se superen los límites de memoria de las instancias.

- [Obligatorio] Analice el acceso a los datos para evaluar qué política aplicar. Identifique una política de memoria máxima adecuada para controlar si se realizan expulsiones en el clúster y cómo se hacen.
 - La expulsión tiene lugar cuando se consume la cantidad máxima de memoria del clúster y existe una política que permite la expulsión. El comportamiento del clúster en esta situación depende de la política de expulsión especificada. Esta política se puede administrar mediante el grupo `maxmemory-policy` de parámetros del clúster ElastiCache (RedisOSS).
 - La política predeterminada `volatile-lru` libera memoria al desalojar las claves con un tiempo de caducidad (valor) establecido. TTL Las políticas utilizadas con menos frecuencia (LFU) y las utilizadas con menos frecuencia (LRU) eliminan las claves en función del uso.
 - Para las cargas de trabajo de Memcached, existe una LRU política predeterminada que controla los desalojos en cada nodo. El número de desalojos de tu ElastiCache clúster de Amazon se puede monitorizar mediante la métrica de desalojos de Amazon. CloudWatch
- [Mejor] Estandarice el comportamiento de eliminación para controlar el impacto en el rendimiento de su clúster y evitar atascos inesperados en el rendimiento.
 - En el ElastiCache caso de las cargas de trabajo (RedisOSS), cuando se eliminan las claves del clúster de forma explícita, `UNLINK` se eliminan las claves especificadas. `DEL` Sin embargo, el comando realiza la recuperación de memoria real en un subproceso diferente, por lo que no es de bloqueo, mientras que `DEL` sí. La eliminación real se realizará más adelante de forma asincrónica.
 - Para las cargas de trabajo de ElastiCache (RedisOSS) 6.x, el comportamiento del `DEL` comando se puede modificar en el grupo de parámetros mediante el parámetro. `lazyfree-lazy-user-del`
- [Recursos]:

- [Configuración de los parámetros de motor mediante los grupos de parámetros](#)
- [UNLINK](#)
- [Gestión financiera en la nube con AWS](#)

PE 6: ¿Cómo se modelan los datos e interactúa con ellos ElastiCache?

Introducción a nivel de preguntas: ElastiCache la aplicación depende en gran medida de las estructuras de datos y del modelo de datos utilizado, pero también debe tener en cuenta el banco de datos subyacente (si está presente). Comprenda las estructuras de datos ElastiCache (RedisOSS) disponibles y asegúrese de utilizar las estructuras de datos más adecuadas para sus necesidades.

Ventaja a nivel de pregunta: el modelado de datos ElastiCache tiene varios niveles, que incluyen el caso de uso de la aplicación, los tipos de datos y las relaciones entre los elementos de datos. Además, cada comando y tipo de datos ElastiCache (RedisOSS) tiene sus propias características de rendimiento bien documentadas.

- [Lo mejor] Una práctica recomendada es reducir la sobrescritura involuntaria de datos. Utilice una convención de nomenclatura que minimice la superposición de nombres de clave. La nomenclatura convencional de las estructuras de datos utiliza un método jerárquico como: APPNAME:CONTEXT:ID, por ejemplo ORDER-APP:CUSTOMER:123.

[Recursos]:

- [Key naming](#)
- Los [mejores] comandos ElastiCache (RedisOSS) tienen una complejidad temporal definida por la notación Big O. Esta complejidad temporal de un comando es una representación algorítmica/ matemática de su impacto. Al introducir un nuevo tipo de datos en su aplicación, debe revisar detenidamente la complejidad temporal de los comandos relacionados. Los comandos con una complejidad temporal de $O(1)$ son constantes y no dependen de la cantidad de datos de entrada.; sin embargo, los comandos con una complejidad temporal de $O(N)$ son lineales y están sujetos a la cantidad de datos de entrada. Debido al diseño de un solo subproceso de ElastiCache (RedisOSS), un gran volumen de operaciones de alta complejidad temporal se traducirá en un menor rendimiento y posibles tiempos de espera de operación.

[Recursos]:

- [Comandos](#)
- [Ideal] Úselo APIs para obtener GUI visibilidad del modelo de datos de su clúster.

[Recursos]:

- [Redis Commander OSS](#)
- [Navegador Redis OSS](#)
- [Redsmin](#)

PE 7: ¿Cómo se registran los comandos de ejecución lenta en el ElastiCache clúster de Amazon?

Introducción a nivel de pregunta: Ventajas del ajuste del rendimiento mediante la captura, la agregación y la notificación de comandos de ejecución prolongada. Si comprende cuánto tiempo tardan en ejecutarse los comandos, puede determinar qué comandos tienen un rendimiento deficiente y qué comandos impiden que el motor funcione de manera óptima. ElastiCache (RedisOSS) también tiene la capacidad de reenviar esta información a Amazon CloudWatch o Amazon Kinesis Data Firehose.

Ventaja a nivel de pregunta: El registro en una ubicación permanente dedicada y la provisión de eventos de notificación para los comandos lentos puede ayudar a realizar un análisis detallado del rendimiento y se puede utilizar para activar eventos automatizados.

- [Obligatorio] Amazon ElastiCache (RedisOSS) ejecuta el motor versión 6.0 o posterior, el grupo de parámetros correctamente configurado y el SLOWLOG registro está habilitado en el clúster.
 - Los parámetros necesarios solo están disponibles cuando la compatibilidad de las versiones del motor está configurada en la OSS versión 6.0 o superior de Redis.
 - SLOWLOG el registro se produce cuando el tiempo de ejecución de un comando en el servidor supera un valor especificado. El comportamiento del clúster depende de los parámetros del grupo de parámetros asociado, que son `slowlog-log-slower-than` y `slowlog-max-len`.
 - Los cambios surten efecto inmediatamente.
- [Lo mejor] Aproveche nuestras capacidades de CloudWatch Kinesis Data Firehose.
 - Utilice las capacidades de filtrado y alarma de CloudWatch CloudWatch Logs Insights y Amazon Simple Notification Services para lograr la supervisión del rendimiento y la notificación de eventos.
 - Utilice las funciones de streaming de Kinesis Data Firehose SLOWLOG para archivar los registros en un almacenamiento permanente o para activar el ajuste automatizado de los parámetros del clúster.

- Determine si JSON el TEXT formato simple se adapta mejor a sus necesidades.
- Proporcione IAM permisos para publicar en CloudWatch o en Kinesis Data Firehose.
- [Mejor] Configure `slowlog-log-slower-than` con un valor distinto al predeterminado.
 - Este parámetro determina cuánto tiempo puede ejecutarse un comando en el OSS motor de Redis antes de que se registre como un comando de ejecución lenta. El valor predeterminado es 10 000 microsegundos (10 milisegundos). El valor predeterminado puede ser demasiado alto para algunas cargas de trabajo.
 - Determine un valor que sea más adecuado para su carga de trabajo en función de las necesidades de la aplicación y los resultados de las pruebas; sin embargo, un valor demasiado bajo puede generar un exceso de datos.
- [Mejor] Deje `slowlog-max-len` como valor predeterminado.
 - Este parámetro determina el límite máximo del número de comandos de ejecución lenta que se capturan en la OSS memoria de Redis en un momento dado. Un valor de 0 desactiva de manera efectiva la captura. Cuanto más alto sea el valor, más entradas se almacenarán en la memoria, lo que reducirá la posibilidad de que se expulse información importante antes de poder revisarla. El valor predeterminado es 128.
 - El valor predeterminado es adecuado para la mayoría de las cargas de trabajo. Si es necesario analizar los datos en una ventana de tiempo ampliada desde `redis-cli` mediante el `SLOWLOG` comando, considere la posibilidad de aumentar este valor. Esto permite que queden más comandos en la memoria de Redis. OSS

Si envía los `SLOWLOG` datos a CloudWatch Logs o a Kinesis Data Firehose, los datos se conservarán y se podrán analizar fuera del sistema, lo que reducirá ElastiCache la necesidad de almacenar grandes cantidades de comandos de ejecución lenta en la memoria de Redis. OSS

- [Recursos]:
 - [¿Cómo activo el registro OSS lento de Redis en un ElastiCache clúster de caché \(Redis\)? OSS](#)
 - [Entrega de registro](#)
 - [Parámetros específicos de Redis OSS](#)
 - <https://aws.amazon.com/cloudwatch/> Amazon CloudWatch
 - [Amazon Kinesis Data Firehose](#)

PE8: ¿Cómo ayuda Auto Scaling a aumentar el rendimiento del ElastiCache clúster?

Introducción a nivel de preguntas: al implementar la función de escalado OSS automático de Redis, ElastiCache sus componentes pueden ajustarse con el tiempo para aumentar o disminuir automáticamente los fragmentos o réplicas deseados. Esto se puede hacer mediante la implementación de políticas de escalado programado o de seguimiento de objetivos.

Ventaja a nivel de pregunta: comprender y planificar los picos de carga de trabajo puede garantizar un mejor rendimiento del almacenamiento en caché y la continuidad empresarial. ElastiCache (RedisOSS) Auto Scaling monitorea continuamente su utilización de CPU /Memory para asegurarse de que su clúster funcione a los niveles de rendimiento deseados.

- [Obligatorio] Al lanzar un clúster para ElastiCache (OSSRedis):
 1. Asegúrese de que el modo Clúster esté habilitado.
 2. Asegúrese de que la instancia pertenezca a una familia de un cierto tipo y tamaño que admita el escalado automático.
 3. Asegúrese de que el clúster no se ejecute en almacenes de datos globales, Outposts o zonas locales.

[Recursos]:

- [Escalar clústeres en Redis OSS \(modo de clúster activado\)](#)
- [Uso de Auto Scaling con particiones](#)
- [Uso de Auto Scaling con réplicas](#)
- [Lo mejor] Identifique si su carga de trabajo requiere mucha lectura o escritura para definir la política de escalado. Para obtener el mejor rendimiento, utilice solo una métrica de seguimiento. Se recomienda evitar tener varias políticas para cada dimensión, ya que las políticas de escalado automático se escalan horizontalmente de forma gradual cuando se alcanza el objetivo, pero solo se reducen horizontalmente cuando todas las políticas de seguimiento de objetivos están listas para reducirse horizontalmente.

[Recursos]:

- [Políticas de Auto Scaling](#)
- [Defining a scaling policy](#)
- [Lo mejor] Supervisar el rendimiento a lo largo del tiempo puede ayudar a detectar cambios en la carga de trabajo que pasarían desapercibidos si se hiciera en un momento determinado. Puede analizar CloudWatch las métricas correspondientes a la utilización del clúster durante un período

de cuatro semanas para determinar el umbral de valor objetivo. Si aún no está seguro de qué valor elegir, se recomienda comenzar con el valor mínimo admitido de métrica predefinida.

[Recursos]:

- [Supervisar el uso con métricas CloudWatch](#)
- [Mejor] Se recomienda probar la aplicación con las cargas de trabajo mínimas y máximas esperadas para identificar la cantidad exacta de particiones o réplicas que necesita el clúster a fin de desarrollar políticas de escalado y mitigar los problemas de disponibilidad.

[Recursos]:

- [Registro de un destino escalable](#)
- [Registro de un destino escalable](#)

El pilar de optimización de ElastiCache costes de Amazon Well-Architected Lens

El pilar de optimización de costes se centra en evitar costos innecesarios. Los temas clave incluyen comprender y controlar dónde se gasta el dinero, seleccionar el tipo de nodo más adecuado (utilice instancias que admitan la organización de datos en niveles en función de las necesidades de la carga de trabajo), el número correcto de tipos de recursos (el número de réplicas leídas), analizar el gasto a lo largo del tiempo y escalar para satisfacer las necesidades empresariales sin gastar de más.

Temas

- [COST1: ¿Cómo identifica y realiza un seguimiento de los costos asociados a sus ElastiCache recursos? ¿Cómo se desarrollan los mecanismos que permiten a los usuarios crear, gestionar y eliminar los recursos creados?](#)
- [COST2: ¿Cómo utiliza las herramientas de monitoreo continuo para ayudarlo a optimizar los costos asociados a sus ElastiCache recursos?](#)
- [COST3: ¿Deberías usar un tipo de instancia que admita la organización de datos en niveles? ¿Cuáles son las ventajas de la organización de datos en niveles? ¿Cuándo no se deben usar instancias de almacenamiento de datos en niveles?](#)

COST1: ¿Cómo identifica y realiza un seguimiento de los costos asociados a sus ElastiCache recursos? ¿Cómo se desarrollan los mecanismos que permiten a los usuarios crear, gestionar y eliminar los recursos creados?

Introducción a nivel de pregunta: Para comprender las métricas de costos es preciso la participación y la colaboración de varios equipos: ingeniería de software, gestión de datos, propietarios de productos, finanzas y liderazgo. Identificar los principales factores de los costos requiere que todas las partes involucradas comprendan las herramientas de control del uso de los servicios y las soluciones de compromiso que conlleva la administración de costos. Con frecuencia es la diferencia clave entre los esfuerzos de optimización de costos fructíferos e infructuosos. Asegurarse de contar con procesos y herramientas para hacer un seguimiento de los recursos creados desde el desarrollo hasta la producción y la retirada le ayuda a gestionar los costes asociados a ellosElastiCache.

Ventaja a nivel de duda: el seguimiento continuo de todos los costes asociados a la carga de trabajo requiere un conocimiento profundo de la arquitectura, que incluye ElastiCache como uno de sus componentes. Además, debe contar con un plan de administración de costos para recopilar y comparar el uso con su presupuesto.

- [Obligatorio] Establezca un centro de excelencia en la nube (CCoE) con uno de sus estatutos fundacionales para definir, rastrear y tomar medidas en función de las métricas relacionadas con el uso de sus organizaciones. ElastiCache Si CCoE existe y funciona, asegúrese de que sepa cómo leer y realizar un seguimiento de los costos asociados. ElastiCache Cuando se creen los recursos, utilice IAM funciones y políticas para validar que solo equipos y grupos específicos puedan crear instancias de los recursos. Esto garantiza que los costos estén asociados a los resultados comerciales y que se establezca una línea clara de responsabilidad, desde la perspectiva de los costos.

1. CCoEdebe identificar, definir y publicar métricas de costos que se actualicen periódicamente (mensualmente) en relación con el ElastiCache uso clave de datos categóricos, como:
 - a. Tipos de nodos utilizados y sus atributos: instancias estándar frente a instancias optimizadas para memoria, instancias bajo demanda o reservadas, regiones y zonas de disponibilidad
 - b. Tipos de entornos: gratuitos, de desarrollo, de pruebas y de producción
 - c. Estrategias de almacenamiento y retención de copias de seguridad
 - d. Transferencia de datos dentro y entre regiones
 - e. Instancias que se ejecutan en Amazon Outposts

2. CCoE consiste en un equipo multifuncional con una representación no exclusiva de los equipos de ingeniería de software, gestión de datos, equipo de producto, finanzas y liderazgo de su organización.

[Recursos]:

- [Create a Cloud Center of Excellence](#)
 - [ElastiCache Precios de Amazon](#)
- [Obligatorio] Utilice etiquetas de asignación de costos para realizar un seguimiento de los costos con un nivel de detalle bajo. Utilice la gestión de AWS costes para visualizar, comprender y gestionar sus AWS costes y su consumo a lo largo del tiempo.
 1. Use etiquetas para organizar sus recursos y etiquetas de asignación de AWS costes para realizar un seguimiento detallado de sus costos. Tras activar las etiquetas de asignación de costes, AWS utiliza las etiquetas de asignación de costes para organizar los costes de los recursos en el informe de asignación de costes, a fin de facilitar la categorización y el seguimiento de AWS los costes. AWS proporciona dos tipos de etiquetas de asignación de costes: las etiquetas AWS generadas y las etiquetas definidas por el usuario. AWS define, crea y aplica automáticamente las etiquetas AWS generadas, y usted define, crea y aplica las etiquetas definidas por el usuario. Debe activar ambos tipos de etiquetas por separado para que puedan aparecer en Cost Management o en un informe de asignación de costos.
 2. Utilice las etiquetas de asignación de costes para organizar la AWS factura y reflejar su propia estructura de costes. Cuando añadas etiquetas de asignación de costes a tus recursos en Amazon ElastiCache, podrás realizar un seguimiento de los costes agrupando los gastos de tus facturas por valores de etiquetas de recursos. Plantéese combinar etiquetas para realizar un seguimiento de los costos con un mayor nivel de detalle.

[Recursos]:

- [Uso de etiquetas AWS de asignación de costes](#)
 - [Monitoreo de costos con etiquetas de asignación de costos](#)
 - [AWS Explorador de costos](#)
- [Lo mejor] Connect ElastiCache el costo con las métricas que llegan a toda la organización.
 1. Tenga en cuenta las métricas empresariales y las métricas operativas, como la latencia: ¿qué conceptos de su modelo de negocio son comprensibles en todos los roles? Las métricas deben ser comprensibles para el mayor número posible de roles de la organización.
 2. Ejemplos: usuarios atendidos simultáneamente, latencia máxima y promedio por operación y usuario, puntuaciones de participación de los usuarios, tasas de retorno de usuarios por

semana, duración de la sesión por usuario, tasa de abandono, tasa de aciertos de caché y seguimiento de las claves.

[Recursos]:

- [Supervisar el uso con CloudWatch métricas](#)
- [Bueno] Mantenga la visibilidad up-to-date arquitectónica y operativa de las métricas y los costos en toda la carga de trabajo que utilice ElastiCache.
 1. Comprenda todo su ecosistema de soluciones, que ElastiCache suele formar parte de un ecosistema completo de AWS servicios en su conjunto tecnológico, desde los clientes hasta API Gateway, Redshift y QuickSight las herramientas de informes (por ejemplo).
 2. Mapee los componentes de su solución, desde los clientes, las conexiones, la seguridad, las operaciones en memoria, el almacenamiento, la automatización de recursos, el acceso y la administración de datos, en su diagrama de arquitectura. Cada capa se conecta a toda la solución y tiene sus propias necesidades y capacidades que aumentan el costo total o ayudan a gestionarlo.
 3. El diagrama debe incluir el uso de la computación, las redes, el almacenamiento, las políticas de ciclo de vida y la recopilación de métricas, así como los ElastiCache elementos operativos y funcionales de la aplicación
 4. Es probable que los requisitos de la carga de trabajo evolucionen con el tiempo por lo que es esencial que siga manteniendo y documentando su comprensión de los componentes subyacentes, así como de sus objetivos funcionales principales, a fin de mantener una actitud proactiva en la gestión de los costes de la carga de trabajo.
 5. El apoyo ejecutivo para garantizar la visibilidad, la responsabilidad, la priorización y los recursos es fundamental para disponer de una estrategia de gestión de costes eficaz. ElastiCache

COST2: ¿Cómo utiliza las herramientas de monitoreo continuo para ayudarlo a optimizar los costos asociados a sus ElastiCache recursos?

Introducción a nivel de preguntas: debe tratar de lograr un equilibrio adecuado entre las métricas de ElastiCache costo y rendimiento de las aplicaciones. Amazon CloudWatch proporciona visibilidad de las métricas operativas clave que pueden ayudarte a evaluar si tus ElastiCache recursos están sobreutilizados o infrautilizados en relación con tus necesidades. Desde el punto de vista de la optimización de costes, debe saber cuándo está sobreaprovisionado y poder desarrollar los mecanismos adecuados para redimensionar sus ElastiCache recursos y, al mismo tiempo, mantener sus necesidades operativas, de disponibilidad, resiliencia y rendimiento.

Ventaja a nivel de pregunta: En un estado ideal, habrá aprovisionado suficientes recursos para satisfacer las necesidades operativas de su carga de trabajo y no tendrá recursos infrautilizados que puedan llevar a un estado de costos deficiente. Debe poder identificar y evitar utilizar ElastiCache recursos sobredimensionados durante períodos prolongados.

- [Obligatorio] Úselo CloudWatch para monitorear sus ElastiCache clústeres y analizar cómo se relacionan estas métricas con sus paneles de AWS Cost Explorer.
 1. ElastiCache proporciona métricas a nivel de host (por ejemplo, el CPU uso) y métricas específicas del software del motor de caché (por ejemplo, las pérdidas y pérdidas de memoria caché). Estas métricas se miden y publican para cada nodo de caché en intervalos de 60 segundos.
 2. ElastiCache Las métricas de rendimiento (CPUUtilization, EngineUtilization SwapUsage CurrConnections, y desalojos) pueden indicar que es necesario escalar hacia arriba o hacia abajo (utilizar tipos de nodos de caché más grandes o más pequeños) o hacia dentro y hacia fuera (añadir más o menos fragmentos). Comprenda las implicaciones financieras de las decisiones de escalado mediante la creación de una matriz de estrategias que estime el costo adicional y los períodos de tiempo mínimos y máximos necesarios para cumplir con los umbrales de rendimiento de sus aplicaciones.

[Recursos]:

- [Supervisar el uso con métricas CloudWatch](#)
- [¿Qué métricas debo monitorear?](#)
- [ElastiCachePrecios de Amazon](#)
- [Obligatorio] Comprenda y documente su estrategia de copias de seguridad y las implicaciones de costos.
 1. Con ElastiCache, las copias de seguridad se almacenan en Amazon S3, que proporciona un almacenamiento duradero. Debe comprender las implicaciones de costos en relación con su capacidad para recuperarse en caso de errores.
 2. Habilite copias de seguridad automáticas que eliminarán los archivos de copia de seguridad que hayan superado el límite de retención.

[Recursos]:

- [Programación de copias de seguridad automáticas](#)
- [Precios de Amazon Simple Storage Service](#)

- [Lo mejor] Utilice nodos reservados para sus instancias como estrategia deliberada para administrar los costos de las cargas de trabajo que se conocen bien y están documentadas. Los nodos reservados se cobran por adelantado en función de una tarifa que depende del tipo de nodo y la duración de la reserva: uno o tres años. Dicho cargo es mucho menor que el cargo por uso por hora que se aplica con los nodos bajo demanda.
 1. Es posible que necesite operar sus ElastiCache clústeres mediante nodos bajo demanda hasta que haya recopilado datos suficientes para estimar los requisitos de las instancias reservadas. Planifique y documente los recursos necesarios para satisfacer sus necesidades y compare los costos esperados entre los tipos de instancias (bajo demanda o reservadas).
 2. Evalúe periódicamente los nuevos tipos de nodos de caché disponibles y determine si tiene sentido, desde una perspectiva de métricas operativas y de costos, migrar su flota de instancias a nuevos tipos de nodos de caché.

COST3: ¿Deberías usar un tipo de instancia que admita la organización de datos en niveles? ¿Cuáles son las ventajas de la organización de datos en niveles? ¿Cuándo no se deben usar instancias de almacenamiento de datos en niveles?

Introducción a nivel de pregunta: Seleccionar el tipo de instancia adecuado no solo puede tener un impacto en el rendimiento y el nivel de servicio, sino también en los aspectos financieros. Los tipos de instancias tienen diferentes costos asociados. Una decisión natural es seleccionar uno o varios tipos de instancias grandes que puedan adaptarse a todas las necesidades de almacenamiento de memoria. Sin embargo, esto podría tener un impacto significativo en los costos a medida que el proyecto madure. Para asegurarse de seleccionar el tipo de instancia correcto, es necesario examinar periódicamente el tiempo de inactividad del ElastiCache objeto.

Ventaja a nivel de pregunta: Debe tener una idea clara de cómo los distintos tipos de instancias afectan a sus costes en el presente y en el futuro. Los cambios marginales o periódicos en la carga de trabajo no deberían provocar cambios desproporcionados en los costos. Si la carga de trabajo lo permite, los tipos de instancias que admiten la organización de datos en niveles ofrecen un mejor precio por almacenamiento disponible. Debido al SSD almacenamiento de datos disponible por instancia, las instancias admiten una capacidad de almacenamiento total de datos por instancia mucho mayor.

- [Obligatorio] Comprenda las limitaciones de las instancias de almacenamiento de datos en niveles.
 1. Solo está disponible para ElastiCache clústeres (RedisOSS).
 2. Solo algunos tipos de instancias admiten la organización de datos en niveles.

3. Solo se ElastiCache admite la versión 6.2 (RedisOSS) o superior
4. Los artículos grandes no se cambian por otros. SSD Los objetos de más de 128 MiB se guardan en la memoria.

[Recursos]:

- [Organización de datos en niveles](#)
 - [ElastiCachePrecios de Amazon](#)
- [Obligatorio] Comprenda a qué porcentaje de su base de datos accede regularmente su carga de trabajo.
 1. Las instancias de almacenamiento de datos en niveles son ideales para cargas de trabajo que, a menudo, acceden a una pequeña parte del conjunto de datos general, pero que aun así requieren un acceso rápido al resto de los datos. En otras palabras, la relación entre datos calientes y templados es de aproximadamente 20:80.
 2. Desarrolle un seguimiento a nivel de clúster del tiempo de inactividad de los objetos.
 3. Las implementaciones grandes de más de 500 Gb de datos son buenas candidatas.
 - [Obligatorio] Comprenda que las instancias de organización de datos en niveles no son opcionales para determinadas cargas de trabajo.
 1. El acceso a los objetos que se utilizan con menos frecuencia conlleva un pequeño coste de rendimiento, ya que se cambian a objetos locales SSD. Si su aplicación depende del tiempo de respuesta, pruebe el impacto en su carga de trabajo.
 2. No resulta adecuado para cachés que almacenan principalmente objetos grandes de más de 128 MiB de tamaño.

[Recursos]:

- [Limitaciones](#)
- [Lo mejor] Los tipos de instancias reservadas admiten la organización de datos en niveles. Esto garantiza un menor costo en términos de cantidad de almacenamiento de datos por instancia.
 1. Es posible que necesite operar sus ElastiCache clústeres con instancias que no estén agrupadas en niveles de datos hasta que comprenda mejor sus requisitos.
 2. Analice el patrón de uso de datos de sus ElastiCache clústeres.
 3. Cree un trabajo automatizado que recopile periódicamente el tiempo de inactividad de los objetos.
 4. Si observa que un gran porcentaje (alrededor del 80 %) de los objetos permanecen inactivos durante un período de tiempo que considere apropiado para su carga de trabajo, documente

los hallazgos y sugiera migrar el clúster a instancias que admitan la organización de datos en niveles.

5. Evalúe periódicamente los nuevos tipos de nodos de caché disponibles y determine si tiene sentido, desde una perspectiva de métricas operativas y de costos, migrar su flota de instancias a nuevos tipos de nodos de caché.

[Recursos]:

- [OBJECT IDLETIME](#)
- [ElastiCachePrecios de Amazon](#)

Pasos comunes para la solución de problemas y prácticas recomendadas

Temas

- [Problemas de conectividad](#)
- [Errores del cliente OSS de Redis](#)
- [Solución de problemas de alta latencia en Serverless ElastiCache](#)
- [Solución de problemas de limitación en Serverless ElastiCache](#)
- [Temas relacionados](#)

Problemas de conectividad

Si no puedes conectarte a la ElastiCache memoria caché, considera una de las siguientes opciones:

1. Uso de TLS: si se produce un bloqueo de la conexión al dispositivo de ElastiCache punto final, es posible que no esté utilizando TLS en su cliente. Si utiliza ElastiCache Serverless, el cifrado en tránsito siempre está activado. Asegúrese de que su cliente utilice TLS para conectarse a la memoria caché. [Obtén más información sobre cómo conectarte a una caché con TLS habilitada aquí.](#)
2. VPC: solo se puede acceder a las ElastiCache cachés desde una VPC. Asegúrese de que la instancia EC2 desde la que accede a la memoria caché y a la ElastiCache memoria caché estén creadas en la misma VPC. Como alternativa, debe habilitar el emparejamiento de [VPC entre](#) la VPC en la que reside la instancia de EC2 y la VPC en la que va a crear la memoria caché.

3. Grupos de seguridad: ElastiCache utiliza grupos de seguridad para controlar el acceso a la memoria caché. Considere lo siguiente:
 - a. Asegúrese de que el grupo de seguridad utilizado por la ElastiCache memoria caché permita el acceso entrante a la misma desde su instancia EC2. Consulte [aquí](#) para obtener información sobre cómo configurar correctamente las reglas de entrada en su grupo de seguridad.
 - b. Asegúrese de que el grupo de seguridad utilizado por la ElastiCache memoria caché permita el acceso a los puertos de la memoria caché (6379 y 6380 para los sistemas sin servidor y 6379 de forma predeterminada para los de diseño propio). ElastiCache utiliza estos puertos para aceptar los comandos de Redis OSS. Obtenga más información sobre cómo configurar el acceso a los puertos [aquí](#).

Errores del cliente OSS de Redis

ElastiCache Solo se puede acceder a Serverless mediante los clientes OSS de Redis que admiten el protocolo de modo de clúster de Redis OSS. Se puede acceder a los clústeres de diseño propio desde los clientes OSS de Redis en cualquier modo, según la configuración del clúster.

Si se producen errores de Redis OSS en su cliente, tenga en cuenta lo siguiente:

1. Modo de clúster: si se producen errores de CROSSLOT o errores con el comando [SELECT](#) Redis OSS, es posible que esté intentando acceder a una caché habilitada en modo de clúster con un cliente de Redis OSS que no sea compatible con el protocolo de clúster OSS de Redis. ElastiCache Serverless solo es compatible con los clientes OSS de Redis que admiten el protocolo de clúster OSS de Redis. Si desea utilizar Redis OSS en un modo de clúster desactivado (CMD), debe diseñar su propio clúster.
2. Errores de CROSSLOT: si se produce el ERR CROSSLOT Keys in request don't hash to the same slot error, es posible que esté intentando acceder a claves que no pertenecen a la misma ranura en una memoria caché en modo clúster. Como recordatorio, ElastiCache Serverless siempre funciona en modo clúster. Las operaciones con varias claves, las transacciones o los scripts de Lua con varias claves solo están permitidas si todas las claves involucradas están en la misma ranura de hash.

[Para obtener más información sobre las mejores prácticas sobre la configuración de los clientes OSS de Redis, consulte esta entrada de blog.](#)

Solución de problemas de alta latencia en Serverless ElastiCache

Si tu carga de trabajo parece experimentar una latencia alta, puedes analizar las `SuccessfulWriteRequestLatency` métricas CloudWatch `SuccessfulReadRequestLatency` y comprobar si la latencia está relacionada con ElastiCache Serverless. Estas métricas miden la latencia, que es interna de ElastiCache Serverless; no se incluyen la latencia del lado del cliente ni los tiempos de recorrido de la red entre el cliente y el terminal ElastiCache Serverless.

Solución de problemas de latencia del lado del cliente

Si observa una latencia elevada en el lado del cliente, pero no un aumento correspondiente en `SuccessfulReadRequestLatency` la latencia del lado del servidor, ni `SuccessfulWriteRequestLatency` métricas que midan la latencia del lado del servidor, tenga en cuenta lo siguiente:

- Asegúrese de que el grupo de seguridad permita el acceso a los puertos 6379 y 6380: ElastiCache Serverless utiliza el puerto 6379 para el punto final principal y el puerto 6380 para el punto final del lector. Algunos clientes establecen la conectividad con ambos puertos para cada nueva conexión, incluso si la aplicación no utiliza la función de lectura desde réplica. Si su grupo de seguridad no permite el acceso entrante a ambos puertos, el establecimiento de la conexión puede tardar más. Obtenga más información sobre cómo configurar el acceso a los puertos [aquí](#).

Solución de problemas de latencia del lado del servidor

Cierta variabilidad y picos ocasionales no deberían ser motivo de preocupación. Sin embargo, si la `Average` estadística muestra un aumento brusco y persiste, consulte el `Personal Health Dashboard` AWS Health Dashboard y su `Personal Health Dashboard` para obtener más información. Si es necesario, considere la posibilidad de abrir un caso de soporte con AWS Support.

Tenga en cuenta las siguientes mejores prácticas y estrategias para reducir la latencia:

- Habilitar la lectura desde réplica: si su aplicación lo permite, le recomendamos que habilite la función de «lectura desde réplica» en su cliente OSS de Redis para escalar las lecturas y lograr una latencia más baja. Cuando está habilitada, ElastiCache Serverless intenta enrutar las solicitudes de lectura a los nodos de caché de réplica que se encuentran en la misma zona de disponibilidad (AZ) que el cliente, lo que evita la latencia de la red entre zonas de disponibilidad. Tenga en cuenta que habilitar la función de lectura desde réplica en su cliente significa que su aplicación acepta, en última instancia, la coherencia de los datos. Es posible que su aplicación reciba datos antiguos durante algún tiempo si intenta leerlos después de escribirlos en una clave.

- Asegúrese de que la aplicación esté desplegada en las mismas zonas de disponibilidad que la memoria caché: si la aplicación no se despliega en las mismas zonas de disponibilidad que la caché, es posible que observe una mayor latencia en el lado del cliente. Al crear una caché sin servidor, puede proporcionar las subredes desde las que la aplicación accederá a la caché, y ElastiCache Serverless crea puntos de enlace de VPC en esas subredes. Asegúrese de que la aplicación esté implementada en las mismas zonas de disponibilidad. De lo contrario, es posible que la aplicación sufra un salto entre zonas de disponibilidad al acceder a la caché, lo que provocará una mayor latencia del lado del cliente.
- Conexiones de reutilización: las solicitudes ElastiCache sin servidor se realizan a través de una conexión TCP habilitada para TLS mediante el protocolo RESP. El inicio de la conexión (incluida la autenticación de la conexión, si está configurada) lleva tiempo, por lo que la latencia de la primera solicitud es superior a la habitual. Las solicitudes a través de una conexión ya inicializada ofrecen una baja ElastiCache latencia constante. Por este motivo, debería considerar la posibilidad de utilizar la agrupación de conexiones o reutilizar las conexiones OSS de Redis existentes.
- Velocidad de escalado: ElastiCache Serverless escala automáticamente a medida que aumenta la tasa de solicitudes. Un aumento importante y repentino de la tasa de solicitudes, más rápido que la velocidad a la que se escala ElastiCache Serverless, puede provocar un aumento de la latencia durante algún tiempo. ElastiCache Por lo general, la tecnología Serverless puede aumentar rápidamente su tasa de solicitudes admitidas, lo que tarda entre 10 y 12 minutos en duplicar la tasa de solicitudes.
- Inspeccione los comandos de ejecución prolongada: algunos comandos de Redis OSS, incluidos los scripts de Lua o los comandos de estructuras de datos de gran tamaño, pueden ejecutarse durante mucho tiempo. Para identificar estos comandos, ElastiCache publica métricas a nivel de comandos. Con [ElastiCache Serverless](#) puedes usar las BasedECPUs métricas.
- Solicitudes limitadas: cuando las solicitudes se limitan en ElastiCache Serverless, es posible que experimente un aumento en la latencia del lado del cliente en su aplicación. [Cuando las solicitudes se limitan en ElastiCache Serverless, deberías ver un aumento en la métrica Serverless.ThrottledRequests ElastiCache](#) Consulta la siguiente sección para solucionar problemas con las solicitudes restringidas.
- Distribución uniforme de las claves y las solicitudes: en ElastiCache Redis OSS, una distribución desigual de las claves o solicitudes por ranura puede provocar una sobrecarga de datos, lo que puede provocar una latencia elevada. ElastiCache La tecnología Serverless admite hasta 30 000 ECPU/segundo (90 000 ECPU/segundo cuando se utiliza Read from Replica) en una sola ranura, lo que supone una carga de trabajo que ejecuta comandos SET/GET sencillos.

Le recomendamos que evalúe la distribución de las claves y las solicitudes entre las ranuras y garantice una distribución uniforme si la tasa de solicitudes supera este límite.

Solución de problemas de limitación en Serverless ElastiCache

En las arquitecturas orientadas a servicios y en los sistemas distribuidos, la limitación de la velocidad a la que los distintos componentes del servicio procesan las llamadas a la API se denomina limitación. Esto suaviza los picos, controla los desajustes en el rendimiento de los componentes y permite realizar recuperaciones más predecibles cuando se produce un evento operativo inesperado. ElastiCache Serverless está diseñado para este tipo de arquitecturas, y la mayoría de los clientes de Redis OSS incorporan reintentos para las solicitudes limitadas. Cierta grado de limitación no es necesariamente un problema para su aplicación, pero la limitación persistente de una parte sensible a la latencia de su flujo de trabajo de datos puede afectar negativamente a la experiencia del usuario y reducir la eficacia general del sistema.

[Cuando las solicitudes se limitan en Serverless, deberías ver un aumento en la ElastiCache métrica Serverless. `ThrottledRequests` ElastiCache](#) Si observa un número elevado de solicitudes limitadas, tenga en cuenta lo siguiente:

- **Velocidad de escalado:** ElastiCache Serverless escala automáticamente a medida que se ingieren más datos o aumenta la tasa de solicitudes. Si la aplicación escala más rápido que la velocidad a la que se escala sin servidor, es posible que sus solicitudes se reduzcan mientras que ElastiCache Serverless escala para adaptarse a su carga de trabajo. ElastiCache ElastiCache Por lo general, Serverless puede aumentar el tamaño de almacenamiento rápidamente, y tarda entre 10 y 12 minutos en duplicar el tamaño de almacenamiento de la memoria caché.
- **Distribución uniforme de las claves y las solicitudes:** en ElastiCache Redis OSS, una distribución desigual de las claves o solicitudes por ranura puede dar lugar a una gran demanda. Una ranura activa puede limitar las solicitudes si la tasa de solicitudes en una sola ranura supera las 30 000 ECPU/segundo, lo que supone una carga de trabajo que ejecuta comandos SET/GET sencillos.
- **Leer desde réplica:** si tu aplicación lo permite, considera utilizar la función «Leer desde réplica». La mayoría de los clientes OSS de Redis se pueden configurar para «escalar las lecturas» y dirigir las lecturas a los nodos de réplica. Esta función le permite escalar el tráfico de lectura. Además, ElastiCache Serverless enruta automáticamente la lectura de las solicitudes de réplica a los nodos de la misma zona de disponibilidad que la aplicación, lo que reduce la latencia. Cuando la función Read from Replica está habilitada, puede alcanzar hasta 90 000 ECPU/segundo en una sola ranura, para cargas de trabajo con simples comandos SET/GET.

Temas relacionados

- [Pasos adicionales de solución de problemas](#)
- [the section called “Procedimientos recomendados y estrategias de almacenamiento en caché”](#)

Pasos adicionales de solución de problemas

Se deben verificar los siguientes elementos al solucionar los problemas de conectividad persistentes con ElastiCache:

Temas

- [Grupos de seguridad](#)
- [ACL de red](#)
- [Tablas de enrutamiento](#)
- [Resolución de los DNS](#)
- [Identificación de los problemas con los diagnósticos del lado del servidor](#)
- [Validación de la conectividad de red](#)
- [Límites relacionados con la red](#)
- [Uso de la CPU](#)
- [Conexiones que terminan desde el lado del servidor](#)
- [Solución de problemas del lado del cliente para instancias de Amazon EC2](#)
- [Análisis del tiempo que se tarda en completar una sola solicitud](#)

Grupos de seguridad

Los grupos de seguridad son firewalls virtuales que protegen su ElastiCache cliente (instancia EC2, AWS Lambda función, contenedor Amazon ECS, etc.) y ElastiCache la memoria caché. Además, son firewalls con estado, lo que significa que, después de permitir el tráfico entrante o saliente, las respuestas para ese tráfico se autorizarán automáticamente en el contexto de ese grupo de seguridad específico.

La característica con estado requiere que el grupo de seguridad realice un seguimiento de todas las conexiones autorizadas. Además, hay un límite para las conexiones monitoreadas. Si se alcanza ese

límite, las conexiones nuevas producirán errores. Consulte la sección de solución de problemas para obtener ayuda sobre cómo identificar si el cliente o ElastiCache el bando han alcanzado los límites.

Puede tener un único grupo de seguridad asignado al mismo tiempo al cliente y al ElastiCache clúster, o grupos de seguridad individuales para cada uno.

En ambos casos, debe permitir el tráfico TCP saliente en el ElastiCache puerto de origen y el tráfico entrante en el mismo puerto. ElastiCache El puerto predeterminado es 11211 para Memcached y 6379 para Redis OSS. De forma predeterminada, los grupos de seguridad permiten el tráfico de salida. En este caso, solo se requiere la regla de entrada en el grupo de seguridad de destino.

Para obtener más información, consulte [Patrones de acceso para acceder a un ElastiCache clúster en una Amazon VPC](#).

ACL de red

Las listas de control de acceso (ACL) de red son reglas sin estado. El tráfico debe estar permitido en ambas direcciones (tanto de entrada como de salida) para tener éxito. Las ACL de red se asignan a subredes, no a recursos específicos. Es posible tener la misma ACL asignada ElastiCache y el mismo recurso del cliente, especialmente si se encuentran en la misma subred.

De forma predeterminada, las ACL de red permiten todo el tráfico. Sin embargo, se pueden modificar para denegar o permitir el tráfico. Además, la evaluación de las reglas de las ACL es secuencial, lo que significa que la regla con el número más bajo que coincida con el tráfico lo permitirá o lo denegará. La configuración mínima para permitir el tráfico de Redis OSS es:

ACL de red del lado del cliente:

- Reglas de entrada:
- Número de regla: preferiblemente inferior a cualquier regla de denegación
- Tipo: regla de TCP personalizada
- Protocolo: TCP
- Intervalo de puertos: 1024-65535
- Fuente: 0.0.0.0/0 (o cree reglas individuales para las subredes del clúster) ElastiCache
- Permitir/Denegar: Permitir

- Reglas salientes

- Número de regla: preferiblemente inferior a cualquier regla de denegación
- Tipo: regla de TCP personalizada
- Protocolo: TCP
- Intervalo de puertos: 6379
- Fuente: 0.0.0.0/0 (o las subredes del clúster). ElastiCache Tenga en cuenta que el uso de direcciones IP específicas puede crear problemas (en caso de conmutación por error o de escalado del clúster)
- Permitir/Denegar: Permitir

ElastiCache ACL de red:

- Reglas de entrada:
 - Número de regla: preferiblemente inferior a cualquier regla de denegación
 - Tipo: regla de TCP personalizada
 - Protocolo: TCP
 - Intervalo de puertos: 6379
 - Fuente: 0.0.0.0/0 (o cree reglas individuales para las subredes del clúster) ElastiCache
 - Permitir/Denegar: Permitir
- Reglas salientes
 - Número de regla: preferiblemente inferior a cualquier regla de denegación
 - Tipo: regla de TCP personalizada
 - Protocolo: TCP
 - Intervalo de puertos: 1024-65535
 - Fuente: 0.0.0.0/0 (o las subredes del clúster). ElastiCache Tenga en cuenta que el uso de direcciones IP específicas puede crear problemas (en caso de conmutación por error o de escalado del clúster)
 - Permitir/Denegar: Permitir

Para obtener más información, consulte las [ACL de red](#).

Tablas de enrutamiento

De manera similar a las ACL de red, cada subred puede tener tablas de enrutamiento diferentes. Si los clientes y el ElastiCache clúster se encuentran en subredes diferentes, asegúrese de que sus tablas de enrutamiento les permitan comunicarse entre sí.

Los entornos más complejos, los cuales implican varias VPC, enrutamiento dinámico o firewalls de red, pueden dificultar la resolución de los problemas. Consulte [Validación de la conectividad de red](#) para confirmar que la configuración de red es adecuada.

Resolución de los DNS

ElastiCache proporciona los puntos finales del servicio en función de los nombres de DNS. Los puntos de enlace disponibles son los puntos Configuration, Primary, Reader y Node. Para obtener más información, consulte [Búsqueda de puntos de enlace de conexión](#).

En caso de conmutación por error o de modificación del clúster, la dirección asociada al nombre del punto de conexión puede cambiar y se actualizará de forma automática.

Es posible que la configuración de DNS personalizada (es decir, que no utilice el servicio DNS de la VPC) no reconozca los nombres de DNS ElastiCache proporcionados. Asegúrese de que el sistema pueda resolver correctamente los ElastiCache puntos finales mediante herramientas del sistema como dig (como se muestra a continuación) o. nslookup

```
$ dig +short example.xxxxxx.ng.0001.use1.cache.amazonaws.com
example-001.xxxxxx.0001.use1.cache.amazonaws.com.
1.2.3.4
```

Además, la resolución de nombres se puede forzar a través del servicio de DNS de la VPC:

```
$ dig +short example.xxxxxx.ng.0001.use1.cache.amazonaws.com @169.254.169.253
example-001.tihewd.0001.use1.cache.amazonaws.com.
1.2.3.4
```

Identificación de los problemas con los diagnósticos del lado del servidor

CloudWatch las métricas y la información sobre el tiempo de ejecución del ElastiCache motor son fuentes o información comunes para identificar las posibles fuentes de problemas de conexión. En general, un buen análisis comienza con los siguientes elementos:

- **Uso de la CPU:** Redis OSS es una aplicación de subprocesos múltiples. Sin embargo, la ejecución de cada comando ocurre en un solo subproceso (principal). Por este motivo, ElastiCache proporciona las métricas `EngineCPUUtilization` y `CPUUtilization`. `EngineCPUUtilization` proporciona el uso de la CPU dedicado al proceso OSS de Redis y `CPUUtilization` el uso en todas las vCPU. Los nodos con más de una vCPU suelen tener valores diferentes para `CPUUtilization` y `EngineCPUUtilization`, el segundo suele ser más alto. Una `EngineCPUUtilization` alta puede producirse por un número elevado de solicitudes u operaciones complejas que toman demasiado tiempo de CPU en completarse. Ambos se pueden identificar por:
 - **Un número elevado de solicitudes:** verifique si hay aumentos en otras métricas que coincidan con el patrón de `EngineCPUUtilization`. Las métricas útiles son:
 - **CacheHits y CacheMisses:** el número de solicitudes correctas o solicitudes que no han encontrado un elemento válido en la caché. Si la proporción de los errores en comparación con los aciertos es alta, la aplicación está perdiendo tiempo y consta de recursos con solicitudes poco útiles.
 - **SetTypeCmds y GetTypeCmds:** estas métricas, las cuales se encuentran correlacionadas con `EngineCPUUtilization`, pueden ayudar a entender si la carga es mucho más elevada para las solicitudes de escritura (medida por `SetTypeCmds`) o lecturas (medida por `GetTypeCmds`). Si la carga son lecturas en su gran mayoría, el uso de varias réplicas de lectura puede equilibrar las solicitudes en varios nodos y ahorrar las principales para las escrituras. En los clústeres con el modo de clúster desactivado, el uso de réplicas de lectura se puede realizar creando una configuración de conexión adicional en la aplicación mediante el punto final del lector. ElastiCache Para obtener más información, consulte [Búsqueda de puntos de enlace de conexión](#). Las operaciones de lectura deben enviarse a esta conexión adicional. Las operaciones de escritura se realizarán a través del punto de conexión principal habitual. En el modo de clúster habilitado, se aconseja utilizar una biblioteca que admita réplicas de lectura por defecto. Con los indicadores correctos, la biblioteca podrá descubrir automáticamente la topología del clúster y los nodos de la réplica, habilitar las operaciones de lectura mediante el comando [READONLY Redis OSS y enviar las solicitudes de lectura](#) a las réplicas.
 - **Número elevado de conexiones:**
 - **CurrConnections y NewConnections:** `CurrConnection` muestra el número de conexiones establecidas en el momento de la recopilación de puntos de datos, mientras que `NewConnections` muestra cuántas conexiones se crearon en el periodo.

La creación y la gestión de las conexiones implica una sobrecarga significativa de la CPU. Además, el protocolo de establecimiento de comunicación de tres canales del TCP que se necesita para crear conexiones nuevas afectará negativamente a los tiempos de respuesta generales.

Un ElastiCache nodo con miles de puntos `NewConnections` por minuto indica que la conexión se crea y se utiliza con solo unos pocos comandos, lo que no es óptimo. Una práctica recomendada es mantener las conexiones establecidas y reutilizarlas para operaciones nuevas. Esto es posible cuando la aplicación de cliente admite e implementa correctamente la agrupación de conexiones o conexiones persistentes. Con la agrupación de conexiones, el número de `currConnections` no tiene grandes variaciones y `NewConnections` debe ser lo más bajo posible. El OSS de Redis proporciona un rendimiento óptimo con un número reducido de `CurrConnections`. Mantener la métrica `currConnections` en el orden de decenas o centenas minimiza el uso de recursos para admitir conexiones individuales tales como los búferes del lado del cliente y los ciclos de la CPU a fin de servir la conexión.

- Rendimiento de la red:
 - Determine el ancho de banda: ElastiCache los nodos tienen un ancho de banda de red proporcional al tamaño del nodo. Dado que las aplicaciones tienen características diferentes, los resultados pueden variar según la carga de trabajo. Como, por ejemplo, las aplicaciones que tengan una alta tasa de solicitudes pequeñas tienden a afectar más al uso de la CPU que al rendimiento de la red, mientras que las claves más grandes generarán un mayor uso de la red. Por esta razón, se aconseja probar los nodos con la carga de trabajo real para una mejor comprensión de los límites.

Simular la carga desde la aplicación proporcionaría resultados más precisos. Sin embargo, las herramientas de punto de referencia pueden transmitir una buena noción de los límites.

- Para los casos en los que las solicitudes son principalmente lecturas, el uso de réplicas para operaciones de lectura aliviará la carga en el nodo primario. Si el caso de uso es predominantemente de escrituras, el uso de muchas réplicas amplificará el uso de la red. Por cada byte escrito en el nodo primario, se enviarán N bytes a las réplicas, siendo N el número de réplicas. La mejor práctica para las cargas de trabajo con un uso intensivo de la escritura es utilizar ElastiCache (Redis OSS) con el modo de clúster habilitado para que las escrituras se puedan equilibrar entre varios fragmentos, o ampliarlas a un tipo de nodo con más capacidades de red.

- Los valores CloudWatchmetrics NetworkBytesIn y NetworkBytesOut proporcionan la cantidad de datos que entran o salen del nodo, respectivamente. ReplicationByteses el tráfico dedicado a la replicación de datos.

Para obtener más información, consulte [Límites relacionados con la red](#).

- Comandos complejos: los comandos OSS de Redis se envían en un único subproceso, lo que significa que las solicitudes se atienden de forma secuencial. Un solo comando lento puede afectar a otras solicitudes y conexiones, lo que genera tiempos de espera. El uso de comandos que actúan sobre varios valores, claves o tipos de datos debe efectuarse con cuidado. Las conexiones pueden bloquearse o interrumpirse en función del número de parámetros o del tamaño de sus valores de entrada o de salida.

Un ejemplo notorio es el comando KEYS. Analiza todo el espacio de claves en la búsqueda de un patrón dado y bloquea la puesta en marcha de otros comandos durante su ejecución. Redis OSS utiliza la notación «O grande» para describir la complejidad de sus comandos.

El comando de claves tiene una complejidad de tiempo $O(N)$, siendo N el número de claves en la base de datos. Por lo tanto, cuanto mayor sea el número de claves, más lento será el comando. Sin embargo, el comando KEYS puede causar problemas de diferentes maneras. Si no se utiliza un patrón de búsqueda, el comando devolverá todos los nombres de clave disponibles. En las bases de datos con miles o millones de elementos, se creará una enorme salida que saturará a los búferes de red.

Si se utiliza un patrón de búsqueda, solo las claves que coincidan con el patrón volverán al cliente. No obstante, el motor todavía barrerá todo el espacio de claves en búsqueda de dicho patrón y el tiempo para completar el comando será el mismo.

Una alternativa para KEYS es el comando SCAN. Vuelve a repetir el proceso sobre el espacio de claves y limita las iteraciones en un número específico de elementos, al evitar bloqueos prolongados en el motor.

El escaneo tiene el parámetro COUNT, el cual se utiliza para establecer el tamaño de los bloques de iteración. El valor predeterminado es 10 (10 elementos por iteración).

En función del número de elementos en la base de datos, los bloques de valores COUNT pequeños requerirán más iteraciones para completar un análisis completo, mientras que los valores más grandes mantendrán al motor ocupado durante más tiempo en cada iteración. Mientras que los valores de conteo pequeños harán SCAN más lento en las bases de datos de

gran tamaño, los valores más elevados pueden causar los mismos problemas presentados para KEYS.

Por ejemplo, ejecutar el comando SCAN con un valor de conteo en 10 requerirá 100 000 repeticiones en una base de datos con 1 millón de claves. Si el tiempo promedio de ida y vuelta de la red es de 0,5 milisegundos, cerca de 50 000 milisegundos (50 segundos) se utilizarán para transferir solicitudes.

Por otro lado, si el valor de conteo fuera 100 000, se requerirá una sola iteración y solo se gastarían 0,5 ms para transferirla. Sin embargo, el motor se encontraría completamente bloqueado para otras operaciones hasta que el comando termine de analizar todo el espacio de claves.

Además de KEYS, existen otros comandos que son potencialmente dañinos si no se utilizan correctamente. Para ver una lista de todos los comandos, junto con su complejidad de tiempo, acceda a <https://redis.io/commands>.

Ejemplos de problemas posibles:

- **Secuencias de comandos de Lua:** Redis OSS proporciona un intérprete de Lua integrado, que permite la ejecución de scripts en el servidor. Los scripts de Lua en Redis OSS se ejecutan a nivel de motor y son atómicos por definición, lo que significa que no se permitirá ejecutar ningún otro comando o script mientras un script esté en ejecución. Los scripts de Lua ofrecen la posibilidad de ejecutar varios comandos, algoritmos de toma de decisiones, análisis de datos y otros directamente en el motor OSS de Redis. Mientras que la atomicidad de los scripts y la posibilidad de descargar la aplicación son tentadoras, los scripts deben emplearse con cuidado y para pequeñas operaciones. Sí ElastiCache, el tiempo de ejecución de los scripts de Lua está limitado a 5 segundos. Los scripts que no se hayan escrito en el espacio de claves se interrumpirán de manera automática después del periodo de 5 segundos. Para evitar la corrupción de datos y las inconsistencias, el nodo realizará una conmutación por error si la ejecución del script no se ha completado en 5 segundos y ha tenido alguna escritura durante su ejecución. Las [transacciones](#) son la alternativa para garantizar la coherencia de las múltiples modificaciones clave relacionadas en Redis OSS. Una transacción permite la ejecución de un bloque de comandos al observar las claves existentes en busca de modificaciones. Si alguna de las claves observadas cambia antes de la finalización de la transacción, se descartan todas las modificaciones.
- **Eliminación masiva de elementos:** el comando DEL acepta varios parámetros, los cuales son los nombres clave que se eliminarán. Las operaciones de eliminación son síncronas y

llevarán mucho tiempo de CPU si la lista de parámetros es grande, o si contiene una lista, un conjunto, un conjunto ordenado o un hash grandes (estructuras de datos que contienen varios subelementos). En otras palabras, incluso la eliminación de una sola clave puede tomar un tiempo considerable si tiene muchos elementos. La alternativa DEL es UNLINK un comando asíncrono disponible desde Redis OSS 4. UNLINK debe preferirse a él siempre que sea posible. DEL A partir de la ElastiCache versión 6x (Redis OSS), el `lazyfree-lazy-user-del` parámetro hace que el DEL comando se comporte igual que UNLINK cuando está activado. Para obtener más información, consulte Cambios de parámetros de [Redis OSS 6.0](#).

- Comandos que actúan sobre varias claves: se mencionó el comando DEL como un comando que acepta varios argumentos y su tiempo de ejecución será directamente proporcional a eso. Sin embargo, Redis OSS proporciona muchos más comandos que funcionan de forma similar. Por ejemplo, los comandos MSET y MGET permiten la inserción o recuperación de varias claves de cadena a la vez. Su uso puede resultar beneficioso para reducir la latencia de la red inherente a varios comandos SET o GET individuales. Sin embargo, una lista de parámetros extensa afectará al uso de la CPU.

Aunque el uso de la CPU por sí sola no es la causa de los problemas de conectividad, dedicar demasiado tiempo a procesar uno o varios comandos a través de varias claves puede causar interrupciones en otras solicitudes y aumentar el uso general de la CPU.

El número y el tamaño de las claves afectarán a la complejidad del comando y, en consecuencia, al tiempo de finalización.

Otros ejemplos de comandos que pueden actuar sobre varias claves son HMGET, HMSET, MSETNX, PFCOUNT, PFMERGE, SDIFF, SDIFFSTORE, SINTER, SINTERSTORE, SUNION, SUNIONSTORE, TOUCH, ZDIFF, ZDIFFSTORE, ZINTER y ZINTERSTORE.

- Comandos que actúan sobre varios tipos de datos: Redis OSS también proporciona comandos que actúan sobre una o varias teclas, independientemente del tipo de datos. ElastiCache (Redis OSS) proporciona la métrica `KeyBasedCmds` para supervisar dichos comandos. Esta métrica suma la ejecución de los siguientes comandos en el periodo seleccionado:

- Complejidad $O(N)$:
 - KEYS
- $O(1)$
 - EXISTS
 - OBJECT

- PTTL
- RANDOMKEY
- TTL
- TYPE
- EXPIRE
- EXPIREAT
- MOVE
- PERSIST
- PEXPIRE
- PEXPIREAT
- UNLINK ($O(N)$) para recuperar la memoria. No obstante, la tarea de recuperación de memoria ocurre en un subproceso aparte y no bloquea el motor.
- Tiempos de complejidad diferentes según el tipo de datos:
 - DEL
 - DUMP
 - Se estima que el comando RENAME tiene una complejidad $O(1)$, pero ejecuta DEL internamente. El tiempo de ejecución variará en función del tamaño de la clave que ha sido renombrada.
 - RENAMENX
 - RESTORE
 - SORT
- Hash de gran tamaño: un hash es un tipo de datos que permite una sola clave con varios subelementos de valor de clave. Cada hash puede almacenar 4 294 967 295 elementos y las operaciones en hash grandes pueden volverse costosas. Del mismo modo que KEYS, los hashes tienen el comando HKEYS con una complejidad de tiempo $O(N)$, siendo N el número de elementos en el hash. Se recomienda emplear HSCAN antes que HKEYS para evitar comandos de larga ejecución. Los comandos HDEL, HGETALL, HMGET, HMSET y HVALS se deben utilizar con precaución en hashes grandes.
- Otras estructuras de big data: además de los hashes, existen otras estructuras de datos que pueden ser pesadas para la CPU. Los conjuntos, las listas, los conjuntos ordenados y los Hyperloglogs también pueden demorar en gestionarse en función del tamaño y de los

comandos utilizados. Para obtener más información sobre los comandos, consulte <https://redis.io/commands>.

Validación de la conectividad de red

Luego de revisar las configuraciones de red relacionadas con la resolución de DNS, los grupos de seguridad, las ACL de red y las tablas de enrutamiento, la conectividad se puede validar con el VPC Reachability Analyzer y las herramientas del sistema.

Reachability Analyzer probará la conectividad de red y confirmará si se cumplen todos los requisitos y permisos. Para las siguientes pruebas, necesitará el ENI ID (identificación de interfaz de red elástica) de uno de los ElastiCache nodos disponibles en su VPC. Para ello, puede realizar lo siguiente:

1. Diríjase a <https://console.aws.amazon.com/ec2/v2/home?#NIC:>
2. Filtre la lista de interfaces por el nombre del ElastiCache clúster o la dirección IP obtenida de las validaciones de DNS anteriores.
3. Anote o guarde el ID de ENI. Si se muestran varias interfaces, revise la descripción para confirmar que pertenecen al ElastiCache clúster correcto y elija una de ellas.
4. Continúe con el siguiente paso.
5. Cree una ruta de análisis en <https://console.aws.amazon.com/vpc/home?#ReachabilityAnalyzer> y elija las siguientes opciones:
 - Tipo de fuente: elija instancia si su ElastiCache cliente se ejecuta en una instancia de Amazon EC2 o en una interfaz de red (si utiliza otro servicio, como AWS Fargate Amazon ECS con red awsvpc AWS Lambda, etc.) y el ID de recurso respectivo (instancia EC2 o ID ENI);
 - Tipo de destino: elija Network Interface (Interfaz de red) y seleccione la ElastiCache ENI (ENI de ElastiCache) de la lista.
 - Puerto de destino: especifique 6379 para ElastiCache (Redis OSS) o 11211 para (Memcached). ElastiCache Estos son los puertos definidos con la configuración predeterminada y en este ejemplo se supone que no se modifican.
 - Protocolo: TCP

Cree la ruta de análisis y espere unos momentos para obtener el resultado. Si no se puede acceder al estado, abra los detalles del análisis y revise el Explorador de análisis para conocer los detalles en los que se bloquearon las solicitudes.

Si se han superado las pruebas de accesibilidad, proceda a la verificación a nivel del sistema operativo.

Para validar la conectividad TCP en el puerto de ElastiCache servicio: en Amazon Linux, Nping está disponible en el paquete nmap y puede probar la conectividad TCP en el ElastiCache puerto, además de proporcionar a la red el tiempo de ida y vuelta para establecer la conexión. Úselo para validar la conectividad de la red y la latencia actual con el ElastiCache clúster, como se muestra a continuación:

```
$ sudo nping --tcp -p 6379 example.xxxxxx.ng.0001.use1.cache.amazonaws.com

Starting Nping 0.6.40 ( http://nmap.org/nping ) at 2020-12-30 16:48 UTC
SENT (0.0495s) TCP ...
(Output suppressed )

Max rtt: 0.937ms | Min rtt: 0.318ms | Avg rtt: 0.449ms
Raw packets sent: 5 (200B) | Rcvd: 5 (220B) | Lost: 0 (0.00%)
Nping done: 1 IP address pinged in 4.08 seconds
```

De forma predeterminada, nping envía 5 sondas con un retraso de 1 segundo entre ellas. Puede utilizar la opción “-c” para aumentar el número de sondas y “-delay” a fin de cambiar el tiempo en que se envía una prueba nueva.

Si las pruebas con el VPC Reachability Analyzer funcionan, pero fracasan con nping, pida al administrador del sistema que revise las reglas de firewall basadas en host, las reglas de enrutamiento asimétrico o cualquier otra restricción posible a nivel de sistema operativo.

En la ElastiCache consola, compruebe si el cifrado en tránsito está activado en los detalles ElastiCache del clúster. Si el cifrado en tránsito se encuentra habilitado, confirme si la sesión de TLS se puede establecer con el siguiente comando:

```
openssl s_client -connect example.xxxxxx.use1.cache.amazonaws.com:6379
```

Se espera un gran resultado si la conexión y la negociación de TLS son exitosas. Verifique el código de retorno que se encuentra disponible en la última línea, el valor debe ser 0 (ok). Si OpenSSL devuelve algo diferente, verifique el motivo del error en <https://www.openssl.org/docs/man1.0.2/man1/verify.html#DIAGNOSTICS>.

Si se han superado todas las pruebas de infraestructura y sistema operativo, pero la aplicación sigue sin poder conectarse a ellos ElastiCache, compruebe si las configuraciones de la aplicación cumplen con los ElastiCache parámetros. Los errores frecuentes son:

- Su aplicación no admite el modo de ElastiCache clúster y ElastiCache tiene el modo de clúster activado;
- Su aplicación no es compatible con TLS/SSL y ElastiCache tiene activado el cifrado en tránsito;
- La aplicación es compatible con TLS/SSL, pero no tiene los indicadores de configuración correctos ni las entidades de certificación de confianza.

Límites relacionados con la red

- Número máximo de conexiones: hay límites estrictos para conexiones simultáneas. Cada ElastiCache nodo permite hasta 65 000 conexiones simultáneas en todos los clientes. Este límite se puede monitorizar a través de las `CurrConnections` métricas CloudWatch activadas. Sin embargo, los clientes también tienen sus límites para las conexiones de salida. En Linux, verifique el rango de puertos efímeros permitido con el comando:

```
# sysctl net.ipv4.ip_local_port_range
net.ipv4.ip_local_port_range = 32768 60999
```

En el ejemplo anterior, se permitirán 28231 conexiones desde el mismo origen, a la misma IP (ElastiCache nodo) y puerto de destino. El siguiente comando muestra cuántas conexiones existen para un ElastiCache nodo específico (IP 1.2.3.4):

```
ss --numeric --tcp state connected "dst 1.2.3.4 and dport == 6379" | grep -vE
'^State' | wc -l
```

Si el número es demasiado alto, es posible que el sistema se sobrecargue al intentar procesar las solicitudes de conexión. Se recomienda considerar la implementación de técnicas tales como la agrupación de conexiones o conexiones persistentes para controlar las conexiones con mayor facilidad. Siempre que sea posible, configure el grupo de conexiones para limitar el número máximo de conexiones a unos pocos cientos. Además, se recomienda seguir la lógica del retardo para controlar los tiempos de espera u otras excepciones de conexión a fin de evitar la pérdida de conexión en caso de problemas.

- Límites de tráfico de red: compruebe las siguientes [CloudWatch métricas de Redis OSS para](#) identificar los posibles límites de red que se estén alcanzando en el nodo: ElastiCache
- `NetworkBandwidthInAllowanceExceeded/NetworkBandwidthOutAllowanceExceeded`: paquetes de red configurados porque el rendimiento superó el límite de banda ancha agregado.

Es importante tener en cuenta que cada byte escrito en el nodo primario se replicará en N réplicas, siendo N el número de réplicas. Es posible que los clústeres con tipos de nodos pequeños, varias réplicas y solicitudes de escritura intensivas no puedan afrontar al retraso de la reproducción. En estos casos, es una práctica recomendada escalar verticalmente (cambiar el tipo de nodo), escalar horizontalmente (agregar particiones en clústeres en modo de clúster habilitado) y disminuir el número de réplicas o el de escrituras.

- `NetworkConntrackAllowanceExceeded`: paquetes configurados porque se ha superado el número máximo de conexiones rastreadas en todos los grupos de seguridad asignados al nodo. Es probable que las conexiones nuevas fallen durante este periodo.
- `NetworkPackets PerSecondAllowanceExceeded`: se ha superado el número máximo de paquetes por segundo. Las cargas de trabajo basadas en una alta tasa de solicitudes muy pequeñas pueden alcanzar este límite antes de la banda ancha máxima.

Las métricas mencionadas son una manera ideal de confirmar que los nodos alcanzan sus límites de red. No obstante, los límites también son identificables por periodos de estancamiento en las métricas de la red.

Si dichos periodos se observan durante largos plazos de tiempo, es probable que se produzca un retraso en la reproducción, un aumento de los bytes utilizados para caché y un deterioro en la memoria que se puede liberar, en el intercambio alto y en el uso de la CPU. Las instancias de Amazon EC2 también tienen límites de red a los que se puede realizar un seguimiento mediante las [Métricas de los controladores de ENA](#). Las instancias de Linux con compatibilidad de red mejorada y los controladores de ENA 2.2.10, o más recientes, pueden controlar los contadores de límites con el comando:

```
# ethtool -S eth0 | grep "allowance_exceeded"
```

Uso de la CPU

La métrica de uso de la CPU es el punto de partida de la investigación, y los siguientes elementos pueden ayudar a reducir los posibles problemas ElastiCache secundarios:

- **Redis OSS SlowLogs:** la configuración ElastiCache predeterminada conserva los últimos 128 comandos que tardaron más de 10 milisegundos en completarse. El historial de comandos lentos se mantiene durante el tiempo de ejecución del motor y se perderá en caso de interrupción o de reinicio. Si la lista alcanza 128 entradas, los eventos antiguos se eliminarán para crear espacio para otros nuevos. El tamaño de la lista de eventos lentos y el tiempo de ejecución que se considera lento puede modificarse a través de los parámetros `slowlog-max-len` y `slowlog-log-slower-than` en un [grupo de parámetros personalizados](#). La lista de registros lentos se puede recuperar al ejecutar `SLOWLOG GET 128` en el motor, siendo 128 los últimos 128 comandos lentos informados. Cada entrada cuenta con los siguientes campos:

```

1) 1) (integer) 1 -----> Sequential ID
   2) (integer) 1609010767 --> Timestamp (Unix epoch time)of the Event
   3) (integer) 4823378 -----> Time in microseconds to complete the command.
   4) 1) "keys" -----> Command
      2) "*" -----> Arguments
   5) "1.2.3.4:57004"-> Source

```

El evento anterior ocurrió el 26 de diciembre, a las 19:26:07 UTC, tardó 4,8 segundos (4823 ms) en completarse y fue causado por el comando KEYS solicitado desde el cliente 1.2.3.4.

En Linux, la marca de tiempo puede convertirse con la fecha del comando:

```

$ date --date='@1609010767'
Sat Dec 26 19:26:07 UTC 2020

```

Con Python:

```

>>> from datetime import datetime
>>> datetime.fromtimestamp(1609010767)
datetime.datetime(2020, 12, 26, 19, 26, 7)

```

O en Windows con PowerShell:

```

PS D:\Users\user> [datetimeoffset]::FromUnixTimeSeconds('1609010767')
DateTime           : 12/26/2020 7:26:07 PM
UtcDateTime        : 12/26/2020 7:26:07 PM
LocalDateTime      : 12/26/2020 2:26:07 PM
Date                : 12/26/2020 12:00:00 AM

```

```

Day           : 26
DayOfWeek     : Saturday
DayOfYear    : 361
Hour         : 19
Millisecond   : 0
Minute       : 26
Month        :
Offset       : 00:00:00Ticks           : 637446075670000000
UtcTicks     : 637446075670000000
TimeOfDay    : 19:26:07
Year         : 2020

```

Muchos comandos lentos en un corto periodo de tiempo (el mismo minuto o menos) son motivo de preocupación. Revise la naturaleza de los comandos y cómo se pueden optimizar (consulte los ejemplos anteriores). Si los comandos con complejidad de tiempo $O(1)$ son frecuentes, verifique los demás factores para el elevado uso de la CPU mencionado anteriormente.

- **Métricas de latencia:** ElastiCache (Redis OSS) proporciona CloudWatch métricas para monitorear la latencia promedio de diferentes clases de comandos. El punto de datos se calcula al dividir el número total de ejecuciones de comandos en la categoría por el tiempo total de ejecución en el periodo. Es importante entender que los resultados de la métrica de latencia son un agregado de varios comandos. Un solo comando puede provocar resultados inesperados, como tiempos de espera, sin un impacto significativo en las métricas. En tales casos, los eventos de registro lento serían una fuente de información más precisa. La siguiente lista contiene las métricas de latencia disponibles y los comandos respectivos que les afectan.
 - **EvalBasedCmdsLatency:** relacionado con los comandos de Lua Script, `eval` `evalsha`
 - **GeoSpatialBasedCmdsLatency:** `geodist`, `geohash`, `geopos`, `georadius`, `georadiusbymember`, `geoadd`;
 - **GetTypeCmdsLatency:** Lee los comandos, independientemente del tipo de datos;
 - **HashBasedCmdsLatency:** `hexists`, `hget`, `hgetall`, `hkeys`, `hlen`, `hmget`, `hvals`, `hstrlen`, `hdel`, `hincrby`, `hincrbyfloat`, `hset`, `hsetnx`;
 - **HyperLogLogBasedCmdsLatency:** `pfselftest`, `pfcount`, `pfdebug`, `pfadd`, `pfmerge`;
 - **KeyBasedCmdsLatency:** Comandos que pueden actuar sobre diferentes tipos de datos: `dump`, `exists`, `keys`, `object`, `pttl`, `randomkey`, `ttdl`, `type`, `del`, `expire`, `expireat`, `move`, `persist`

- **ListBasedCmdsLatency:** lindex, len, lrange, lpop, broplpush, linsert, pop, push, pushx, lrem, let, ltrim, rpop, proplpush, rpushx;
- **PubSubBasedCmdsLatency:** psubscribe, publish, pubsub, punsubscribe, subscribe, unsubscribe;
- **SetBasedCmdsLatency:** scard, sdiff, sinter, sismember, smembers, srandmember, union, sadd, sdiffstore, sinterstore, smove, spop, srem, sunionstore;
- **SetTypeCmdsLatency:** Escribe comandos, independientemente del tipo de datos;
- **SortedSetBasedCmdsLatency:** zcard, zcount, zrange, zrangebyscore, zrank, zrevrange, zrevrangebyscore, zrevrank, zscore, zrangebylex, zrevrangebylex, zlexcount, zadd, zincrby, zinterstore, zrem, zremrangebyrank, zremrangebyscore, zunionstore, zremrangebylex, zpopmax, zpopmin, bzpopmin, bzpopmax;
- **StringBasedCmdsLatency:** bitcount, get, getbit, getrange, mget, strlen, substr, bitpos, append, bitop, bitfield, decr, decrby, getset, incr, incrby, incrbyfloat, mset, msetnx, psetex, set, setbit, setex, setnx, setrange;
- **StreamBasedCmdsLatency:** xrange, xrevrange, xlen, xread, xpending, xinfo, xadd, xgroup, readgroup, xack, xclaim, xdel, xtrim, xsetid;
- **Comandos de tiempo de ejecución de Redis OSS:**
 - **info commandstats:** proporciona una lista de los comandos ejecutados desde que se inició el motor OSS de Redis, su número acumulado de ejecuciones, el tiempo total de ejecución y el tiempo medio de ejecución por comando;
 - **client list:** proporciona una lista de clientes conectados actualmente e información relevante como el uso de los búferes, el último comando ejecutado, entre otros.
- **Backup y replicación:** las versiones ElastiCache (Redis OSS) anteriores a la 2.8.22 utilizan un proceso bifurcado para crear copias de seguridad y procesar sincronizaciones completas con las réplicas. Este método puede incurrir en una sobrecarga significativa de la memoria para casos de uso intensivos de escritura.

A partir de la versión 2.8.22 de ElastiCache Redis OSS, se introdujo un método de copia de seguridad y replicación sin bifurcación. AWS El método nuevo puede retrasar las escrituras a fin de evitar errores. Ambos métodos pueden causar periodos de mayor uso de la CPU y dar lugar a tiempos de respuesta más altos, lo que, en consecuencia, conlleva a tiempos de espera del cliente durante su ejecución. Siempre verifique si los errores del cliente se producen durante el periodo de copia de seguridad o si la métrica `SaveInProgress` fue 1 en el periodo. Se aconseja programar

el periodo de copia de seguridad para periodos de baja utilización con el objetivo de minimizar los posibles problemas con los clientes o los errores de la copia de seguridad.

Conexiones que terminan desde el lado del servidor

La configuración predeterminada ElastiCache (Redis OSS) mantiene las conexiones de los clientes establecidas indefinidamente. Sin embargo, en algunos casos, la interrupción de la conexión puede ser deseable. Por ejemplo:

- Los errores en la aplicación cliente pueden hacer que se olviden las conexiones y que se mantengan establecidas con un estado inactivo. Esto se denomina “fuga de conexión”. Su consecuencia es un aumento constante en el número de conexiones establecidas que se observaron en la métrica `CurConnections`. Este comportamiento puede provocar una saturación en el cliente o ElastiCache en el lado. Cuando no es posible realizar una solución inmediata desde el lado del cliente, algunos administradores establecen un valor de «tiempo de espera» en su grupo de ElastiCache parámetros. El tiempo de espera es el tiempo permitido (medido en segundos) para que las conexiones inactivas persistan. Si el cliente no envía ninguna solicitud durante el período, el motor OSS de Redis finalizará la conexión en cuanto la conexión alcance el valor de tiempo de espera. Los valores de tiempo de espera pequeños pueden dar lugar a desconexiones innecesarias de manera que los clientes necesitarán ocuparse correctamente de ellas y volver a conectarse, lo que genera retrasos.
- La memoria empleada para almacenar claves se comparte con los búferes del cliente. Los clientes lentos con grandes solicitudes o respuestas pueden exigir una cantidad significativa de memoria para operar sus búferes. La configuración predeterminada ElastiCache (Redis OSS) no restringe el tamaño de los búferes de salida normales de los clientes. Si se alcanza el límite de `maxmemory`, el motor intentará expulsar elementos para cumplir con el uso del búfer. En condiciones de muy poca memoria, ElastiCache (Redis OSS) puede optar por desconectar los clientes que consumen grandes búferes de salida para liberar memoria y conservar el estado del clúster.

Se puede limitar el tamaño de los búferes de cliente mediante configuraciones personalizadas por lo que cuando un cliente alcance ese límite se desconectará. No obstante, los clientes deben ser capaces de resolver desconexiones inesperadas. Los parámetros para manejar el tamaño de los búferes para los clientes regulares son los siguientes:

- `client-query-buffer-limit`: Tamaño máximo de una sola solicitud de entrada;
- `client-output-buffer-limit-normal-soft-limit`: Límite flexible para las conexiones de los clientes. La conexión finalizará si se mantiene por encima del límite flexible durante más tiempo del tiempo

en segundos definido normal-soft-seconds o si sobrepasa el límite estricto; client-output-buffer-limit

- client-output-buffer-limit-normal-soft-seconds: Tiempo permitido para las conexiones que superen el client-output-buffer-limit -normal-soft-limit;
- client-output-buffer-limit-normal-hard-limit: Una conexión que alcance este límite finalizará inmediatamente.

Además de los búferes de los clientes frecuentes, las siguientes opciones controlan el búfer para los nodos de réplica y los clientes de publicación/suscripción:

- client-output-buffer-limit-replica-hard-limit;
- client-output-buffer-limit-replica-soft-seconds;
- client-output-buffer-limit-replica-hard-limit;
- client-output-buffer-limit-pubsub-soft-limit;
- client-output-buffer-limit-pubsub-soft-seconds;
- client-output-buffer-limit-pubsub-hard-limit;

Solución de problemas del lado del cliente para instancias de Amazon EC2

La carga y la capacidad de respuesta por parte del cliente también pueden afectar a las solicitudes. ElastiCache Los límites de la instancia EC2 y del sistema operativo deben revisarse con cuidado mientras se solucionan problemas de conectividad intermitente o de tiempo de espera. Algunos puntos clave a observar:

- CPU:
 - El uso de la CPU de la instancia EC2: asegúrese de que la CPU no se encuentre saturada o cerca del 100 %. El análisis histórico se puede realizar mediante CloudWatch, sin embargo, tenga en cuenta que la granularidad de los puntos de datos es de 1 minuto (con la monitorización detallada habilitada) o 5 minutos;
 - Si utiliza [instancias EC2 ampliables](#), asegúrese de que no se haya agotado el saldo de crédito de la CPU. Esta información está disponible en la CPUCreditBalance CloudWatch métrica.
 - Los períodos cortos de uso intensivo de la CPU pueden provocar tiempos de espera sin que se refleje en el 100 por ciento de uso. CloudWatch Estos casos requieren un monitoreo en tiempo real con herramientas de sistema operativo como top, ps y mpstat.
- Network

- Verifique si el rendimiento de la red se encuentra por debajo de los valores aceptables de acuerdo con las capacidades de la instancia. Para obtener más información, consulte [Tipos de instancia de Amazon EC2](#)
- En instancias que dispongan de un controlador de red mejorado ena, verifique las [estadísticas de ENA](#) para los tiempos de espera o los límites excedidos. Las siguientes estadísticas son útiles para confirmar la saturación de los límites de red:
 - `bw_in_allowance_exceeded/bw_out_allowance_exceeded`: número de paquetes moldeados debido a un rendimiento excesivo de entrada o de salida;
 - `conntrack_allowance_exceeded`: número de paquetes descartados debido a los [límites de seguimiento de conexiones](#) de los grupos de seguridad. Las conexiones nuevas fallarán cuando este límite se encuentre saturado.
 - `linklocal_allowance_exceeded`: número de paquetes descartados debido a solicitudes excesivas de metadatos de instancia, NTP a través de DNS de la VPC. El límite es de 1024 paquetes por segundo para todos los servicios.
 - `pps_allowance_exceeded`: número de paquetes descartados debido a una proporción excesiva de paquetes por segundo. Se puede alcanzar el límite de PPS cuando el tráfico de red consiste en miles o millones de solicitudes muy pequeñas por segundo. ElastiCache el tráfico se puede optimizar para aprovechar mejor los paquetes de red mediante canalizaciones o comandos que realizan varias operaciones a la vez, por ejemplo, en MGET lugar de hacerlo. GET

Análisis del tiempo que se tarda en completar una sola solicitud

- En la red: `Tcpdump` y `Wireshark` (`tshark` en la línea de comandos) son herramientas útiles para saber cuánto tiempo tardó la solicitud en recorrer la red, activarse ElastiCache y recibir respuesta. En el próximo ejemplo se resalta una sola solicitud que se creó con el siguiente comando:

```
$ echo ping | nc example.xxxxxx.ng.0001.use1.cache.amazonaws.com 6379
+PONG
```

Del mismo modo que el comando anterior, `tcpdump` se encontraba en ejecución y volvió:

```
$ sudo tcpdump -i any -nn port 6379 -tt
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on any, link-type LINUX_SLL (Linux cooked), capture size 262144 bytes
1609428918.917869 IP 172.31.11.142.40966
```

```

> 172.31.11.247.6379: Flags [S], seq 177032944, win 26883, options [mss
8961,sackOK,TS val 27819440 ecr 0,nop,wscale 7], length 0
1609428918.918071 IP 172.31.11.247.6379 > 172.31.11.142.40966: Flags [S.], seq
53962565, ack 177032945, win
28960, options [mss 1460,sackOK,TS val 3788576332 ecr 27819440,nop,wscale 7],
length 0
1609428918.918091 IP 172.31.11.142.40966 > 172.31.11.247.6379: Flags [.], ack 1, win
211, options [nop,nop,TS val 27819440 ecr 3788576332], length 0
1609428918.918122
IP 172.31.11.142.40966 > 172.31.11.247.6379: Flags [P.], seq 1:6, ack 1, win 211,
options [nop,nop,TS val 27819440 ecr 3788576332], length 5: RESP "ping"
1609428918.918132 IP 172.31.11.142.40966 > 172.31.11.247.6379: Flags [F.], seq 6, ack
1, win 211, options [nop,nop,TS val 27819440 ecr 3788576332], length 0
1609428918.918240 IP 172.31.11.247.6379 > 172.31.11.142.40966: Flags [.], ack 6, win
227, options [nop,nop,TS val 3788576332 ecr 27819440], length 0
1609428918.918295
IP 172.31.11.247.6379 > 172.31.11.142.40966: Flags [P.], seq 1:8, ack 7, win 227,
options [nop,nop,TS val 3788576332 ecr 27819440], length 7: RESP "PONG"
1609428918.918300 IP 172.31.11.142.40966 > 172.31.11.247.6379: Flags [.], ack 8, win
211, options [nop,nop,TS val 27819441 ecr 3788576332], length 0
1609428918.918302 IP 172.31.11.247.6379 > 172.31.11.142.40966: Flags [F.], seq 8, ack
7, win 227, options [nop,nop,TS val 3788576332 ecr 27819440], length 0
1609428918.918307
IP 172.31.11.142.40966 > 172.31.11.247.6379: Flags [.], ack 9, win 211, options
[nop,nop,TS val 27819441 ecr 3788576332], length 0
^C
10 packets captured
10 packets received by filter
0 packets dropped by kernel

```

De la salida anterior podemos confirmar que el protocolo de establecimiento de comunicación de tres canales de TCP se completó en 222 microsegundos (918091 - 917869) y el comando ping se envió y devolvió en 173 microsegundos (918295 - 918122).

Desde la solicitud hasta la interrupción de la conexión pasaron 438 microsegundos (918307 - 917869). Esos resultados confirmarían que los tiempos de respuesta de la red y del motor son buenos por lo que la investigación puede centrarse en otros componentes.

- En el sistema operativo: `Strace` puede ayudar a identificar brechas de tiempo a nivel de sistema operativo. El análisis de las aplicaciones reales sería mucho más extenso y se aconseja utilizar depuradores o perfiles de aplicaciones especializados. El siguiente ejemplo solo muestra si los componentes del sistema operativo base funcionan como se esperaba, de lo contrario,

podría requerirse una investigación adicional. Usando el mismo PING comando OSS de Redis obtenemos trace:

```
$ echo ping | strace -f -tttt -r -e trace=execve,socket,open,recvfrom,sendto
nc example.xxxxxx.ng.0001.use1.cache.amazonaws.com (http://
example.xxxxxx.ng.0001.use1.cache.amazonaws.com/)
 6379
1609430221.697712 (+ 0.000000) execve("/usr/bin/nc", ["nc",
"example.xxxxxx.ng.0001.use"..., "6379"], 0x7ffffede7cc38 /* 22 vars */) = 0
1609430221.708955 (+ 0.011231) socket(AF_UNIX, SOCK_STREAM|SOCK_CLOEXEC|
SOCK_NONBLOCK, 0) = 3
1609430221.709084
(+ 0.000124) socket(AF_UNIX, SOCK_STREAM|SOCK_CLOEXEC|SOCK_NONBLOCK, 0) = 3
1609430221.709258 (+ 0.000173) open("/etc/nsswitch.conf", O_RDONLY|O_CLOEXEC) = 3
1609430221.709637 (+ 0.000378) open("/etc/host.conf", O_RDONLY|O_CLOEXEC) = 3
1609430221.709923
(+ 0.000286) open("/etc/resolv.conf", O_RDONLY|O_CLOEXEC) = 3
1609430221.711365 (+ 0.001443) open("/etc/hosts", O_RDONLY|O_CLOEXEC) = 3
1609430221.713293 (+ 0.001928) socket(AF_INET, SOCK_DGRAM|SOCK_CLOEXEC|SOCK_NONBLOCK,
IPPROTO_IP) = 3
1609430221.717419
(+ 0.004126) recvfrom(3, "\362|
\201\200\0\1\0\2\0\0\0\0\rnotls20201224\6tihew"..., 2048, 0, {sa_family=AF_INET,
sin_port=htons(53), sin_addr=inet_addr("172.31.0.2")}, [28->16]) = 155
1609430221.717890 (+ 0.000469) recvfrom(3,
"\204\207\201\200\0\1\0\1\0\0\0\0\rnotls20201224\6tihew"...,
65536, 0, {sa_family=AF_INET, sin_port=htons(53),
sin_addr=inet_addr("172.31.0.2")}, [28->16]) = 139
1609430221.745659 (+ 0.027772) socket(AF_INET, SOCK_STREAM, IPPROTO_TCP) = 3
1609430221.747548 (+ 0.001887) recvfrom(0, 0x7ffcf2f2ca50, 8192,
0, 0x7ffcf2f2c9d0, [128]) = -1 ENOTSOCK (Socket operation on non-socket)
1609430221.747858 (+ 0.000308) sendto(3, "ping\n", 5, 0, NULL, 0) = 5
1609430221.748048 (+ 0.000188) recvfrom(0, 0x7ffcf2f2ca50, 8192, 0, 0x7ffcf2f2c9d0,
[128]) = -1 ENOTSOCK
(Socket operation on non-socket)
1609430221.748330 (+ 0.000282) recvfrom(3, "+PONG\r\n", 8192, 0, 0x7ffcf2f2c9d0,
[128->0]) = 7
+PONG
1609430221.748543 (+ 0.000213) recvfrom(3, "", 8192, 0, 0x7ffcf2f2c9d0, [128->0]) = 0
1609430221.752110
(+ 0.003569) +++ exited with 0 +++
```

En el ejemplo anterior, el comando tardó un poco más de 54 milisegundos en completarse (752110 - 697712 = 54398 microsegundos).

Se necesitó una cantidad significativa de tiempo, cerca de 20 ms, para representar a nc y realizar la resolución de nombres (de 697712 a 717890). Luego, se necesitaron 2 ms para crear el socket de TCP (745659 a 747858) y 0,4 ms (747858 a 748330) a fin de enviar y recibir la respuesta para la solicitud.

Seguridad en Amazon ElastiCache

La seguridad en la nube AWS es la máxima prioridad. Como AWS cliente, usted se beneficia de una arquitectura de centro de datos y red diseñada para cumplir con los requisitos de las organizaciones más sensibles a la seguridad.

La seguridad es una responsabilidad compartida entre usted AWS y usted. El [modelo de responsabilidad compartida](#) la describe como seguridad de la nube y seguridad en la nube:

- Seguridad de la nube: AWS es responsable de proteger la infraestructura que ejecuta AWS los servicios en la AWS nube. AWS también le proporciona servicios que puede utilizar de forma segura. Auditores independientes prueban y verifican periódicamente la eficacia de nuestra seguridad en el marco de los [programas de conformidad de AWS](#). Para obtener más información sobre los programas de conformidad que se aplican a Amazon ElastiCache, consulta [AWS Servicios incluidos en el ámbito de aplicación por programa de conformidad](#).
- Seguridad en la nube: tu responsabilidad viene determinada por el AWS servicio que utilices. Usted también es responsable de otros factores, incluida la confidencialidad de los datos, los requisitos de la empresa y la legislación y los reglamentos aplicables.

Esta documentación te ayuda a entender cómo aplicar el modelo de responsabilidad compartida cuando utilizas Amazon ElastiCache. En los temas siguientes, se muestra cómo configurar Amazon ElastiCache para que cumpla con sus objetivos de seguridad y conformidad. También aprenderás a utilizar otros AWS servicios que te ayudan a supervisar y proteger tus ElastiCache recursos de Amazon.

Temas

- [Protección de datos en Amazon ElastiCache](#)
- [Privacidad del tráfico entre redes](#)
- [Identity and Access Management para Amazon ElastiCache](#)
- [Validación de conformidad para Amazon ElastiCache](#)
- [Resiliencia en Amazon ElastiCache](#)
- [Seguridad de la infraestructura en ElastiCache de AWS](#)
- [Actualizaciones de servicios en ElastiCache](#)
- [Vulnerabilidades y exposiciones comunes \(CVE\): vulnerabilidades de seguridad abordadas en ElastiCache \(Redis OSS\)](#)

Protección de datos en Amazon ElastiCache

El [modelo de responsabilidad compartida](#) de AWS se aplica a la protección de datos de ElastiCache de AWS (ElastiCache). Tal como se describe en este modelo, AWS es responsable de proteger la infraestructura global que ejecuta toda la nube de AWS. Usted es responsable de mantener el control sobre el contenido alojado en esta infraestructura. Este contenido incluye la configuración de seguridad y las tareas de administración de los servicios de AWS que usted utiliza. Para obtener más información sobre la privacidad de datos, consulte las [Preguntas frecuentes sobre la privacidad de datos](#).

Para fines de protección de datos, recomendamos proteger las credenciales de la cuenta de AWS y configurar cuentas individuales con AWS Identity and Access Management (IAM). De esta manera, cada usuario recibe únicamente los permisos necesarios para cumplir con sus obligaciones laborales. También recomendamos proteger sus datos de las siguientes maneras:

- Utilice autenticación multifactor (MFA) en cada cuenta.
- Utilice TLS para comunicarse con los recursos de AWS.
- Configure la API y el registro de actividad del usuario con AWS CloudTrail.
- Utilice las soluciones de cifrado de AWS, junto con todos los controles de seguridad predeterminados dentro de los servicios de AWS.
- Utilice avanzados servicios de seguridad administrados, como Amazon Macie, que lo ayuden a detectar y proteger los datos personales almacenados en Amazon S3.

Le recomendamos encarecidamente que nunca introduzca información de identificación confidencial, como, por ejemplo, números de cuenta de sus clientes, en los campos de formato libre, como el campo Nombre. Esto incluye cuando se trabaja con ElastiCache u otros servicios de AWS con la consola, la API, la AWS CLI o los SDK de AWS. Es posible que cualquier dato que escriba en ElastiCache o en otros servicios se incluya en los registros de diagnóstico. Cuando proporcione una URL a un servidor externo, no incluya información de credenciales en la URL para validar la solicitud para ese servidor.

Temas

- [Seguridad de datos en Amazon ElastiCache](#)

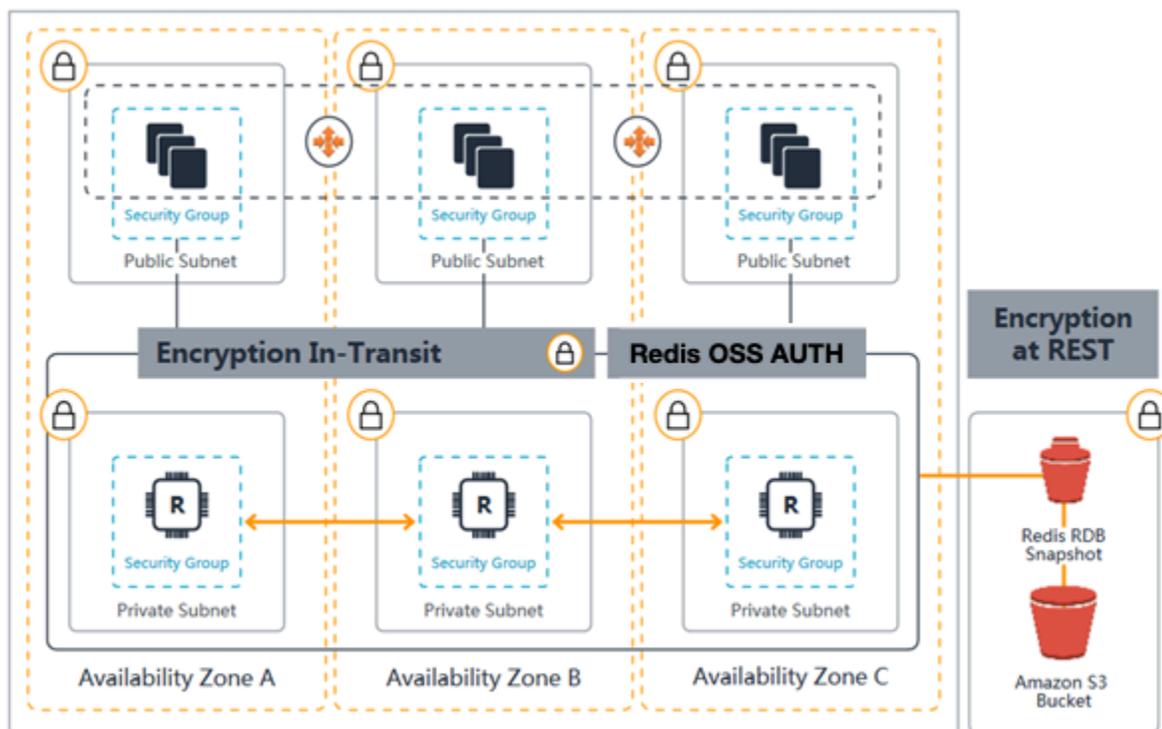
Seguridad de datos en Amazon ElastiCache

Para ayudar a mantener sus datos seguros, Amazon ElastiCache y Amazon EC2 proporcionan mecanismos para protegerlos del acceso no autorizado a sus datos en el servidor.

Amazon ElastiCache (Redis OSS) proporciona funciones de cifrado para los datos de las cachés que ejecutan las versiones 3.2.6 (programada para EOL, consulte el [calendario de fin de vida de las versiones de Redis OSS](#)), 4.0.10 o posteriores:

- El cifrado en tránsito cifra los datos mientras se mueven de un lugar a otro; por ejemplo, entre los nodos del clúster o entre la caché y la aplicación.
- El cifrado en reposo cifra los datos en el disco durante las operaciones de sincronización y copia de seguridad.

Amazon ElastiCache (Redis OSS) también admite la autenticación de usuarios con IAM o Redis OSS AUTH, y la autorización de las operaciones de los usuarios mediante el control de acceso basado en roles (RBAC).



ElastiCache Diagrama de seguridad (Redis OSS)

Temas

- [ElastiCache cifrado en tránsito \(\) TLS](#)
- [Cifrado en reposo en ElastiCache](#)
- [Autenticación y autorización](#)

ElastiCache cifrado en tránsito () TLS

Para ayudar a mantener sus datos seguros, Amazon ElastiCache y Amazon EC2 proporcionan mecanismos para protegerlos del acceso no autorizado a sus datos en el servidor. Al proporcionar la capacidad de cifrado en tránsito, ElastiCache le brinda una herramienta que puede usar para ayudar a proteger sus datos cuando se mueven de un lugar a otro.

Todas las cachés sin servidor tienen activado el cifrado en tránsito. En el caso de los clústeres de diseño propio, puede habilitar el cifrado en tránsito en un grupo de replicación configurando el parámetro `TransitEncryptionEnabled` en `true` (CLI: `--transit-encryption-enabled`) al crear el grupo de replicación. Puede hacerlo tanto si va a crear el grupo de replicación mediante el AWS Management Console AWS CLI, el o el. ElastiCache API

Temas

- [Información general sobre el cifrado en tránsito](#)
- [Condiciones del cifrado en tránsito](#)
- [Prácticas recomendadas de cifrado en tránsito](#)
- [Véase también](#)
- [Habilitación del cifrado en tránsito](#)
- [Conexión a Amazon ElastiCache \(RedisOSS\) con cifrado en tránsito mediante redis-cli](#)
- [Habilitación del cifrado en tránsito en un clúster Redis OSS de diseño propio mediante Python](#)
- [Procedimientos recomendados para habilitar el cifrado en tránsito](#)

Información general sobre el cifrado en tránsito

El cifrado ElastiCache en tránsito de Amazon es una función que te permite aumentar la seguridad de tus datos en los puntos más vulnerables, cuando están en tránsito de un lugar a otro. Como se requiere cierto procesamiento para cifrar y descifrar los datos en los puntos de enlace, habilitar el cifrado en tránsito puede afectar al rendimiento. Debe comparar los datos con y sin cifrado en tránsito para determinar el impacto en el rendimiento de los casos de uso.

ElastiCache el cifrado en tránsito implementa las siguientes funciones:

- Conexiones de cliente cifradas: las conexiones de los clientes a los nodos de caché están TLS cifradas.
- Conexiones de servidor cifradas: los datos que se mueven entre los nodos de un clúster están cifrados.
- Autenticación de servidores: los clientes pueden autenticar que se encuentran conectados al servidor correcto.
- Autenticación de clientes: mediante la OSS AUTH función Redis, el servidor puede autenticar los clientes.

Condiciones del cifrado en tránsito

Al planificar la implementación de un clúster de diseño propio, debes tener en cuenta las siguientes restricciones del cifrado en ElastiCache tránsito de Amazon:

- El cifrado en tránsito es compatible con los grupos de replicación que ejecutan OSS las versiones 3.2.6, 4.0.10 y posteriores de Redis.
- Los grupos de replicación que ejecutan la versión 7 y versiones posteriores de Redis admiten la modificación de la configuración de cifrado en tránsito para un clúster existente. OSS
- El cifrado en tránsito solo se admite para los grupos de replicación que se ejecutan en AmazonVPC.
- El cifrado en tránsito no es compatible con los grupos de replicación que ejecutan los siguientes tipos de nodos: M1, M2.

Para obtener más información, consulte [Tipos de nodos compatibles](#).

- El cifrado en tránsito se habilita estableciendo explícitamente el parámetro `TransitEncryptionEnabled` en `true`.
- Asegúrese de que su cliente de almacenamiento en caché sea compatible con la TLS conectividad y de que la haya habilitado en la configuración del cliente.
- El uso de las versiones TLS 1.0 y TLS 1.1 antiguas está obsoleto en todas AWS las regiones para las ElastiCache versiones 6 y posteriores. ElastiCache seguirá siendo compatible con las versiones TLS 1.0 y 1.1 hasta el 8 de mayo de 2025. Los clientes deben actualizar su software de cliente antes de esa fecha.

Prácticas recomendadas de cifrado en tránsito

- Debido al procesamiento requerido para cifrar y descifrar los datos en los puntos de enlace, la implementación del cifrado en tránsito puede reducir el rendimiento. Compare sus datos con y sin cifrado en tránsito para determinar el impacto en el rendimiento de la implementación.
- Dado que crear nuevas conexiones puede resultar caro, puede reducir el impacto del cifrado en tránsito en el rendimiento si conserva las conexiones. TLS

Véase también

- [Cifrado en reposo en ElastiCache](#)
- [Autenticación con el comando REDIS OSS AUTH](#)
- [Autenticación de usuarios con un control de acceso basado en roles \(\) RBAC](#)
- [Seguridad de ElastiCache y Amazon VPC](#)
- [Identity and Access Management para Amazon ElastiCache](#)

Habilitación del cifrado en tránsito

Todas las cachés sin servidor tienen activado el cifrado en tránsito. En un clúster de diseño propio, puede habilitar el cifrado en tránsito mediante la AWS Management Console AWS CLI o la ElastiCache API.

Habilitar el cifrado en tránsito mediante AWS Management Console

Habilitar el cifrado en tránsito para un nuevo clúster de diseño propio mediante AWS Management Console

Al diseñar su propia caché, las configuraciones de “desarrollo/prueba” y “producción” con el método “Creación sencilla” tienen el cifrado en tránsito habilitado. Al elegir la configuración, seleccione estas opciones:

- Elija la versión del motor 3.2.6, 4.0.10 o posterior.
- Haga clic en la casilla de verificación situada junto a Habilitar en la opción Cifrado en tránsito.

Para ver el step-by-step proceso, consulte lo siguiente:

- [Creación de un clúster de Redis OSS \(modo de clúster desactivado\) \(consola\)](#)

- [Creación de un clúster de Redis OSS \(modo de clúster activado\) \(consola\)](#)

Habilitar el cifrado en tránsito para un clúster de diseño propio existente mediante AWS Management Console

Habilitar el cifrado en tránsito es un proceso de dos pasos, primero debe configurar el modo de cifrado de tránsito en `preferred`. Este modo permite que sus clientes OSS de Redis se conecten mediante conexiones cifradas y no cifradas. Tras migrar todos sus clientes OSS de Redis para utilizar conexiones cifradas, podrá modificar la configuración del clúster para establecer el modo de cifrado de tránsito en `required`. Al configurar el modo de cifrado de tránsito en `required` se eliminarán todas las conexiones no cifradas y solo se permitirán las conexiones cifradas.

Paso 1: establezca el Transit encryption mode (Modo de cifrado de tránsito) como Preferred (Preferido)

1. Inicia sesión en la ElastiCache consola de Amazon AWS Management Console y ábrela en <https://console.aws.amazon.com/elasticache/>.
2. Elija las cachés OSS de Redis de los ElastiCache recursos que aparecen en el panel de navegación, en la parte izquierda.
3. Elija la caché OSS de Redis que desee actualizar.
4. Elija Actions (Acciones) y después Modify (Modificar).
5. Elija Enable (Habilitar) en Encryption in transit (Cifrado en tránsito), en la sección Security (Seguridad).
6. Elija Preferred (Preferido) como Transit encryption mode (Modo de cifrado de tránsito).
7. Elija Preview changes (Vista previa de cambios) y guarde los cambios.

Después de migrar todos sus clientes OSS de Redis para utilizar conexiones cifradas:

Paso 2: establezca el Transit encryption mode (Modo de cifrado de tránsito) en Preferred (preferido)

1. Inicia sesión en la ElastiCache consola de Amazon AWS Management Console y ábrela en <https://console.aws.amazon.com/elasticache/>.
2. Elija las cachés OSS de Redis de los ElastiCache recursos que aparecen en el panel de navegación, en la parte izquierda.
3. Elija la caché OSS de Redis que desee actualizar.
4. Elija Actions (Acciones) y después Modify (Modificar).

5. Elija **Required (Obligatorio)** como **Transit encryption mode (Modo de cifrado de tránsito)**, en la sección **Security (Seguridad)**.
6. Elija **Preview changes (Vista previa de cambios)** y guarde los cambios.

Habilitar el cifrado en tránsito mediante AWS CLI

Para habilitar el cifrado en tránsito al crear un grupo de replicación de Redis OSS mediante el AWS CLI, utilice el parámetro `transit-encryption-enabled`

Habilitar el cifrado en tránsito en un nuevo clúster de diseño propio para Redis OSS (modo de clúster desactivado) (CLI)

Utilice la AWS CLI operación `create-replication-group` y los siguientes parámetros para crear un grupo de replicación de Redis OSS con réplicas que tengan activado el cifrado en tránsito:

Parámetros clave:

- **--engine**: debe ser `redis`.
- **--engine-version**: debe ser 3.2.6, 4.0.10 o posterior.
- **--transit-encryption-enabled**: obligatorio. Si habilita el cifrado en tránsito, también deberá proporcionar un valor para el parámetro `--cache-subnet-group`.
- **--num-cache-clusters**: debe ser, como mínimo, 1. El valor máximo para este parámetro es seis.

Para más información, consulte los siguientes temas:

- [Crear un grupo de replicación de Redis OSS \(modo de clúster desactivado\) desde cero \(AWS CLI\)](#)
- [create-replication-group](#)

Habilitar el cifrado en tránsito en un nuevo clúster de diseño propio para Redis OSS (modo de clúster activado) (CLI)

Utilice la AWS CLI operación `create-replication-group` y los siguientes parámetros para crear un grupo de replicación de Redis OSS (habilitado en modo de clúster) que tenga activado el cifrado en tránsito:

Parámetros clave:

- **--engine**: debe ser `redis`.
- **--engine-version**: debe ser 3.2.6, 4.0.10 o posterior.
- **--transit-encryption-enabled**: obligatorio. Si habilita el cifrado en tránsito, también deberá proporcionar un valor para el parámetro `--cache-subnet-group`.
- Utilice uno de los siguientes conjuntos de parámetros para especificar la configuración de los grupos de nodos del grupo de reproducción:
 - **--num-node-groups**: especifica el número de particiones (grupos de nodos) de este grupo de reproducción. El valor máximo de este parámetro es 500.
 - **--replicas-per-node-group**: especifica el número de nodos de réplica de cada grupo de nodos. El valor especificado aquí se aplica a todos los fragmentos de este grupo de reproducción. El valor máximo de este parámetro es 5.
 - **--node-group-configuration**: especifica por separado la configuración de cada partición.

Para más información, consulte los siguientes temas:

- [Crear un grupo de replicación de Redis OSS \(modo de clúster activado\) desde cero \(AWS CLI\)](#)
- [create-replication-group](#)

Habilitación del cifrado en tránsito para un clúster existente con la AWS CLI

Habilitar el cifrado en tránsito es un proceso de dos pasos, primero debe configurar el modo de cifrado de tránsito en `preferred`. Este modo permite que sus clientes de Redis OSS se conecten mediante conexiones cifradas y no cifradas. Tras migrar todos sus clientes OSS de Redis para utilizar conexiones cifradas, podrá modificar la configuración del clúster para establecer el modo de cifrado de tránsito en `required`. Al configurar el modo de cifrado de tránsito en `required` se eliminarán todas las conexiones no cifradas y solo se permitirán las conexiones cifradas.

Utilice la AWS CLI operación `modify-replication-group` y los siguientes parámetros para actualizar un grupo de replicación de Redis OSS (habilitado en modo de clúster) que tenga desactivado el cifrado en tránsito.

Para habilitar el cifrado en tránsito

1. `transit-encryption-mode` Configúrelo en `preferred`, mediante los siguientes parámetros

- **--transit-encryption-enabled**: obligatorio.
 - **--transit-encryption-mode**: debe establecerse en `preferred`.
2. `transit-encryption-mode required` Configúrelo en, mediante los siguientes parámetros:
- **--transit-encryption-enabled**: obligatorio.
 - **--transit-encryption-mode**: debe establecerse en `required`.

Conexión a Amazon ElastiCache (RedisOSS) con cifrado en tránsito mediante redis-cli

Para acceder a los datos de las cachés ElastiCache (RedisOSS) habilitadas con el cifrado en tránsito, utilice clientes que funcionan con Secure Socket Layer (SSL). También puede usar `redis-cli con/TLS` en Amazon Linux y Amazon Linux 2. Si su cliente no lo admite, puede usar el `stunnel` comando del host del cliente para crear un SSL túnel hacia los nodos de Redis. OSS

Conexión cifrada con Linux

Para usar `redis-cli` para conectarse a un OSS clúster de Redis habilitado con cifrado en tránsito en Amazon Linux 2023, Amazon Linux 2 o Amazon Linux, siga estos pasos.

1. Descargue y compile la utilidad `redis-cli`. Esta utilidad se incluye en la distribución del software Redis. OSS
2. En la línea de comandos de la EC2 instancia, escriba los comandos correspondientes a la versión de Linux que esté utilizando.

Amazon Linux 2023

Si utiliza Amazon Linux 2023, introduzca lo siguiente:

```
sudo yum install redis6 -y
```

A continuación, escriba el siguiente comando y sustituya el punto final del clúster y el puerto por lo que se muestra en este ejemplo.

```
redis-cli -h Primary or Configuration Endpoint --tls -p 6379
```

Para obtener más información sobre cómo encontrar el punto de conexión, consulte [Encontrar los puntos de conexión de los nodos](#).

Amazon Linux 2

Si utiliza Amazon Linux 2, introduzca lo siguiente:

```
sudo yum -y install openssl-devel gcc
wget https://download.redis.io/releases/redis-7.2.5.tar.gz
tar xvzf redis-7.2.5.tar.gz
cd redis-7.2.5
make distclean
make redis-cli BUILD_TLS=yes
sudo install -m 755 src/redis-cli /usr/local/bin/
```

Amazon Linux

Si utiliza Amazon Linux, introduzca lo siguiente:

```
sudo yum install gcc jemalloc-devel openssl-devel tcl tcl-devel clang wget
wget http://download.redis.io/redis-7.2.5.tar.gz
tar xvzf redis-7.2.5.tar.gz
cd redis-7.2.5
make redis-cli CC=clang BUILD_TLS=yes
sudo install -m 755 src/redis-cli /usr/local/bin/
```

En Amazon Linux, es posible que también deba ejecutar los siguientes pasos adicionales:

```
sudo yum install clang
CC=clang make
sudo make install
```

3. Tras descargar e instalar la utilidad redis-cli, se recomienda ejecutar el comando opcional.
make-test
4. Para conectarse a un clúster con el cifrado y la autenticación habilitados, introduzca este comando:

```
redis-cli -h Primary or Configuration Endpoint --tls -a 'your-password' -p 6379
```

Note

Si instala redis6 en Amazon Linux 2023, ahora puede usar `redis6-cli` el comando en lugar de: `redis-cli`

```
redis6-cli -h Primary or Configuration Endpoint --tls -p 6379
```

Conexión cifrada con stunnel

Para usar `redis-cli` para conectarse a un OSS clúster de Redis habilitado con cifrado en tránsito mediante `stunnel`, sigue estos pasos.

1. Úselo para conectarse SSH a su cliente e instalarlo. `stunnel`

```
sudo yum install stunnel
```

2. Ejecute el siguiente comando para crear y editar un archivo `/etc/stunnel/redis-cli.conf` simultáneamente y añadir un punto final de clúster ElastiCache (RedisOSS) a uno o más parámetros de conexión, utilizando el resultado que se proporciona a continuación como plantilla.

```
vi /etc/stunnel/redis-cli.conf

fips = no
setuid = root
setgid = root
pid = /var/run/stunnel.pid
debug = 7
delay = yes
options = NO_SSLv2
options = NO_SSLv3
[redis-cli]
  client = yes
  accept = 127.0.0.1:6379
  connect = primary.ssltest.wif01h.use1.cache.amazonaws.com:6379
[redis-cli-replica]
  client = yes
  accept = 127.0.0.1:6380
```

```
connect = ssltest-02.ssltest.wif01h.use1.cache.amazonaws.com:6379
```

En este ejemplo, el archivo de configuración tiene dos conexiones, `redis-cli` y `redis-cli-replica`. Los parámetros se establecen como sigue:

- `client` se establece en sí para especificar que esta instancia de stunnel es un cliente.
- `accept` se establece en la IP del cliente. En este ejemplo, el puerto principal tiene el OSS valor 127.0.0.1 predeterminado de Redis en el puerto 6379. La réplica debe llamar a otro puerto y establecerlo en 6380. Puede utilizar los puertos efímeros 1024-65535. Para obtener más información, consulta [Puertos efímeros](#) en la Guía VPC del usuario de Amazon.
- `connect` está configurado en el punto final del servidor RedisOSS. Para obtener más información, consulte [Búsqueda de puntos de conexión](#).

3. Inicie stunnel.

```
sudo stunnel /etc/stunnel/redis-cli.conf
```

Use el comando `netstat` para confirmar que los túneles se han iniciado.

```
sudo netstat -tulnp | grep -i stunnel

tcp        0      0 127.0.0.1:6379          0.0.0.0:*               LISTEN
           3189/stunnel
tcp        0      0 127.0.0.1:6380          0.0.0.0:*               LISTEN
           3189/stunnel
```

4. Conéctese al OSS nodo Redis cifrado mediante el punto final local del túnel.

- Si no se utilizó ninguna AUTH contraseña durante la creación del clúster ElastiCache (RedisOSS), en este ejemplo se utiliza `redis-cli` para conectarse al servidor ElastiCache (RedisOSS) mediante la ruta completa de `redis-cli`, en Amazon Linux:

```
/home/ec2-user/redis-7.2.5.tar.gz/src/redis-cli -h localhost -p 6379
```

Si se utilizó la AUTH contraseña durante la creación del OSS clúster de Redis, en este ejemplo se utiliza `redis-cli` para conectarse al OSS servidor de Redis mediante la ruta completa de `redis-cli`, en Amazon Linux:

```
/home/ec2-user/redis-7.2.5.tar.gz/src/redis-cli -h localhost -p 6379 -a my-secret-password
```

OR

- Cambie el directorio a redis-7.2.5 y haga lo siguiente:

Si no se utilizó ninguna AUTH contraseña durante la creación del clúster ElastiCache (RedisOSS), en este ejemplo se utiliza redis-cli para conectarse al servidor ElastiCache (RedisOSS) mediante la ruta completa de redis-cli, en Amazon Linux:

```
src/redis-cli -h localhost -p 6379
```

Si se utilizó la AUTH contraseña durante la creación del OSS clúster de Redis, en este ejemplo se utiliza redis-cli para conectarse al OSS servidor de Redis mediante la ruta completa de redis-cli, en Amazon Linux:

```
src/redis-cli -h localhost -p 6379 -a my-secret-password
```

En este ejemplo, se utiliza Telnet para conectarse al servidor Redis. OSS

```
telnet localhost 6379

Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
auth MySecretPassword
+OK
get foo
$3
bar
```

5. Para detener y cerrar los SSL túneles, `kill` el proceso de túnel.

```
sudo kill stunnel
```

Habilitación del cifrado en tránsito en un clúster Redis OSS de diseño propio mediante Python

La siguiente guía mostrará cómo habilitar el cifrado en tránsito en un clúster de Redis OSS 7.0 que se creó originalmente con el cifrado en tránsito desactivado. Los clientes de TCP y TLS seguirán comunicándose con el clúster durante este proceso sin tiempo de inactividad.

Boto3 obtendrá las credenciales que necesita

(`aws_access_key_id`, `aws_secret_access_key` y `aws_session_token`) de las variables de entorno. Esas credenciales se pegarán previamente en el mismo terminal bash donde ejecutaremos `python3` para procesar el código Python que se muestra en esta guía. El código del ejemplo siguiente se procesó a partir de una instancia EC2 que se lanzó en la misma VPC que se utilizará para crear ElastiCache el clúster OSS de Redis en ella.

Note

- En los ejemplos siguientes, se utiliza el SDK boto3 para las operaciones de ElastiCache administración (creación de clústeres o usuarios) y `redis-py/ redis-py-cluster` para la gestión de datos.
- Debe utilizar al menos la versión 1.26.39 de boto3 (=~) para utilizar la migración de TLS en línea con la API de modificación del clúster.
- ElastiCache admite la migración de TLS en línea solo para los clústeres OSS de Redis con la versión 7.0 o superior. Por lo tanto, si tiene un clúster que ejecuta una versión de Redis OSS anterior a la 7.0, tendrá que actualizar la versión de Redis OSS de su clúster. Para obtener más información sobre las diferencias de las versiones, consulte [Diferencias principales de compatibilidad y comportamiento de versiones](#).

Temas

- [Defina las constantes de cadena que lanzarán el clúster de Redis OSS ElastiCache](#)
- [Definir las clases para la configuración del clúster](#)
- [Definir una clase que representará el propio clúster](#)
- [\(Opcional\) Cree una clase contenedora para demostrar la conexión del cliente al clúster OSS de Redis](#)
- [Cree la función principal que muestre el proceso de cambio de la configuración de cifrado en tránsito](#)

Defina las constantes de cadena que lanzarán el clúster de Redis OSS ElastiCache

Primero, definamos algunas constantes de cadena de Python simples que contendrán los nombres de las AWS entidades necesarias para crear el ElastiCache clúster `security-group`, como `Cache Subnet group`, y `a default parameter group`. Todas estas AWS entidades deben crearse con antelación en su AWS cuenta en la región que desee utilizar.

```
#Constants definitions
SECURITY_GROUP = "sg-0492aa0a29c558427"
CLUSTER_DESCRIPTION = "This cluster has been launched as part of the online TLS
migration user guide"
EC_SUBNET_GROUP = "client-testing"
DEFAULT_PARAMETER_GROUP_REDIS_7_CLUSTER_MODE_ENABLED = "default.redis7.cluster.on"
```

Definir las clases para la configuración del clúster

Ahora, definamos algunas clases sencillas de Python que representarán la configuración de un clúster, que contendrá los metadatos sobre el clúster, como la versión OSS de Redis, el tipo de instancia y si el cifrado en tránsito (TLS) está habilitado o deshabilitado.

```
#Class definitions

class Config:
    def __init__(
        self,
        instance_type: str = "cache.t4g.small",
        version: str = "7.0",
        multi_az: bool = True,
        TLS: bool = True,
        name: str = None,
    ):
        self.instance_type = instance_type
        self.version = version
        self.multi_az = multi_az
        self.TLS = TLS
        self.name = name or f"tls-test"

    def create_base_launch_request(self):
        return {
            "ReplicationGroupId": self.name,
            "TransitEncryptionEnabled": self.TLS,
            "MultiAZEnabled": self.multi_az,
```

```

        "CacheNodeType": self.instance_type,
        "Engine": "redis",
        "EngineVersion": self.version,
        "CacheSubnetGroupName": EC_SUBNET_GROUP ,
        "CacheParameterGroupName":
DEFAULT_PARAMETER_GROUP_REDIS_7_CLUSTER_MODE_ENABLED ,
        "ReplicationGroupDescription": CLUSTER_DESCRIPTION,
        "SecurityGroupIds": [SECURITY_GROUP],
    }

```

```

class ConfigCME(Config):
    def __init__(
        self,
        instance_type: str = "cache.t4g.small",
        version: str = "7.0",
        multi_az: bool = True,
        TLS: bool = True,
        name: str = None,
        num_shards: int = 2,
        num_replicas_per_shard: int = 1,
    ):
        super().__init__(instance_type, version, multi_az, TLS, name)
        self.num_shards = num_shards
        self.num_replicas_per_shard = num_replicas_per_shard

    def create_launch_request(self) -> dict:
        launch_request = self.create_base_launch_request()
        launch_request["NumNodeGroups"] = self.num_shards
        launch_request["ReplicasPerNodeGroup"] = self.num_replicas_per_shard
        return launch_request

```

Definir una clase que representará el propio clúster

Ahora, definamos algunas clases sencillas de Python que representarán el propio clúster ElastiCache OSS de Redis. Esta clase tendrá un campo de cliente que contendrá un cliente boto3 para las operaciones de ElastiCache administración, como la creación del clúster y la consulta de la API. ElastiCache

```

import botocore.config
import boto3

# Create boto3 client
def init_client(region: str = "us-east-1"):

```

```
config = boto3.config.Config(retries={"max_attempts": 10, "mode": "standard"})
init_request = dict()
init_request["config"] = config
init_request["service_name"] = "elasticache"
init_request["region_name"] = region
return boto3.client(**init_request)
```

```
class ElastiCacheClusterBase:
    def __init__(self, name: str):
        self.name = name
        self.elasticache_client = init_client()

    def get_first_replication_group(self):
        return self.elasticache_client.describe_replication_groups(
            ReplicationGroupId=self.name
        )["ReplicationGroups"][0]

    def get_status(self) -> str:
        return self.get_first_replication_group()["Status"]

    def get_transit_encryption_enabled(self) -> bool:
        return self.get_first_replication_group()["TransitEncryptionEnabled"]

    def is_available(self) -> bool:
        return self.get_status() == "available"

    def is_modifying(self) -> bool:
        return self.get_status() == "modifying"

    def wait_for_available(self):
        while True:
            if self.is_available():
                break
            else:
                time.sleep(5)

    def wait_for_modifying(self):
        while True:
            if self.is_modifying():
                break
            else:
                time.sleep(5)
```

```

def delete_cluster(self) -> bool:
    self.elasticache_client.delete_replication_group(
        ReplicationGroupId=self.name, RetainPrimaryCluster=False
    )

def modify_transit_encryption_mode(self, new_transit_encryption_mode: str):
    # generate api call to migrate the cluster to TLS preferred or to TLS required
    self.elasticache_client.modify_replication_group(
        ReplicationGroupId=self.name,
        TransitEncryptionMode=new_transit_encryption_mode,
        TransitEncryptionEnabled=True,
        ApplyImmediately=True,
    )
    self.wait_for_modifying()

class ElastiCacheClusterCME(ElastiCacheClusterBase):
    def __init__(self, name: str):
        super().__init__(name)

    @classmethod
    def launch(cls, config: ConfigCME = None) -> ElastiCacheClusterCME:
        config = config or ConfigCME()
        print(config)
        new_cluster = ElastiCacheClusterCME(config.name)
        launch_request = config.create_launch_request()
        new_cluster.elasticache_client.create_replication_group(**launch_request)
        new_cluster.wait_for_available()
        return new_cluster

    def get_configuration_endpoint(self) -> str:
        return self.get_first_replication_group()["ConfigurationEndpoint"]["Address"]

#Since the code can throw exceptions, we define this class to make the code more
#readable and
#so we won't forget to delete the cluster
class ElastiCacheCMEManager:
    def __init__(self, config: ConfigCME = None):
        self.config = config or ConfigCME()

    def __enter__(self) -> ElastiCacheClusterCME:
        self.cluster = ElastiCacheClusterCME.launch(self.config)
        return self.cluster

    def __exit__(self, exc_type, exc_val, exc_tb):

```

```
self.cluster.delete_cluster()
```

(Opcional) Cree una clase contenedora para demostrar la conexión del cliente al clúster OSS de Redis

Ahora, vamos a crear una clase contenedora para el cliente de `redis-py-cluster`. Esta clase contenedora permitirá rellenar previamente el clúster con algunas claves y luego ejecutar comandos `get` repetidos aleatorios.

Note

Este es un paso opcional, pero simplifica el código de la función principal que viene en un paso posterior.

```
import redis
import random
from time import perf_counter_ns, time

class DowntimeTestClient:
    def __init__(self, client):
        self.client = client

        # num of keys prefilled
        self.prefilled = 0
        # percent of get above prefilled
        self.percent_get_above_prefilled = 10 # nil result expected when get hit above
prefilled
        # total downtime in nano seconds
        self.downtime_ns = 0
        # num of success and fail operations
        self.success_ops = 0
        self.fail_ops = 0
        self.connection_errors = 0
        self.timeout_errors = 0

    def replace_client(self, client):
        self.client = client

    def prefill_data(self, timelimit_sec=60):
```

```

    end_time = time() + timelimit_sec
    while time() < end_time:
        self.client.set(self.prefilled, self.prefilled)
        self.prefilled += 1

# unsuccessful operations throw exceptions
def _exec(self, func):
    try:
        start_ns = perf_counter_ns()
        func()
        self.success_ops += 1
        elapsed_ms = (perf_counter_ns() - start_ns) // 10 ** 6
        # upon succesful execution of func
        # reset random_key to None so that the next command
        # will use a new random key
        self.random_key = None

    except Exception as e:
        elapsed_ns = perf_counter_ns() - start_ns
        self.downtime_ns += elapsed_ns
        # in case of failure- increment the relevant counters so that we will keep
track
        # of how many connection issues we had while trying to communicate with
        # the cluster.
        self.fail_ops += 1
        if e.__class__ is redis.exceptions.ConnectionError:
            self.connection_errors += 1
        if e.__class__ is redis.exceptions.TimeoutError:
            self.timeout_errors += 1

def _repeat_exec(self, func, seconds):
    end_time = time() + seconds
    while time() < end_time:
        self._exec(func)

def _new_random_key_if_needed(self, percent_above_prefilled):
    if self.random_key is None:
        max = int((self.prefilled * (100 + percent_above_prefilled)) / 100)
        return random.randint(0, max)
    return self.random_key

def _random_get(self):
    key = self._new_random_key_if_needed(self.percent_get_above_prefilled)
    result = self.client.get(key)

```

```
# we know the key was set for sure only in the case key < self.prefilled
if key < self.prefilled:
    assert result.decode("UTF-8") == str(key)

def repeat_get(self, seconds=60):
    self._repeat_exec(self._random_get, seconds)

def get_downtime_ms(self) -> int:
    return self.downtime_ns // 10 ** 6

def do_get_until(self, cond_check):
    while not cond_check():
        self.repeat_get()
    # do one more get cycle once condition is met
    self.repeat_get()
```

Cree la función principal que muestre el proceso de cambio de la configuración de cifrado en tránsito

Ahora, definamos la función principal, que hará lo siguiente:

1. Cree el clúster con el cliente ElastiCache boto3.
2. Inicialice el cliente de `redis-py-cluster` que se conectará al clúster con una conexión TCP clara sin TLS.
3. El cliente de `redis-py-cluster` rellena previamente el clúster con algunos datos.
4. El cliente de boto3 activará la migración de TLS de sin TLS a TLS preferido.
5. Mientras se migra el clúster a TLS Preferred, el cliente de TCP de `redis-py-cluster` enviará operaciones `get` repetidas al clúster hasta que finalice la migración.
6. Una vez finalizada la migración a TLS Preferred, confirmaremos que el clúster admite el cifrado en tránsito. Después, crearemos un cliente de `redis-py-cluster` que se conectará al clúster con TLS.
7. Enviaremos algunos comandos `get` utilizando el nuevo cliente de TLS y el antiguo cliente TCP.
8. El cliente de boto3 activará la migración de TLS de TLS Preferred a TLS requerido.
9. Mientras se migra el clúster a TLS (es necesario), el cliente `redis-py-cluster` TLS enviará `get` operaciones repetidas al clúster hasta que finalice la migración.

```
import redis
```

```
def init_cluster_client(
    cluster: ElastiCacheClusterCME, prefill_data: bool, TLS: bool = True) ->
DowntimeTestClient:
    # we must use for the host name the cluster configuration endpoint.
    redis_client = redis.RedisCluster(
        host=cluster.get_configuration_endpoint(), ssl=TLS, socket_timeout=0.25,
socket_connect_timeout=0.1
    )
    test_client = DowntimeTestClient(redis_client)
    if prefill_data:
        test_client.prefill_data()
    return test_client

if __name__ == '__main__':
    config = ConfigCME(TLS=False, instance_type="cache.m5.large")

    with ElastiCacheCMEManager(config) as cluster:
        # create a client that will connect to the cluster with clear tcp connection
        test_client_tcp = init_cluster_client(cluster, prefill_data=True, TLS=False)

        # migrate the cluster to TLS Preferred
        cluster.modify_transit_encryption_mode(new_transit_encryption_mode="preferred")

        # do repeated get commands until the cluster finishes the migration to TLS
        Preferred
        test_client_tcp.do_get_until(cluster.is_available)

        # verify that in transit encryption is enabled so that clients will be able to
        connect to the cluster with TLS
        assert cluster.get_transit_encryption_enabled() == True

        # create a client that will connect to the cluster with TLS connection.
        # we must first make sure that the cluster indeed supports TLS
        test_client_tls = init_cluster_client(cluster, prefill_data=True, TLS=True)

        # by doing get commands with the tcp client for 60 more seconds
        # we can verify that the existing tcp connection to the cluster still works
        test_client_tcp.repeat_get(seconds=60)

        # do get commands with the new TLS client for 60 more seconds
        test_client_tcp.repeat_get(seconds=60)

        # migrate the cluster to TLS required
```

```
cluster.modify_transit_encryption_mode(new_transit_encryption_mode="required")

# from this point the tcp clients will be disconnected and we must not use them
anymore.
# do get commands with the TLS client until the cluster finishes migration to
TLS required mode.
test_client_tls.do_get_until(cluster.is_available)
```

Procedimientos recomendados para habilitar el cifrado en tránsito

Antes de habilitar el cifrado en tránsito: asegúrese de gestionar correctamente los registros de DNS

Note

Durante este proceso, cambiaremos y eliminaremos los puntos de conexión antiguos. El uso incorrecto de los puntos de conexión puede provocar que el cliente OSS de Redis utilice puntos de conexión antiguos y eliminados, lo que impedirá que se conecte al clúster.

Mientras se migra el clúster de sin TLS a TLS preferido, se conservan los registros de DNS antiguos por nodo y los nuevos registros de DNS por nodo se generan en un formato diferente. Los clústeres habilitados para TLS utilizan un formato de registros DNS diferente al de los clústeres no habilitados para TLS. ElastiCache conservará ambos registros DNS cuando un clúster esté configurado en modo de cifrado: se prefiere para que Applications y otros clientes de Redis OSS puedan cambiar de uno a otro. Durante el proceso de migración de TLS se producen los siguientes cambios en los registros de DNS:

Descripción de los cambios en los registros de DNS que se producen al habilitar el cifrado en tránsito

Para clústeres de CME

Cuando un clúster está configurado en “modo de cifrado de tránsito: preferido”:

- Los puntos de conexión del clúster original para el clúster no habilitado para TLS permanecerán activos. No habrá tiempo de inactividad cuando el clúster se vuelva a configurar del modo de cifrado TLS “ninguno” a “preferido”.
- Se generarán nuevos puntos de conexión TLS Redis OSS cuando el clúster se configure en el modo TLS preferido. Estos nuevos puntos de conexión se resolverán con las mismas IP que los anteriores (no TLS).

- El nuevo punto final de configuración de TLS Redis OSS se mostrará en la ElastiCache consola y en la respuesta a la API. `describe-replication-group`

Cuando un clúster está configurado en “modo de cifrado de tránsito: requerido”:

- Se eliminarán los puntos de conexión antiguos que no estén habilitados para TLS. No habrá tiempo de inactividad en los puntos de conexión del clúster de TLS.
- Puede recuperar uno nuevo `cluster-configuration-endpoint` desde la ElastiCache consola o desde la `describe-replication-group` API.

Para clústeres de CMD con la conmutación por error automática habilitada o la conmutación por error automática desactivada

Cuando el grupo de replicación está establecido en “modo de cifrado de tránsito: preferido”:

- El punto de conexión principal y el punto de conexión del lector originales del clúster que no admite TLS permanecerán activos.
- Se generarán nuevos puntos de conexión principales y del lector de TLS cuando el clúster se establezca en el modo Preferred de TLS. Estos nuevos puntos de conexión se resolverán con las mismas IP que los anteriores (sin TLS).
- El nuevo punto final principal y el punto final del lector se mostrarán en la ElastiCache consola y en la respuesta a la `describe-replication-group` API.

Cuando el grupo de replicación está establecido en “modo de cifrado de tránsito: requerido”:

- Se eliminarán los puntos de conexión principales y de lector antiguos que no sean de TLS. No habrá tiempo de inactividad en los puntos de conexión del clúster de TLS.
- Puede recuperar los nuevos puntos de conexión principales y de lectura desde la ElastiCache consola o desde la `describe-replication-group` API.

El uso sugerido de los registros de DNS

Para clústeres de CME

- Utilice el punto de conexión de configuración del clúster en lugar de los registros de DNS por nodo en el código de la aplicación. No se recomienda utilizar nombres de DNS por nodo directamente, ya que es posible que cambien al agregar o eliminar particiones.

- No codifique el punto de conexión de configuración del clúster en la aplicación, ya que cambiará durante este proceso.
- Tener el punto de conexión de configuración del clúster codificado en la aplicación no es conveniente, ya que se puede cambiar durante este proceso. Una vez completado el cifrado en tránsito, consulte el punto de conexión de configuración del clúster con la API `describe-replication-group` (como se ha demostrado anteriormente [en negrita]) y utilice el DNS que obtendrá como respuesta a partir de ese momento.

Para clústeres de CMD con la conmutación por error automática habilitada

- Utilice el punto de conexión principal y el punto de conexión del lector en lugar de los nombres de DNS por nodo del código de la aplicación, ya que los nombres de DNS antiguos por nodo se eliminan y se generan otros nuevos al migrar el clúster de sin TLS a TLS preferido. No se recomienda utilizar nombres de DNS por nodo directamente, ya que es posible que agregue réplicas al clúster en el futuro. Además, cuando la conmutación por error automática está habilitada, el ElastiCache servicio cambia automáticamente las funciones del clúster principal y de las réplicas, por lo que se recomienda utilizar el punto final principal y el punto final del lector para poder realizar un seguimiento de esos cambios. Por último, utilizar el punto de conexión del lector le ayudará a distribuir las lecturas de las réplicas de manera equitativa entre las réplicas del clúster.
- Tener el punto de conexión principal y el punto de conexión del lector codificados en la aplicación no es conveniente, ya que se pueden cambiar durante el proceso de migración de TLS. Cuando se complete el cambio de migración a TLS-Preferred, consulte el punto final principal y el punto final del lector con la `describe-replication-group` API y utilice el DNS que obtenga como respuesta a partir de ese momento. De esta forma, podrá realizar un seguimiento de los cambios en los puntos de conexión de forma dinámica.

Para clústeres de CMD con la conmutación por error automática desactivada

- Utilice el punto de conexión principal y el punto de conexión del lector en lugar de los nombres de DNS por nodo del código de la aplicación. Si la conmutación por error automática está desactivada, tú eres quien realiza el escalado, los parches, la conmutación por error y otros procedimientos que el ElastiCache servicio gestiona automáticamente cuando la conmutación automática por error está habilitada. Esto le facilita la realización manual del seguimiento de los distintos puntos de conexión. Dado que los nombres de DNS antiguos por nodo se eliminan y se generan otros nuevos al migrar el clúster de sin TLS a TLS preferido, no utilice los nombres

de DNS por nodo directamente. Esto es obligatorio para que los clientes puedan conectarse al clúster durante la migración de TLS. Además, se beneficiará de distribuir las lecturas de manera uniforme entre las réplicas cuando utilice el punto de conexión del lector y de llevar un registro de los registros de DNS al agregar o eliminar réplicas del clúster.

- Tener el punto de conexión de configuración del clúster codificado en la aplicación no es conveniente, ya que se puede cambiar durante el proceso de migración de TLS.

Durante el cifrado en tránsito: preste atención a cuándo finaliza el proceso de migración

El cambio del modo de cifrado de tránsito no es inmediato y puede llevar algún tiempo. Esto es especialmente cierto en el caso de los clústeres grandes. Solo cuando el clúster finalice la migración a TLS preferido podrá aceptar y ofrecer conexiones de TCP y TLS. Por lo tanto, no debe crear clientes que intenten establecer conexiones de TLS con el clúster hasta que se complete el cifrado en tránsito.

Hay varias formas de recibir notificaciones cuando el cifrado en tránsito se completa correctamente o no funciona: (No se muestra en el ejemplo de código anterior):

- Uso del servicio de SNS para recibir una notificación cuando se complete el cifrado
- Uso de la API `describe-events` que emitirá un evento cuando se complete el cifrado
- Ver un mensaje en la ElastiCache consola que indica que el cifrado ha finalizado

También puede implementar la lógica en la aplicación para saber si se ha completado el cifrado. En el ejemplo anterior, vimos varias formas de garantizar que el clúster finalice la migración:

- Esperar a que comience el proceso de migración (el estado del clúster cambia a “modificándose”) y esperar a que finalice la modificación (el estado del clúster vuelve a ser “disponible”)
- Afirmar que el clúster ha establecido `transit_encryption_enabled` en Verdadero mediante una consulta a la API `describe-replication-group`.

Después de habilitar el cifrado en tránsito: asegúrese de que los clientes que utiliza estén configurados correctamente

Mientras el clúster esté en el modo TLS preferido, la aplicación debe abrir conexiones de TLS al clúster y utilizar solo esas conexiones. De esta forma, la aplicación no sufrirá tiempo de inactividad al habilitar el cifrado en tránsito. Puede asegurarse de que no haya conexiones TCP más claras al motor OSS de Redis mediante el comando `Redis OSS info` de la sección SSL.

```
# SSL
ssl_enabled:yes
ssl_current_certificate_not_before_date:Mar 20 23:27:07 2017 GMT
ssl_current_certificate_not_after_date:Feb 24 23:27:07 2117 GMT
ssl_current_certificate_serial:D8C7DEA91E684163
tls_mode_connected_tcp_clients:0 (should be zero)
tls_mode_connected_tls_clients:100
```

Cifrado en reposo en ElastiCache

Para ayudar a proteger sus datos, Amazon ElastiCache y Amazon S3 ofrecen diferentes formas de restringir el acceso a los datos de la memoria caché. Para obtener más información, consulte [Seguridad de ElastiCache y Amazon VPC](#) y [Identity and Access Management para Amazon ElastiCache](#).

ElastiCache El cifrado en reposo es una función que permite aumentar la seguridad de los datos mediante el cifrado de los datos del disco. Siempre está habilitada en una caché sin servidor. Si está habilitada, cifra los siguientes elementos:

- Disco durante las operaciones de sincronización, copia de seguridad o intercambio
- Copias de seguridad almacenadas en Amazon S3

Los datos almacenados en los SSD (unidades de estado sólido) en clústeres habilitados para la organización de datos en niveles siempre se cifran.

ElastiCache ofrece un cifrado predeterminado (gestionado por el servicio) en reposo, así como la posibilidad de utilizar sus propias claves KMS simétricas gestionadas por el cliente en AWS el [Servicio de administración](#) de claves (AWS KMS). Cuando se haga una copia de seguridad de la caché, elija, en las opciones de cifrado, si desea usar la clave de cifrado predeterminada o una clave administrada por el cliente. Para obtener más información, consulte [Activación del cifrado en reposo](#).

Note

El cifrado predeterminado (gestionado por el servicio) es la única opción disponible en las regiones GovCloud (EE. UU.).

Important

La activación del cifrado en reposo en un clúster OSS de Redis de diseño propio implica eliminar el grupo de replicación existente, tras ejecutar la copia de seguridad y la restauración en el grupo de replicación.

El cifrado en reposo solo se puede habilitar en una caché en el momento de su creación. Como se requiere cierto procesamiento para cifrar y descifrar los datos, habilitar el cifrado en reposo durante

estas operaciones puede afectar al rendimiento. Debe comparar los datos con y sin cifrado en reposo para determinar el impacto en el rendimiento de los casos de uso.

Temas

- [Condiciones del cifrado en reposo](#)
- [Uso de claves administradas por el cliente desde AWS KMS](#)
- [Activación del cifrado en reposo](#)
- [Véase también](#)

Condiciones del cifrado en reposo

Al planificar la implementación del cifrado en ElastiCache reposo, se deben tener en cuenta las siguientes restricciones en relación con el ElastiCache cifrado en reposo:

- El cifrado en reposo se admite en los grupos de replicación que ejecutan las versiones 4.0.10 o posteriores de Redis OSS (la 3.2.6 está programada para su fin de vida útil; consulte el calendario de [fin de vida de las versiones de Redis OSS](#)).
- El cifrado en reposo solo es compatible con los grupos de reproducción que se ejecutan en una Amazon VPC.
- El cifrado en reposo solo se admite en los grupos de reproducción que ejecutan los siguientes tipos de nodos.
 - R6gd, R6g, R5, R4, R3
 - M6g, M5, M4, M3
 - T4g, T3, T2

Para obtener más información, consulte [Tipos de nodos compatibles](#).

- El cifrado en reposo se habilita estableciendo explícitamente el parámetro `AtRestEncryptionEnabled` en `true`.
- El cifrado en reposo solamente se puede habilitar en un grupo de reproducción en el momento de su creación. Por tanto, el cifrado en reposo no se puede habilitar y desactivar modificando un grupo de reproducción. Para obtener información acerca de la implementación del cifrado en reposo en un grupo de reproducción existente, consulte [Activación del cifrado en reposo](#).
- Si un clúster utiliza un tipo de nodo de la familia r6gd, los datos almacenados en SSD se cifran independientemente de si el cifrado en reposo está habilitado o no.

- La opción de utilizar una clave gestionada por el cliente para el cifrado en reposo no está disponible en AWS GovCloud las regiones (-1 y -1). us-gov-east us-gov-west
- Si un clúster utiliza un tipo de nodo de la familia r6gd, los datos almacenados en la SSD se cifran con la clave AWS KMS administrada por el cliente elegida (o con el cifrado administrado por el servicio en Regions). AWS GovCloud

Implementar el cifrado en reposo puede reducir el rendimiento durante las operaciones de backup y sincronización de nodos. Compare sus datos con y sin cifrado en reposo para determinar el impacto en el rendimiento de la implementación.

Uso de claves administradas por el cliente desde AWS KMS

ElastiCache admite claves AWS KMS simétricas administradas por el cliente (clave KMS) para el cifrado en reposo. Las claves KMS administradas por el cliente son claves de cifrado que usted crea, posee y administra en su cuenta. AWS Para obtener más información, consulte [Claves de AWS KMS](#) en la Guía para desarrolladores de AWS Key Management Service. Las claves se deben crear en AWS KMS para poder usarlas con ellas. ElastiCache

Para obtener información sobre cómo crear claves raíz de AWS KMS, consulte [Creación de claves](#) en la Guía AWS para desarrolladores del Servicio de administración de claves.

ElastiCache le permite integrarse con AWS KMS. Para obtener más información, consulte [Uso de concesiones](#) en la Guía para desarrolladores de AWS Key Management Service. No es necesaria ninguna acción por parte del cliente para habilitar ElastiCache la integración de Amazon con AWS KMS.

La `kms:ViaService` clave condicional limita el uso de una clave AWS KMS (clave KMS) a las solicitudes de AWS servicios específicos. Para usarla `kms:ViaService` con ElastiCache, incluye ambos `ViaService` nombres en el valor de la clave de condición: `elasticache.AWS_region.amazonaws.com` y `dax.AWS_region.amazonaws.com`. Para obtener más información, consulte [kms: ViaService](#).

Puedes utilizarla [AWS CloudTrail](#) para hacer un seguimiento de las solicitudes que Amazon ElastiCache envía AWS Key Management Service en tu nombre. Todas las llamadas a la API AWS Key Management Service relacionadas con las claves gestionadas por el cliente tienen CloudTrail los registros correspondientes. También puede ver las concesiones que se ElastiCache crean al llamar a la API de [ListGrantsKMS](#).

Una vez que se cifra un grupo de reproducción mediante la clave administrada por el cliente, todas las copias de seguridad para el grupo de reproducción se cifran de la siguiente manera:

- Las copias de seguridad diarias automáticas se cifran mediante la clave administrada por el cliente asociada con el clúster.
- La copia de seguridad final creada cuando se elimina el grupo de reproducción también se cifra mediante la clave administrada por el cliente asociada con el grupo de reproducción.
- Las copias de seguridad creadas de forma manual se cifran de manera predeterminada para utilizar la clave de KMS asociada con el grupo de reproducción. Puede anular esto al elegir otra clave administrada por el cliente.
- Al copiar una copia de seguridad se utiliza de forma predeterminada una clave administrada por el cliente asociada a la copia de seguridad de origen. Puede anular esto al elegir otra clave administrada por el cliente.

Note

- Las claves administradas por el cliente no se pueden utilizar cuando se exportan copias de seguridad al bucket de Amazon S3 seleccionado. Sin embargo, todas las copias de seguridad exportadas a Amazon S3 se cifran mediante el [cifrado del lado del servidor](#). Puede optar por copiar el archivo de copia de seguridad en un objeto de S3 nuevo y cifrarlo mediante una clave de KMS administrada por el cliente, copiar el archivo a otro bucket de S3 que se ha configurado con el cifrado predeterminado mediante una clave de KMS o cambiar una opción de cifrado en el propio archivo.
- También puede utilizar claves administradas por el cliente a fin de cifrar copias de seguridad creadas de forma manual para grupos de reproducción que no utilicen claves administradas por el cliente para el cifrado. Con esta opción, el archivo de copia de seguridad almacenado en Amazon S3 se cifra mediante una clave de KMS, aunque los datos no se cifren en el grupo de reproducción original.

La restauración desde una copia de seguridad le permite elegir entre las opciones de cifrado disponibles, similares a las opciones de cifrado disponibles cuando se crea un nuevo grupo de reproducción.

- Si elimina la clave o [deshabilita](#) la clave y [revoca las concesiones](#) para la clave que utilizó para cifrar una caché, esta se vuelve irrecuperable. En otras palabras, no se puede modificar ni recuperar después de un fallo de hardware. AWS KMS elimina las claves raíz solo después de un período de espera de al menos siete días. Después de eliminar la clave, puede utilizar una clave administrada por el cliente diferente para crear una copia de seguridad con fines de archivo.
- La rotación automática de claves preserva las propiedades de las claves raíz de AWS KMS, por lo que la rotación no afecta a su capacidad de acceder a sus ElastiCache datos. ElastiCache Las cachés cifradas de Amazon no admiten la rotación manual de claves, lo que implica crear una nueva clave raíz y actualizar cualquier referencia a la clave anterior. Para obtener más información, consulte [Rotación de claves AWS KMS](#) en la Guía AWS para desarrolladores del servicio de administración de claves.
- El cifrado de una ElastiCache caché mediante una clave KMS requiere una concesión por caché. Esa concesión se utiliza durante toda la vida útil de la caché. Además, se utiliza una concesión por copia de seguridad durante la creación de la copia de seguridad. Dicha concesión se retira una vez que se crea la copia de seguridad.
- Para obtener más información sobre las concesiones y los límites de AWS KMS, consulte [los límites](#) en la Guía para desarrolladores del servicio de administración de AWS claves.

Activación del cifrado en reposo

Todas las cachés sin servidor tienen activado el cifrado en reposo.

Al crear un clúster de autodiseño, puede habilitar el cifrado en reposo configurando el parámetro `AtRestEncryptionEnabled` en `true`. No puede habilitar el cifrado en reposo en grupos de reproducción existentes.

Puede habilitar el cifrado en reposo al crear una ElastiCache memoria caché. Puede hacerlo mediante la AWS Management Console AWS CLI, la o la ElastiCache API.

Cuando cree una caché, puede elegir una de las siguientes opciones:

- Default (Predeterminado): esta opción utiliza el cifrado administrado por el servicio en reposo.
- Clave administrada por el cliente: esta opción le permite proporcionar el ID/ARN de clave de AWS KMS para el cifrado en reposo.

Para obtener información sobre cómo crear claves raíz de AWS KMS, consulte [Crear claves](#) en la Guía para desarrolladores del Servicio de administración de AWS claves

Contenido

- [Habilitar el cifrado en reposo mediante AWS Management Console](#)
- [Habilitación del cifrado en reposo mediante AWS CLI](#)

Habilitación del cifrado en reposo en un clúster OSS de Redis de diseño propio existente

Solo puede habilitar el cifrado en reposo al crear un grupo de replicación de Redis OSS. Si tiene un grupo de reproducción existente en el que desea habilitar el cifrado en reposo, haga lo siguiente.

Para habilitar el cifrado en reposo en un grupo de reproducción existente

1. Cree un backup manual del grupo de reproducción existente. Para obtener más información, consulte [Copias de seguridad manuales](#).
2. Cree un nuevo grupo de reproducción a partir de este backup. En el nuevo grupo de reproducción, habilite el cifrado en reposo. Para obtener más información, consulte [Restauración desde una copia de seguridad a una nueva caché](#).
3. Actualice los puntos de enlace de la aplicación para que apunten al nuevo grupo de reproducción.
4. Elimine el grupo de reproducción antiguo. Para obtener más información, consulte [Eliminación de un clúster](#) o [Eliminación de un grupo de reproducción](#).

Habilitar el cifrado en reposo mediante AWS Management Console

Habilitación del cifrado en reposo en una caché sin servidor (consola)

Todas las cachés sin servidor tienen activado el cifrado en reposo. De forma predeterminada, se utiliza una AWS clave KMS propia para cifrar los datos. Para elegir su propia AWS KMS clave, realice las siguientes selecciones:

- Amplíe la sección Configuración predeterminada.
- Seleccione Personalizar la configuración predeterminada en la sección Configuración predeterminada.
- Seleccione Personalice su configuración de seguridad en la sección Seguridad.
- Elija CMK administrada por el cliente en Configuración de clave de cifrado.
- Seleccione una clave en el ajuste Clave de AWS KMS .

Activación del cifrado en reposo en un clúster de autodiseño (consola)

Al diseñar su propia caché, las configuraciones de “desarrollo y prueba” y “producción” con el método “Creación sencilla” habilitan el cifrado en reposo mediante la clave Predeterminada. Al elegir la configuración, seleccione estas opciones:

- Seleccione la versión 3.2.6, 4.0.10 o posterior como versión del motor.
- Haga clic en la casilla de verificación situada junto a Habilitar en la opción Cifrado en reposo.
- Elija una Clave predeterminada o una CMK administrada por el cliente.

Para conocer el step-by-step procedimiento, consulte lo siguiente:

- [Creación de un clúster de Redis OSS \(modo de clúster desactivado\) \(consola\)](#)
- [Creación de un clúster de Redis OSS \(modo de clúster activado\) \(consola\)](#)

Habilitación del cifrado en reposo mediante AWS CLI

Para habilitar el cifrado en reposo al crear un clúster de Redis OSS mediante el AWS CLI, utilice el `at-rest-encryption-enabled` parámetro -- al crear un grupo de replicación.

Habilitación del cifrado en reposo en un clúster (CLI) de Redis OSS (modo de clúster desactivado)

La siguiente operación crea el grupo de replicación Redis OSS (modo de clúster desactivado) `my-classic-rg` con tres nodos (`-- num-cache-clusters`), una réplica principal y dos réplicas de lectura. El cifrado en reposo está habilitado para este grupo de replicación (`--`). `at-rest-encryption-enabled`

Los siguientes parámetros y sus valores son necesarios para habilitar el cifrado de este grupo de reproducción:

Parámetros clave

- **--engine:** debe ser `redis`.
- **--engine-version:** debe ser 3.2.6, 4.0.10 o posterior.
- **--at-rest-encryption-enabled:** obligatorio para habilitar el cifrado en reposo.

Example 1: Clúster de Redis OSS (modo de clúster desactivado) con réplicas

Para Linux, macOS o Unix:

```
aws elasticache create-replication-group \  
  --replication-group-id my-classic-rg \  
  --replication-group-description "3 node replication group" \  
  --cache-node-type cache.m4.large \  
  --engine redis \  
  --at-rest-encryption-enabled \  
  --num-cache-clusters 3
```

Para Windows:

```
aws elasticache create-replication-group ^  
  --replication-group-id my-classic-rg ^  
  --replication-group-description "3 node replication group" ^  
  --cache-node-type cache.m4.large ^  
  --engine redis ^  
  --at-rest-encryption-enabled ^  
  --num-cache-clusters 3 ^
```

Para obtener más información, consulte los siguientes temas:

- [Crear un grupo de replicación de Redis OSS \(modo de clúster desactivado\) desde cero \(AWS CLI\)](#)
- [create-replication-group](#)

Habilitación del cifrado en reposo en un clúster para Redis OSS (modo de clúster habilitado) (CLI)

La siguiente operación crea el grupo de replicación Redis OSS (modo de clúster activado) *my-clustered-rg* con tres grupos de nodos o fragmentos (--num-node-groups). Cada uno tiene tres nodos, una réplica principal y dos réplicas de lectura (--replicas-per-node-group). El cifrado en reposo está habilitado para este grupo de replicación (--at-rest-encryption-enabled).

Los siguientes parámetros y sus valores son necesarios para habilitar el cifrado de este grupo de reproducción:

Parámetros clave

- **--engine:** debe ser `redis`.
- **--engine-version:** debe ser 4.0.10 o posterior.
- **--at-rest-encryption-enabled:** obligatorio para habilitar el cifrado en reposo.

- **--cache-parameter-group**: debe ser `default-redis4.0.cluster.on` o un valor derivado de este que lo convierta en un grupo de reproducción con el modo de clúster habilitado.

Example 2: Un clúster de Redis OSS (modo de clúster activado)

Para Linux, macOS o Unix:

```
aws elasticache create-replication-group \  
  --replication-group-id my-clustered-rg \  
  --replication-group-description "redis clustered cluster" \  
  --cache-node-type cache.m3.large \  
  --num-node-groups 3 \  
  --replicas-per-node-group 2 \  
  --engine redis \  
  --engine-version 6.2 \  
  --at-rest-encryption-enabled \  
  --cache-parameter-group default.redis6.x.cluster.on
```

Para Windows:

```
aws elasticache create-replication-group ^  
  --replication-group-id my-clustered-rg ^  
  --replication-group-description "redis clustered cluster" ^  
  --cache-node-type cache.m3.large ^  
  --num-node-groups 3 ^  
  --replicas-per-node-group 2 ^  
  --engine redis ^  
  --engine-version 6.2 ^  
  --at-rest-encryption-enabled ^  
  --cache-parameter-group default.redis6.x.cluster.on
```

Para obtener más información, consulte los siguientes temas:

- [Crear un grupo de replicación de Redis OSS \(modo de clúster activado\) desde cero \(AWS CLI\)](#)
- [create-replication-group](#)

Véase también

- [Seguridad de ElastiCache y Amazon VPC](#)
- [Identity and Access Management para Amazon ElastiCache](#)

Autenticación y autorización

ElastiCache admite la autenticación de usuarios mediante IAM y el comando AUTH OSS de Redis, y la autorización de las operaciones de los usuarios mediante el control de acceso basado en roles (RBAC).

Temas

- [Control de acceso basado en roles \(RBAC\)](#)
- [Autenticación con el comando REDIS OSS AUTH](#)
- [Deshabilitar el control de acceso en una caché ElastiCache OSS de Redis](#)

Control de acceso basado en roles (RBAC)

En lugar de autenticar a los usuarios con el comando AUTH OSS de Redis como se describe en [Autenticación con el comando REDIS OSS AUTH](#), a partir de la versión 6.0 de Redis, puede utilizar una función denominada Control de acceso basado en roles (RBAC). El RBAC también es la única forma de controlar el acceso a las cachés sin servidor.

A diferencia de Redis OSS AUTH, donde todos los clientes autenticados tienen acceso total a la caché si su token está autenticado, RBAC permite controlar el acceso a la caché a través de grupos de usuarios. Estos grupos de usuarios se han diseñado como una forma de organizar el acceso a las cachés.

Con el RBAC, puede crear usuarios y asignarles permisos específicos mediante una cadena de acceso, como se describe a continuación. Los usuarios se asignan a grupos de usuarios alineados con una función específica (administradores, recursos humanos) que, a continuación, se despliegan en una o más ElastiCache cachés (Redis OSS). De este modo, puede establecer límites de seguridad entre los clientes que utilizan la misma caché o cachés de Redis OSS y evitar que los clientes accedan a los datos de los demás.

El RBAC está diseñado para respaldar la introducción de la [ACL de Redis OSS en Redis OSS 6](#). Cuando utiliza RBAC con su caché ElastiCache (Redis OSS), existen algunas limitaciones:

- No puede especificar contraseñas en una cadena de acceso. Las contraseñas se configuran con o con las llamadas [CreateUser](#). [ModifyUser](#)
- Para los derechos de usuario, pasa on y off como parte de la cadena de acceso. Si no se especifica ninguno en la cadena de acceso, se le asigna off al usuario y no tiene derechos de acceso a la caché.

- No se pueden utilizar comandos prohibidos y renombrados. Si especifica un comando prohibido o renombrado, se generará una excepción. Si desea utilizar listas de control de acceso (ACL) para un comando renombrado, especifique el nombre original del comando, es decir, el nombre del comando antes de cambiarle el nombre.
- No puede utilizar el comando `reset` como parte de una cadena de acceso. Las contraseñas se especifican con los parámetros de la API y ElastiCache (Redis OSS) las gestiona. Por lo tanto, no puede utilizar `reset` porque eliminará todas las contraseñas de un usuario.
- Redis OSS 6 presenta el comando [ACL LIST](#). Este comando devuelve una lista de usuarios junto con las reglas de ACL aplicadas a cada usuario. ElastiCache (Redis OSS) admite el ACL LIST comando, pero no incluye la compatibilidad con los hashes de contraseñas, como sí lo hace Redis OSS. Con ElastiCache (Redis OSS), puede utilizar la operación [describe-users](#) para obtener información similar, incluidas las reglas contenidas en la cadena de acceso. Sin embargo, [describe-users](#) no recupera una contraseña de usuario.

[Otros comandos de solo lectura compatibles con ElastiCache \(Redis OSS\) son ACL WHOAMI, ACL USERS y ACL CAT.](#) ElastiCache (Redis OSS) no admite ningún otro comando ACL basado en la escritura.

- Existen las siguientes limitaciones:

Recurso	Máximo permitido
Usuarios por grupo de usuarios	100
Número de usuarios	1 000
Número de grupos de usuarios	100

El uso de RBAC con ElastiCache (Redis OSS) se describe con más detalle a continuación.

Temas

- [Especificación de permisos mediante una cadena de acceso](#)
- [Aplicar RBAC a una caché para ElastiCache \(Redis OSS\)](#)
- [Migración de Redis OSS AUTH a RBAC](#)
- [Migración de RBAC a Redis OSS AUTH](#)
- [Rotación automática de contraseñas para usuarios](#)

- [Autenticación con IAM](#)

Especificación de permisos mediante una cadena de acceso

Para especificar los permisos de una caché ElastiCache (Redis OSS), debe crear una cadena de acceso y asignarla a un usuario mediante la tecla o. AWS CLI AWS Management Console

Las cadenas de acceso se definen como una lista de reglas delimitadas por espacios que se aplican al usuario. Definen qué comandos puede ejecutar un usuario y qué claves puede operar. Para ejecutar un comando, un usuario debe tener acceso al comando que se ejecuta y a todas las claves a las que accede el comando. Las reglas se aplican de izquierda a derecha de forma acumulativa y se puede utilizar una cadena más simple en lugar de la proporcionada si hay redundancias en la cadena proporcionada.

Para obtener más información sobre la sintaxis de las reglas de ACL, consulte [ACL](#).

En el siguiente ejemplo, la cadena de acceso representa un usuario activo con acceso a todas las claves y comandos disponibles.

```
on ~* +@all
```

La sintaxis de la cadena de acceso se desglosa de la siguiente manera:

- `on`: el usuario es un usuario activo.
- `~*`: se brinda acceso a todas las claves disponibles.
- `+@all`: se brinda acceso a todos los comandos disponibles.

La configuración anterior es la menos restrictiva. Puede modificar esta configuración para hacerla más segura.

En el siguiente ejemplo, la cadena de acceso representa a un usuario con acceso restringido al acceso de lectura en claves que comienzan por el espacio de claves “app:”

```
on ~app:* -@all +@read
```

Puede refinar aún más estos permisos al enumerar comandos a los que el usuario tiene acceso:

`+command1`: el acceso del usuario a los comandos se encuentra limitado a *command1*.

`+@category`: el acceso del usuario a los comandos se encuentra limitado a la categoría de comandos.

Para obtener información sobre cómo asignar una cadena de acceso a un usuario, consulte [Creación de usuarios y grupos de usuarios con la consola y la CLI](#).

Si va a migrar una carga de trabajo existente a ElastiCache, puede recuperar la cadena de acceso mediante una llamada `ACL LIST`, excluyendo los hashes de usuario y de contraseña.

En la versión 6.2 y superior de Redis OSS, también se admite la siguiente sintaxis de cadena de acceso:

- `&*`: se brinda acceso a todos los canales disponibles.

En la versión 7.0 y superior de Redis OSS, también se admite la siguiente sintaxis de cadena de acceso:

- `|`: se puede usar para bloquear subcomandos (por ejemplo, “-config|set”).
- `%R~<pattern>`: agregue el patrón de claves de lectura especificado. Esto se comporta de forma similar al patrón de claves normal, pero solo permite leer las claves que coinciden con el patrón dado. Consulte [key permissions](#) (permisos de clave) para obtener más información.
- `%W~<pattern>`: agregue el patrón de claves de escritura especificado. Esto se comporta de forma similar al patrón de claves normal, pero solo concede permiso de escritura a las claves que coinciden con el patrón dado. Consulte [key permissions](#) (permisos de clave) para obtener más información.
- `%RW~<pattern>`: alias para `~<pattern>`.
- `(<rule list>)`: cree un nuevo selector con el que comparar las reglas. Los selectores se evalúan después de los permisos del usuario y según el orden en que se definen. Si un comando coincide con los permisos del usuario o con algún selector, está permitido. Consulte la página sobre [selectores de ACL](#) para obtener más información.
- `clearselectors`: elimine todos los selectores asociados al usuario.

Aplicar RBAC a una caché para ElastiCache (Redis OSS)

Para utilizar el ElastiCache RBAC (Redis OSS), siga los siguientes pasos:

1. Cree uno o más usuarios.
2. Cree un grupo de usuarios y agregue usuarios al grupo.
3. Asigne el grupo de usuarios a una caché que tenga el cifrado en tránsito habilitado.

Estos pasos se describen en la siguiente tabla.

Temas

- [Creación de usuarios y grupos de usuarios con la consola y la CLI](#)
- [Administración de grupos de usuarios con la consola y la CLI](#)
- [Asignación de grupos de usuarios a cachés sin servidor](#)
- [Asignación de grupos de usuarios a grupos de reproducción](#)

Creación de usuarios y grupos de usuarios con la consola y la CLI

La información de usuario para los usuarios de RBAC es un ID de usuario, nombre de usuario y, opcionalmente, una contraseña y una cadena de acceso. La cadena de acceso proporciona el nivel de permisos en las claves y comandos. El ID de usuario es exclusivo del usuario y el nombre de usuario es lo que se pasa al motor.

Asegúrese de que los permisos de usuario que proporcione tengan sentido con el propósito previsto del grupo de usuarios. Por ejemplo, si crea un grupo de usuarios denominado `Administrators`, cualquier usuario que agregue a ese grupo debe tener su cadena de acceso establecida en el acceso completo a las claves y comandos. Para los usuarios de un grupo de usuarios `e-commerce`, puede establecer las cadenas de acceso en acceso de solo lectura.

ElastiCache configura automáticamente un usuario predeterminado con un ID de usuario y un nombre de usuario y lo añade a `default` todos los grupos de usuarios. No puede modificar ni eliminar este usuario. Este usuario está diseñado para ser compatible con el comportamiento predeterminado de las versiones anteriores de Redis OSS y tiene una cadena de acceso que le permite llamar a todos los comandos y acceder a todas las teclas.

Para agregar un control de acceso adecuado a una caché, reemplace este usuario predeterminado por uno nuevo que no se encuentre habilitado o utilice una contraseña segura. Para cambiar el usuario predeterminado, cree un usuario nuevo con el nombre de usuario establecido en `default`. A continuación, puede intercambiarlo con el usuario predeterminado original.

En los siguientes procedimientos se muestra cómo intercambiar el usuario `default` original con otro usuario `default` que tiene una cadena de acceso modificada.

Para modificar el usuario predeterminado de la consola

1. Inicia sesión en la ElastiCache consola de Amazon AWS Management Console y ábrela en <https://console.aws.amazon.com/elasticache/>.

2. Seleccione Administración de grupos de usuarios en el panel de navegación.
3. En ID de grupo de usuarios, elija el ID que desea modificar. Asegúrese de elegir el enlace y no la casilla de verificación.
4. Elija Modificar.
5. En la ventana Modificar, elija Administrar y elija el usuario que desee como usuario predeterminado con el Nombre de usuario como predeterminado.
6. Seleccione Elegir.
7. Elija Modificar. Al hacerlo, se terminan todas las conexiones existentes que el usuario predeterminado tenga a una caché.

Para modificar el usuario predeterminado con el AWS CLI

1. Cree un usuario nuevo con el nombre de usuario default mediante los siguientes comandos.

Para Linux, macOS o Unix:

```
aws elasticache create-user \  
  --user-id "new-default-user" \  
  --user-name "default" \  
  --engine "REDIS" \  
  --passwords "a-strong-password" \  
  --access-string "off +get ~keys*"
```

Para Windows:

```
aws elasticache create-user ^  
  --user-id "new-default-user" ^  
  --user-name "default" ^  
  --engine "REDIS" ^  
  --passwords "a-strong-password" ^  
  --access-string "off +get ~keys*"
```

2. Cree un grupo de usuarios y agregue el usuario que creó anteriormente.

Para Linux, macOS o Unix:

```
aws elasticache create-user-group \  
  --user-group-id "new-group-2" \  
  --engine "REDIS" \  
  --passwords "a-strong-password" \  
  --access-string "off +get ~keys*"
```

```
--user-ids "new-default-user"
```

Para Windows:

```
aws elasticache create-user-group ^
  --user-group-id "new-group-2" ^
  --engine "REDIS" ^
  --user-ids "new-default-user"
```

3. Intercambie el usuario default nuevo por el usuario default.

Para Linux, macOS o Unix:

```
aws elasticache modify-user-group \
  --user-group-id test-group \
  --user-ids-to-add "new-default-user" \
  --user-ids-to-remove "default"
```

Para Windows:

```
aws elasticache modify-user-group ^
  --user-group-id test-group ^
  --user-ids-to-add "new-default-user" ^
  --user-ids-to-remove "default"
```

Cuando se llama a esta operación de modificación, se terminan todas las conexiones existentes que el usuario predeterminado tenga a una caché.

Al crear un usuario, puede configurar hasta dos contraseñas. Al modificar una contraseña, se mantienen todas las conexiones existentes a las cachés.

En particular, tenga en cuenta estas restricciones de contraseña de usuario cuando utilice RBAC para ElastiCache (Redis OSS):

- Las contraseñas deben tener entre 16 y 128 caracteres imprimibles.
- No se admiten los siguientes caracteres no alfanuméricos: , " " / @.

Administración de usuarios con la consola y la CLI

Utilice el siguiente procedimiento para administrar usuarios en la consola.

Para administrar usuarios con la consola

1. Inicia sesión en la ElastiCache consola de Amazon AWS Management Console y ábrela en <https://console.aws.amazon.com/elasticache/>.
2. En el ElastiCache panel de Amazon, selecciona Gestión de usuarios. Están disponibles las siguientes opciones:

- Crear usuario: al crear un usuario, introduce un ID de usuario, un nombre de usuario, un modo de autenticación y una cadena de acceso. La cadena de acceso establece el nivel de permisos para qué claves y comandos se permite al usuario.

Al crear un usuario, puede configurar hasta dos contraseñas. Al modificar una contraseña, se mantienen todas las conexiones existentes a las cachés.

- Modificar usuario: permite actualizar la configuración de autenticación de un usuario o cambiar su cadena de acceso.
- Eliminar usuario: la cuenta se eliminará de cualquier grupo de usuarios al que pertenezca.

Utilice los siguientes procedimientos para administrar usuarios con la AWS CLI.

Para modificar un usuario mediante la CLI

- Utilice el comando `modify-user` para actualizar la contraseña o contraseñas de un usuario o cambiar los permisos de acceso de un usuario.

Cuando se modifica un usuario, se actualizan los grupos de usuarios asociados al usuario, junto con las cachés asociadas al grupo de usuarios. Se mantienen todas las conexiones existentes. A continuación se muestran algunos ejemplos.

Para Linux, macOS o Unix:

```
aws elasticache modify-user \  
  --user-id user-id-1 \  
  --access-string "~objects:* ~items:* ~public:*" \  
  --no-password-required
```

Para Windows:

```
aws elasticache modify-user ^  
  --user-id user-id-1 ^  
  --access-string "~objects:* ~items:* ~public:*" ^  
  --no-password-required
```

Note

No recomendamos utilizar la opción `nopass`. Si lo hace, recomendamos que configure los permisos del usuario en solo lectura con acceso a un conjunto limitado de claves.

Para eliminar un usuario mediante la CLI

- Ejecute el comando `delete-user` para eliminar un usuario. La cuenta se borra y elimina de todos los grupos de usuarios a los que pertenezca. A continuación, se muestra un ejemplo.

Para Linux, macOS o Unix:

```
aws elasticache delete-user \  
  --user-id user-id-2
```

Para Windows:

```
aws elasticache delete-user ^  
  --user-id user-id-2
```

Para ver una lista de usuarios, llame a la operación [describe-users](#).

```
aws elasticache describe-users
```

Administración de grupos de usuarios con la consola y la CLI

Puede crear grupos de usuarios para organizar y controlar el acceso de los usuarios a una o más cachés, como se muestra a continuación.

Utilice el siguiente procedimiento para administrar grupos de usuarios en la consola.

Para administrar grupos de usuarios mediante la consola

1. Inicia sesión en la ElastiCache consola de Amazon AWS Management Console y ábrela en <https://console.aws.amazon.com/elasticache/>.
2. En el ElastiCache panel de Amazon, selecciona Administración de grupos de usuarios.

A continuación, se muestran las operaciones disponibles para crear grupos de usuarios nuevos:

- **Crear:** al crear un grupo de usuarios, se agregan usuarios y, a continuación, se asignan los grupos de usuarios a las cachés. Por ejemplo, puede crear un grupo de usuarios Admin para los usuarios que tienen roles administrativos en una caché.

 **Important**

Cuando crea un grupo de usuarios, debe incluir el usuario predeterminado.

- **Agregar usuarios:** agregue usuarios al grupo de usuarios.
- **Eliminar usuarios:** elimine usuarios del grupo de usuarios. Cuando se quitan usuarios de un grupo de usuarios, se terminan todas las conexiones existentes que tengan a una caché.
- **Eliminar:** utilice esta opción para eliminar un grupo de usuarios. Tenga en cuenta que se eliminará el grupo de usuarios en sí, no los usuarios que pertenecen al grupo.

Para los grupos de usuarios existentes, puede realizar lo siguiente:

- **Agregar usuarios:** agregue usuarios existentes al grupo de usuarios.
- **Eliminar usuarios:** elimine los usuarios existentes del grupo de usuarios.

 **Note**

Los usuarios se eliminan del grupo de usuarios, pero no se eliminan del sistema.

Utilice el siguiente procedimiento para administrar grupos de usuarios en la CLI.

Para crear un grupo de usuarios nuevo y agregar un usuario mediante la CLI

- Utilice el comando `create-user-group`, tal y como se muestra a continuación:

Para Linux, macOS o Unix:

```
aws elasticache create-user-group \  
  --user-group-id "new-group-1" \  
  --engine "REDIS" \  
  --user-ids user-id-1, user-id-2
```

Para Windows:

```
aws elasticache create-user-group ^  
  --user-group-id "new-group-1" ^  
  --engine "REDIS" ^  
  --user-ids user-id-1, user-id-2
```

Para modificar un grupo de usuarios al agregar usuarios nuevos o eliminar miembros actuales mediante la CLI

- Utilice el comando `modify-user-group`, tal y como se muestra a continuación:

Para Linux, macOS o Unix:

```
aws elasticache modify-user-group --user-group-id new-group-1 \  
  --user-ids-to-add user-id-3 \  
  --user-ids-to-remove user-id-2
```

Para Windows:

```
aws elasticache modify-user-group --user-group-id new-group-1 ^  
  --user-ids-to-add userid-3 ^  
  --user-ids-to-remove user-id-2
```

Note

Cualquier conexión abierta que pertenezca a un usuario eliminado de un grupo de usuarios finaliza con este comando.

Para eliminar un grupo de usuarios mediante la CLI

- Utilice el comando `delete-user-group`, tal y como se muestra a continuación: Se elimina el grupo de usuarios en sí, no los usuarios que pertenecen al grupo.

Para Linux, macOS o Unix:

```
aws elasticache delete-user-group /  
  --user-group-id
```

Para Windows:

```
aws elasticache delete-user-group ^  
  --user-group-id
```

Para ver una lista de grupos de usuarios, puedes llamar a la [describe-user-groups](#) operación.

```
aws elasticache describe-user-groups \  
  --user-group-id test-group
```

Asignación de grupos de usuarios a cachés sin servidor

Después de crear un grupo de usuarios y agregar usuarios, el paso final para implementar el RBAC es asignar el grupo de usuarios a una caché sin servidor.

Asignación de grupos de usuarios a cachés sin servidor mediante la consola

Para añadir un grupo de usuarios a una caché sin servidor mediante el AWS Management Console, haga lo siguiente:

- Para el modo de clúster deshabilitado, consulte [Creación de un clúster de Redis OSS \(modo de clúster desactivado\) \(consola\)](#)

- Para el modo de clúster habilitado, consulte [Creación de un clúster de Redis OSS \(modo de clúster activado\) \(consola\)](#)

Asignación de grupos de usuarios a cachés sin servidor mediante el AWS CLI

La siguiente AWS CLI operación crea una caché sin servidor mediante el `user-group-id` parámetro con el valor. *my-user-group-id* Reemplace el grupo de subredes `sng-test` por otro existente.

Parámetros clave

- **--engine**: debe ser `redis`.
- **--user-group-id**: este valor proporciona el ID del grupo de usuarios, compuesto por usuarios con permisos de acceso especificados para la caché.

Para Linux, macOS o Unix:

```
aws elasticache create-serverless-cache \  
  --serverless-cache-name "new-serverless-cache" \  
  --description "new-serverless-cache" \  
  --engine "redis" \  
  --user-group-id "new-group-1"
```

Para Windows:

```
aws elasticache create-serverless-cache ^  
  --serverless-cache-name "new-serverless-cache" ^  
  --description "new-serverless-cache" ^  
  --engine "redis" ^  
  --user-group-id "new-group-1"
```

La siguiente AWS CLI operación modifica una caché sin servidor con el `user-group-id` parámetro con el valor. *my-user-group-id*

Para Linux, macOS o Unix:

```
aws elasticache modify-serverless-cache \  
  --serverless-cache-name serverless-cache-1 \  
  --user-group-id "new-group-2"
```

Para Windows:

```
aws elasticache modify-serverless-cache ^
  --serverless-cache-name serverless-cache-1 ^
  --user-group-id "new-group-2"
```

Tenga en cuenta que cualquier modificación que se haga en una caché se actualizará de forma asíncrona. Puede monitorear el progreso al visualizar los eventos. Para obtener más información, consulte [Visualización de eventos de ElastiCache](#).

Asignación de grupos de usuarios a grupos de reproducción

Después de crear un grupo de usuarios y agregar usuarios, el paso final para implementar el RBAC es asignar el grupo de usuarios a un grupo de reproducción.

Asignación de grupos de usuarios a grupos de reproducción mediante la consola

Para añadir un grupo de usuarios a una replicación mediante el AWS Management Console, haga lo siguiente:

- Para el modo de clúster deshabilitado, consulte [Creación de un clúster de Redis OSS \(modo de clúster desactivado\) \(consola\)](#)
- Para el modo de clúster habilitado, consulte [Creación de un clúster de Redis OSS \(modo de clúster activado\) \(consola\)](#)

Asignación de grupos de usuarios a grupos de replicación mediante el AWS CLI

La siguiente AWS CLI operación crea un grupo de replicación con el cifrado en tránsito (TLS) activado y el `user-group-ids` parámetro con el valor `my-user-group-id`. Reemplace el grupo de subredes `sng-test` por otro existente.

Parámetros clave

- **--engine:** debe ser `redis`.
- **--engine-version:** debe ser 6.0 o posterior.
- **--transit-encryption-enabled:** obligatorio para la autenticación y a fin de asociar un grupo de usuarios.
- **--user-group-ids:** este valor proporciona el ID del grupo de usuarios, compuesto por usuarios con permisos de acceso especificados para la caché.

- **--cache-subnet-group**: obligatorio para asociar un grupo de usuarios.

Para Linux, macOS o Unix:

```
aws elasticache create-replication-group \  
  --replication-group-id "new-replication-group" \  
  --replication-group-description "new-replication-group" \  
  --engine "redis" \  
  --cache-node-type cache.m5.large \  
  --transit-encryption-enabled \  
  --user-group-ids "new-group-1" \  
  --cache-subnet-group "cache-subnet-group"
```

Para Windows:

```
aws elasticache create-replication-group ^\  
  --replication-group-id "new-replication-group" ^\  
  --replication-group-description "new-replication-group" ^\  
  --engine "redis" ^\  
  --cache-node-type cache.m5.large ^\  
  --transit-encryption-enabled ^\  
  --user-group-ids "new-group-1" ^\  
  --cache-subnet-group "cache-subnet-group"
```

La siguiente AWS CLI operación modifica un grupo de replicación con el cifrado en tránsito (TLS) habilitado y el user-group-ids parámetro con el valor. *my-user-group-id*

Para Linux, macOS o Unix:

```
aws elasticache modify-replication-group \  
  --replication-group-id replication-group-1 \  
  --user-group-ids-to-remove "new-group-1" \  
  --user-group-ids-to-add "new-group-2"
```

Para Windows:

```
aws elasticache modify-replication-group ^\  
  --replication-group-id replication-group-1 ^\  
  --user-group-ids-to-remove "new-group-1" ^\  
  --user-group-ids-to-add "new-group-2"
```

Tenga en cuenta los PendingChanges en la respuesta. Las modificaciones que se hagan en una caché se actualizarán de forma asíncrona. Puede monitorear el progreso al visualizar los eventos. Para obtener más información, consulte [Visualización de eventos de ElastiCache](#).

Migración de Redis OSS AUTH a RBAC

Si utiliza Redis OSS AUTH como se describe en [Autenticación con el comando REDIS OSS AUTH](#) y desea migrar a RBAC, utilice los siguientes procedimientos.

Utilice el siguiente procedimiento para migrar de Redis OSS AUTH a RBAC mediante la consola.

Para migrar de Redis OSS AUTH a RBAC mediante la consola

1. [Inicie sesión en la consola AWS Management Console y ábrala en https://console.aws.amazon.com/elasticache/](https://console.aws.amazon.com/elasticache/). ElastiCache
2. En la lista de la esquina superior derecha, selecciona la AWS región en la que se encuentra la memoria caché que deseas modificar.
3. En el panel de navegación, elija el motor que se ejecuta en la caché que desea modificar.

Aparecerá una lista de las cachés del motor elegido.

4. En la lista de cachés, elija el nombre de la caché que desea modificar.
5. Para Actions (Acciones), elija Modify (Modificar).

Aparecerá la ventana Modificar.

6. En Opción de control de acceso, elija Lista de control de acceso de grupos de usuarios.
7. En Lista de control de acceso de grupos de usuarios, elija un grupo de usuarios.
8. Seleccione Vista previa de los cambios y seleccione Modificar en la siguiente pantalla.

Utilice el siguiente procedimiento para migrar de Redis OSS AUTH a RBAC mediante la CLI.

Para migrar de Redis OSS AUTH a RBAC mediante la CLI

- Utilice el comando `modify-replication-group`, tal y como se muestra a continuación:

Para Linux, macOS o Unix:

```
aws elasticache modify-replication-group --replication-group-id test \
```

```
--auth-token-update-strategy DELETE \  
--user-group-ids-to-add user-group-1
```

Para Windows:

```
aws elasticache modify-replication-group --replication-group-id test ^  
--auth-token-update-strategy DELETE ^  
--user-group-ids-to-add user-group-1
```

Migración de RBAC a Redis OSS AUTH

Si utiliza RBAC y desea migrar a Redis OSS AUTH, consulte. [Migración de RBAC a Redis OSS AUTH](#)

Note

Si necesita deshabilitar el control de acceso en una ElastiCache memoria caché, deberá hacerlo a través del. AWS CLI Para obtener más información, consulte [the section called “Deshabilitar el control de acceso en una caché ElastiCache OSS de Redis”](#).

Rotación automática de contraseñas para usuarios

Con AWS Secrets Manager, puedes reemplazar las credenciales codificadas de tu código (incluidas las contraseñas) por una llamada a la API a Secrets Manager para recuperar el secreto mediante programación. Esto ayuda a garantizar la integridad del secreto si alguien examina el código, dado que el secreto sencillamente no está allí. Asimismo, puede configurar Secrets Manager para rotar el secreto automáticamente de acuerdo con la programación que especifique. Esto le permite reemplazar secretos a largo plazo con secretos a corto plazo, lo que contribuye a reducir significativamente el riesgo de peligro.

Con Secrets Manager, puede rotar automáticamente sus contraseñas ElastiCache (es decir, secretas) (es decir, secretas) (es decir, secretas) automáticamente mediante una AWS Lambda función que proporciona Secrets Manager.

Para obtener más información al respecto AWS Secrets Manager, consulte [¿Qué es? AWS Secrets Manager](#)

¿Cómo ElastiCache usa los secretos

Con Redis OSS 6 ElastiCache (Redis OSS) se introdujo [Control de acceso basado en roles \(RBAC\)](#) para proteger el clúster de Redis OSS. Esta característica permite limitar determinadas conexiones en términos de los comandos que se pueden ejecutar y las claves a las que se puede acceder. Con RBAC, mientras el cliente crea un usuario con contraseñas, los valores de la contraseña se deben ingresar manualmente en texto no cifrado y estar visibles para el operador.

Con Secrets Manager, las aplicaciones obtienen la contraseña de Secrets Manager en lugar de ingresarla manualmente y almacenarla en la configuración de la aplicación. Para obtener información sobre cómo hacerlo, consulte [Cómo se asocian ElastiCache los usuarios al secreto](#).

El uso de secretos conlleva un coste. Para obtener información sobre precios, consulte [Precios de AWS Secrets Manager](#).

Cómo se asocian ElastiCache los usuarios al secreto

Secrets Manager guardará una referencia del usuario asociado en el campo `SecretString` del secreto. No habrá ninguna referencia al secreto desde ElastiCache un lado.

```
{
  "password": "strongpassword",
  "username": "user1",
  "user_arn": "arn:aws:elasticache:us-east-1:xxxxxxxxxx918:user:user1" //this is the
  bond between the secret and the user
}
```

Función de rotación de Lambda

Para habilitar la rotación automática de contraseñas de Secrets Manager, se creará una función de Lambda que interactuará con la API [modificar-usuario](#) para actualizar las contraseñas del usuario.

Para obtener información sobre cómo funciona esto, consulte [Cómo funciona la rotación](#).

Note

En el caso de algunos AWS servicios, para evitar el confuso escenario de los suplentes, se AWS recomienda utilizar tanto la `aws:SourceArn` clave de condición como la clave de condición `aws:SourceAccount` global. No obstante, si se incluye la condición `aws:SourceArn` en la política de la función de rotación, la función de rotación solo se puede utilizar para rotar el secreto especificado por ese ARN. Se recomienda incluir solo la clave

de contexto `aws:SourceAccount`, para poder utilizar la función de rotación para varios secretos.

Para cualquier problema que puedas encontrar, consulta [Solucionar problemas de rotación de AWS Secrets Manager](#).

Cómo crear un ElastiCache usuario y asociarlo a Secrets Manager

Los siguientes pasos muestran cómo crear un usuario y asociarlo con Secrets Manager:

1. Crear un usuario inactivo

Para Linux, macOS o Unix:

```
aws elasticache create-user \  
  --user-id user1 \  
  --user-name user1 \  
  --engine "REDIS" \  
  --no-password \ // no authentication is required  
  --access-string "*off* +get ~keys*" // this disables the user
```

Para Windows:

```
aws elasticache create-user ^  
  --user-id user1 ^  
  --user-name user1 ^  
  --engine "REDIS" ^  
  --no-password ^ // no authentication is required  
  --access-string "*off* +get ~keys*" // this disables the user
```

Verá una respuesta similar a la siguiente:

```
{  
  "UserId": "user1",  
  "UserName": "user1",  
  "Status": "active",  
  "Engine": "redis",  
  "AccessString": "off ~keys* -@all +get",  
  "UserGroupIds": [],  
  "Authentication": {
```

```

    "Type": "no_password"
  },
  "ARN": "arn:aws:elasticache:us-east-1:xxxxxxxxxx918:user:user1"
}

```

2. Crear un secreto

Para Linux, macOS o Unix:

```

aws secretsmanager create-secret \
--name production/ec/user1 \
--secret-string \
'{"user_arn": "arn:aws:elasticache:us-east-1:123456xxxx:user:user1",
  "username": "user1"}'

```

Para Windows:

```

aws secretsmanager create-secret ^
--name production/ec/user1 ^
--secret-string ^
'{"user_arn": "arn:aws:elasticache:us-east-1:123456xxxx:user:user1",
  "username": "user1"}'

```

Verá una respuesta similar a la siguiente:

```

{
  "ARN": "arn:aws:secretsmanager:us-east-1:123456xxxx:secret:production/ec/user1-
eaFois",
  "Name": "production/ec/user1",
  "VersionId": "aae5b963-1e6b-4250-91c6-ebd6c47d0d95"
}

```

3. Configurar una función de Lambda para rotar la contraseña

- a. [Inicie sesión en la consola Lambda AWS Management Console y ábrala en https://console.aws.amazon.com/lambda/](https://console.aws.amazon.com/lambda/)

- b. Elija Functions (Funciones) en el panel de navegación y, a continuación, elija la función que ha creado. Elija el nombre de la función, no la casilla de verificación situada a la izquierda.
- c. Elija la pestaña Configuración.
- d. En General configuration (Configuración general), elija Edit (Editar) y, a continuación, establezca Timeout (Tiempo de espera) en al menos 12 minutos.
- e. Seleccione Guardar.
- f. Elija Environment variables (Variables de entorno) y, a continuación, establezca lo siguiente:
 - i. SECRETS_MANAGER_ENDPOINT: <https://secretsmanager.REGION.amazonaws.com>
 - ii. SECRET_ARN: el nombre de recurso de Amazon (ARN) del secreto que creó en el paso 2.
 - iii. USER_NAME: nombre de usuario del usuario, ElastiCache
 - iv. Seleccione Guardar.
- g. Elegir Permissions (Permisos)
- h. En Execution role (Rol de ejecución), elija el nombre del rol de la función de Lambda para verla en la consola de IAM.
- i. La función de Lambda necesitará el siguiente permiso para modificar los usuarios y establecer la contraseña:

ElastiCache

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:DescribeUsers",
        "elasticache:ModifyUser"
      ],
      "Resource": "arn:aws:elasticache:us-east-1:xxxxxxxxxxx918:user:user1"
    }
  ]
}
```

Secrets Manager

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "secretsmanager:GetSecretValue",
        "secretsmanager:DescribeSecret",
        "secretsmanager:PutSecretValue",
        "secretsmanager:UpdateSecretVersionStage"
      ],
      "Resource": "arn:aws:secretsmanager:us-
east-1:xxxxxxxxxxxx:secret:XXXX"
    },
    {
      "Effect": "Allow",
      "Action": "secretsmanager:GetRandomPassword",
      "Resource": "*"
    }
  ]
}
```

4. Configurar la rotación de secretos de Secrets Manager

- a. Uso de AWS Management Console, consulte [Configurar la rotación automática de los AWS secretos de Secrets Manager mediante la consola](#)

Para obtener más información sobre la configuración de un programa de rotación, consulte [Programación de expresiones en la rotación de Secrets Manager](#).

- b. Con el AWS CLI, consulte [Configurar la rotación automática para AWS Secrets Manager usar el AWS Command Line Interface](#)

Autenticación con IAM

Temas

- [Información general](#)
- [Limitaciones](#)
- [Configuración](#)
- [Conexión](#)

Información general

Con la autenticación de IAM, puede autenticar una conexión a ElastiCache (Redis OSS) mediante identidades de AWS IAM, si la memoria caché está configurada para usar Redis OSS versión 7 o superior. Esto le permite reforzar el modelo de seguridad y simplificar muchas tareas de seguridad administrativa. También puede utilizar la autenticación de IAM para configurar un control de acceso detallado para cada caché y usuario individuales ElastiCache , siguiendo los principios de permisos con privilegios mínimos. ElastiCache La autenticación de IAM para ElastiCache (Redis OSS) funciona proporcionando un token de autenticación de IAM de corta duración en lugar de una contraseña de usuario de larga duración en el OSS o el comando de Redis. ElastiCache AUTH HELLO Para obtener más información sobre el token de autenticación de IAM, consulte el [proceso de firma de la versión 4](#) en la Guía de referencia AWS general y el ejemplo de código que aparece a continuación.

Puede utilizar las identidades de IAM y sus políticas asociadas para restringir aún más el acceso a Redis OSS. También puede conceder acceso a los usuarios desde sus proveedores de identidad federados directamente a las cachés de Redis OSS.

Para usar AWS IAM con ElastiCache (Redis OSS), primero debe crear un ElastiCache usuario con el modo de autenticación configurado en IAM. A continuación, puede crear o reutilizar una identidad de IAM. La identidad de IAM necesita una política asociada para conceder la `elasticache:Connect` acción a la ElastiCache memoria caché y ElastiCache al usuario. Una vez configurado, puede crear un token de autenticación de IAM con AWS las credenciales del usuario o rol de IAM. Por último, debe proporcionar el token de autenticación de IAM de corta duración como contraseña en su cliente OSS de Redis cuando se conecte a la memoria caché de Redis OSS. Un cliente OSS de Redis con soporte para el proveedor de credenciales puede generar automáticamente las credenciales temporales automáticamente para cada nueva conexión. ElastiCache (Redis OSS) autenticará con IAM las solicitudes de conexión de ElastiCache los usuarios con IAM y las validará con IAM.

Limitaciones

Si utiliza la autenticación de IAM, se aplicarán las siguientes limitaciones:

- La autenticación de IAM está disponible cuando se utiliza la versión 7.0 o superior ElastiCache (Redis OSS).
- En el caso de ElastiCache los usuarios con IAM, las propiedades del nombre de usuario y el identificador de usuario deben ser idénticas.

- El token de autenticación de IAM es válido durante 15 minutos. Para conexiones de larga duración, recomendamos utilizar un cliente OSS de Redis que admita una interfaz de proveedor de credenciales.
- Una conexión autenticada por IAM a ElastiCache (Redis OSS) se desconectará automáticamente después de 12 horas. La conexión se puede prolongar durante 12 horas enviando un comando AUTH o HELLO con un nuevo token de autenticación de IAM.
- Los comandos MULTI EXEC no admiten la autenticación de IAM.
- Actualmente, la autenticación de IAM admite las siguientes claves de contexto de condición global:
 - Cuando se utiliza la autenticación de IAM con cachés sin servidor, se admiten `aws:VpcSourceIp`, `aws:SourceVpc`, `aws:SourceVpce`, `aws:CurrentTime`, `aws:EpochTime` y `aws:ResourceTag/%s` (desde usuarios y cachés sin servidor asociados).
 - Cuando se utiliza la autenticación de IAM con grupos de réplica, se admiten `aws:SourceIp` y `aws:ResourceTag/%s` (desde los usuarios y grupos de réplica asociados).

Para obtener más información sobre las claves de contexto de condición globales, consulte [Claves de contexto de condición globales de AWS](#) en la Guía del usuario de IAM.

Configuración

Para configurar la autenticación de IAM:

1. Creación de una caché

```
aws elasticache create-serverless-cache \  
  --serverless-cache-name cache-01 \  
  --description "ElastiCache IAM auth application" \  
  --engine redis
```

2. Cree un documento de política de confianza de IAM, como se muestra a continuación, para el rol que permita a la cuenta asumir el nuevo rol. Guarde la política en un archivo denominado `trust-policy.json`.

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Principal": { "AWS": "arn:aws:iam::123456789012:root" },  
      "Action": "sts:AssumeRole"  
    }  
  ]  
}
```

```
}
}
```

3. Cree un documento de política de IAM, como se muestra a continuación. Guarde la política en un archivo denominado `policy.json`.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect" : "Allow",
      "Action" : [
        "elasticache:Connect"
      ],
      "Resource" : [
        "arn:aws:elasticache:us-east-1:123456789012:serverlesscache:cache-01",
        "arn:aws:elasticache:us-east-1:123456789012:user:iam-user-01"
      ]
    }
  ]
}
```

4. Crear un rol de IAM.

```
aws iam create-role \
  --role-name "elasticache-iam-auth-app" \
  --assume-role-policy-document file://trust-policy.json
```

5. Cree la política de IAM.

```
aws iam create-policy \
  --policy-name "elasticache-allow-all" \
  --policy-document file://policy.json
```

6. Adjunte la política de IAM al rol.

```
aws iam attach-role-policy \
  --role-name "elasticache-iam-auth-app" \
  --policy-arn "arn:aws:iam::123456789012:policy/elasticache-allow-all"
```

7. Cree un nuevo usuario habilitado para IAM.

```
aws elasticache create-user \
```

```
--user-name iam-user-01 \  
--user-id iam-user-01 \  
--authentication-mode Type=iam \  
--engine redis \  
--access-string "on ~* +@all"
```

8. Cree un grupo de usuarios y asocie al usuario.

```
aws elasticache create-user-group \  
  --user-group-id iam-user-group-01 \  
  --engine redis \  
  --user-ids default iam-user-01  
  
aws elasticache modify-serverless-cache \  
  --serverless-cache-name cache-01 \  
  --user-group-id iam-user-group-01
```

Conexión

Conectar con el token como contraseña

Primero debe generar el token de autenticación de IAM de corta duración mediante una [solicitud prefirmada SigV4 de AWS](#). Después, proporciona el token de autenticación de IAM como contraseña cuando se conecta a una memoria caché de Redis OSS, como se muestra en el siguiente ejemplo.

```
String userId = "insert user id";  
String cacheName = "insert cache name";  
boolean isServerless = true;  
String region = "insert region";  
  
// Create a default AWS Credentials provider.  
// This will look for AWS credentials defined in environment variables or system  
// properties.  
AWSCredentialsProvider awsCredentialsProvider = new  
  DefaultAWSCredentialsProviderChain();  
  
// Create an IAM authentication token request and signed it using the AWS credentials.  
// The pre-signed request URL is used as an IAM authentication token for ElastiCache  
// (Redis OSS).  
IAMAuthTokenRequest iamAuthTokenRequest = new IAMAuthTokenRequest(userId, cacheName,  
  region, isServerless);
```

```
String iamAuthToken =
    iamAuthTokenRequest.toSignedRequestUri(awsCredentialsProvider.getCredentials());

// Construct Redis OSS URL with IAM Auth credentials provider
RedisURI redisURI = RedisURI.builder()
    .withHost(host)
    .withPort(port)
    .withSsl(ssl)
    .withAuthentication(userId, iamAuthToken)
    .build();

// Create a new Lettuce Redis OSS client
RedisClient client = RedisClient.create(redisURI);
client.connect();
```

A continuación, se muestra la definición de `IAMAuthTokenRequest`.

```
public class IAMAuthTokenRequest {
    private static final HttpMethodName REQUEST_METHOD = HttpMethodName.GET;
    private static final String REQUEST_PROTOCOL = "http://";
    private static final String PARAM_ACTION = "Action";
    private static final String PARAM_USER = "User";
    private static final String PARAM_RESOURCE_TYPE = "ResourceType";
    private static final String RESOURCE_TYPE_SERVERLESS_CACHE = "ServerlessCache";
    private static final String ACTION_NAME = "connect";
    private static final String SERVICE_NAME = "elasticache";
    private static final long TOKEN_EXPIRY_SECONDS = 900;

    private final String userId;
    private final String cacheName;
    private final String region;
    private final boolean isServerless;

    public IAMAuthTokenRequest(String userId, String cacheName, String region, boolean
isServerless) {
        this.userId = userId;
        this.cacheName = cacheName;
        this.region = region;
        this.isServerless = isServerless;
    }

    public String toSignedRequestUri(AWSCredentials credentials) throws
URISyntaxException {
```

```

    Request<Void> request = getSignableRequest();
    sign(request, credentials);
    return new URIBuilder(request.getEndpoint())
        .addParameters(toNamedValuePair(request.getParameters()))
        .build()
        .toString()
        .replace(REQUEST_PROTOCOL, "");
}

private <T> Request<T> getSignableRequest() {
    Request<T> request = new DefaultRequest<>(SERVICE_NAME);
    request.setHttpMethod(REQUEST_METHOD);
    request.setEndpoint(getRequestUri());
    request.addParameters(PARAM_ACTION, Collections.singletonList(ACTION_NAME));
    request.addParameters(PARAM_USER, Collections.singletonList(userId));
    if (isServerless) {
        request.addParameters(PARAM_RESOURCE_TYPE,
Collections.singletonList(RESOURCE_TYPE_SERVERLESS_CACHE));
    }
    return request;
}

private URI getRequestUri() {
    return URI.create(String.format("%s%s/", REQUEST_PROTOCOL, cacheName));
}

private <T> void sign(SignableRequest<T> request, AWSCredentials credentials) {
    AWS4Signer signer = new AWS4Signer();
    signer.setRegionName(region);
    signer.setServiceName(SERVICE_NAME);

    DateTime dateTime = DateTime.now();
    dateTime = dateTime.plus(Duration.standardSeconds(TOKEN_EXPIRY_SECONDS));

    signer.presignRequest(request, credentials, dateTime.toDate());
}

private static List<NameValuePair> toNamedValuePair(Map<String, List<String>> in) {
    return in.entrySet().stream()
        .map(e -> new BasicNameValuePair(e.getKey(), e.getValue().get(0)))
        .collect(Collectors.toList());
}
}

```

Conectar con el proveedor de credenciales

El siguiente código muestra cómo autenticarse con ElastiCache (Redis OSS) mediante el proveedor de credenciales de autenticación de IAM.

```
String userId = "insert user id";
String cacheName = "insert cache name";
boolean isServerless = true;
String region = "insert region";

// Create a default AWS Credentials provider.
// This will look for AWS credentials defined in environment variables or system
// properties.
AWSCredentialsProvider awsCredentialsProvider = new
    DefaultAWSCredentialsProviderChain();

// Create an IAM authentication token request. Once this request is signed it can be
// used as an
// IAM authentication token for ElastiCache (Redis OSS).
IAMAuthTokenRequest iamAuthTokenRequest = new IAMAuthTokenRequest(userId, cacheName,
    region, isServerless);

// Create a Redis OSS credentials provider using IAM credentials.
RedisCredentialsProvider redisCredentialsProvider = new
    RedisIAMAuthCredentialsProvider(
        userId, iamAuthTokenRequest, awsCredentialsProvider);

// Construct Redis OSS URL with IAM Auth credentials provider
RedisURI redisURI = RedisURI.builder()
    .withHost(host)
    .withPort(port)
    .withSsl(ssl)
    .withAuthentication(redisCredentialsProvider)
    .build();

// Create a new Lettuce Redis OSS client
RedisClient client = RedisClient.create(redisURI);
client.connect();
```

A continuación, se muestra un ejemplo de un cliente OSS de Lettuce Redis que incluye el IAM AuthTokenRequest en un proveedor de credenciales para generar automáticamente credenciales temporales cuando es necesario.

```
public class RedisIAMAAuthCredentialsProvider implements RedisCredentialsProvider {
    private static final long TOKEN_EXPIRY_SECONDS = 900;

    private final AWSCredentialsProvider awsCredentialsProvider;
    private final String userId;
    private final IAMAuthTokenRequest iamAuthTokenRequest;
    private final Supplier<String> iamAuthTokenSupplier;

    public RedisIAMAAuthCredentialsProvider(String userId,
        IAMAuthTokenRequest iamAuthTokenRequest,
        AWSCredentialsProvider awsCredentialsProvider) {
        this.userName = userName;
        this.awsCredentialsProvider = awsCredentialsProvider;
        this.iamAuthTokenRequest = iamAuthTokenRequest;
        this.iamAuthTokenSupplier =
        Suppliers.memoizeWithExpiration(this::getIamAuthToken, TOKEN_EXPIRY_SECONDS,
        TimeUnit.SECONDS);
    }

    @Override
    public Mono<RedisCredentials> resolveCredentials() {
        return Mono.just(RedisCredentials.just(userId, iamAuthTokenSupplier.get()));
    }

    private String getIamAuthToken() {
        return
        iamAuthTokenRequest.toSignedRequestUri(awsCredentialsProvider.getCredentials());
    }
}
```

Autenticación con el comando REDIS OSS AUTH

Note

Redis OSS AUTH ha sido reemplazado por [the section called “Control de acceso basado en roles \(RBAC\)”](#) Todas las cachés sin servidor deben usar RBAC para la autenticación.

Los identificadores o contraseñas de autenticación de Redis OSS permiten a Redis OSS solicitar una contraseña antes de permitir que los clientes ejecuten comandos, lo que mejora la seguridad de los datos. Redis OSS solo AUTH está disponible para clústeres de diseño propio.

Temas

- [Descripción general de AUTH en ElastiCache \(Redis OSS\)](#)
- [Aplicar la autenticación a un clúster ElastiCache \(Redis OSS\)](#)
- [Modificar el token AUTH en un clúster existente ElastiCache \(Redis OSS\)](#)
- [Migración de RBAC a Redis OSS AUTH](#)

Descripción general de AUTH en ElastiCache (Redis OSS)

Al utilizar el OSS de Redis AUTH con su clúster ElastiCache (Redis OSS), hay algunas mejoras.

En concreto, tenga en cuenta estas restricciones de contraseña o token de AUTH cuando utilice el AUTH con ElastiCache (Redis OSS):

- Los tokens o contraseñas, deben tener entre 16 y 128 caracteres imprimibles.
- Los caracteres no alfanuméricos están restringidos a (!, &, #, \$, ^, <, >, -).
- La autenticación AUTH solo se puede habilitar para clústeres con cifrado en tránsito ElastiCache (Redis OSS).

Para configurar un token seguro, recomendamos que siga una política de contraseña estricta, como las que exigen las siguientes condiciones:

- Los tokens o contraseñas deben incluir al menos tres de los siguientes tipos de caracteres:
 - Caracteres en mayúsculas
 - Caracteres en minúsculas
 - Dígitos
 - Caracteres no alfanuméricos (!, &, #, \$, ^, <, >, -)
- Los identificadores o contraseñas no deben contener una palabra del diccionario ni una palabra del diccionario ligeramente modificada.
- Los identificadores o contraseñas no deben ser iguales o similares a los identificadores usados recientemente.

Aplicar la autenticación a un clúster ElastiCache (Redis OSS)

Puede exigir a los usuarios que introduzcan un token (contraseña) en un servidor Redis OSS protegido por un token. Para ello, incluya el parámetro `--auth-token` (API: AuthToken) con el

token correcto al crear el clúster o grupo de reproducción. Inclúyalo también en todos los comandos posteriores para el clúster o grupo de reproducción.

La siguiente AWS CLI operación crea un grupo de replicación con el cifrado en tránsito (TLS) habilitado y el token. AUTH *This-is-a-sample-token* Reemplace el grupo de subredes *sng-test* por otro existente.

Parámetros clave

- **--engine**: debe ser `redis`.
- **--engine-version**: debe ser 3.2.6, 4.0.10 o posterior.
- **--transit-encryption-enabled**: obligatorio para la autenticación y elegibilidad para la HIPAA.
- **--auth-token**: obligatorio para la conformidad con la HIPAA. Este valor debe ser el token correcto para este servidor Redis OSS protegido por un token.
- **--cache-subnet-group**: obligatorio para la conformidad con la HIPAA.

Para Linux, macOS o Unix:

```
aws elasticache create-replication-group \  
  --replication-group-id authtestgroup \  
  --replication-group-description authtest \  
  --engine redis \  
  --cache-node-type cache.m4.large \  
  --num-node-groups 1 \  
  --replicas-per-node-group 2 \  
  --transit-encryption-enabled \  
  --auth-token This-is-a-sample-token \  
  --cache-subnet-group sng-test
```

Para Windows:

```
aws elasticache create-replication-group ^  
  --replication-group-id authtestgroup ^  
  --replication-group-description authtest ^  
  --engine redis ^  
  --cache-node-type cache.m4.large ^  
  --num-node-groups 1 ^  
  --replicas-per-node-group 2 ^
```

```
--transit-encryption-enabled ^  
--auth-token This-is-a-sample-token ^  
--cache-subnet-group sng-test
```

Modificar el token AUTH en un clúster existente ElastiCache (Redis OSS)

Para facilitar la actualización de la autenticación, puede modificar el AUTH token utilizado en un clúster ElastiCache (Redis OSS). Puede realizar esta modificación si la versión del motor es 5.0.6 o superior y si ElastiCache (Redis OSS) tiene activado el cifrado en tránsito.

La modificación del token de AUTH admite dos estrategias: ROTATE y SET. La estrategia ROTATE añade un token AUTH adicional al servidor y conserva el token anterior. La estrategia SET actualiza el servidor para que solo admita un token AUTH. Realice estas llamadas de modificación con el parámetro `--apply-immediately` para aplicar los cambios de inmediato.

Rotación del token AUTH

Para actualizar un servidor OSS de Redis con un nuevo token AUTH, llame a la `ModifyReplicationGroup` API con el `--auth-token` parámetro como nuevo AUTH token y el `--auth-token-update-strategy` valor ROTATE. Una vez completada la modificación ROTATE, el clúster admitirá el token AUTH anterior además del especificado en el parámetro `auth-token`. Si no se configuró ningún token AUTH en el grupo de replicación antes de la rotación del token AUTH, el clúster admite el token AUTH especificado en el `--auth-token` parámetro, además de admitir la conexión sin autenticación. Consulte [Configuración del token AUTH](#) para actualizar el token AUTH que sea necesario mediante la estrategia de actualización SET.

Note

Si no configura el token AUTH antes, una vez completada la modificación, el clúster no admitirá ningún token AUTH además del especificado en el parámetro `auth-token`.

Si esta modificación se realiza en un servidor que ya admite dos tokens AUTH, el token AUTH más antiguo también se eliminará durante esta operación. Esto permite que un servidor admita hasta los dos tokens AUTH más recientes a la vez.

En este punto, puede continuar actualizando el cliente para que utilice el último token AUTH. Una vez que se actualizan los clientes, puede utilizar la estrategia SET para la rotación del token AUTH (que se explica en la sección siguiente) a fin de comenzar a utilizar en exclusiva el token nuevo.

La siguiente AWS CLI operación modifica un grupo de replicación para rotar el AUTH token. *This-is-the-rotated-token*

Para Linux, macOS o Unix:

```
aws elasticache modify-replication-group \  
--replication-group-id authtestgroup \  
--auth-token This-is-the-rotated-token \  
--auth-token-update-strategy ROTATE \  
--apply-immediately
```

Para Windows:

```
aws elasticache modify-replication-group ^\  
--replication-group-id authtestgroup ^\  
--auth-token This-is-the-rotated-token ^\  
--auth-token-update-strategy ROTATE ^\  
--apply-immediately
```

Configuración del token AUTH

Para actualizar un servidor OSS de Redis para que admita un único AUTH token obligatorio, llame a la operación de ModifyReplicationGroup API con el `--auth-token` parámetro con el mismo valor que el último token de AUTH y el `--auth-token-update-strategy` parámetro con ese valor. SET La estrategia SET solo se puede usar con un clúster que tenga 2 tokens AUTH o 1 token AUTH opcional si antes se utilizaba una estrategia ROTATE. Una vez completada la modificación, el servidor OSS de Redis solo admite el token AUTH especificado en el parámetro `auth-token`.

La siguiente AWS CLI operación modifica un grupo de replicación en el que se establece el token AUTH. *This-is-the-set-token*

Para Linux, macOS o Unix:

```
aws elasticache modify-replication-group \  
--replication-group-id authtestgroup \  
--auth-token This-is-the-set-token \  
--auth-token-update-strategy SET \  
--apply-immediately
```

Para Windows:

```
aws elasticache modify-replication-group ^  
--replication-group-id authtestgroup ^  
--auth-token This-is-the-set-token ^  
--auth-token-update-strategy SET ^  
--apply-immediately
```

Habilitar la autenticación en un clúster existente ElastiCache (Redis OSS)

Para habilitar la autenticación en un servidor OSS de Redis existente, llame a la operación de ModifyReplicationGroup API. Llame a ModifyReplicationGroup con el parámetro `--auth-token` como el nuevo token y `--auth-token-update-strategy` con el valor ROTATE.

Una vez completada la modificación ROTATE, el clúster admite el AUTH token especificado en el `--auth-token` parámetro, además de admitir la conexión sin autenticación. Una vez que todas las aplicaciones cliente se hayan actualizado para autenticarse en Redis OSS con el token AUTH, utilice la estrategia SET para marcar el token AUTH como necesario. La activación de la autenticación solo se admite en los servidores OSS de Redis con el cifrado en tránsito (TLS) habilitado.

Migración de RBAC a Redis OSS AUTH

Si va a autenticar usuarios con el control de acceso basado en roles (RBAC) de Redis OSS, tal como se describe en [Control de acceso basado en roles \(RBAC\)](#), y desea migrar a Redis OSS AUTH, utilice los siguientes procedimientos. Puede migrar mediante la consola o la CLI.

Para migrar de RBAC a Redis OSS AUTH mediante la consola

1. [Inicie sesión en la consola AWS Management Console y ábrala en https://console.aws.amazon.com/elasticache/](https://console.aws.amazon.com/elasticache/). [ElastiCache](#)
2. En la lista de la esquina superior derecha, elija la AWS región en la que se encuentra el clúster que desea modificar.
3. En el panel de navegación, elija el motor que se ejecuta en el clúster que desea modificar.

Se mostrará una lista de los clústeres del motor elegido.

4. En la lista de clústeres, para el clúster que desea modificar, elija su nombre.
5. Para Actions (Acciones), elija Modify (Modificar).

Aparecerá la ventana Modificar.

6. Para el control de acceso, elija el acceso de usuario predeterminado de Redis OSS AUTH.
7. En el token AUTH OSS de Redis, defina un nuevo token.

8. Seleccione Vista previa de los cambios y seleccione Modificar en la siguiente pantalla.

Para migrar de RBAC a Redis OSS AUTH mediante el AWS CLI

Utilice uno de los siguientes comandos para configurar un nuevo AUTH token opcional para su grupo de replicación de Redis OSS. Tenga en cuenta que un token de autenticación opcional permitirá el acceso no autenticado al grupo de replicación hasta que el token de autenticación se marque como obligatorio, siguiendo la estrategia SET de actualización que se describe en el siguiente paso.

Para Linux, macOS o Unix:

```
aws elasticache modify-replication-group \  
  --replication-group-id test \  
  --remove-user-groups \  
  --auth-token This-is-a-sample-token \  
  --auth-token-update-strategy ROTATE \  
  --apply-immediately
```

Para Windows:

```
aws elasticache modify-replication-group ^  
  --replication-group-id test ^  
  --remove-user-groups ^  
  --auth-token This-is-a-sample-token ^  
  --auth-token-update-strategy ROTATE ^  
  --apply-immediately
```

Tras ejecutar el comando anterior, puede actualizar sus aplicaciones de Redis OSS para que se autenticuen en el grupo de ElastiCache replicación mediante el token de AUTH opcional recién configurado. Para completar la rotación del token de autenticación, utilice la estrategia de actualización del siguiente comando que aparece a SET continuación. Esto marcará el token AUTH opcional como obligatorio. Cuando se complete la actualización del token de autenticación, se mostrará el estado del grupo de replicación ACTIVE y todas las conexiones de Redis OSS a este grupo de replicación requerirán autenticación.

Para Linux, macOS o Unix:

```
aws elasticache modify-replication-group \  
  --replication-group-id test \  
  --auth-token This-is-a-sample-token \  
  --auth-token-update-strategy SET
```

```
--auth-token-update-strategy SET \  
--apply-immediately
```

Para Windows:

```
aws elasticache modify-replication-group ^  
  --replication-group-id test ^  
  --remove-user-groups ^  
  --auth-token This-is-a-sample-token ^  
  --auth-token-update-strategy SET ^  
  --apply-immediately
```

Para obtener más información, consulte [Autenticación con el comando REDIS OSS AUTH](#).

 Note

Si necesita deshabilitar el control de acceso en un ElastiCache clúster, consulte [the section called “Deshabilitar el control de acceso en una caché ElastiCache OSS de Redis”](#)

Deshabilitar el control de acceso en una caché ElastiCache OSS de Redis

Siga las instrucciones que aparecen a continuación para deshabilitar el control de acceso en una memoria caché compatible con Redis OSS TLS. La caché OSS de Redis tendrá uno de dos tipos diferentes de configuraciones: el acceso de usuario predeterminado de Redis OSS AUTH o la lista de control de acceso a grupos de usuarios (RBAC). Si la caché se creó con la configuración AUTH, debe cambiarla a la configuración RBAC antes de poder deshabilitar la caché mediante la eliminación de los grupos de usuarios. Si la caché se creó con la configuración RBAC, puede proceder directamente a deshabilitarla.

Para deshabilitar una caché sin servidor de Redis OSS configurada con RBAC

1. Elimine los grupos de usuarios para deshabilitar el control de acceso.

```
aws elasticache modify-serverless-cache --serverless-cache-name <serverless-cache>  
  --remove-user-group
```

2. (Opcional) Compruebe que no haya ningún grupo de usuarios asociado a la caché sin servidor.

```
aws elasticache describe-serverless-caches --serverless-cache-name <serverless-  
cache>
```

```
{
  "...
  "UserGroupId": ""
  "...
}
```

Para deshabilitar una caché OSS de Redis configurada con un token AUTH

1. Cambie el token AUTH a RBAC y especifique un grupo de usuarios que agregar.

```
aws elasticache modify-replication-group --replication-group-id <replication-group-id-value> --auth-token-update-strategy DELETE --user-group-ids-to-add <user-group-value>
```

2. Compruebe que el token AUTH se haya desactivado y que se haya agregado un grupo de usuarios.

```
aws elasticache describe-replication-groups --replication-group-id <replication-group-id-value>
{
  "...
  "AuthTokenEnabled": false,
  "UserGroupIds": [
    "<user-group-value>"
  ]
  "...
}
```

3. Elimine los grupos de usuarios para deshabilitar el control de acceso.

```
aws elasticache modify-replication-group --replication-group-id <replication-group-value> --user-group-ids-to-remove <user-group-value>
{
  "...
  "PendingModifiedValues": {
    "UserGroups": {
      "UserGroupIdsToAdd": [],
      "UserGroupIdsToRemove": [
        "<user-group-value>"
      ]
    }
  }
  "...
}
```

```
}

```

4. (Opcional) Compruebe que no haya ningún grupo de usuarios asociado al clúster. Además, el campo `AuthTokenEnabled` debe decir `false`.

```
aws elasticache describe-replication-groups --replication-group-id <replication-
group-value>
"AuthTokenEnabled": false

```

Para deshabilitar un clúster de OSS de Redis configurado con RBAC

1. Elimine los grupos de usuarios para deshabilitar el control de acceso.

```
aws elasticache modify-replication-group --replication-group-id <replication-group-
value> --user-group-ids-to-remove <user-group-value>
{
  "...
  "PendingModifiedValues": {
    "UserGroups": {
      "UserGroupIdsToAdd": [],
      "UserGroupIdsToRemove": [
        "<user-group-value>"
      ]
    }
  }
  "...
}
```

2. (Opcional) Compruebe que no haya ningún grupo de usuarios asociado al clúster. Además, el campo `AuthTokenEnabled` debe decir `false`.

```
aws elasticache describe-replication-groups --replication-group-id <replication-
group-value>
"AuthTokenEnabled": false

```

Privacidad del tráfico entre redes

Amazon ElastiCache utiliza las siguientes técnicas para proteger los datos de su caché frente a accesos no autorizados:

- [Seguridad de ElastiCache y Amazon VPC](#) explica el tipo de grupo de seguridad que necesita para su instalación.
- [Identity and Access Management para Amazon ElastiCache](#) para conceder y limitar las acciones de los usuarios, grupos y roles.

Seguridad de ElastiCache y Amazon VPC

Puesto que la seguridad de los datos es importante, ElastiCache ofrece medios para que pueda controlar quién tiene acceso a los datos. El modo en que controla el acceso a sus datos depende de si ha lanzado o no sus clústeres en una Amazon Virtual Private Cloud (Amazon VPC) o en Amazon EC2-Classic.

Important

Hemos dado de baja la utilización de Amazon EC2-Classic para lanzar clústeres de ElastiCache. Todos los nodos de la generación actual se lanzan exclusivamente en Amazon Virtual Private Cloud.

El servicio de Amazon Virtual Private Cloud (Amazon VPC) define una red virtual que se parece mucho a un centro de datos tradicional. Al configurar su Amazon VPC, puede seleccionar su rango de direcciones IP, crear subredes y configurar las tablas de enrutamiento, las gateways de red y la configuración de seguridad. También puede agregar un clúster de caché a la red virtual y controlar el acceso al clúster de caché mediante grupos de seguridad de Amazon VPC.

En esta sección se explica cómo configurar un clúster de ElastiCache manualmente en una Amazon VPC. Esta información se ha pensado para usuarios que desean tener conocimientos más amplios acerca de cómo funcionan ElastiCache y Amazon VPC de manera conjunta.

Temas

- [Comprensión de ElastiCache y Amazon VPC](#)
- [Patrones de acceso para acceder a una ElastiCache caché en una Amazon VPC](#)
- [Creación de una Virtual Private Cloud \(VPC\)](#)
- [Conexión a una caché que se ejecuta en una Amazon VPC](#)

Comprensión de ElastiCache y Amazon VPC

ElastiCache se encuentra totalmente integrado con la Amazon Virtual Private Cloud (Amazon VPC). Para los usuarios de ElastiCache, esto significa lo siguiente:

- Si su cuenta de AWS solo admite la plataforma EC2-VPC, ElastiCache siempre lanza el clúster en una Amazon VPC.
- Si es la primera vez que utiliza AWS, los clústeres se implementarán en una Amazon VPC. Para ello, se creará una VPC predeterminada automáticamente.
- Si tiene una VPC predeterminada y no especifica una subred al lanzar un clúster, el clúster se lanzará en su Amazon VPC predeterminada.

Para obtener más información, consulte [Detección de plataformas compatibles y comprobación de si tiene una VPC predeterminada](#).

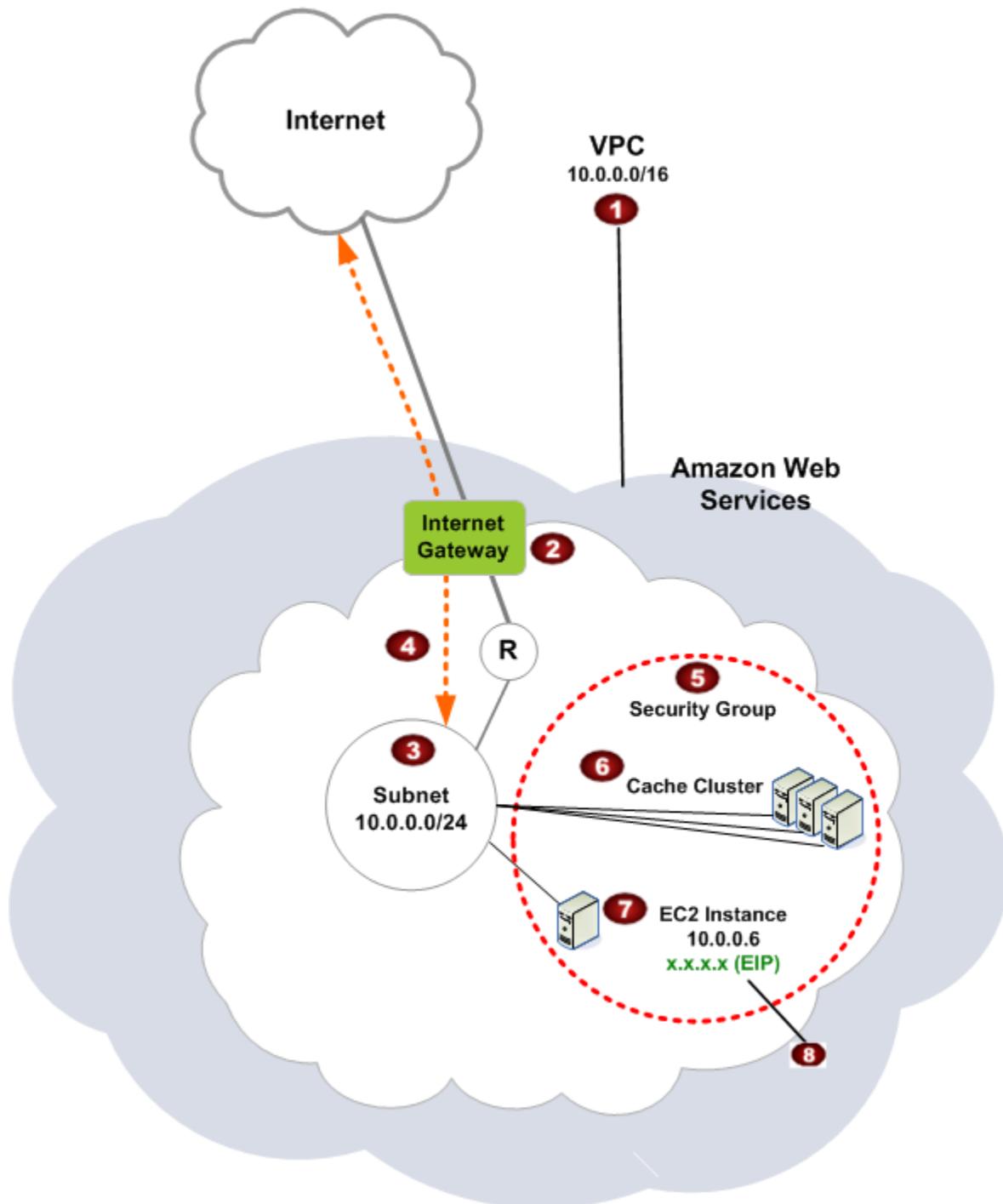
Con Amazon Virtual Private Cloud, podrá crear una red virtual en la nube de AWS que se parece mucho a un centro de datos tradicional. Puede configurar su Amazon VPC, incluido seleccionar el rango de direcciones IP, crear subredes y configurar las tablas de enrutamiento, las gateways de red y la configuración de seguridad.

La funcionalidad básica de ElastiCache funciona exactamente igual en una nube virtual privada; ElastiCache administra las actualizaciones de software, la aplicación de parches, la detección de errores y la recuperación, tanto si los clústeres se han implementado dentro o fuera de una Amazon VPC.

Los nodos de caché de ElastiCache implementados fuera de una Amazon VPC tienen asignada una dirección IP en la que se resuelve el punto de conexión/nombre de DNS. Esto proporciona conectividad de instancias de Amazon Elastic Compute Cloud (Amazon EC2). Al lanzar un clúster de ElastiCache en una subred privada de Amazon VPC, se asigna a todos los nodos de caché una dirección IP privada dentro de dicha subred.

Información general sobre ElastiCache en una Amazon VPC

A continuación, se proporcionan un diagrama y una tabla que describen el entorno de Amazon VPC, junto con los clústeres de ElastiCache y las instancias de Amazon EC2 que se lanzaron en la Amazon VPC.



1

La Amazon VPC es una parte aislada de la nube de AWS a la que se asigna su propio bloque de direcciones IP.

2

Una gateway de Internet conecta la Amazon VPC directamente a Internet y proporciona acceso a otros recursos de AWS como Amazon Simple Storage Service (Amazon S3) que se ejecutan fuera de su Amazon VPC.

3

Una subred de Amazon VPC es un segmento del rango de direcciones IP de una Amazon VPC donde podrá aislar recursos de AWS en función de sus necesidades operativas y de seguridad.

4

Una tabla de enrutamiento en la Amazon VPC dirige el tráfico de red entre la subred e Internet. La Amazon VPC tiene un enrutador implícito, representado en este diagrama mediante el círculo que contiene la letra R.

5

Un grupo de seguridad de Amazon VPC controla el tráfico entrante y saliente de sus clústeres de ElastiCache y sus instancias de Amazon EC2.

6

Puede lanzar un clúster de ElastiCache en la subred. Los nodos de caché tienen direcciones IP privadas del rango de direcciones de la subred.

7

También puede lanzar instancias de Amazon EC2 en la subred. Cada instancia de Amazon EC2 tiene una dirección IP privada del rango de direcciones de la subred. La instancia de Amazon EC2 puede conectarse a cualquier nodo de caché de la misma subred.

8

Para que se pueda obtener acceso a una instancia de Amazon EC2 de su Amazon VPC desde Internet, deberá asignar a la instancia una dirección pública y estática denominada dirección IP elástica.

Requisitos previos

Para crear un clúster de ElastiCache en una Amazon VPC, esta Amazon VPC debe cumplir los siguientes requisitos:

- La Amazon VPC debe permitir instancias de Amazon EC2 no dedicadas. No puede utilizar ElastiCache en una Amazon VPC configurada para la tenencia de instancias dedicadas.

- Debe definir un grupo de subredes de caché para su Amazon VPC. ElastiCache utiliza dicho grupo de subredes de caché para seleccionar una subred y direcciones IP pertenecientes a ella y asociarlas a sus puntos de conexión de VPC o nodos de caché.
- Los bloques de CIDR de cada subred deben ser lo suficientemente grandes como para proporcionar direcciones IP auxiliares para que pueda utilizarlas durante las actividades de mantenimiento.

Enrutamiento y seguridad

Puede configurar el enrutamiento en la Amazon VPC para controlar dónde fluye el tráfico (por ejemplo, a la gateway de Internet o la gateway privada virtual). Con una gateway de Internet, la Amazon VPC tiene acceso directo a otros recursos de AWS que no se ejecutan en su Amazon VPC. Si decide tener solo una gateway privada virtual con una conexión a la red local de su organización, puede enrutar el tráfico vinculado a Internet a través de la VPN y utilizar políticas de seguridad locales y firewalls para controlar las salidas. En ese caso, se cobrarán tarifas de banda ancha adicionales al obtener acceso a los recursos de AWS a través de Internet.

Puede utilizar grupos de seguridad de Amazon VPC para ayudar a proteger los clústeres de ElastiCache y las instancias de Amazon EC2 de la Amazon VPC. Los grupos de seguridad actúan como un firewall en el ámbito de la instancia, no en el de la subred.

Note

Recomendamos utilizar nombres de DNS para conectarse a los nodos de caché, ya que la dirección IP subyacente puede cambiar.

Documentación de Amazon VPC

Amazon VPC tiene su propia serie de documentación que describe cómo crear y utilizar una Amazon VPC. En la siguiente tabla, se proporcionan enlaces a las guías de Amazon VPC.

Descripción	Documentación
Cómo comenzar a utilizar Amazon VPC	Introducción a Amazon VPC
Cómo utilizar Amazon VPC a través de la AWS Management Console	Guía del usuario de Amazon VPC

Descripción	Documentación
Descripciones completas de todos los comandos de Amazon VPC	Referencia de línea de comandos de Amazon EC2 (los comandos de Amazon VPC se encuentran en la referencia de Amazon EC2)
Descripciones completas de las operaciones, los tipos de datos y los errores de la API de Amazon VPC	Referencia de la API de Amazon EC2 (las operaciones de la API de Amazon VPC se encuentran en la referencia de Amazon EC2)
Información para el administrador de red que necesita configurar la gateway en su extremo para disponer de una conexión de VPN IPsec opcional	¿Qué es AWS Site-to-Site VPN?

Para obtener información más detallada sobre Amazon Virtual Private Cloud, consulte [Amazon Virtual Private Cloud](#).

Patrones de acceso para acceder a una ElastiCache caché en una Amazon VPC

Amazon ElastiCache admite los siguientes escenarios para acceder a una caché en una Amazon VPC:

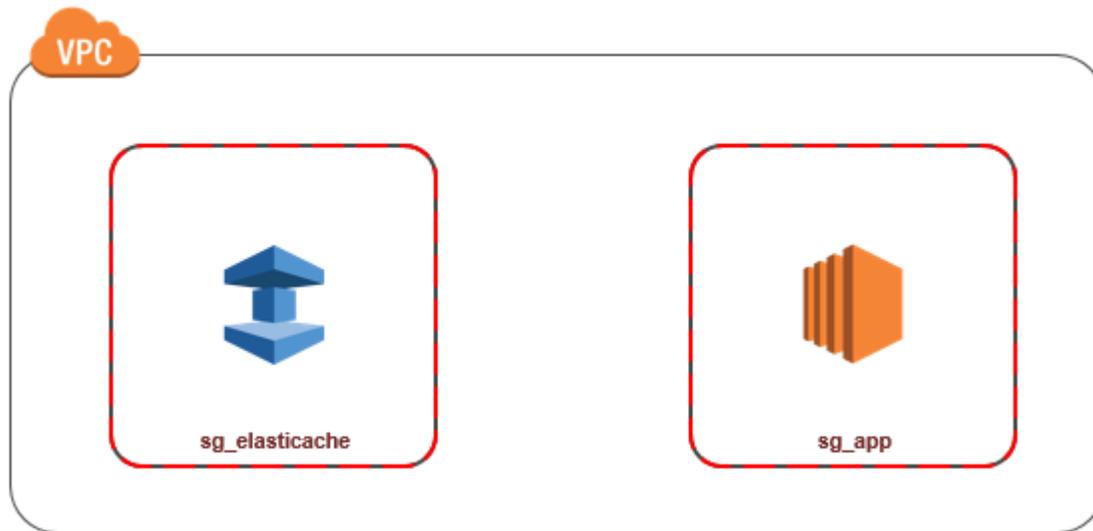
Contenido

- [Acceso a una ElastiCache caché cuando ésta y la instancia de Amazon EC2 se encuentran en la misma Amazon VPC](#)
- [Acceso a una ElastiCache memoria caché cuando ésta y la instancia de Amazon EC2 se encuentran en distintas Amazon VPC](#)
 - [Acceso a una ElastiCache caché cuando ésta y la instancia de Amazon EC2 se encuentran en distintas Amazon VPC en la misma región](#)
 - [Uso de Transit Gateway](#)
 - [Acceso a una ElastiCache caché cuando ésta y la instancia de Amazon EC2 se encuentran en distintas Amazon VPC en distintas regiones](#)
 - [Uso de la VPC de tránsito](#)
- [Acceso a una ElastiCache memoria caché desde una aplicación que se ejecuta en el centro de datos de un cliente](#)
 - [Acceder a una ElastiCache memoria caché desde una aplicación que se ejecuta en el centro de datos de un cliente mediante la conectividad VPN](#)
 - [Acceso a una ElastiCache memoria caché desde una aplicación que se ejecuta en el centro de datos de un cliente mediante Direct Connect](#)

Acceso a una ElastiCache caché cuando ésta y la instancia de Amazon EC2 se encuentran en la misma Amazon VPC

El caso de uso más común es cuando una aplicación implementada en una instancia EC2 debe conectarse a una caché en la misma VPC.

En el siguiente diagrama se ilustra este escenario.



La forma más sencilla de administrar el acceso entre instancias EC2 y cachés de la misma VPC es hacer lo siguiente:

1. Crear un grupo de seguridad VPC para la caché. Este grupo de seguridad se puede utilizar para restringir el acceso a la caché. Por ejemplo, puede crear una regla personalizada para este grupo de seguridad que permita el acceso mediante TCP utilizando el puerto que asignó a la caché cuando la creó y una dirección IP que se utilizará para obtener acceso a la caché.

El puerto predeterminado para las cachés OSS de Redis es. 6379

2. Cree un grupo de seguridad de VPC para sus instancias EC2 (servidores web y de aplicaciones). Si es necesario, este grupo de seguridad puede permitir el acceso a la instancia EC2 desde Internet a través de la tabla de enrutamiento de la VPC. Por ejemplo, puede establecer reglas en este grupo de seguridad para permitir el acceso mediante TCP a la instancia EC2 a través del puerto 22.
3. Crear reglas personalizadas en el grupo de seguridad para la caché que permitan las conexiones desde el grupo de seguridad que ha creado para las instancias EC2. Esto permitirá a cualquier miembro del grupo de seguridad obtener acceso a las cachés.

Note

Si planea utilizar [Local Zones](#), asegúrese de que las ha habilitado. Cuando crea un grupo de subredes en esa zona local, la VPC se amplía a dicha zona local y la VPC tratará la subred

como cualquier subred de cualquier otra zona de disponibilidad. Todas las gateways y tablas de enrutamiento relevantes se ajustarán de forma automática.

Para crear una regla en un grupo de seguridad de VPC que permita establecer conexiones desde otro grupo de seguridad

1. [Inicie sesión en la consola AWS de administración y abra la consola de Amazon VPC en https://console.aws.amazon.com/vpc.](https://console.aws.amazon.com/vpc)
2. En el panel de navegación, elija Grupos de seguridad.
3. Seleccione o cree un grupo de seguridad que utilizará para la caché En Inbound Rules (Reglas de entrada), seleccione Edit Inbound Rules (Editar reglas de entrada) y, a continuación, seleccione Add Rule (Agregar regla). Este grupo de seguridad permitirá el acceso a los miembros de otro grupo de seguridad.
4. En Type (Tipo), elija Custom TCP Rule (Personalizar regla de TCP).
 - a. En Rango de puerto, especifique el puerto que utilizó al crear la caché.

El puerto predeterminado para las cachés y los grupos de replicación de Redis OSS es. 6379
 - b. En el cuadro Source (Fuente), comience a escribir el ID del grupo de seguridad. Desde la lista, seleccione el grupo de seguridad que utilizará para sus instancias de Amazon EC2.
5. Cuando haya terminado, elija Save (Guardar).

	Name	Security group rule...	IP version	Type	Protocol	Port range	Source	Description
<input type="checkbox"/>	-	sg-...	IPv4	SSH	TCP	22	0.0.0.0/0	-
<input type="checkbox"/>	-	sg-...	-	Custom TCP	TCP	6379	sg-... / default	-

Acceso a una ElastiCache memoria caché cuando ésta y la instancia de Amazon EC2 se encuentran en distintas Amazon VPC

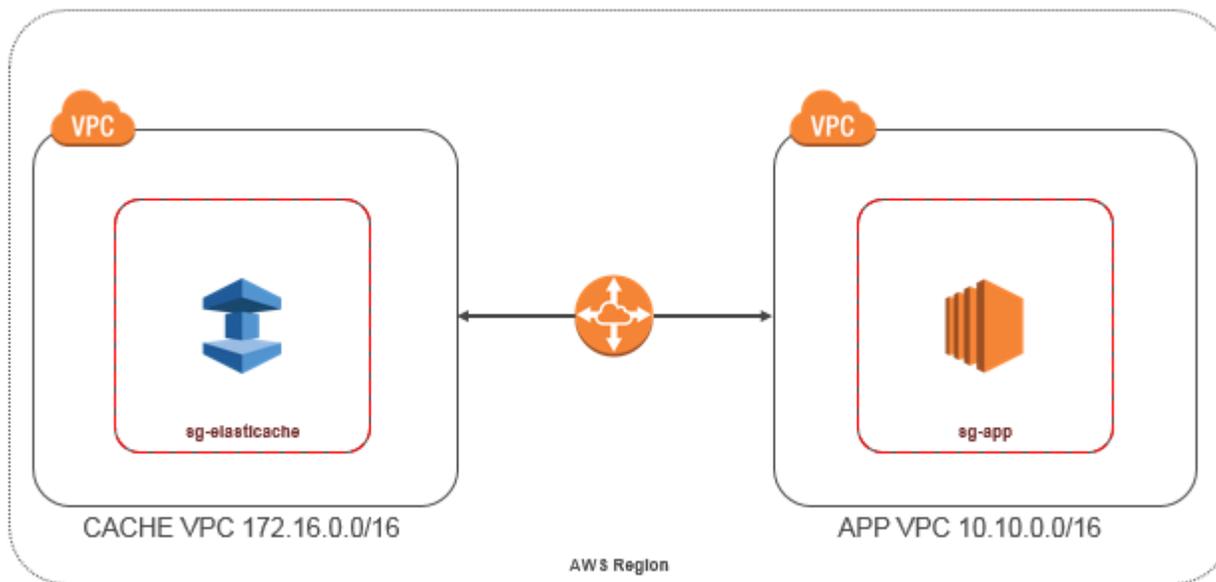
Cuando una caché está en una VPC que no coincide con la de la instancia EC2 que se está utilizando para acceder a ella, hay varias formas de obtener acceso a la caché. Si la caché y la instancia EC2 están en VPC distintas, pero en la misma región, puede utilizar la conexión de emparejamiento de VPC. Si la caché y la instancia EC2 se encuentran en distintas regiones, puede crear conectividad de VPN entre regiones.

Temas

- [Acceso a una ElastiCache caché cuando ésta y la instancia de Amazon EC2 se encuentran en distintas Amazon VPC en la misma región](#)
- [Acceso a una ElastiCache caché cuando ésta y la instancia de Amazon EC2 se encuentran en distintas Amazon VPC en distintas regiones](#)

Acceso a una ElastiCache caché cuando ésta y la instancia de Amazon EC2 se encuentran en distintas Amazon VPC en la misma región

En el siguiente diagrama, se ilustra el acceso a una caché por parte de una instancia de Amazon EC2 en una Amazon VPC diferente y en la misma región a través de una conexión de emparejamiento de VPC.



Acceso a caché de una instancia de Amazon EC2 en una Amazon VPC diferente dentro de la misma región: conexión de emparejamiento de VPC

Una conexión de emparejamiento de VPC es una conexión de redes entre dos VPC que permite direccionar el tráfico entre ellas mediante direcciones IP privadas. Las instancias de ambas VPC se pueden comunicar entre sí siempre que se encuentren en la misma red. Puede crear una conexión de emparejamiento de VPC entre sus propias Amazon VPC o con una Amazon VPC de otra AWS cuenta de una sola región. Para obtener más información sobre la conexión de emparejamiento de Amazon VPC, consulte la [documentación de VPC](#).

Note

La resolución de nombres DNS puede fallar en las VPC emparejadas, según las configuraciones aplicadas a la ElastiCache VPC. Para solucionar esto, ambas VPC deben tener habilitados los nombres de host DNS y la resolución de DNS. Para obtener más información, consulte [Habilitación de la resolución de DNS para la conexión de emparejamiento de VPC](#).

Para obtener acceso a una caché en una Amazon VPC diferente a través de emparejamiento

1. Asegúrese de que las dos VPC no tengan rangos de IP solapados o no podrá interconectarlas.
2. Coloque las dos VPC al mismo nivel. Para obtener más información, consulte [Creación y aceptación de interconexiones de Amazon VPC](#).
3. Actualice su tabla de ruteo. Para obtener más información, consulte [Actualización de las tablas de ruteo para interconexiones de VPC](#)

A continuación, se muestra el aspecto que tienen las tablas de ruteo para el ejemplo del diagrama anterior. Tenga en cuenta que pcx-a894f1c1 es la conexión de emparejamiento.

Destination	Target	Destination	Target
172.16.0.0/16	local	10.10.0.0/16	local
10.10.0.0/16	pcx-a894f1c1	0.0.0.0/0	igw-bfdcccd8
		172.16.0.0/16	pcx-a894f1c1

Tabla de enrutamiento de VPC

4. Modifique el grupo de seguridad de la ElastiCache memoria caché para permitir la conexión entrante desde el grupo de seguridad de la aplicación en la VPC interconectada. Para obtener más información, consulte [Actualización de los grupos de seguridad para que hagan referencia a grupos de la VPC del mismo nivel](#).

El acceso a una caché a través de una conexión de emparejamiento generará costos de transferencia de datos adicionales.

Uso de Transit Gateway

Una puerta de enlace de tránsito le permite conectar VPC y conexiones VPN en la misma AWS región y enrutar el tráfico entre ellas. Una pasarela de transporte público funciona en todas AWS las cuentas y puedes usar AWS Resource Access Manager para compartir tu pasarela de transporte público con otras cuentas. Después de compartir una pasarela de transporte público con otra AWS cuenta, el propietario de la cuenta puede adjuntar sus VPC a la pasarela de transporte público. Un usuario de cualquiera de las cuentas puede eliminar la vinculación en cualquier momento.

Puede habilitar la multidifusión en una puerta de enlace de tránsito y, a continuación, crear un dominio de multidifusión de transit puerta de enlace que permita que el tráfico de multidifusión se envíe desde el origen de multidifusión a los miembros del grupo de multidifusión a través de conexiones de la VPC que asocie con el dominio.

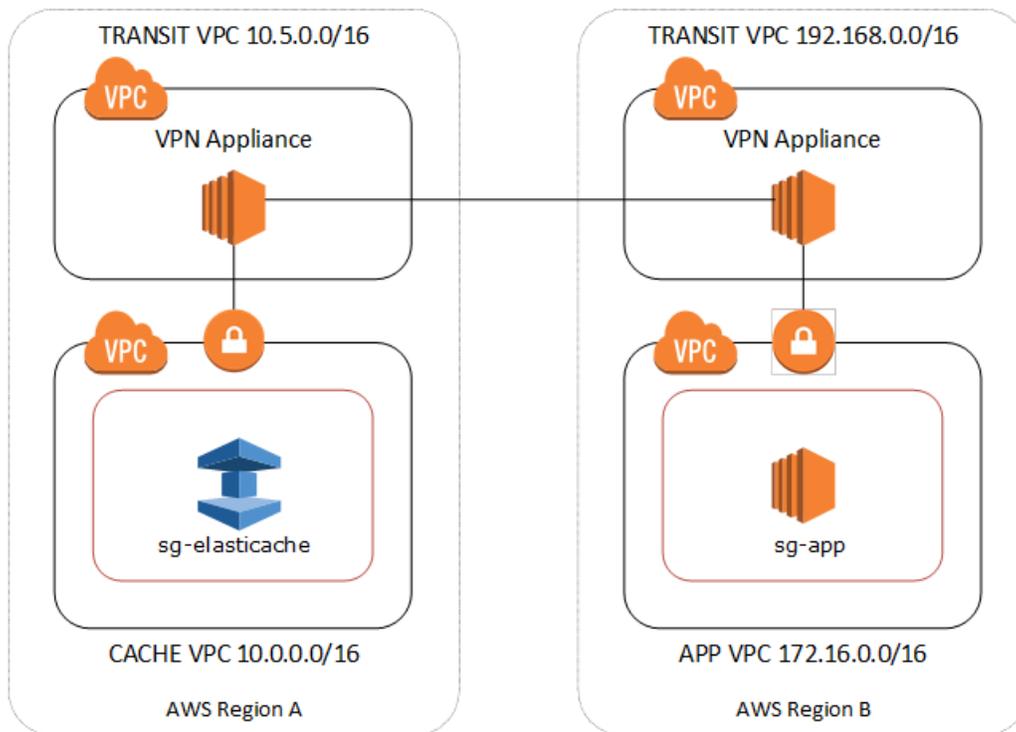
También puedes crear un adjunto de conexión entre pasarelas de tránsito de distintas regiones. AWS Esto le permite dirigir el tráfico entre las vinculaciones de las transit gateways a través de diferentes regiones.

Para obtener más información, consulte [Transit gateways](#).

Acceso a una ElastiCache caché cuando ésta y la instancia de Amazon EC2 se encuentran en distintas Amazon VPC en distintas regiones

Uso de la VPC de tránsito

Una alternativa a la utilización de la conexión de emparejamiento de VPC, otra estrategia común para conectar varias VPC y redes remotas dispersas geográficamente es crear una VPC de tránsito que sirva como un centro de tránsito de red global. Una VPC de tránsito simplifica la administración de la red y minimiza el número de conexiones necesarias para conectar varias VPC y redes remotas. Este diseño puede ahorrar tiempo y esfuerzo, además de reducir los costos, ya que se implementa prácticamente sin los gastos tradicionales de establecer una presencia física en un hub de tránsito de coubicación o de implementar un equipo de red física.



Conexión entre VPC diferentes en distintas regiones

Una vez establecida la VPC Amazon de Transit, una aplicación desplegada en una VPC «radial» de una región puede conectarse a una ElastiCache caché de una VPC «radial» de otra región.

Para acceder a una caché en una VPC diferente dentro de una región diferente AWS

1. Implemente una solución de VPC de tránsito. Para obtener más información, consulte [AWS Transit Gateway](#).
2. Actualice las tablas de enrutamiento de la VPC en la aplicación y las VPC de la caché para direccionar el tráfico a través de la VGW (gateway privada virtual) y el dispositivo de VPN. En caso de que se produzca el enrutamiento dinámico con el protocolo de gateway fronteriza (BGP), las rutas se pueden propagar automáticamente.
3. Modifique el grupo de seguridad de su ElastiCache caché para permitir la conexión entrante desde el rango de IP de las instancias de aplicación. Tenga en cuenta que no podrá remitirse al grupo de seguridad de servidor de la aplicación en este caso.

El acceso a una caché entre regiones conllevará latencias de red y costos adicionales de transferencia de datos entre regiones.

Acceso a una ElastiCache memoria caché desde una aplicación que se ejecuta en el centro de datos de un cliente

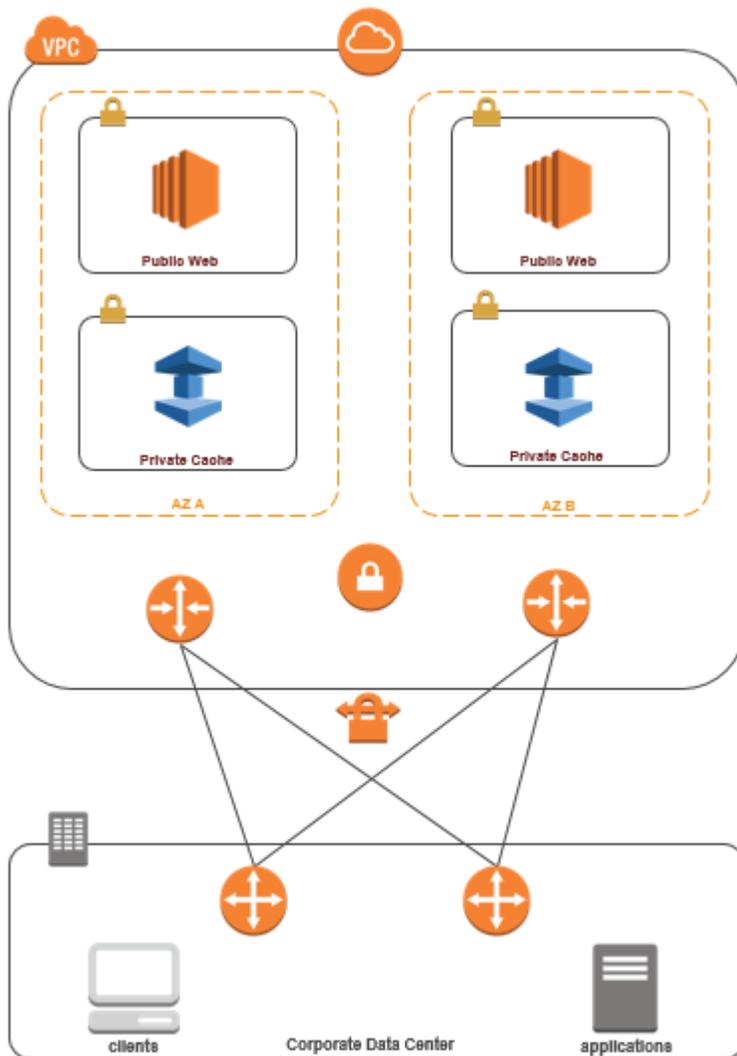
Otro escenario posible es una arquitectura híbrida en la que los clientes o las aplicaciones del centro de datos del cliente puedan necesitar acceder a una ElastiCache memoria caché en la VPC. Esta situación también se admite, siempre que haya conectividad entre la VPC del cliente y el centro de datos, ya sea a través de la VPN como de Direct Connect.

Temas

- [Acceder a una ElastiCache memoria caché desde una aplicación que se ejecuta en el centro de datos de un cliente mediante la conectividad VPN](#)
- [Acceso a una ElastiCache memoria caché desde una aplicación que se ejecuta en el centro de datos de un cliente mediante Direct Connect](#)

Acceder a una ElastiCache memoria caché desde una aplicación que se ejecuta en el centro de datos de un cliente mediante la conectividad VPN

El siguiente diagrama ilustra el acceso a una ElastiCache memoria caché desde una aplicación que se ejecuta en la red corporativa mediante conexiones VPN.



Conexión ElastiCache desde su centro de datos a través de una VPN

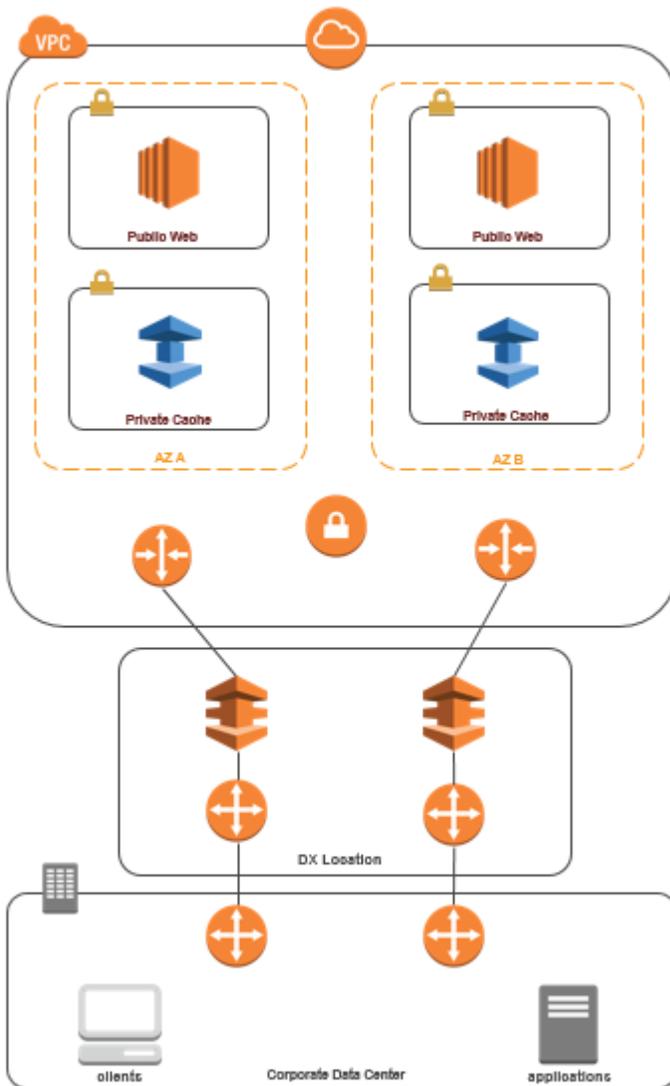
Para obtener acceso a una caché en una VPC desde una aplicación local a través de una conexión de VPN

1. Para establecer la conectividad de VPN, agregue una gateway privada virtual de hardware a su VPC. Para obtener más información, consulte [Adición de una gateway privada virtual de hardware a la VPC](#).
2. Actualice la tabla de enrutamiento de VPC de la subred en la que se implementa la ElastiCache caché para permitir el tráfico desde el servidor de aplicaciones local. En caso de que se produzca el enrutamiento dinámico con BGP, las rutas se pueden propagar automáticamente.
3. Modifique el grupo de seguridad de su ElastiCache caché para permitir la conexión entrante desde los servidores de aplicaciones locales.

El acceso a una caché a través de una conexión de VPN conllevará latencias de red y costos adicionales de transferencia de datos.

Acceso a una ElastiCache memoria caché desde una aplicación que se ejecuta en el centro de datos de un cliente mediante Direct Connect

El siguiente diagrama ilustra el acceso a una ElastiCache memoria caché desde una aplicación que se ejecuta en la red corporativa mediante Direct Connect.



Conexión ElastiCache desde su centro de datos a través de Direct Connect

Para acceder a una ElastiCache memoria caché desde una aplicación que se ejecuta en la red mediante Direct Connect

1. Establezca la conectividad de Direct Connect. Para obtener más información, consulte [Introducción a AWS Direct Connect](#).
2. Modifique el grupo de seguridad de la ElastiCache memoria caché para permitir la conexión entrante desde los servidores de aplicaciones locales.

El acceso a una caché a través de una conexión de DX puede conllevar latencias de red y cargos adicionales por transferencia de datos.

Creación de una Virtual Private Cloud (VPC)

En este ejemplo, creará una Amazon VPC con una subred privada para cada zona de disponibilidad.

Creación de una Amazon VPC (consola)

1. Inicie sesión en la consola de administración de AWS y abra la consola de Amazon VPC en <https://console.aws.amazon.com/vpc/>.
2. En el panel de VPC, elija Create VPC (Crear VPC).
3. En Recursos para crear elija VPC y más.
4. En Number of Availability Zones (AZs) (Número de zonas de disponibilidad), seleccione el número de zonas de disponibilidad que quiere usar con las subredes.
5. En Number of public subnets (Número de subredes públicas), elija el número de subredes públicas que desea agregar a la VPC.
6. En Number of private subnets (Número de subredes privadas), elija el número de subredes públicas que desea agregar a la VPC.

Tip

Anote los identificadores de las subredes e identifique cuál es pública y cuál es privada. Necesitará esta información más adelante al lanzar sus clústeres y agregar una instancia de Amazon EC2 a su Amazon VPC.

7. Cree un grupo de seguridad de Amazon VPC. Utilizará este grupo para su clúster de caché y su instancia de Amazon EC2.
 - a. En el panel de navegación de la consola de administración de Amazon VPC, elija Security Groups (Grupos de seguridad).
 - b. Elija Crear grupo de seguridad.
 - c. Escriba un nombre y una descripción para el grupo de seguridad en los cuadros correspondientes. En el cuadro VPC, elija el identificador de su Amazon VPC.

Create security group [Info](#)

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

Basic details

Security group name [Info](#)

Name cannot be edited after creation.

Description [Info](#)

VPC [Info](#)

Inbound rules [Info](#)

This security group has no inbound rules.

Outbound rules [Info](#)

Type	Protocol	Port range	Destination	Description - optional	
All traffic	All	All	Custom	<input type="text"/>	<input type="button" value="Delete"/>
			<input type="text" value="0.0.0.0"/>		

- d. Una vez que la configuración sea la deseada, elija Yes, Create (Sí, crear).
8. Defina una regla de entrada de red para su grupo de seguridad. Esta regla permitirá conectarse a su instancia de Amazon EC2 mediante Secure Shell (SSH).
 - a. En la lista de navegación, elija Security Groups (Grupos de seguridad).
 - b. Busque el grupo de seguridad en la lista y, a continuación, elíjalo.
 - c. En Security Group (Grupo de seguridad), elija la pestaña Inbound (Entrada). En el cuadro Create a new rule (Crear una nueva regla), elija SSH y, a continuación, elija Add Rule (Agregar regla).
 - d. Establezca los siguientes valores para la regla de entrada nueva a fin de permitir el acceso HTTP:
 - Tipo: HTTP
 - Fuente: 0.0.0.0/0

Elija Apply Rule Changes (Aplicar cambios de regla).

Ahora se encuentra preparado para crear un grupo de subredes de caché y lanzar un clúster de caché en su Amazon VPC.

- [Creación de un grupo de subredes](#)
- [Creación de un clúster de Redis OSS \(modo de clúster desactivado\) \(consola\)](#).

Conexión a una caché que se ejecuta en una Amazon VPC

En este ejemplo, se muestra cómo lanzar una instancia de Amazon EC2 en su Amazon VPC. A continuación, puede iniciar sesión en esta instancia y acceder a la ElastiCache caché que se ejecuta en la Amazon VPC.

Conexión a una caché que se ejecuta en una Amazon VPC (consola)

En este ejemplo, crea una instancia de Amazon EC2 en su Amazon VPC. Puede utilizar esta instancia de Amazon EC2 para conectarse a nodos de caché que se ejecutan en la Amazon VPC.

Note

Para obtener información sobre la utilización de Amazon EC2, consulte la [Guía de introducción a Amazon EC2](#) en la [Documentación de Amazon EC2](#).

Para crear una instancia de Amazon EC2 en su Amazon VPC mediante la consola de Amazon EC2

1. [Inicie sesión en la consola Amazon EC2 AWS Management Console y ábrala en https://console.aws.amazon.com/ec2/](https://console.aws.amazon.com/ec2/).
2. En la consola, elija Launch Instance (Lanzar instancia) y siga los pasos siguientes:
3. En la página Choose an Amazon Machine Image (AMI) (Elegir una Amazon Machine Image [AMI]), elija una AMI de Amazon Linux de 64 bits y, a continuación, elija Select (Seleccionar).
4. En la página de Choose an Instance Type (Elegir un tipo de instancias), elija 3. Configure Instance (Configurar instancia).
5. En la página Configure Instance Details (Configurar detalles de instancia), seleccione lo siguiente:
 - a. En la lista de Network (Red), elija su Amazon VPC.
 - b. En la lista Subred, elija su subred pública.

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Tag Instance 6. Configure Security Group 7. Review

Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot Instances to take advantage pricing, assign an access management role to the instance, and more.

Number of instances

Purchasing option Request Spot Instances

Network

Subnet
250 IP Addresses available

Public IP Automatically assign a public IP address to your instances

Cuando esté conforme con los ajustes, elija 4. Add Storage (Agregar almacenamiento).

6. En la página de Add Storage (Agregar almacenamiento), elija 5. Tag Instance (Etiquetar instancia).
7. En la página de Tag Instance (Etiquetar instancia), escriba un nombre para su instancia de Amazon EC2 y, a continuación, elija 6. Configure Security Group (Configurar grupo de seguridad).
8. En la página Configure Security Group (Configurar grupo de seguridad), elija Select an existing security group (Seleccionar un grupo de seguridad existente). Para obtener más información sobre los grupos de seguridad, consulte [Grupos de seguridad de Amazon EC2 para instancias Linux](#).

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Tag Instance 6. Configure Security Group 7. Review

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP. You can create a new security group or select from an existing one below. [Learn more about Amazon EC2 security groups.](#)

Assign a security group: Create a new security group
 Select an existing security group

Security Group ID	Name	Description
<input type="checkbox"/> sg-1a3d2178	default	default VPC security group
<input checked="" type="checkbox"/> sg-f13d2193	my-vpc-security-group	Testing

Elija el nombre de su grupo de seguridad de Amazon VPC y, a continuación, elija Review and Launch (Revisar y lanzar).

9. En la página Review Instance and Launch (Revisar instancia y lanzar), elija Launch (Lanzar).

10. En la ventana **Select an existing key pair or create a new key pair** (Seleccionar un par de claves existente o crear uno nuevo), especifique el par de claves que desea utilizar con esta instancia.

 Note

Para obtener información sobre la administración de pares de claves, consulte la [Guía de introducción a Amazon EC2](#).

11. Cuando se encuentre listo para lanzar la instancia de Amazon EC2, elija **Launch** (Lanzar).

Ahora puede asignar una dirección IP elástica a la instancia de Amazon EC2 que acaba de crear. Necesitará utilizar esta dirección IP para conectarse a la instancia de Amazon EC2.

Para asignar una dirección IP elástica (consola)

1. Abra la consola de Amazon VPC en <https://console.aws.amazon.com/vpc/>.
2. En la lista de navegación, elija **Elastic IPs** (IP elásticas).
3. Elija **Asignar dirección IP elástica**.
4. En el cuadro de diálogo **Allocate Elastic IP address** (Asignar dirección IP elástica) acepte el **Network Border Group** (Grupo de bordes de red) predeterminado y elija **Allocate** (Asignar).
5. Elija la dirección IP elástica que acaba de asignar de la lista y elija **Associate Address** (Asociar dirección).
6. En el cuadro de diálogo de **Associate Address** (Asociar dirección), en el cuadro de **Instance** (Instancia), elija el ID de la instancia de Amazon EC2 que ha lanzado.

En el cuadro **Private IP address** (Dirección IP privada), seleccione el cuadro para obtener la dirección IP privada y, a continuación, elija **Associate** (Asociar).

Ahora puede utilizar SSH para conectarse a la instancia de Amazon EC2 mediante la dirección IP elástica que ha creado.

Para conectarse con la instancia de Amazon EC2, siga estos pasos:

- Abra una ventana del sistema. En el símbolo del sistema, envíe el comando siguiente y reemplace `mykeypair.pem` por el nombre de su archivo de par de claves y `54.207.55.251` con su dirección IP elástica.

```
ssh -i mykeypair.pem ec2-user@54.207.55.251
```

⚠ Important

Todavía no cierre la sesión de su instancia de Amazon EC2.

Ahora está listo para interactuar con su ElastiCache clúster. Para ello, debe instalar la utilidad telnet, si es que todavía no lo ha hecho.

Para instalar telnet e interactuar con su clúster de caché (AWS CLI)

1. Abra una ventana del sistema. En el símbolo del sistema, envíe el comando que se indica a continuación. En el mensaje de confirmación, escriba y.

```
sudo yum install telnet
Loaded plugins: priorities, security, update-motd, upgrade-helper
Setting up Install Process
Resolving Dependencies
--> Running transaction check

...(output omitted)...

Total download size: 63 k
Installed size: 109 k
Is this ok [y/N]: y
Downloading Packages:
telnet-0.17-47.7.amzn1.x86_64.rpm                | 63 kB      00:00

...(output omitted)...

Complete!
```

2. Utilice telnet para conectarse a su punto de conexión de nodo de caché a través del puerto 6379. Reemplace el nombre de host que se muestra a continuación con el nombre de host de su nodo de caché.

```
telnet my-cache-cluster.7wufxa.0001.use1.cache.amazonaws.com 6379
```

Ya está conectado a la caché del motor y puede enviar comandos. En este ejemplo, se añade un elemento de datos a la caché para, a continuación, obtenerlo inmediatamente. Por último, desconéctese del nodo de caché.

Para almacenar una clave y un valor, escriba las dos líneas siguientes:

```
set mykey myvalue
```

El motor de caché responde con lo siguiente:

```
OK
```

Para recuperar el valor de mykey, escriba lo siguiente:

```
get mykey
```

Para desconectarse del motor de caché, escriba lo siguiente:

```
quit
```

3. Ve a la ElastiCache consola en <https://console.aws.amazon.com/elasticache/> y obtén el punto final de uno de los nodos de tu clúster de caché. Para obtener más información, consulte [Búsqueda de puntos de conexión](#) para Redis OSS.
4. Utilice telnet para conectarse a su punto de conexión de nodo de caché a través del puerto 6379. Reemplace el nombre de host que se muestra a continuación con el nombre de host de su nodo de caché.

```
telnet my-cache-cluster.7wufxa.0001.use1.cache.amazonaws.com 6379
```

Ya está conectado a la caché del motor y puede enviar comandos. En este ejemplo, se añade un elemento de datos a la caché para, a continuación, obtenerlo inmediatamente. Por último, desconéctese del nodo de caché.

Para almacenar una clave y un valor, escriba lo siguiente:

```
set mykey myvalue
```

El motor de caché responde con lo siguiente:

```
OK
```

Para recuperar el valor de mykey, escriba lo siguiente:

```
get mykey
```

El motor de caché responde con lo siguiente:

```
get mykey  
myvalue
```

Para desconectarse del motor de caché, escriba lo siguiente:

```
quit
```

Important

Para evitar incurrir en cargos adicionales en su AWS cuenta, asegúrese de eliminar AWS los recursos que ya no desee utilizar después de probar estos ejemplos.

Puntos de conexión de VPC de interfaz y API de Amazon ElastiCache (AWS PrivateLink)

Puede establecer una conexión privada entre los puntos de conexión de la VPC y la API de Amazon ElastiCache mediante la creación de un punto de conexión de VPC de interfaz . Los puntos de conexión de interfaz que cuentan con [AWS PrivateLink](#). AWS PrivateLink le permite acceder de forma privada a las operaciones de la API de Amazon ElastiCache sin una puerta de enlace de Internet, un dispositivo NAT, una conexión VPN o una conexión AWS Direct Connect.

Las instancias de la VPC no necesitan direcciones IP públicas para comunicarse con el punto de conexión de la API de Amazon ElastiCache. Sus instancias tampoco necesitan direcciones IP públicas para utilizar ninguna de las operaciones de la API ElastiCache de RDS disponibles. El

tráfico entre su VPC y Amazon ElastiCache no sale de la red de Amazon. Cada punto de conexión de la interfaz está representado por una o más interfaces de red elásticas en las subredes. Para obtener más información sobre las interfaces de red elásticas, consulte [Interfaces de red elásticas](#) en la Guía del usuario de Amazon EC2.

- Para obtener más información sobre puntos de conexión de la VPC, consulte [Puntos de enlace de la VPC de tipo interfaz \(AWS PrivateLink\)](#) en la Guía del usuario de Amazon VPC.
- Para obtener más información sobre las operaciones de la API de ElastiCache, consulte [Operaciones de API de ElastiCache](#).

Después de crear un punto de conexión de VPC de tipo interfaz, si habilita nombres de host [DNS privados](#) para el punto de conexión, el punto de conexión predeterminado de ElastiCache (`https://elasticache.Region.amazonaws.com`) se resuelve en el punto de conexión de VPC. Si no habilita nombres de host de DNS privados, Amazon VPC proporciona un nombre de punto de conexión de DNS que puede utilizar en el siguiente formato:

```
VPC_Endpoint_ID.elasticache.Region.vpce.amazonaws.com
```

Para obtener más información, consulte [Puntos de conexión de la VPC de tipo interfaz \(AWS PrivateLink\)](#) en la Guía del usuario de Amazon VPC. ElastiCache permite realizar llamadas a todas sus [acciones de API](#) dentro de su VPC.

Note

Los nombres de host DNS privados solo se pueden habilitar para un punto de conexión de VPC en la VPC. Si quiere crear un punto de conexión de VPC adicional, el nombre de host DNS privado debe estar deshabilitado para ello.

Consideraciones para los puntos de conexión de VPC de

Antes de configurar un punto de conexión de VPC de tipo interfaz para los puntos de conexión de la API de Amazon ElastiCache, asegúrese de revisar [Propiedades y limitaciones del puntos de conexión de la interfaz](#) en la Guía del usuario de Amazon VPC. Todas las operaciones de API de ElastiCache relevantes para la administración de los recursos Amazon ElastiCache están disponibles desde la VPC mediante el uso de AWS PrivateLink.

Las políticas de puntos de conexión de VPC son compatibles con los puntos de conexión de API de ElastiCache. De forma predeterminada, se permite el acceso completo a las operaciones de API de ElastiCache a través del punto de conexión. Para obtener más información, consulte [Control del acceso a los servicios con puntos de conexión de la VPC](#) en la guía del usuario de Amazon VPC.

Creación de un punto de conexión de VPC de interfaz para la API de ElastiCache

Puede crear un punto de conexión de VPC para la API de Amazon ElastiCache mediante la consola de Amazon VPC o la AWS CLI. Para obtener más información, consulte [Creación de un punto de conexión de interfaz](#) en la Guía del usuario de Amazon VPC.

Después de crear un punto de conexión de VPC de interfaz, puede habilitar nombres de host de DNS privados para el punto de conexión. Cuando lo haga, el punto de conexión predeterminado de Amazon ElastiCache (<https://elasticache.Region.amazonaws.com>) se resuelve en el punto de conexión de VPC. En las regiones de China (Pekín) y China (Ningxia) de AWS, puede realizar solicitudes de la API con el punto de conexión de VPC mediante elasticache.cn-north-1.amazonaws.com.cn para Pekín y elasticache.cn-northwest-1.amazonaws.com.cn para Ningxia. Para obtener más información, consulte [Acceso a un servicio a través de un punto de conexión de interfaz](#) en la Guía del usuario de Amazon VPC.

Creación de una política de punto de conexión de VPC para la API de Amazon ElastiCache

Puede asociar una política de punto de conexión con el punto de conexión de VPC que controla el acceso a la API de ElastiCache. La política especifica lo siguiente:

- La entidad de seguridad que puede realizar acciones.
- Las acciones que se pueden realizar.
- Los recursos en los que se pueden llevar a cabo las acciones.

Para obtener más información, consulte [Control del acceso a los servicios con puntos de enlace de la VPC](#) en la guía del usuario de Amazon VPC.

Example Política de punto de conexión de VPC para acciones de la API de ElastiCache

A continuación, se muestra un ejemplo de una política de punto de conexión para la API de ElastiCache. Cuando se asocia a un punto de conexión, esta política concede acceso a las acciones de la API de ElastiCache enumeradas para todos las entidades principales de todos los recursos.

```
{
  "Statement": [{
    "Principal": "*",
    "Effect": "Allow",
    "Action": [
      "elasticache:CreateCacheCluster",
      "elasticache:ModifyCacheCluster",
      "elasticache:CreateSnapshot"
    ],
    "Resource": "*"
  }]
}
```

Example Política de punto de conexión de VPC que deniega todo el acceso desde una cuenta de AWS especificada

La siguiente política de punto de conexión de VPC deniega a la cuenta de AWS **123456789012** todo el acceso a los recursos mediante el punto de conexión. La política permite todas las acciones de otras cuentas.

```
{
  "Statement": [{
    "Action": "*",
    "Effect": "Allow",
    "Resource": "*",
    "Principal": "*"
  },
  {
    "Action": "*",
    "Effect": "Deny",
    "Resource": "*",
    "Principal": {
      "AWS": [
        "123456789012"
      ]
    }
  }
]
}
```

Subredes y grupos de subredes

Un grupo de subredes es una colección de subredes (normalmente, privadas) que puede designar para los clústeres de autodiseño que se ejecutan en un entorno de nube privada virtual (VPC).

Si crea un clúster de autodiseño en una VPC de Amazon, debe usar un grupo de subredes. ElastiCache utiliza dicho grupo de subredes para elegir una subred y direcciones IP pertenecientes a dicha subred a fin de asociarlas a sus nodos.

ElastiCache proporciona un grupo de subredes IPv4 predeterminado o puede optar por crear uno nuevo. Para IPv6, debe crear un grupo de subred con un bloque de CIDR IPv6. Si elige dual stack (doble pila), debe seleccionar un tipo de IP de detección, IPv6 o IPv4.

ElastiCache Serverless no utiliza un recurso de grupo de subredes, sino que toma una lista de subredes directamente durante la creación.

En esta sección, se explica cómo crear y aprovechar las subredes y los grupos de subredes para administrar el acceso a los recursos de ElastiCache.

Para obtener más información sobre la utilización de grupos de subredes en entornos de Amazon VPC, consulte [Acceso al clúster o al grupo de reproducción](#).

Temas

- [Creación de un grupo de subredes](#)
- [Asignación de un grupo de subredes a una caché](#)
- [Modificación de un grupo de subredes](#)
- [Eliminación de un grupo de subredes](#)

Creación de un grupo de subredes

Un grupo de subredes de caché es una colección de subredes que quizá quiera designar para sus cachés en una VPC. Al lanzar una caché en una VPC, tiene que seleccionar un grupo de subredes de caché. A continuación, ElastiCache utilizará dicho grupo de subredes de caché a fin de asignar direcciones IP de esta subred para cada nodo de caché en la caché.

Cuando cree un nuevo grupo de subredes, tenga en cuenta el número de direcciones IP disponibles. Si la subred tiene pocas direcciones IP libres, es posible que el número de nodos que pueda agregar a un clúster sea limitado. Para solucionar este problema, puede asignar una o varias subredes a un grupo de subredes para, de este modo, disponer de suficientes direcciones IP en la zona de disponibilidad de su clúster. Hecho esto, podrá agregar más nodos a su clúster.

Si elige IPv4 como tipo de red, estará disponible un grupo de subredes predeterminado o puede crear uno nuevo. ElastiCache utiliza dicho grupo de subredes para elegir una subred y direcciones IP pertenecientes a dicha subred a fin de asociarlas a sus nodos. Si elige IPv6 o doble pila, se le indicará que cree subredes IPv6 o de doble pila. Para obtener más información sobre los tipos de red, consulte [Tipo de red](#). Para obtener más información, consulte [Create a subnet in your VPC](#) (Crear una subred en la VPC).

En los siguientes procedimientos, se muestra cómo crear un grupo de subredes denominado mysubnetgroup (consola), la AWS CLI y la API de ElastiCache.

Creación de un grupo de subredes (consola)

En el siguiente procedimiento, se muestra cómo crear un grupo de subredes (consola).

Para crear un grupo de subredes (consola)

1. Inicie sesión en la AWS Management Console y abra la consola de ElastiCache en <https://console.aws.amazon.com/elasticache/>.
2. En la lista de navegación, elija Grupos de subredes.
3. Elija Create subnet group (Crear grupo de subredes).
4. En el asistente Crear grupo de subredes, haga lo siguiente. Cuando esté conforme con todos los ajustes, elija Crear.
 - a. En el cuadro Name (Nombre), escriba un nombre para el grupo de subredes.
 - b. En el cuadro Description(Descripción), escriba la descripción del grupo de subredes.
 - c. En el cuadro ID de VPC, elija su Amazon VPC.

- d. Todas las subredes están seleccionadas de forma predeterminada. En el panel Subredes seleccionadas, haga clic en Administrar y seleccione las zonas de disponibilidad o las [Zonas Locales](#) y los ID de sus subredes privadas; luego, seleccione Elegir.
5. En el mensaje de confirmación que aparece, elija Close (Cerrar).

El grupo de subredes nuevo aparecerá en la lista Subnet Groups (Grupos de subredes) de la consola de ElastiCache. En la parte inferior de la ventana, podrá elegir el grupo de subredes para ver detalles tales como todas las subredes asociadas al grupo.

Creación de un grupo de subredes (AWS CLI)

En el símbolo del sistema, utilice el comando `create-cache-subnet-group` para crear un grupo de subredes.

Para Linux, macOS o Unix:

```
aws elasticache create-cache-subnet-group \  
  --cache-subnet-group-name mysubnetgroup \  
  --cache-subnet-group-description "Testing" \  
  --subnet-ids subnet-53df9c3a
```

Para Windows:

```
aws elasticache create-cache-subnet-group ^  
  --cache-subnet-group-name mysubnetgroup ^  
  --cache-subnet-group-description "Testing" ^  
  --subnet-ids subnet-53df9c3a
```

Este comando debería producir un resultado similar al siguiente:

```
{  
  "CacheSubnetGroup": {  
    "VpcId": "vpc-37c3cd17",  
    "CacheSubnetGroupDescription": "Testing",  
    "Subnets": [  
      {  
        "SubnetIdentifier": "subnet-53df9c3a",  
        "SubnetAvailabilityZone": {  
          "Name": "us-west-2a"  
        }  
      }  
    ]  
  }  
}
```

```
    }  
  ],  
  "CacheSubnetGroupName": "mysubnetgroup"  
}  
}
```

Para obtener más información, consulte el tema de la AWS CLI [create-cache-subnet-group](#).

Asignación de un grupo de subredes a una caché

Una vez que ha creado un grupo de subredes, puede lanzar una caché en una Amazon VPC. Para obtener más información, consulte lo siguiente.

- Clúster OSS de Redis independiente: para lanzar un clúster de OSS de Redis de un solo nodo, consulte. [Creación de un clúster de Redis OSS \(modo de clúster desactivado\) \(consola\)](#) En el paso 7.a (Configuración avanzada de Redis OSS), elija un grupo de subredes de VPC.
- Grupo de replicación de Redis OSS (modo de clúster desactivado): para lanzar un grupo de replicación de Redis OSS (modo de clúster desactivado) en una VPC, consulte. [Creación de un grupo de replicación de Redis OSS \(modo de clúster desactivado\) desde cero](#) En el paso 7.b (Configuración avanzada de Redis OSS), elija un grupo de subredes de VPC.
- Grupo de replicación Redis OSS (modo de clúster activado) —. [Creación de un clúster de Redis OSS \(modo de clúster activado\) \(consola\)](#) En el paso 6.i (Configuración avanzada de Redis OSS), elija un grupo de subredes de VPC.

Modificación de un grupo de subredes

Puede modificar la descripción de un grupo de subredes o modificar la lista de los ID de subred asociados al grupo de subredes. No puede eliminar un ID de subred desde un grupo de subredes si hay una caché que utiliza actualmente dicha subred.

Los procedimientos que se describen a continuación muestran cómo modificar un grupo de subredes.

Modificación de grupos de subredes (consola)

Para modificar un grupo de subredes

1. Inicie sesión en la AWS Management Console y abra la consola de ElastiCache en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación, elija Subnet groups (grupos de subredes).
3. En la lista de grupos de subredes, seleccione el botón de opción del grupo que desee modificar y seleccione Modificar.
4. En el panel Subredes seleccionadas, elija Administrar.
5. Haga los cambios que desee en las subredes seleccionadas y haga clic en Elegir.
6. Haga clic en Guardar cambios para guardar los cambios.

Modificación de grupos de subredes (AWS CLI)

En el símbolo del sistema, use el comando `modify-cache-subnet-group` para modificar un grupo de subredes.

Para Linux, macOS o Unix:

```
aws elasticache modify-cache-subnet-group \  
  --cache-subnet-group-name mysubnetgroup \  
  --cache-subnet-group-description "New description" \  
  --subnet-ids "subnet-42df9c3a" "subnet-48fc21a9"
```

Para Windows:

```
aws elasticache modify-cache-subnet-group ^  
  --cache-subnet-group-name mysubnetgroup ^  
  --cache-subnet-group-description "New description" ^
```

```
--subnet-ids "subnet-42df9c3a" "subnet-48fc21a9"
```

Este comando debería producir un resultado similar al siguiente:

```
{
  "CacheSubnetGroup": {
    "VpcId": "vpc-73cd3c17",
    "CacheSubnetGroupDescription": "New description",
    "Subnets": [
      {
        "SubnetIdentifier": "subnet-42dcf93a",
        "SubnetAvailabilityZone": {
          "Name": "us-west-2a"
        }
      },
      {
        "SubnetIdentifier": "subnet-48fc12a9",
        "SubnetAvailabilityZone": {
          "Name": "us-west-2a"
        }
      }
    ],
    "CacheSubnetGroupName": "mysubnetgroup"
  }
}
```

Para obtener más información, consulte el tema de la AWS CLI [modify-cache-subnet-group](#).

Eliminación de un grupo de subredes

Si decide que ya no necesita su grupo de subredes, puede eliminarlo. No puede eliminar un grupo de subredes si hay una caché que lo utiliza actualmente.

Los procedimientos que se describen a continuación muestran cómo eliminar un grupo de subredes.

Eliminación de un grupo de subredes (consola)

Para eliminar un grupo de subredes

1. Inicie sesión en la AWS Management Console y abra la consola de ElastiCache en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación, elija Subnet groups (grupos de subredes).
3. En la lista de grupos de subredes, elija el grupo que desea eliminar y, a continuación, elija Delete(Eliminar).
4. Cuando se le pida que confirme esta operación, escriba el nombre del grupo de subredes en el campo de entrada de texto y elija Eliminar.

Eliminación de un grupo de subredes (AWS CLI)

Mediante la AWS CLI, llame al comando `delete-cache-subnet-group` con el siguiente parámetro:

- `--cache-subnet-group-name mysubnetgroup`

Para Linux, macOS o Unix:

```
aws elasticache delete-cache-subnet-group \  
  --cache-subnet-group-name mysubnetgroup
```

Para Windows:

```
aws elasticache delete-cache-subnet-group ^  
  --cache-subnet-group-name mysubnetgroup
```

Este comando no genera ninguna salida.

Para obtener más información, consulte el tema de la AWS CLI [delete-cache-subnet-group](#).

Identity and Access Management para Amazon ElastiCache

AWS Identity and Access Management (IAM) es un Servicio de AWS que ayuda al administrador a controlar de forma segura el acceso a AWS los recursos. IAM los administradores controlan quién puede autenticarse (iniciar sesión) y quién puede autorizarse (tener permisos) para usar ElastiCache los recursos. IAM es un Servicio de AWS que puede utilizar sin coste adicional.

Temas

- [Público](#)
- [Autenticación con identidades](#)
- [Administración de acceso mediante políticas](#)
- [Cómo ElastiCache trabaja Amazon con IAM](#)
- [Ejemplos de políticas basadas en identidades de Amazon ElastiCache](#)
- [Solución de problemas de ElastiCache identidad y acceso a Amazon](#)
- [Control de acceso](#)
- [Información general sobre la administración de los permisos de acceso a los recursos de ElastiCache](#)

Público

La forma de usar AWS Identity and Access Management (IAM) varía según el trabajo en el que se realice ElastiCache.

Usuario del servicio: si utiliza el ElastiCache servicio para realizar su trabajo, el administrador le proporcionará las credenciales y los permisos que necesita. A medida que vaya utilizando más ElastiCache funciones para realizar su trabajo, es posible que necesite permisos adicionales. Entender cómo se administra el acceso puede ayudarlo a solicitar los permisos correctos al administrador. Si no puede acceder a una función en ElastiCache, consulte [Solución de problemas de ElastiCache identidad y acceso a Amazon](#).

Administrador de servicios: si está a cargo de ElastiCache los recursos de su empresa, probablemente tenga acceso total a ellos ElastiCache. Su trabajo consiste en determinar a qué ElastiCache funciones y recursos deben acceder los usuarios del servicio. A continuación, debe

enviar solicitudes a su IAM administrador para cambiar los permisos de los usuarios del servicio. Revise la información de esta página para comprender los conceptos básicos de IAM. Para obtener más información sobre cómo su empresa puede utilizar IAM con ElastiCache, consulte [Cómo ElastiCache trabaja Amazon con IAM](#).

IAM administrador: si es IAM administrador, puede que desee obtener más información sobre cómo puede redactar políticas para administrar el acceso a ellas ElastiCache. Para ver ejemplos de políticas ElastiCache basadas en la identidad que puede utilizar IAM, consulte. [Ejemplos de políticas basadas en identidades de Amazon ElastiCache](#)

Autenticación con identidades

La autenticación es la forma de iniciar sesión AWS con sus credenciales de identidad. Debe estar autenticado (con quien haya iniciado sesión AWS) como IAM usuario o asumiendo un IAM rol.

Usuario raíz de la cuenta de AWS

Puede iniciar sesión AWS como una identidad federada mediante las credenciales proporcionadas a través de una fuente de identidad. AWS IAM Identity Center Los usuarios (IAM Identity Center), la autenticación de inicio de sesión único de su empresa y sus credenciales de Google o Facebook son ejemplos de identidades federadas. Al iniciar sesión como una identidad federada, el administrador configuró previamente la federación de identidades mediante roles. IAM Cuando accede AWS mediante la federación, asume indirectamente un rol.

Según el tipo de usuario que sea, puede iniciar sesión en el portal AWS Management Console o en el de AWS acceso. Para obtener más información sobre cómo iniciar sesión AWS, consulte [Cómo iniciar sesión Cuenta de AWS en su](#) Guía del AWS Sign-In usuario.

Si accede AWS mediante programación, AWS incluye un kit de desarrollo de software (SDK) y una interfaz de línea de comandos (CLI) para firmar criptográficamente sus solicitudes con sus credenciales. Si no utilizas AWS herramientas, debes firmar las solicitudes tú mismo. Para obtener más información sobre cómo usar el método recomendado para firmar las solicitudes usted mismo, consulte [Firmar AWS API las solicitudes](#) en la Guía del IAM usuario.

Independientemente del método de autenticación que use, es posible que deba proporcionar información de seguridad adicional. Por ejemplo, le AWS recomienda que utilice la autenticación multifactorial (MFA) para aumentar la seguridad de su cuenta. Para obtener más información, consulte [Autenticación multifactorial](#) en la Guía del AWS IAM Identity Center usuario y [Uso de la autenticación multifactorial \(MFA\) AWS en](#) la Guía del IAM usuario.

Cuenta de AWS usuario root

Al crear una Cuenta de AWS, comienza con una identidad de inicio de sesión que tiene acceso completo a todos Servicios de AWS los recursos de la cuenta. Esta identidad se denomina usuario Cuenta de AWS raíz y se accede a ella iniciando sesión con la dirección de correo electrónico y la contraseña que utilizaste para crear la cuenta. Recomendamos encarecidamente que no utilice el usuario raíz para sus tareas diarias. Proteja las credenciales del usuario raíz y utilícelas solo para las tareas que solo el usuario raíz pueda realizar. Para ver la lista completa de tareas que requieren que inicie sesión como usuario root, consulte [Tareas que requieren credenciales de usuario root](#) en la Guía del IAM usuario.

Identidad federada

Como práctica recomendada, exija a los usuarios humanos, incluidos los que requieren acceso de administrador, que utilicen la federación con un proveedor de identidades para acceder Servicios de AWS mediante credenciales temporales.

Una identidad federada es un usuario del directorio de usuarios de su empresa, un proveedor de identidades web AWS Directory Service, el directorio del Centro de Identidad o cualquier usuario al que acceda Servicios de AWS mediante las credenciales proporcionadas a través de una fuente de identidad. Cuando las identidades federadas acceden Cuentas de AWS, asumen funciones y las funciones proporcionan credenciales temporales.

Para una administración de acceso centralizada, le recomendamos que utilice AWS IAM Identity Center. Puede crear usuarios y grupos en IAM Identity Center, o puede conectarse y sincronizarse con un conjunto de usuarios y grupos de su propia fuente de identidad para usarlos en todas sus aplicaciones Cuentas de AWS . Para obtener información sobre IAM Identity Center, consulte [¿Qué es IAM Identity Center?](#) en la Guía AWS IAM Identity Center del usuario.

Usuarios y grupos de IAM

Un [IAMusuario](#) es una identidad dentro de tu Cuenta de AWS que tiene permisos específicos para una sola persona o aplicación. Siempre que sea posible, recomendamos utilizar credenciales temporales en lugar de crear IAM usuarios con credenciales de larga duración, como contraseñas y claves de acceso. Sin embargo, si tiene casos de uso específicos que requieren credenciales a largo plazo con IAM los usuarios, le recomendamos que rote las claves de acceso. Para obtener más información, consulte [Rotar las claves de acceso con regularidad para los casos de uso que requieran credenciales de larga duración](#) en la Guía del IAM usuario.

Un [IAMgrupo](#) es una identidad que especifica un conjunto de IAM usuarios. No puede iniciar sesión como grupo. Puede usar los grupos para especificar permisos para varios usuarios a la vez. Los grupos facilitan la administración de los permisos para grandes conjuntos de usuarios. Por ejemplo, puede asignar un nombre a un grupo IAMAdminsy concederle permisos para administrar IAM los recursos.

Los usuarios son diferentes de los roles. Un usuario se asocia exclusivamente a una persona o aplicación, pero la intención es que cualquier usuario pueda asumir un rol que necesite. Los usuarios tienen credenciales de larga duración permanentes; no obstante, los roles proporcionan credenciales temporales. Para obtener más información, consulte [Cuándo crear un IAM usuario \(en lugar de un rol\)](#) en la Guía del IAM usuario.

IAMroles

Un [IAMrol](#) es una identidad dentro de tu Cuenta de AWS que tiene permisos específicos. Es similar a un IAM usuario, pero no está asociado a una persona específica. Puede asumir temporalmente un IAM rol en el AWS Management Console [cambiando de rol](#). Puede asumir un rol llamando a una AWS API operación AWS CLI o utilizando una operación personalizadaURL. Para obtener más información sobre los métodos de uso de roles, consulte [Uso de IAM roles](#) en la Guía del IAM usuario.

IAMlos roles con credenciales temporales son útiles en las siguientes situaciones:

- **Acceso de usuario federado:** para asignar permisos a una identidad federada, puede crear un rol y definir sus permisos. Cuando se autentica una identidad federada, se asocia la identidad al rol y se le conceden los permisos define el rol. Para obtener información sobre los roles para la federación, consulte [Creación de un rol para un proveedor de identidad externo](#) en la Guía del IAM usuario. Si usa IAM Identity Center, configura un conjunto de permisos. Para controlar a qué pueden acceder sus identidades después de autenticarse, IAM Identity Center correlaciona el conjunto de permisos con un rol en IAM. Para obtener información acerca de los conjuntos de permisos, consulte [Conjuntos de permisos](#) en la Guía del usuario de AWS IAM Identity Center .
- **Permisos IAM de usuario temporales:** un IAM usuario o rol puede asumir un IAM rol para asumir temporalmente diferentes permisos para una tarea específica.
- **Acceso multicuenta:** puedes usar un IAM rol para permitir que alguien (un responsable de confianza) de una cuenta diferente acceda a los recursos de tu cuenta. Los roles son la forma principal de conceder acceso entre cuentas. Sin embargo, con algunos Servicios de AWS, puedes adjuntar una política directamente a un recurso (en lugar de usar un rol como proxy). Para conocer

la diferencia entre las funciones y las políticas basadas en recursos para el acceso multicuenta, consulta el tema sobre el acceso a los [recursos entre cuentas IAM en](#) la Guía del IAM usuario.

- Acceso entre servicios: algunos Servicios de AWS utilizan funciones en otros. Servicios de AWS Por ejemplo, cuando realizas una llamada en un servicio, es habitual que ese servicio ejecute aplicaciones en Amazon EC2 o almacene objetos en Amazon S3. Es posible que un servicio haga esto usando los permisos de la entidad principal, usando un rol de servicio o usando un rol vinculado al servicio.
- Sesiones de acceso directo (FAS): cuando utilizas un IAM usuario o un rol para realizar acciones en AWS ellas, se te considera director. Cuando utiliza algunos servicios, es posible que realice una acción que desencadene otra acción en un servicio diferente. FAS utiliza los permisos del principal que llama a un Servicio de AWS, junto con los que solicitan, Servicio de AWS para realizar solicitudes a los servicios descendentes. FAS las solicitudes solo se realizan cuando un servicio recibe una solicitud que requiere interacciones con otros Servicios de AWS recursos para completarse. En este caso, debe tener permisos para realizar ambas acciones. Para obtener información detallada sobre la política a la hora de realizar FAS solicitudes, consulte [Reenviar las sesiones de acceso](#).
- Función de servicio: una función de servicio es una [IAM función](#) que un servicio asume para realizar acciones en su nombre. Un IAM administrador puede crear, modificar y eliminar un rol de servicio desde dentro IAM. Para obtener más información, consulte [Crear un rol para delegar permisos Servicio de AWS en un rol](#) en el IAM Manual del usuario.
- Función vinculada a un servicio: una función vinculada a un servicio es un tipo de función de servicio que está vinculada a un. Servicio de AWS El servicio puede asumir el rol para realizar una acción en su nombre. Los roles vinculados al servicio aparecen en usted Cuenta de AWS y son propiedad del servicio. Un IAM administrador puede ver los permisos de los roles vinculados al servicio, pero no editarlos.
- Aplicaciones que se ejecutan en Amazon EC2: puedes usar un IAM rol para administrar las credenciales temporales de las aplicaciones que se ejecutan en una EC2 instancia y que realizan AWS CLI o AWS API solicitan. Esto es preferible a almacenar las claves de acceso dentro de la EC2 instancia. Para asignar un AWS rol a una EC2 instancia y ponerlo a disposición de todas sus aplicaciones, debe crear un perfil de instancia adjunto a la instancia. Un perfil de instancia contiene el rol y permite que los programas que se ejecutan en la EC2 instancia obtengan credenciales temporales. Para obtener más información, consulte [Uso de un IAM rol para conceder permisos a aplicaciones que se ejecutan en EC2 instancias de Amazon](#) en la Guía del IAM usuario.

Para saber si se deben usar IAM roles o IAM usuarios, consulte [Cuándo crear un IAM rol \(en lugar de un usuario\)](#) en la Guía del IAM usuario.

Administración de acceso mediante políticas

El acceso se controla AWS creando políticas y adjuntándolas a AWS identidades o recursos. Una política es un objeto AWS que, cuando se asocia a una identidad o un recurso, define sus permisos. AWS evalúa estas políticas cuando un director (usuario, usuario raíz o sesión de rol) realiza una solicitud. Los permisos en las políticas determinan si la solicitud se permite o se deniega. La mayoría de las políticas se almacenan AWS como JSON documentos. Para obtener más información sobre la estructura y el contenido de los documentos de JSON políticas, consulte [Descripción general de JSON las políticas](#) en la Guía del IAM usuario.

Los administradores pueden usar AWS JSON las políticas para especificar quién tiene acceso a qué. Es decir, qué entidad principal puede realizar acciones en qué recursos y en qué condiciones.

De forma predeterminada, los usuarios y los roles no tienen permisos. Para conceder a los usuarios permiso para realizar acciones en los recursos que necesitan, un IAM administrador puede crear IAM políticas. A continuación, el administrador puede añadir las IAM políticas a las funciones y los usuarios pueden asumir las funciones.

IAM las políticas definen los permisos para una acción independientemente del método que se utilice para realizar la operación. Por ejemplo, suponga que dispone de una política que permite la acción `iam:GetRole`. Un usuario con esa política puede obtener información sobre el rol de AWS Management Console AWS CLI, el o el AWS API.

Políticas basadas en identidad

Las políticas basadas en la identidad son documentos de política de JSON permisos que se pueden adjuntar a una identidad, como un IAM usuario, un grupo de usuarios o un rol. Estas políticas controlan qué acciones pueden realizar los usuarios y los roles, en qué recursos y en qué condiciones. Para obtener información sobre cómo crear una política basada en la identidad, consulte [Creación de IAM políticas](#) en la Guía del usuario. IAM

Las políticas basadas en identidades pueden clasificarse además como políticas insertadas o políticas administradas. Las políticas insertadas se integran directamente en un único usuario, grupo o rol. Las políticas administradas son políticas independientes que puede adjuntar a varios usuarios, grupos y funciones de su empresa. Cuenta de AWS Las políticas administradas incluyen políticas AWS administradas y políticas administradas por el cliente. Para saber cómo elegir entre una política

gestionada o una política integrada, consulte [Elegir entre políticas gestionadas y políticas integradas en la Guía del IAM](#) usuario.

Políticas basadas en recursos

Las políticas basadas en recursos son documentos de JSON política que se adjuntan a un recurso. Algunos ejemplos de políticas basadas en recursos son las políticas de confianza de IAM roles y las políticas de bucket de Amazon S3. En los servicios que admiten políticas basadas en recursos, los administradores de servicios pueden utilizarlos para controlar el acceso a un recurso específico. Para el recurso al que se asocia la política, la política define qué acciones puede realizar una entidad principal especificada en ese recurso y en qué condiciones. Debe [especificar una entidad principal](#) en una política en función de recursos. Los principales pueden incluir cuentas, usuarios, roles, usuarios federados o. Servicios de AWS

Las políticas basadas en recursos son políticas insertadas que se encuentran en ese servicio. No puede usar políticas AWS administradas desde una política IAM basada en recursos.

Listas de control de acceso () ACLs

Las listas de control de acceso (ACLs) controlan qué responsables (miembros de la cuenta, usuarios o roles) tienen permisos para acceder a un recurso. ACLs son similares a las políticas basadas en recursos, aunque no utilizan el formato de documento de JSON políticas.

Amazon S3 AWS WAF y Amazon VPC son ejemplos de servicios compatibles ACLs. Para obtener más información ACLs, consulte la [descripción general de la lista de control de acceso \(ACL\)](#) en la Guía para desarrolladores de Amazon Simple Storage Service.

Otros tipos de políticas

AWS admite tipos de políticas adicionales y menos comunes. Estos tipos de políticas pueden establecer el máximo de permisos que los tipos de políticas más frecuentes le conceden.

- **Límites de permisos:** un límite de permisos es una función avanzada en la que se establecen los permisos máximos que una política basada en la identidad puede conceder a una IAM entidad (IAM usuario o rol). Puede establecer un límite de permisos para una entidad. Los permisos resultantes son la intersección de las políticas basadas en la identidad de la entidad y los límites de permisos. Las políticas basadas en recursos que especifiquen el usuario o rol en el campo `Principal` no estarán restringidas por el límite de permisos. Una denegación explícita en cualquiera de estas políticas anulará el permiso. Para obtener más información sobre los límites

de los permisos, consulte los [límites de los permisos para IAM las entidades](#) en la Guía del IAM usuario.

- **Políticas de control de servicios (SCPs):** SCPs son JSON políticas que especifican los permisos máximos para una organización o unidad organizativa (OU) AWS Organizations. AWS Organizations es un servicio para agrupar y administrar de forma centralizada varios de los Cuentas de AWS que son propiedad de su empresa. Si habilitas todas las funciones de una organización, puedes aplicar políticas de control de servicios (SCPs) a una o a todas tus cuentas. SCP limita los permisos de las entidades en las cuentas de los miembros, incluidas las de cada una Usuario raíz de la cuenta de AWS. Para obtener más información sobre Organizations SCPs, consulte las [políticas de control de servicios](#) en la Guía del AWS Organizations usuario.
- **Políticas de sesión:** las políticas de sesión son políticas avanzadas que se pasan como parámetro cuando se crea una sesión temporal mediante programación para un rol o un usuario federado. Los permisos de la sesión resultantes son la intersección de las políticas basadas en identidades del rol y las políticas de la sesión. Los permisos también pueden proceder de una política en función de recursos. Una denegación explícita en cualquiera de estas políticas anulará el permiso. Para obtener más información, consulte [las políticas de sesión](#) en la Guía del IAM usuario.

Varios tipos de políticas

Cuando se aplican varios tipos de políticas a una solicitud, los permisos resultantes son más complicados de entender. Para saber cómo se AWS determina si se debe permitir una solicitud cuando se trata de varios tipos de políticas, consulte la [lógica de evaluación de políticas](#) en la Guía del IAM usuario.

Cómo ElastiCache trabaja Amazon con IAM

Antes de gestionar el IAM acceso a ElastiCache, infórmate sobre IAM las funciones disponibles para su uso ElastiCache.

IAM funciones que puedes usar con Amazon ElastiCache

IAM característica	ElastiCache apoyo
Políticas basadas en identidades	Sí
Políticas basadas en recursos	No

IAM característica	ElastiCache apoyo
Acciones de políticas	Sí
Recursos de políticas	Sí
Claves de condición de política	Sí
ACLs	Sí
ABAC(etiquetas en las políticas)	Sí
Credenciales temporales	Sí
Permisos de entidades principales	Sí
Roles de servicio	Sí
Roles vinculados al servicio	Sí

Para obtener una visión general de cómo ElastiCache funcionan otros AWS servicios con la mayoría de las IAM funciones, consulte [AWS los servicios con los que funcionan IAM](#) en la Guía del IAM usuario.

Políticas basadas en la identidad para ElastiCache

Compatibilidad con las políticas basadas en identidad: sí

Las políticas basadas en la identidad son documentos de política de JSON permisos que se pueden adjuntar a una identidad, como un IAM usuario, un grupo de usuarios o un rol. Estas políticas controlan qué acciones pueden realizar los usuarios y los roles, en qué recursos y en qué condiciones. Para obtener información sobre cómo crear una política basada en la identidad, consulte [Creación de IAM políticas](#) en la Guía del usuario. IAM

Con las políticas IAM basadas en la identidad, puede especificar las acciones y los recursos permitidos o denegados, así como las condiciones en las que se permiten o deniegan las acciones. No es posible especificar la entidad principal en una política basada en identidad porque se aplica al usuario o rol al que está adjunto. Para obtener más información sobre todos los elementos que puede utilizar en una JSON política, consulte la [referencia sobre los elementos de la IAM JSON política](#) en la Guía del IAM usuario.

Ejemplos de políticas basadas en identidades de ElastiCache

Para ver ejemplos de políticas ElastiCache basadas en la identidad, consulte. [Ejemplos de políticas basadas en identidades de Amazon ElastiCache](#)

Políticas basadas en recursos de ElastiCache

Admite políticas basadas en recursos: no

Las políticas basadas en recursos son JSON documentos de políticas que se adjuntan a un recurso. Algunos ejemplos de políticas basadas en recursos son las políticas de confianza de IAM roles y las políticas de bucket de Amazon S3. En los servicios que admiten políticas basadas en recursos, los administradores de servicios pueden utilizarlos para controlar el acceso a un recurso específico. Para el recurso al que se asocia la política, la política define qué acciones puede realizar una entidad principal especificada en ese recurso y en qué condiciones. Debe [especificar una entidad principal](#) en una política en función de recursos. Los principales pueden incluir cuentas, usuarios, roles, usuarios federados o. Servicios de AWS

Para habilitar el acceso entre cuentas, puede especificar una cuenta completa o IAM entidades de otra cuenta como principales en una política basada en recursos. Añadir a una política en función de recursos una entidad principal entre cuentas es solo una parte del establecimiento de una relación de confianza. Cuando el principal y el recurso son diferentes Cuentas de AWS, el IAM administrador de la cuenta de confianza también debe conceder permiso a la entidad principal (usuario o rol) para acceder al recurso. Para conceder el permiso, adjunte la entidad a una política basada en identidad. Sin embargo, si la política en función de recursos concede el acceso a una entidad principal de la misma cuenta, no es necesaria una política basada en identidad adicional. Para obtener más información, consulte el [tema Acceso a recursos entre cuentas IAM en](#) la Guía del IAM usuario.

Acciones políticas para ElastiCache

Compatibilidad con las acciones de política: sí

Los administradores pueden usar AWS JSON políticas para especificar quién tiene acceso a qué. Es decir, qué entidad principal puede realizar acciones en qué recursos y en qué condiciones.

El `Action` elemento de una JSON política describe las acciones que puede utilizar para permitir o denegar el acceso en una política. Las acciones de política suelen tener el mismo nombre que la AWS API operación asociada. Hay algunas excepciones, como las acciones que solo permiten permisos y que no tienen una operación coincidente. API También hay algunas operaciones que

requieren varias acciones en una política. Estas acciones adicionales se denominan acciones dependientes.

Incluya acciones en una política para conceder permisos y así llevar a cabo la operación asociada.

Para ver una lista de ElastiCache acciones, consulta [Acciones definidas por Amazon ElastiCache](#) en la Referencia de autorización de servicio.

Las acciones políticas ElastiCache utilizan el siguiente prefijo antes de la acción:

```
elasticache
```

Para especificar varias acciones en una única instrucción, sepárelas con comas.

```
"Action": [  
  "elasticache:action1",  
  "elasticache:action2"  
]
```

Puede utilizar caracteres comodín (*) para especificar varias acciones . Por ejemplo, para especificar todas las acciones que comiencen con la palabra Describe, incluya la siguiente acción:

```
"Action": "elasticache:Describe*"
```

Para ver ejemplos de políticas ElastiCache basadas en la identidad, consulte. [Ejemplos de políticas basadas en identidades de Amazon ElastiCache](#)

Recursos de políticas para ElastiCache

Compatibilidad con los recursos de políticas: sí

Los administradores pueden usar AWS JSON políticas para especificar quién tiene acceso a qué. Es decir, qué entidad principal puede realizar acciones en qué recursos y en qué condiciones.

El elemento Resource JSON de política especifica el objeto o los objetos a los que se aplica la acción. Las instrucciones deben contener un elemento Resource o NotResource. Como práctica recomendada, especifique un recurso mediante su [nombre de recurso de Amazon \(ARN\)](#). Puede

hacerlo para acciones que admitan un tipo de recurso específico, conocido como permisos de nivel de recurso.

Para las acciones que no admiten permisos de nivel de recurso, como las operaciones de descripción, utilice un carácter comodín (*) para indicar que la instrucción se aplica a todos los recursos.

```
"Resource": "*"
```

Para ver una lista de los tipos de ElastiCache recursos y sus respectivos tiposARNs, consulta [Recursos definidos por Amazon ElastiCache](#) en la Referencia de autorización de servicio. Para saber con qué acciones puede especificar cada recurso, consulte [Acciones definidas por Amazon ElastiCache](#). ARN

Para ver ejemplos de políticas ElastiCache basadas en la identidad, consulte. [Ejemplos de políticas basadas en identidades de Amazon ElastiCache](#)

Claves de condición de políticas para ElastiCache

Compatibilidad con claves de condición de políticas específicas del servicio: sí

Los administradores pueden usar AWS JSON políticas para especificar quién tiene acceso a qué. Es decir, qué entidad principal puede realizar acciones en qué recursos y en qué condiciones.

El elemento `Condition` (o bloque de `Condition`) permite especificar condiciones en las que entra en vigor una instrucción. El elemento `Condition` es opcional. Puede crear expresiones condicionales que utilicen [operadores de condición](#), tales como igual o menor que, para que la condición de la política coincida con los valores de la solicitud.

Si especifica varios elementos de `Condition` en una instrucción o varias claves en un único elemento de `Condition`, AWS las evalúa mediante una operación AND lógica. Si especifica varios valores para una única clave de condición, AWS evalúa la condición mediante una OR operación lógica. Se deben cumplir todas las condiciones antes de que se concedan los permisos de la instrucción.

También puede utilizar variables de marcador de posición al especificar condiciones. Por ejemplo, puede conceder a un IAM usuario permiso para acceder a un recurso solo si está etiquetado con su nombre de IAM usuario. Para obtener más información, consulte [los elementos IAM de la política: variables y etiquetas](#) en la Guía del IAM usuario.

AWS admite claves de condición globales y claves de condición específicas del servicio. Para ver todas las claves de condición AWS globales, consulte las claves de [contexto de condición AWS globales](#) en la Guía del IAMusuario.

Para ver una lista de claves de ElastiCache estado, consulta [Claves de estado de Amazon ElastiCache](#) en la Referencia de autorización de servicio. Para saber con qué acciones y recursos puede utilizar una clave de condición, consulte [Acciones definidas por Amazon ElastiCache](#).

Para ver ejemplos de políticas ElastiCache basadas en la identidad, consulte. [Ejemplos de políticas basadas en identidades de Amazon ElastiCache](#)

Listas de control de acceso () ACLs en ElastiCache

SoportaACLs: Sí

Las listas de control de acceso (ACLs) controlan qué directores (miembros de la cuenta, usuarios o roles) tienen permisos para acceder a un recurso. ACLs son similares a las políticas basadas en recursos, aunque no utilizan el formato de documento de JSON políticas.

Control de acceso basado en atributos () con ABAC ElastiCache

Soportes ABAC (etiquetas en las políticas): Sí

El control de acceso basado en atributos (ABAC) es una estrategia de autorización que define los permisos en función de los atributos. En AWS, estos atributos se denominan etiquetas. Puede adjuntar etiquetas a IAM entidades (usuarios o roles) y a muchos AWS recursos. Etiquetar entidades y recursos es el primer paso de ABAC. Luego, diseñe ABAC políticas para permitir las operaciones cuando la etiqueta del principal coincida con la etiqueta del recurso al que está intentando acceder.

ABAC es útil en entornos de rápido crecimiento y ayuda en situaciones en las que la administración de políticas se vuelve engorrosa.

Para controlar el acceso en función de etiquetas, debe proporcionar información de las etiquetas en el [elemento de condición](#) de una política utilizando las claves de condición `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` o `aws:TagKeys`.

Si un servicio admite las tres claves de condición para cada tipo de recurso, el valor es Sí para el servicio. Si un servicio admite las tres claves de condición solo para algunos tipos de recursos, el valor es Parcial.

Para obtener más información al respecto ABAC, consulte [¿Qué es? ABAC](#) en la Guía IAM del usuario. Para ver un tutorial con los pasos de configuración ABAC, consulte [Usar el control de acceso basado en atributos \(ABAC\)](#) en la Guía del IAM usuario.

Uso de credenciales temporales con ElastiCache

Compatibilidad con credenciales temporales: sí

Algunos Servicios de AWS no funcionan cuando inicias sesión con credenciales temporales. Para obtener información adicional, incluida la información sobre cuáles Servicios de AWS funcionan con credenciales temporales, consulta Servicios de AWS la guía del IAM usuario sobre cómo [trabajar con IAM](#) ellas.

Está utilizando credenciales temporales si inicia sesión en ellas AWS Management Console mediante cualquier método excepto un nombre de usuario y una contraseña. Por ejemplo, cuando accedes AWS mediante el enlace de inicio de sesión único (SSO) de tu empresa, ese proceso crea automáticamente credenciales temporales. También crea credenciales temporales de forma automática cuando inicia sesión en la consola como usuario y luego cambia de rol. Para obtener más información sobre el cambio de rol, consulte [Cambiar a un rol \(consola\)](#) en la Guía del IAM usuario.

Puede crear credenciales temporales manualmente con la tecla AWS CLI o AWS API. A continuación, puede utilizar esas credenciales temporales para acceder AWS. AWS recomienda generar credenciales temporales de forma dinámica en lugar de utilizar claves de acceso a largo plazo. Para obtener más información, consulte [Credenciales de seguridad temporales en IAM](#).

Permisos de entidades principales entre servicios de ElastiCache

Admite sesiones de acceso directo (FAS): Sí

Cuando utilizas un IAM usuario o un rol para realizar acciones en AWS él, se te considera director. Cuando utiliza algunos servicios, es posible que realice una acción que desencadene otra acción en un servicio diferente. FAS utiliza los permisos del principal que llama a un Servicio de AWS, junto con los que solicita, Servicio de AWS para realizar solicitudes a los servicios descendentes. FAS las solicitudes solo se realizan cuando un servicio recibe una solicitud que requiere interacciones con otros recursos Servicios de AWS o para completarse. En este caso, debe tener permisos para realizar ambas acciones. Para obtener información detallada sobre la política a la hora de realizar FAS solicitudes, consulte [Reenviar las sesiones de acceso](#).

Funciones de servicio para ElastiCache

Compatibilidad con roles de servicio: sí

Una función de servicio es una [IAMfunción](#) que asume un servicio para realizar acciones en su nombre. Un IAM administrador puede crear, modificar y eliminar un rol de servicio desde dentro IAM. Para obtener más información, consulte [Crear un rol para delegar permisos Servicio de AWS en un rol](#) en el IAMManual del usuario.

Warning

Cambiar los permisos de un rol de servicio podría interrumpir ElastiCache la funcionalidad. Edite las funciones de servicio solo cuando se ElastiCache proporcionen instrucciones para hacerlo.

Funciones vinculadas al servicio para ElastiCache

Admite roles vinculados al servicio: sí

Un rol vinculado a un servicio es un tipo de rol de servicio que está vinculado a un. Servicio de AWS El servicio puede asumir el rol para realizar una acción en su nombre. Los roles vinculados al servicio aparecen en usted Cuenta de AWS y son propiedad del servicio. Un IAM administrador puede ver los permisos de los roles vinculados al servicio, pero no editarlos.

Para obtener más información sobre la creación o la administración de funciones vinculadas a un servicio, consulte los [AWS servicios](#) que funcionan con. IAM Busque un servicio en la tabla que incluya Yes en la columna Rol vinculado a un servicio. Seleccione el vínculo Sí para ver la documentación acerca del rol vinculado a servicios para ese servicio.

Ejemplos de políticas basadas en identidades de Amazon ElastiCache

De forma predeterminada, los usuarios y roles no tienen permiso para crear ni modificar los recursos de ElastiCache. Tampoco pueden realizar tareas mediante la AWS Management Console, la AWS Command Line Interface (AWS CLI) o la API de AWS. Para conceder permiso a los usuarios para realizar acciones en los recursos que necesiten, un administrador de IAM puede crear políticas de IAM. A continuación, el administrador puede añadir las políticas de IAM a roles, y los usuarios pueden asumirlos.

Para obtener información sobre cómo crear una política basada en identidad de IAM mediante el uso de estos documentos de políticas JSON de ejemplo, consulte [Creación de políticas de IAM](#) en la Guía del usuario de IAM.

Para obtener más información sobre las acciones y los tipos de recursos definidos por ElastiCache, incluido el formato de los ARN para cada tipo de recurso, consulte [Acciones, recursos y claves de condición para Amazon ElastiCache](#), en la Referencia de autorizaciones de servicio.

Temas

- [Prácticas recomendadas sobre las políticas](#)
- [Uso de la consola de ElastiCache](#)
- [Cómo permitir a los usuarios consultar sus propios permisos](#)

Prácticas recomendadas sobre las políticas

Las políticas basadas en identidades determinan si alguien puede crear, acceder o eliminar los recursos de ElastiCache de la cuenta. Estas acciones pueden generar costes adicionales para su Cuenta de AWS. Siga estas directrices y recomendaciones al crear o editar políticas basadas en identidades:

- Comience con las políticas administradas de AWS y continúe con los permisos de privilegio mínimo: a fin de comenzar a conceder permisos a los usuarios y las cargas de trabajo, utilice las políticas administradas de AWS, que conceden permisos para muchos casos de uso comunes. Están disponibles en su Cuenta de AWS. Se recomienda definir políticas administradas por el cliente de AWS específicas para los casos de uso a fin de reducir aún más los permisos. Con el fin de obtener más información, consulte las [políticas administradas por AWS](#) o las [políticas administradas por AWS para funciones de trabajo](#) en la Guía del usuario de IAM.
- Aplique permisos de privilegio mínimo: cuando establezca permisos con políticas de IAM, conceda solo los permisos necesarios para realizar una tarea. Para ello, debe definir las acciones que se pueden llevar a cabo en determinados recursos en condiciones específicas, también conocidos como permisos de privilegios mínimos. Con el fin de obtener más información sobre el uso de IAM para aplicar permisos, consulte [Políticas y permisos en IAM](#) en la Guía de usuario de IAM.
- Utilice condiciones en las políticas de IAM para restringir aún más el acceso: puede agregar una condición a sus políticas para limitar el acceso a las acciones y los recursos. Por ejemplo, puede escribir una condición de políticas para especificar que todas las solicitudes deben enviarse utilizando SSL. También puede usar condiciones para conceder acceso a acciones de servicios si se emplean a través de un Servicio de AWS determinado, como por ejemplo AWS CloudFormation. Para obtener más información, consulte [Elementos de la política de JSON de IAM: Condición](#) en la Guía del usuario de IAM.

- Utilice el analizador de acceso de IAM para validar las políticas de IAM con el fin de garantizar la seguridad y funcionalidad de los permisos: el analizador de acceso de IAM valida políticas nuevas y existentes para que respeten el lenguaje (JSON) de las políticas de IAM y las prácticas recomendadas de IAM. El analizador de acceso de IAM proporciona más de 100 verificaciones de políticas y recomendaciones procesables para ayudar a crear políticas seguras y funcionales. Para más información, consulte la [Política de validación del analizador de acceso de IAM](#) en la Guía de usuario de IAM.
- Solicite la autenticación multifactor (MFA): si se encuentra en una situación en la que necesita usuarios raíz o de IAM en su Cuenta de AWS, active la MFA para mayor seguridad. Para solicitar la MFA cuando se invocan las operaciones de la API, agregue las condiciones de la MFA a sus políticas. Para obtener más información, consulte [Configuración de acceso a una API protegida por MFA](#) en la Guía del usuario de IAM.

Para obtener más información sobre las prácticas recomendadas de IAM, consulte las [Prácticas recomendadas de seguridad en IAM](#) en la Guía del usuario de IAM.

Uso de la consola de ElastiCache

Para acceder a la consola de Amazon ElastiCache, debe tener un conjunto mínimo de permisos. Estos permisos deben permitirle mostrar y consultar los detalles acerca de los recursos de ElastiCache en la Cuenta de AWS. Si crea una política basada en identidades que sea más restrictiva que el mínimo de permisos necesarios, la consola no funcionará del modo esperado para las entidades (usuarios o roles) que tengan esa política.

No es necesario que conceda permisos mínimos para la consola a los usuarios que solo realizan llamadas a la AWS CLI o a la API de AWS. En su lugar, permite acceso únicamente a las acciones que coincidan con la operación de API que intentan realizar.

Para asegurarse de que los usuarios y los roles puedan seguir utilizando la consola de ElastiCache, asocie también la política administrada `ConsoleAccess` o `ReadOnly` de AWS de ElastiCache a las entidades. Para obtener más información, consulte [Adición de permisos a un usuario](#) en la Guía del usuario de IAM.

Cómo permitir a los usuarios consultar sus propios permisos

En este ejemplo, se muestra cómo podría crear una política que permita a los usuarios de IAM ver las políticas administradas e insertadas que se asocian a la identidad de sus usuarios. Esta política

incluye permisos para realizar esta acción en la consola o mediante programación con la AWS CLI o la API de AWS.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupsForUser",
        "iam:ListAttachedUserPolicies",
        "iam:ListUserPolicies",
        "iam:GetUser"
      ],
      "Resource": ["arn:aws:iam::*:user/${aws:username}"]
    },
    {
      "Sid": "NavigateInConsole",
      "Effect": "Allow",
      "Action": [
        "iam:GetGroupPolicy",
        "iam:GetPolicyVersion",
        "iam:GetPolicy",
        "iam:ListAttachedGroupPolicies",
        "iam:ListGroupPolicies",
        "iam:ListPolicyVersions",
        "iam:ListPolicies",
        "iam:ListUsers"
      ],
      "Resource": "*"
    }
  ]
}
```

Solución de problemas de ElastiCache identidad y acceso a Amazon

Usa la siguiente información para ayudarte a diagnosticar y solucionar los problemas más comunes que puedes encontrar al trabajar con un ElastiCache IAM.

Temas

- [No estoy autorizado a realizar ninguna acción en ElastiCache](#)
- [No estoy autorizado a realizar lo siguiente: PassRole](#)
- [Quiero permitir que personas ajenas a mi AWS cuenta accedan a mis ElastiCache recursos](#)

No estoy autorizado a realizar ninguna acción en ElastiCache

Si AWS Management Console le indica que no está autorizado a realizar una acción, debe ponerse en contacto con su administrador para obtener ayuda. Su administrador es la persona que le facilitó su nombre de usuario y contraseña.

En el siguiente ejemplo, el error se produce cuando el usuario `mateojackson` intenta utilizar la consola para consultar los detalles acerca de un recurso ficticio `my-example-widget`, pero no tiene los permisos ficticios `elasticache:GetWidget`.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
elasticache:GetWidget on resource: my-example-widget
```

En este caso, Mateo pide a su administrador que actualice sus políticas de forma que pueda obtener acceso al recurso `my-example-widget` mediante la acción `elasticache:GetWidget`.

No estoy autorizado a realizar lo siguiente: PassRole

Si recibes un mensaje de error en el que se indica que no estás autorizado a realizar la `iam:PassRole` acción, debes actualizar tus políticas para que puedas transferirle ElastiCache una función.

Algunos Servicios de AWS permiten transferir una función existente a ese servicio en lugar de crear una nueva función de servicio o una función vinculada al servicio. Para ello, debe tener permisos para transferir el rol al servicio.

El siguiente ejemplo de error se produce cuando un usuario de IAM denominado `marymajor` intenta utilizar la consola para realizar una acción en ella. ElastiCache Sin embargo, la acción requiere que el servicio cuente con permisos que otorguen un rol de servicio. Mary no tiene permisos para transferir el rol al servicio.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

En este caso, las políticas de Mary se deben actualizar para permitirle realizar la acción `iam:PassRole`.

Si necesita ayuda, póngase en contacto con su AWS administrador. El administrador es la persona que le proporcionó las credenciales de inicio de sesión.

Quiero permitir que personas ajenas a mi AWS cuenta accedan a mis ElastiCache recursos

Puede crear un rol que los usuarios de otras cuentas o las personas externas a la organización puedan utilizar para acceder a sus recursos. Puede especificar una persona de confianza para que asuma el rol. En el caso de los servicios que admitan las políticas basadas en recursos o las listas de control de acceso (ACL), puede utilizar dichas políticas para conceder a las personas acceso a sus recursos.

Para más información, consulte lo siguiente:

- Para saber si ElastiCache es compatible con estas funciones, consulte [Cómo ElastiCache trabaja Amazon con IAM](#).
- Para obtener información sobre cómo proporcionar acceso a los recursos de su Cuentas de AWS propiedad, consulte [Proporcionar acceso a un usuario de IAM en otro usuario de su propiedad Cuenta de AWS en](#) la Guía del usuario de IAM.
- Para obtener información sobre cómo proporcionar acceso a tus recursos a terceros Cuentas de AWS, consulta [Cómo proporcionar acceso a recursos que Cuentas de AWS son propiedad de terceros](#) en la Guía del usuario de IAM.
- Para obtener información sobre cómo proporcionar acceso mediante una federación de identidades, consulte [Proporcionar acceso a usuarios autenticados externamente \(identidad federada\)](#) en la Guía del usuario de IAM.
- Para conocer la diferencia entre usar roles y políticas basadas en recursos para el acceso entre cuentas, consulte el tema Acceso a [recursos entre cuentas en IAM en la Guía del usuario de IAM](#).

Control de acceso

Puede tener credenciales válidas para autenticar sus solicitudes, pero a menos que tenga permisos, no podrá crear ElastiCache recursos ni acceder a ellos. Por ejemplo, debe tener permisos para crear un ElastiCache clúster.

En las siguientes secciones se describe cómo administrar los permisos para ElastiCache. Recomendamos que lea primero la información general.

- [Información general sobre la administración de los permisos de acceso a los recursos de ElastiCache](#)
- [Uso de políticas basadas en identidad \(políticas de IAM\) para Amazon ElastiCache](#)

Información general sobre la administración de los permisos de acceso a los recursos de ElastiCache

Cada recurso de AWS es propiedad de una cuenta de AWS y los permisos para crear o tener acceso a un recurso se rigen por las políticas de permisos. Un administrador de cuentas puede asociar políticas de permisos a identidades de IAM (es decir, usuarios, grupos y funciones). Además, Amazon ElastiCache también permite adjuntar políticas de permisos a los recursos.

Note

Un administrador de cuentas (o usuario administrador) es un usuario que tiene privilegios de administrador. Para obtener más información, consulte [Prácticas recomendadas de IAM](#) en la Guía del usuario de IAM.

Para dar acceso, añada permisos a los usuarios, grupos o roles:

- Usuarios y grupos en AWS IAM Identity Center:

Cree un conjunto de permisos. Siga las instrucciones de [Creación de un conjunto de permisos](#) en la Guía del usuario de AWS IAM Identity Center.

- Usuarios administrados en IAM a través de un proveedor de identidades:

Cree un rol para la federación de identidades. Siga las instrucciones de [Creación de un rol para un proveedor de identidades de terceros \(federación\)](#) en la Guía del usuario de IAM.

- Usuarios de IAM:

- Cree un rol que el usuario pueda aceptar. Siga las instrucciones descritas en [Creación de un rol para un usuario de IAM](#) en la Guía del usuario de IAM.
- (No recomendado) Adjunte una política directamente a un usuario o añada un usuario a un grupo de usuarios. Siga las instrucciones descritas en [Adición de permisos a un usuario \(consola\)](#) de la Guía del usuario de IAM.

Temas

- [Operaciones y recursos de Amazon ElastiCache](#)
- [Titularidad de los recursos](#)
- [Administrar el acceso a los recursos](#)

- [Políticas administradas por AWS para Amazon ElastiCache](#)
- [Uso de políticas basadas en identidad \(políticas de IAM\) para Amazon ElastiCache](#)
- [Permisos de nivel de recursos](#)
- [Uso de claves de condición](#)
- [Uso de roles vinculados a servicios para Amazon ElastiCache](#)
- [ElastiCache Permisos de API: referencia de acciones, recursos y condiciones](#)

Operaciones y recursos de Amazon ElastiCache

Para ver una lista de tipos de recursos y ARN de ElastiCache, consulte [Tipos de recurso definidos por Amazon ElastiCache](#), en la Referencia de autorizaciones de servicio. Para obtener información sobre las acciones con las que puede especificar el ARN de cada recurso, consulte [Acciones definidas por Amazon ElastiCache](#).

Titularidad de los recursos

El propietario de los recursos es la cuenta de AWS que crea el recurso. Es decir, el propietario del recurso es la cuenta de AWS de la entidad principal que autentica la solicitud que creó el recurso. Una entidad principal puede ser la cuenta raíz, un usuario de IAM o un rol de IAM. Los siguientes ejemplos ilustran cómo funciona:

- Supongamos que utiliza las credenciales de la cuenta raíz de la cuenta de AWS para crear un clúster de caché. En este caso, la cuenta de AWS es la propietaria del recurso. En ElastiCache, el recurso es el clúster de caché.
- Supongamos que crea un usuario de IAM en la cuenta de AWS y concede permisos a fin de crear un clúster de caché para dicho usuario. En este caso, el usuario puede crear un clúster de la caché. Sin embargo, la cuenta de AWS, a la que pertenece el usuario, es la propietaria del recurso de clúster de caché.
- Supongamos que crea un rol de IAM en la cuenta de AWS con permisos para crear un clúster de caché. En este caso, cualquiera que pueda asumir el rol puede crear un clúster de la caché. La cuenta de AWS, a la que pertenece el rol, es la propietaria del recurso de clúster de caché.

Administrar el acceso a los recursos

Una política de permisos describe quién tiene acceso a qué. En la siguiente sección se explican las opciones disponibles para crear políticas de permisos.

Note

En esta sección, también se explica el uso de IAM en el contexto de Amazon ElastiCache. No se proporciona información detallada sobre el servicio de IAM. Para ver la documentación completa de IAM, consulte [¿Qué es IAM?](#) en la Guía del usuario de IAM. Para obtener más información acerca de la sintaxis y las descripciones de las políticas del IAM, consulte [Referencia de políticas de IAM de AWS](#) en la Guía del usuario de IAM.

Las políticas que se asocian a una identidad de IAM se denominan políticas basadas en identidades (o políticas de IAM). Las políticas que se adjuntan a un recurso se denominan políticas basadas en recursos.

Temas

- [Políticas basadas en identidades \(políticas de IAM\)](#)
- [Especificación de elementos de política: acciones, efectos, recursos y entidades principales](#)
- [Especificar las condiciones de una política](#)

Políticas basadas en identidades (políticas de IAM)

Puede asociar políticas a identidades de IAM. Por ejemplo, puede hacer lo siguiente:

- Asociar una política de permisos a un usuario o grupo de la cuenta: un administrador de la cuenta puede utilizar una política de permisos asociada a un usuario determinado para concederle permisos. En este caso, los permisos son para que ese usuario cree un recurso de ElastiCache, como un clúster de caché, un grupo de parámetros o un grupo de seguridad.
- Asociar una política de permisos a un rol (conceder permisos entre cuentas): puede asociar una política de permisos basada en identidad a un rol de IAM; para conceder permisos entre cuentas. Por ejemplo, el administrador de la Cuenta A puede crear un rol para conceder permisos entre cuentas a otra cuenta de AWS (por ejemplo, a la Cuenta B) o a un servicio de AWS, tal y como se indica a continuación:
 1. El administrador de la AccountA crea un rol de IAM y asocia una política de permisos a dicho rol, que concede permisos sobre los recursos de la AccountA.
 2. El administrador de la AccountA asocia una política de confianza al rol que identifica la AccountB como la entidad principal que puede asumir el rol.

3. El administrador de la Cuenta B puede delegar permisos para asumir el rol a cualquier usuario de la Cuenta B. Hacer esto permite a los usuarios de la Cuenta B crear o acceder a los recursos de la Cuenta A. En algunos casos, es posible que desee conceder a un servicio de AWS permisos para asumir el rol. Para respaldar este enfoque, la entidad principal de la política de confianza también puede ser la entidad principal de un servicio de AWS.

Para obtener más información sobre el uso de IAM para delegar permisos, consulte [Administración de accesos](#) en la Guía del usuario de IAM.

A continuación se ofrece un ejemplo de una política que permite a un usuario realizar la acción `DescribeCacheClusters` para la cuenta de AWS. ElastiCache también permite identificar recursos específicos mediante los ARN de recurso para realizar acciones de la API. Este enfoque también se conoce como "permisos a nivel de recursos".

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "DescribeCacheClusters",
    "Effect": "Allow",
    "Action": [
      "elasticache:DescribeCacheClusters"],
    "Resource": resource-arn
  ]
}
```

Para obtener más información sobre el uso de políticas basadas en identidades con ElastiCache, consulte [Uso de políticas basadas en identidad \(políticas de IAM\) para Amazon ElastiCache](#). Para obtener más información sobre usuarios, grupos, roles y permisos, consulte [Identidades \(usuarios, grupos y roles\)](#) en la Guía del usuario de IAM.

Especificación de elementos de política: acciones, efectos, recursos y entidades principales

En cada recurso de Amazon ElastiCache (consulte [Operaciones y recursos de Amazon ElastiCache](#)), el servicio define un conjunto de operaciones de la API (consulte [Acciones](#)). A fin de conceder permisos para estas operaciones de API, ElastiCache define un conjunto de acciones que puede especificar en una política. Por ejemplo, para el recurso del clúster de ElastiCache, se definen las siguientes acciones: `CreateCacheCluster`, `DeleteCacheCluster` y `DescribeCacheCluster`. Para realizar una operación API pueden ser necesarios permisos para más de una acción.

A continuación se indican los elementos más básicos de la política:

- **Recurso:** en una política, se usa un nombre de recurso de Amazon (ARN) para identificar el recurso al que se aplica la política. Para obtener más información, consulte [Operaciones y recursos de Amazon ElastiCache](#).
- **Acción:** utilice palabras clave de acción para identificar las operaciones del recurso que desea permitir o denegar. Por ejemplo, en función del elemento `Effect` especificado, el permiso `elasticache:CreateCacheCluster` permite o deniega al usuario los permisos para realizar la operación `CreateCacheCluster` de Amazon ElastiCache.
- **Efecto:** especifique el efecto que se producirá cuando el usuario solicite la acción específica; puede ser permitir o denegar. Si no concede acceso de forma explícita (permitir) a un recurso, el acceso se deniega implícitamente. También puede denegar explícitamente el acceso a un recurso. Por ejemplo, esto puede servir para asegurarse de que un usuario no pueda tener acceso al recurso, aunque otra política le conceda acceso.
- **Entidad principal:** en las políticas basadas en identidades (políticas de IAM), el usuario al que se asocia esta política es la entidad principal implícita. Para las políticas basadas en recursos, debe especificar el usuario, la cuenta, el servicio u otra entidad que desee que reciba permisos (se aplica solo a las políticas basadas en recursos).

Para obtener más información sobre la sintaxis y descripciones de las políticas de IAM, consulte [Referencia de la política de IAM de AWS](#) en la Guía del usuario de IAM.

Para ver una tabla con todas las acciones de API de Amazon ElastiCache, consulte [ElastiCache Permisos de API: referencia de acciones, recursos y condiciones](#).

Especificar las condiciones de una política

Al conceder permisos, puede utilizar el lenguaje de la política de IAM para especificar las condiciones en la que se debe aplicar una política. Por ejemplo, es posible que desee que solo se aplique una política después de una fecha específica. Para obtener más información sobre cómo especificar condiciones en un lenguaje de política, consulte [Condition](#) en la Guía del usuario de IAM.

Para expresar condiciones, se usan claves de condición predefinidas. Para utilizar las claves de condición específicas de ElastiCache, consulte [Uso de claves de condición](#). Existen claves de condición que se aplican a todo AWS que puede utilizar cuando corresponda. Para ver una lista completa de las claves que se aplican a todo AWS, consulte [Claves disponibles para las condiciones](#) en la Guía del usuario de IAM.

Políticas administradas por AWS para Amazon ElastiCache

Una política administrada de AWS es una política independiente que AWS crea y administra. Las políticas administradas de AWS se diseñan para ofrecer permisos para muchos casos de uso comunes, por lo que puede empezar a asignar permisos a los usuarios, grupos y roles.

Considere que es posible que las políticas administradas por AWS no concedan permisos de privilegio mínimo para los casos de uso concretos, ya que están disponibles para que las utilicen todos los clientes de AWS. Se recomienda definir [políticas administradas por el cliente](#) para los casos de uso a fin de reducir aún más los permisos.

No puede cambiar los permisos definidos en las políticas administradas de AWS. Si AWS actualiza los permisos definidos en una política administrada de AWS, la actualización afecta a todas las identidades de entidades principales (usuarios, grupos y roles) a las que está adjunta la política. Lo más probable es que AWS actualice una política administrada de AWS cuando se lance un nuevo Servicio de AWS o las operaciones de la API nuevas estén disponibles para los servicios existentes.

Para obtener más información, consulte [Políticas administradas por AWS](#) en la Guía del usuario de IAM.

Política administrada por AWS: ElastiCacheServiceRolePolicy

No puede adjuntar ElastiCacheServiceRolePolicy a las entidades de IAM. Esta política está adjunta a un rol vinculado a servicios que permite a ElastiCache realizar acciones en su nombre.

Esta política permite que ElastiCache administre recursos de AWS en su nombre, según sea necesario para administrar la caché:

- `ec2`: administre los recursos de red de EC2 para conectarlos a los nodos de caché, incluidos los puntos de conexión de VPC (para cachés sin servidor), las interfaces de red elásticas (ENI) (para clústeres de autodiseño) y los grupos de seguridad.
- `cloudwatch`: emita datos sobre métricas del servicio a CloudWatch.
- `outposts`: permita la creación de nodos de caché en AWS Outposts.

Puede encontrar la política [ElastiCacheServiceRolePolicy](#) en la consola IAM, y puede encontrar la política [ElastiCacheServiceRolePolicy](#) en la Guía de referencia de políticas administradas de AWS.

Política administrada por AWS: AmazonElastiCacheFullAccess

Puede adjuntar la política de AmazonElastiCacheFullAccess a las identidades de IAM.

Esta política les da a las entidades principales el acceso total a ElastiCache mediante la consola de administración de AWS:

- `elasticache`: acceso completo a todas las API.
- `iam`: cree el rol vinculado al servicio necesario para el funcionamiento del servicio.
- `ec2`: describa los recursos de EC2 dependientes necesarios para la creación de caché (VPC, subred, grupo de seguridad) y permita la creación de puntos de conexión de VPC (para cachés sin servidor).
- `kms`: permita el uso de CMK administradas por el cliente para el cifrado en reposo.
- `cloudwatch`: permita el acceso a métricas a fin de mostrar las métricas de ElastiCache en la consola.
- `application-autoscaling`: permita el acceso para describir las políticas de escalado automático de las cachés.
- `logs`: se utiliza para rellenar los flujos de registros para la funcionalidad de entrega de registros en la consola.
- `firehose`: se utiliza para rellenar los flujos para la funcionalidad de entrega de registros en la consola.
- `s3`: se utiliza a fin de rellenar los buckets S3 para la funcionalidad de restauración de instantáneas en la consola.
- `outposts`: se utiliza a fin de rellenar AWS Outposts para la creación de cachés en la consola.
- `sns`: se utiliza para rellenar los temas de SNS para la funcionalidad de notificación en la consola.

Puede encontrar la política [AmazonElastiCacheFullAccess](#) en la consola de IAM y puede encontrar la política [AmazonElastiCacheFullAccess](#) en la Guía de referencia de las políticas administradas por AWS.

Política administrada por AWS: AmazonElastiCacheReadOnlyAccess

Puede adjuntar la política de AmazonElastiCacheReadOnlyAccess a las identidades de IAM.

Esta política les da a las entidades principales el acceso de solo lectura a ElastiCache mediante la consola de administración de AWS:

- `elasticache`: acceda a las API `Describe` de solo lectura.

Puede encontrar la política [AmazonElastiCacheReadOnlyAccess](#) en la consola de IAM y puede encontrar la política [AmazonElastiCacheReadOnlyAccess](#) en la Guía de referencia de las políticas administradas por AWS.

Actualizaciones de ElastiCache a las políticas administradas de AWS

Es posible consultar los detalles sobre las actualizaciones de las políticas administradas de AWS para ElastiCache debido a que este servicio comenzó a realizar un seguimiento de estos cambios. Para obtener alertas automáticas sobre los cambios realizados en esta página, suscríbase a la fuente RSS en la página del historial de documentos de ElastiCache.

Cambio	Descripción	Fecha
AmazonElastiCacheFullAccess : actualización de una política existente.	ElastiCache agregó nuevos permisos para permitir la administración de cachés sin servidor y para permitir el uso de todas las características del servicio a través de la consola.	27 de noviembre de 2023
ElastiCacheServiceRolePolicy : actualización de una política existente.	ElastiCache agregó nuevos permisos para permitir la administración de los puntos de conexión de VPC para los recursos de caché sin servidor.	27 de noviembre de 2023
ElastiCache comenzó a hacer un seguimiento de los cambios	ElastiCache comenzó a realizar el seguimiento de los cambios de las políticas administradas de AWS.	7 de febrero de 2020

Uso de políticas basadas en identidad (políticas de IAM) para Amazon ElastiCache

Este tema contiene ejemplos de políticas basadas en identidades, donde los administradores de cuentas pueden asociar políticas de permisos a identidades de IAM (es decir, a usuarios, grupos y funciones).

Important

Recomendamos que primero lea los temas en los que se explican los conceptos básicos y las opciones para administrar el acceso a los recursos de Amazon ElastiCache. Para obtener más información, consulte [Información general sobre la administración de los permisos de acceso a los recursos de ElastiCache](#).

En las secciones de este tema se explica lo siguiente:

- [Políticas administradas por AWS para Amazon ElastiCache](#)
- [Ejemplos de políticas administradas por los clientes](#)

A continuación se muestra un ejemplo de una política de permisos.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowClusterPermissions",
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateServerlessCache",
        "elasticache:CreateCacheCluster",
        "elasticache:DescribeServerlessCaches",
        "elasticache:DescribeReplicationGroups",
        "elasticache:DescribeCacheClusters",
        "elasticache:ModifyServerlessCache",
        "elasticache:ModifyReplicationGroup",
        "elasticache:ModifyCacheCluster"
      ],
      "Resource": "*"
    }
  ],
}
```

```
        "Sid": "AllowUserToPassRole",
        "Effect": "Allow",
        "Action": [ "iam:PassRole" ],
        "Resource": "arn:aws:iam::123456789012:role/EC2-roles-for-cluster"
    }
]
}
```

La política tiene dos instrucciones:

- La primera declaración concede permisos para las acciones de Amazon ElastiCache (`elasticache:Create*`, `elasticache:Describe*`, `elasticache:Modify*`)
- La segunda declaración concede permisos para la acción de IAM (`iam:PassRole`) en el nombre de rol de IAM especificado al final del valor `Resource`.

La política no especifica el elemento `Principal`, ya que en una política basada en la identidad no se especifica el elemento principal que obtiene el permiso. Al asociar una política a un usuario, el usuario es la entidad principal implícita. Cuando asocia una política de permisos a un rol de IAM, el elemento principal identificado en la política de confianza de rol obtiene los permisos.

Para ver una tabla con todas las acciones de la API de Amazon ElastiCache y los recursos a los que se aplican, consulte [ElastiCache Permisos de API: referencia de acciones, recursos y condiciones](#).

Ejemplos de políticas administradas por los clientes

Si no está utilizando una política predeterminada y elige utilizar una política administrada de forma personalizada, asegúrese de una de las dos cosas. Debería tener permisos para llamar a `iam:createServiceLinkedRole` (para obtener más información, consulte [Ejemplo 4: permitir que un usuario llame a la API `CreateServiceLinkedRole` de IAM](#)). También debe haber creado un rol vinculado al servicio de ElastiCache.

Combinadas con los permisos mínimos necesarios para utilizar la consola de Amazon ElastiCache, las políticas de ejemplo de esta sección conceden permisos adicionales. Los ejemplos también son relevantes para los SDK de AWS y la AWS CLI.

Para obtener instrucciones sobre la configuración de grupos y usuarios de IAM, consulte [Creación del primer grupo y usuario administrador de IAM](#) en la Guía del usuario de IAM.

⚠ Important

Pruebe siempre sus políticas de IAM antes de utilizarlas en entornos de producción. Algunas acciones de ElastiCache que parecen sencillas pueden requerir otras acciones de apoyo cuando se utiliza la consola de ElastiCache. Por ejemplo, `elasticache:CreateCacheCluster` concede permisos para crear clústeres de caché de ElastiCache. Sin embargo, para realizar esta operación, la consola de ElastiCache utiliza varias acciones `Describe` y `List` a fin de completar las listas de la consola.

Ejemplos de

- [Ejemplo 1: permitir al usuario acceso de solo lectura a los recursos de ElastiCache](#)
- [Ejemplo 2: conceder a un usuario permiso para realizar tareas comunes de administrador del sistema de ElastiCache](#)
- [Ejemplo 3: conceder a un usuario permiso para tener acceso a todas las acciones de API de ElastiCache](#)
- [Ejemplo 4: permitir que un usuario llame a la API `CreateServiceLinkedRole` de IAM](#)
- [Ejemplo 5: permitir que un usuario se conecte a una caché sin servidor mediante la autenticación de IAM](#)

Ejemplo 1: permitir al usuario acceso de solo lectura a los recursos de ElastiCache

La siguiente política concede permisos para utilizar acciones de ElastiCache que permiten a un usuario mostrar recursos. Normalmente, este tipo de política de permisos se adjunta a un grupo de administradores.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "ECReadOnly",
    "Effect": "Allow",
    "Action": [
      "elasticache:Describe*",
      "elasticache:List*"
    ],
    "Resource": "*"
  ]
}
```

```
}
```

Ejemplo 2: conceder a un usuario permiso para realizar tareas comunes de administrador del sistema de ElastiCache

Las tareas comunes del administrador del sistema incluyen la modificación de los recursos. También es posible que el administrador del sistema necesite obtener información acerca de los eventos de ElastiCache. La siguiente política concede a un usuario permisos a fin de realizar acciones de ElastiCache para estas tareas comunes de administrador del sistema. Normalmente, este tipo de política de permisos se adjunta al grupo de administradores del sistema.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "ECAAllowMutations",
    "Effect": "Allow",
    "Action": [
      "elasticache:Modify*",
      "elasticache:Describe*",
      "elasticache:ResetCacheParameterGroup"
    ],
    "Resource": "*"
  }
]
```

Ejemplo 3: conceder a un usuario permiso para tener acceso a todas las acciones de API de ElastiCache

La siguiente política permite a un usuario acceder a todas las acciones de ElastiCache. Recomendamos que conceda este tipo de política de permisos solo a un usuario administrador.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "ECAAllowAll",
    "Effect": "Allow",
    "Action": [
      "elasticache:*"
    ],
    "Resource": "*"
  }
]
```

```

]
}

```

Ejemplo 4: permitir que un usuario llame a la API CreateServiceLinkedRole de IAM

La siguiente política permite al usuario llamar a la API CreateServiceLinkedRole de IAM.

Recomendamos que conceda este tipo de política de permisos al usuario que invoca las operaciones de ElastiCache mutantes.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "CreateSLRAllows",
      "Effect": "Allow",
      "Action": [
        "iam:CreateServiceLinkedRole"
      ],
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "iam:AWSServiceName": "elasticache.amazonaws.com"
        }
      }
    }
  ]
}

```

Ejemplo 5: permitir que un usuario se conecte a una caché sin servidor mediante la autenticación de IAM

La siguiente política permite que cualquier usuario se conecte a cualquier caché sin servidor mediante la autenticación de IAM entre el 1 de abril de 2023 y el 30 de junio de 2023.

```

{
  "Version" : "2012-10-17",
  "Statement" :
  [
    {
      "Effect" : "Allow",
      "Action" : ["elasticache:Connect"],
      "Resource" : [

```

```
    "arn:aws:elasticache:us-east-1:123456789012:serverlesscache:*"
  ],
  "Condition": {
    "DateGreaterThan": {"aws:CurrentTime": "2023-04-01T00:00:00Z"},
    "DateLessThan": {"aws:CurrentTime": "2023-06-30T23:59:59Z"}
  }
},
{
  "Effect" : "Allow",
  "Action" : ["elasticache:Connect"],
  "Resource" : [
    "arn:aws:elasticache:us-east-1:123456789012:user:*"
  ]
}
]
```

Permisos de nivel de recursos

Puede restringir el alcance de los permisos de un usuario mediante la especificación de recursos en una política de IAM. Muchas acciones de la API de ElastiCache admiten un tipo de recurso que varía en función del comportamiento de la acción. Cada instrucción de una política de IAM concede permiso para realizar una acción en un recurso. Cuando la acción no actúa sobre un recurso nombrado, o cuando se concede permiso para realizar la acción sobre todos los recursos, el valor del recurso en la política es un comodín (*). Para muchas acciones de API, puede restringir los recursos que un usuario puede modificar si especifica el nombre de recurso de Amazon (ARN) de un recurso o un patrón de ARN que coincida con varios recursos. Para restringir los permisos por recurso, especifique el recurso por ARN.

Para ver una lista de tipos de recursos y ARN de ElastiCache, consulte [Tipos de recurso definidos por Amazon ElastiCache](#), en la Referencia de autorizaciones de servicio. Para obtener información sobre las acciones con las que puede especificar el ARN de cada recurso, consulte [Acciones definidas por Amazon ElastiCache](#).

Ejemplos de

- [Ejemplo 1: permitir a un usuario acceso completo a tipos de recursos específicos de ElastiCache](#)
- [Ejemplo 2: denegarle a un usuario el acceso a una memoria caché sin servidor.](#)

Ejemplo 1: permitir a un usuario acceso completo a tipos de recursos específicos de ElastiCache

La siguiente política permite de forma explícita todos los recursos del tipo caché sin servidor.

```
{
  "Sid": "Example1",
  "Effect": "Allow",
  "Action": "elasticache:*",
  "Resource": [
    "arn:aws:elasticache:us-east-1:account-id:serverlesscache:*"
  ]
}
```

Ejemplo 2: denegarle a un usuario el acceso a una memoria caché sin servidor.

En el siguiente ejemplo, se deniega de forma explícita el acceso a una determinada caché sin servidor.

```
{
  "Sid": "Example2",
  "Effect": "Deny",
  "Action": "elasticache:*",
  "Resource": [
    "arn:aws:elasticache:us-east-1:account-id:serverlesscache:name"
  ]
}
```

Uso de claves de condición

Puede especificar condiciones que determinan cómo se aplica una política de IAM. En ElastiCache, puede utilizar el `Condition` elemento de una política de JSON para comparar las claves del contexto de la solicitud con los valores clave que especifique en la política. Para obtener más información, consulte [Elementos de la política de JSON de IAM: Condición](#).

Para ver una lista de claves de ElastiCache estado, consulta [Claves de estado de Amazon ElastiCache](#) en la Referencia de autorización de servicio.

Para obtener una lista de todas las claves de condición globales, consulte [Claves de contexto de condición globales de AWS](#).

Especificación de condiciones: uso de claves de condición

Para implementar el control detallado, hay que escribir una política de permisos de IAM que especifique las condiciones a fin de controlar un conjunto de parámetros individuales en determinadas solicitudes. A continuación, se aplica la política de IAM a los usuarios, los grupos o los roles creados con la consola de IAM.

Para aplicar una condición, agregue la información de condición a la declaración de la política de IAM. En el siguiente ejemplo, se especifica la condición de que cualquier clúster de caché de autodiseño creado sea del tipo de nodo cache .r5.large.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateCacheCluster",
        "elasticache:CreateReplicationGroup"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:parametergroup:*",
        "arn:aws:elasticache:*:*:subnetgroup:*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateCacheCluster",
        "elasticache:CreateReplicationGroup"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:cluster:*",
        "arn:aws:elasticache:*:*:replicationgroup:*"
      ],
      "Condition": {
        "StringEquals": {
          "elasticache:CacheNodeType": [
            "cache.r5.large"
          ]
        }
      }
    }
  ]
}
```

```
]
}
```

Para obtener más información, consulte [Ejemplos de políticas de control de acceso basadas en etiquetas](#).

Para obtener más información sobre el uso de operadores de condición de política, consulte [ElastiCache Permisos de API: referencia de acciones, recursos y condiciones](#).

Ejemplos de políticas: uso de condiciones para el control de parámetros de precisión

En esta sección se muestran ejemplos de políticas para implementar un control de acceso detallado en los parámetros enumerados anteriormente. ElastiCache

1. `elasticache:MaximumDataStorage`: Especifique el almacenamiento máximo de datos de una caché sin servidor. Mediante las condiciones proporcionadas, el cliente no puede crear cachés que puedan almacenar más de una cantidad específica de datos.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowDependentResources",
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateServerlessCache"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:serverlesscachesnapshot:*",
        "arn:aws:elasticache:*:*:snapshot:*",
        "arn:aws:elasticache:*:*:usergroup:*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateServerlessCache"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:serverlesscache:*"
      ],
      "Condition": {
        "NumericLessThanEquals": {
```

```

        "elasticache:MaximumDataStorage": "30"
      },
      "StringEquals": {
        "elasticache:DataStorageUnit": "GB"
      }
    }
  ]
}

```

2. `ElastiCache:MaximumECPUPerSecond`: especifique el valor máximo de ECPU por segundo de una caché sin servidor. Mediante las condiciones proporcionadas, el cliente no puede crear cachés que puedan ejecutar más de un número específico de ECPU por segundo.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowDependentResources",
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateServerlessCache"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:serverlesscachesnapshot:*",
        "arn:aws:elasticache:*:*:snapshot:*",
        "arn:aws:elasticache:*:*:usergroup:*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateServerlessCache"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:serverlesscache:*"
      ],
      "Condition": {
        "NumericLessThanEquals": {
          "elasticache:MaximumECPUPerSecond": "100000"
        }
      }
    }
  ]
}

```

```
]
}
```

3. `elasticache::CacheNodeType` Especifique cuáles puede crear un usuario. `NodeType` Mediante las condiciones proporcionadas, el cliente puede especificar un valor único o un valor de rango para un tipo de nodo.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateCacheCluster",
        "elasticache:CreateReplicationGroup"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:parametergroup:*",
        "arn:aws:elasticache:*:*:subnetgroup:*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateCacheCluster",
        "elasticache:CreateReplicationGroup"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:cluster:*",
        "arn:aws:elasticache:*:*:replicationgroup:*"
      ],
      "Condition": {
        "StringEquals": {
          "elasticache:CacheNodeType": [
            "cache.t2.micro",
            "cache.t2.medium"
          ]
        }
      }
    }
  ]
}
```

4. elasticache:NumNodeGroups: Cree un grupo de replicación con menos de 20 grupos de nodos.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateReplicationGroup"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:parametergroup:*",
        "arn:aws:elasticache:*:*:subnetgroup:*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateReplicationGroup"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:replicationgroup:*"
      ],
      "Condition": {
        "NumericLessThanEquals": {
          "elasticache:NumNodeGroups": "20"
        }
      }
    }
  ]
}
```

5. elasticache:ReplicasPerNodeGroup: Especifique las réplicas por nodo entre 5 y 10.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateReplicationGroup"
      ],
    },
  ],
}
```

```

    "Resource": [
      "arn:aws:elasticache:*:*:parametergroup:*",
      "arn:aws:elasticache:*:*:subnetgroup:*"
    ],
  },
  {
    "Effect": "Allow",
    "Action": [
      "elasticache:CreateReplicationGroup"
    ],
    "Resource": [
      "arn:aws:elasticache:*:*:replicationgroup:*"
    ],
    "Condition": {
      "NumericGreaterThanEquals": {
        "elasticache:ReplicasPerNodeGroup": "5"
      },
      "NumericLessThanEquals": {
        "elasticache:ReplicasPerNodeGroup": "10"
      }
    }
  }
]
}

```

6. elasticache:EngineVersion: Especifique el uso de la versión 5.0.6 del motor.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateCacheCluster",
        "elasticache:CreateReplicationGroup"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:parametergroup:*",
        "arn:aws:elasticache:*:*:subnetgroup:*"
      ]
    },
  ],
}

```

```

    "Effect": "Allow",
    "Action": [
        "elasticache:CreateCacheCluster",
        "elasticache:CreateReplicationGroup"
    ],
    "Resource": [
        "arn:aws:elasticache:*:*:cluster:*",
        "arn:aws:elasticache:*:*:replicationgroup:*"
    ],
    "Condition": {
        "StringEquals": {
            "elasticache:EngineVersion": "5.0.6"
        }
    }
}
]
}

```

7. elasticache:EngineType: Especifique utilizando únicamente el motor OSS de Redis.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateCacheCluster",
        "elasticache:CreateReplicationGroup"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:parametergroup:*",
        "arn:aws:elasticache:*:*:subnetgroup:*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateCacheCluster",
        "elasticache:CreateReplicationGroup"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:cluster:*",
        "arn:aws:elasticache:*:*:replicationgroup:*"
      ]
    }
  ]
}

```

```

    ],
    "Condition": {
      "StringEquals": {
        "elasticache:EngineType": "redis"
      }
    }
  }
]
}

```

8. `elasticache:AtRestEncryptionEnabled`: Especifique que los grupos de replicación se crearán solo con el cifrado habilitado.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateReplicationGroup"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:parametergroup:*",
        "arn:aws:elasticache:*:*:subnetgroup:*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateReplicationGroup"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:replicationgroup:*"
      ],
      "Condition": {
        "Bool": {
          "elasticache:AtRestEncryptionEnabled": "true"
        }
      }
    }
  ]
}

```

}

9. elasticache: TransitEncryptionEnabled

- a. Defina la clave de `elasticache:TransitEncryptionEnabled` condición `false` para que la [CreateReplicationGroup](#) acción especifique que los grupos de replicación solo se pueden crear cuando no se utilice TLS:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateReplicationGroup"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:parametergroup:*",
        "arn:aws:elasticache:*:*:subnetgroup:*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateReplicationGroup"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:replicationgroup:*"
      ],
      "Condition": {
        "Bool": {
          "elasticache:TransitEncryptionEnabled": "false"
        }
      }
    }
  ]
}
```

Si la clave de `elasticache:TransitEncryptionEnabled` condición se establece `false` en una política para la [CreateReplicationGroup](#) acción, solo se permitirá una `CreateReplicationGroup` solicitud si no se utiliza TLS (es decir, si la solicitud no

incluye un `TransitEncryptionEnabled` parámetro establecido como `true` o un `TransitEncryptionMode` parámetro establecido como `required`

- b. Defina la clave de `elasticache:TransitEncryptionEnabled` condición `true` para que la [CreateReplicationGroup](#) acción especifique que los grupos de replicación solo se pueden crear cuando se utiliza TLS:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateReplicationGroup"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:parametergroup:*",
        "arn:aws:elasticache:*:*:subnetgroup:*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateReplicationGroup"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:replicationgroup:*"
      ],
      "Condition": {
        "Bool": {
          "elasticache:TransitEncryptionEnabled": "true"
        }
      }
    }
  ]
}
```

Cuando la clave de `elasticache:TransitEncryptionEnabled` condición esté establecida `true` en una política para la [CreateReplicationGroup](#) acción, solo se permitirá una `CreateReplicationGroup` solicitud si la solicitud incluye un `TransitEncryptionEnabled`

parámetro establecido en `true` y un `TransitEncryptionMode` parámetro establecido en `required`

- c. Defina `elasticache:TransitEncryptionEnabled` en `true` para que la acción `ModifyReplicationGroup` especifique que los grupos de replicación solo se pueden modificar cuando se usa TLS:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:ModifyReplicationGroup"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:replicationgroup:*"
      ],
      "Condition": {
        "BoolIfExists": {
          "elasticache:TransitEncryptionEnabled": "true"
        }
      }
    }
  ]
}
```

Cuando la clave de `elasticache:TransitEncryptionEnabled` condición esté establecida `true` en una política para la [ModifyReplicationGroup](#) acción, solo se permitirá una `ModifyReplicationGroup` solicitud si la solicitud incluye un `TransitEncryptionMode` parámetro establecido en `required`. El parámetro `TransitEncryptionEnabled` establecido en `true` también se puede incluir de forma opcional, pero no es necesario en este caso para habilitar TLS.

10. `elasticache:AutomaticFailoverEnabled`: Especifique que los grupos de replicación se crearán solo con la conmutación por error automática habilitada.

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

```

    "Action": [
      "elasticache:CreateReplicationGroup"
    ],
    "Resource": [
      "arn:aws:elasticache:*:*:parametergroup:*",
      "arn:aws:elasticache:*:*:subnetgroup:*"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "elasticache:CreateReplicationGroup"
    ],
    "Resource": [
      "arn:aws:elasticache:*:*:replicationgroup:*"
    ],
    "Condition": {
      "Bool": {
        "elasticache:AutomaticFailoverEnabled": "true"
      }
    }
  }
]
}

```

11 `elasticache:MultiAZEnabled`: especifique que no se pueden crear grupos de reproducción con Multi-AZ deshabilitado.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateCacheCluster",
        "elasticache:CreateReplicationGroup"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:parametergroup:*",
        "arn:aws:elasticache:*:*:subnetgroup:*"
      ]
    }
  ],
}

```

```

    {
      "Effect": "Deny",
      "Action": [
        "elasticache:CreateCacheCluster",
        "elasticache:CreateReplicationGroup"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:cluster:*",
        "arn:aws:elasticache:*:*:replicationgroup:*"
      ],
      "Condition": {
        "Bool": {
          "elasticache:MultiAZEnabled": "false"
        }
      }
    }
  ]
}

```

12 `elasticache:ClusterModeEnabled`: Especifique que los grupos de replicación solo se pueden crear con el modo de clúster activado.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateReplicationGroup"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:parametergroup:*",
        "arn:aws:elasticache:*:*:subnetgroup:*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateReplicationGroup"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:replicationgroup:*"
      ]
    }
  ]
}

```

```

    ],
    "Condition": {
      "Bool": {
        "elasticache:ClusterModeEnabled": "true"
      }
    }
  }
]
}

```

13. `elasticache:AuthTokenEnabled`: Especifique que los grupos de replicación solo se pueden crear con el token AUTH habilitado.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateCacheCluster",
        "elasticache:CreateReplicationGroup"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:parametergroup:*",
        "arn:aws:elasticache:*:*:subnetgroup:*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateCacheCluster",
        "elasticache:CreateReplicationGroup"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:cluster:*",
        "arn:aws:elasticache:*:*:replicationgroup:*"
      ],
      "Condition": {
        "Bool": {
          "elasticache:AuthTokenEnabled": "true"
        }
      }
    }
  ]
}

```

```

    }
  }
]
}

```

14 `elasticache:SnapshotRetentionLimit`: Especifique el número de días (o mínimo/máximo) para conservar la instantánea. A continuación, la política impone el almacenamiento de copias de seguridad durante al menos 30 días.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateCacheCluster",
        "elasticache:CreateReplicationGroup"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:parametergroup:*",
        "arn:aws:elasticache:*:*:subnetgroup:*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateCacheCluster",
        "elasticache:CreateReplicationGroup",
        "elasticache:CreateServerlessCache"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:cluster:*",
        "arn:aws:elasticache:*:*:replicationgroup:*",
        "arn:aws:elasticache:*:*:serverlesscache:*"
      ],
      "Condition": {
        "NumericGreaterThanEquals": {
          "elasticache:SnapshotRetentionLimit": "30"
        }
      }
    }
  ]
}

```

```
]
}
```

15 `elasticache:KmsKeyId`: Especifique el uso de las claves de KMS administradas por el cliente. AWS

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowDependentResources",
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateServerlessCache"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:serverlesscachesnapshot:*",
        "arn:aws:elasticache:*:*:snapshot:*",
        "arn:aws:elasticache:*:*:usergroup:*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateServerlessCache"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:serverlesscache:*"
      ],
      "Condition": {
        "StringEquals": {
          "elasticache:KmsKeyId": "my-key"
        }
      }
    }
  ]
}
```

16 `elasticache:CacheParameterGroupName`: Especifique un grupo de parámetros no predeterminado con parámetros específicos de una organización en sus clústeres. También puede especificar un patrón de nomenclatura para los grupos de parámetros o eliminar bloques en un nombre de grupo de parámetros específico. El siguiente es un ejemplo que restringe el uso de solo "». `my-org-param-group`

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateCacheCluster",
        "elasticache:CreateReplicationGroup"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:parametergroup:*",
        "arn:aws:elasticache:*:*:subnetgroup:*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateCacheCluster",
        "elasticache:CreateReplicationGroup"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:cluster:*",
        "arn:aws:elasticache:*:*:replicationgroup:*"
      ],
      "Condition": {
        "StringEquals": {
          "elasticache:CacheParameterGroupName": "my-org-param-group"
        }
      }
    }
  ]
}

```

17 `elasticache:CreateCacheCluster`: Denegar la `CreateCacheCluster` acción si falta la etiqueta `Project` de solicitud o no es igual a `Dev`, o. `QA Prod`

```

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

```

```

    "Effect": "Allow",
    "Action": [
        "elasticache:CreateCacheCluster"
    ],
    "Resource": [
        "arn:aws:elasticache:*:*:parametergroup:*",
        "arn:aws:elasticache:*:*:subnetgroup:*",
        "arn:aws:elasticache:*:*:securitygroup:*",
        "arn:aws:elasticache:*:*:replicationgroup:*"
    ]
},
{
    "Effect": "Deny",
    "Action": [
        "elasticache:CreateCacheCluster"
    ],
    "Resource": [
        "arn:aws:elasticache:*:*:cluster:*"
    ],
    "Condition": {
        "Null": {
            "aws:RequestTag/Project": "true"
        }
    }
},
{
    "Effect": "Allow",
    "Action": [
        "elasticache:CreateCacheCluster",
        "elasticache:AddTagsToResource"
    ],
    "Resource": "arn:aws:elasticache:*:*:cluster:*",
    "Condition": {
        "StringEquals": {
            "aws:RequestTag/Project": [
                "Dev",
                "Prod",
                "QA"
            ]
        }
    }
}
]

```

}

18. elasticache:CacheNodeType: Permitir CreateCacheCluster con cacheNodeType cache.r5.large o cache.r6g.4xlarge y etiqueta. Project=XYZ

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateCacheCluster",
        "elasticache:CreateReplicationGroup"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:parametergroup:*",
        "arn:aws:elasticache:*:*:subnetgroup:*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateCacheCluster"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:cluster:*"
      ],
      "Condition": {
        "StringEqualsIfExists": {
          "elasticache:CacheNodeType": [
            "cache.r5.large",
            "cache.r6g.4xlarge"
          ]
        },
        "StringEquals": {
          "aws:RequestTag/Project": "XYZ"
        }
      }
    }
  ]
}
```

Note

Cuando se crean políticas a fin de imponer etiquetas y otras claves de condición juntas, el condicional `IfExists` puede ser necesario en los elementos de la clave de condición debido a los requisitos extra de la política `elasticache:AddTagsToResource` para las solicitudes de creación con el parámetro `--tags`.

Uso de roles vinculados a servicios para Amazon ElastiCache

Amazon ElastiCache utiliza [roles vinculados a servicios](#) de AWS Identity and Access Management (IAM). Un rol vinculado a servicios es un tipo único de rol de IAM que se encuentra vinculado directamente a un servicio de AWS, como Amazon ElastiCache. Amazon ElastiCache predefine los roles vinculados a servicios de Amazon ElastiCache. Incluyen todos los permisos que requiere el servicio para llamar a otros servicios de AWS en nombre de los clústeres.

Un rol vinculado a servicios simplifica la configuración de Amazon ElastiCache porque ya no tendrá que agregar manualmente los permisos necesarios. Los roles ya existen dentro de la cuenta de AWS, pero se encuentran vinculados a los casos de uso de Amazon ElastiCache y tienen permisos predefinidos. Solo Amazon ElastiCache puede asumir estos roles y solo estos roles pueden utilizar la política de permisos predefinida. Las funciones se pueden eliminar únicamente después de eliminar primero sus recursos relacionados. De esta forma se protegen los recursos de Amazon ElastiCache porque no puede eliminar los permisos necesarios para acceder a los recursos.

Para obtener información acerca de otros servicios que admiten roles vinculados a servicios, consulte [Servicios de AWS que funcionan con IAM](#) y busque los servicios que muestran Sí en la columna Roles vinculado a servicios. Elija una opción Sí con un enlace para ver la documentación acerca del rol vinculado a servicios en cuestión.

Contenido

- [Permisos de roles vinculados a servicios para Amazon ElastiCache](#)
 - [Permisos para crear un rol vinculado a servicios](#)
- [Creación de un rol vinculado a servicios \(IAM\)](#)
 - [Creación de un rol vinculado a servicios \(consola de IAM\)](#)
 - [Creación de un rol vinculado a servicios \(CLI de IAM\)](#)
 - [Creación de un rol vinculado a servicios \(API de IAM\)](#)
- [Edición de la descripción de un rol vinculado a servicios para Amazon ElastiCache](#)

- [Edición de la descripción de un rol vinculado a servicios \(consola de IAM\)](#)
- [Edición de la descripción de un rol vinculado a servicios \(CLI de IAM\)](#)
- [Edición de la descripción de un rol vinculado a servicios \(API de IAM\)](#)
- [Eliminar un rol vinculado a servicios para Amazon ElastiCache](#)
 - [Limpiar un rol vinculado a un servicio](#)
 - [Eliminación de un rol vinculado a servicios \(consola de IAM\)](#)
 - [Eliminación de un rol vinculado a servicios \(CLI de IAM\)](#)
 - [Eliminación de un rol vinculado a servicios \(API de IAM\)](#)

Permisos de roles vinculados a servicios para Amazon ElastiCache

Permisos para crear un rol vinculado a servicios

Para permitir a una entidad de IAM crear roles vinculados a servicios
AWSServiceRoleForElastiCache

Agregue la siguiente instrucción de política a los permisos para esa entidad de IAM:

```
{
  "Effect": "Allow",
  "Action": [
    "iam:CreateServiceLinkedRole",
    "iam:PutRolePolicy"
  ],
  "Resource": "arn:aws:iam::*:role/aws-service-role/elasticache.amazonaws.com/AWSServiceRoleForElastiCache*",
  "Condition": {"StringLike": {"iam:AWSServiceName": "elasticache.amazonaws.com"}}
}
```

Para permitir a una entidad de IAM eliminar roles vinculados a servicios
AWSServiceRoleForElastiCache

Agregue la siguiente instrucción de política a los permisos para esa entidad de IAM:

```
{
  "Effect": "Allow",
  "Action": [
    "iam>DeleteServiceLinkedRole",
  ]
}
```

```
    "iam:GetServiceLinkedRoleDeletionStatus"
  ],
  "Resource": "arn:aws:iam::*:role/aws-service-role/elasticache.amazonaws.com/AWSServiceRoleForElastiCache*",
  "Condition": {"StringLike": {"iam:AWSServiceName": "elasticache.amazonaws.com"}}
}
```

También puede utilizar una política administrada por AWS para proporcionar acceso completo a Amazon ElastiCache.

Creación de un rol vinculado a servicios (IAM)

Puede crear un rol vinculado a servicios mediante la consola de IAM, la CLI o la API.

Creación de un rol vinculado a servicios (consola de IAM)

Puede utilizar la consola de IAM para crear un rol vinculado a un servicio.

Para crear un rol vinculado a un servicio (consola)

1. Inicie sesión en la AWS Management Console y abra la consola de IAM en <https://console.aws.amazon.com/iam/>.
2. En el panel de navegación de la consola de IAM, elija Roles. A continuación, elija Create new role (Crear nuevo rol).
3. En Select type of trusted entity (Seleccionar el tipo de entidad de confianza), elija AWS Service (Servicio de).
4. En Or select a service to view its use cases (O seleccionar un servicio para ver sus casos de uso), elija ElastiCache.
5. Elija Siguiente: permisos.
6. En Policy name (Nombre de la política), tenga en cuenta que ElastiCacheServiceRolePolicy es necesario para este rol. Elija Siguiente:Etiquetas.
7. Tenga en cuenta que las etiquetas no son compatibles con los roles vinculados a servicios. Elija Next: Review.
8. (Opcional) En Descripción del rol, edite la descripción del nuevo rol vinculado al servicio.
9. Revise el rol y, a continuación, seleccione Crear rol.

Creación de un rol vinculado a servicios (CLI de IAM)

Puede utilizar las operaciones de IAM desde la AWS Command Line Interface para crear un rol vinculado a servicios. Este rol puede incluir la política de confianza y las políticas insertadas que el servicio necesita para asumir el rol.

Para crear un rol vinculado a un servicio (CLI)

Use la operación siguiente:

```
$ aws iam create-service-linked-role --aws-service-name elasticache.amazonaws.com
```

Creación de un rol vinculado a servicios (API de IAM)

Puede utilizar la API de IAM para crear un rol vinculado a servicios. Este rol puede contener la política de confianza y las políticas insertadas que el servicio necesita para asumir el rol.

Para crear un rol vinculado a un servicio (API)

Use la llamada de API de [CreateServiceLinkedRole](#). En la solicitud, especifique el nombre del servicio de `elasticache.amazonaws.com`.

Edición de la descripción de un rol vinculado a servicios para Amazon ElastiCache

Amazon ElastiCache no permite editar el rol vinculado a servicios de `AWSServiceRoleForElastiCache`. Después de crear un rol vinculado al servicio, no podrá cambiar el nombre del rol, ya que varias entidades podrían hacer referencia al rol. Sin embargo, sí puede editar la descripción del rol con IAM.

Edición de la descripción de un rol vinculado a servicios (consola de IAM)

Puede utilizar la consola de IAM para editar una descripción de rol vinculado a servicios.

Para editar la descripción de un rol vinculado a un servicio (consola)

1. En el panel de navegación de la consola de IAM, elija Roles.
2. Seleccione el nombre del rol que desea modificar.
3. En el extremo derecho de Role description, seleccione Edit.
4. Ingrese una descripción nueva en el cuadro Save (Guardar).

Edición de la descripción de un rol vinculado a servicios (CLI de IAM)

Puede utilizar operaciones de IAM desde la AWS Command Line Interface para editar una descripción de rol vinculado a servicios.

Para cambiar la descripción de un rol vinculado a un servicio (CLI)

1. (Opcional) A fin de ver la descripción actual de un rol, utilice la AWS CLI para la operación de IAM [get-role](#).

Example

```
$ aws iam get-role --role-name AWSServiceRoleForElastiCache
```

Utilice el nombre del rol, no el ARN, para hacer referencia a los roles con las operaciones de la CLI. Por ejemplo, si una función tiene el ARN `arn:aws:iam::123456789012:role/myrole`, debe referirse a él como **myrole**.

2. Para actualizar la descripción de un rol vinculado a servicios, utilice la AWS CLI para la operación de IAM [update-role-description](#).

Para Linux, macOS o Unix:

```
$ aws iam update-role-description \  
  --role-name AWSServiceRoleForElastiCache \  
  --description "new description"
```

Para Windows:

```
$ aws iam update-role-description ^  
  --role-name AWSServiceRoleForElastiCache ^  
  --description "new description"
```

Edición de la descripción de un rol vinculado a servicios (API de IAM)

Puede utilizar la API de IAM para editar una descripción de rol vinculado a servicios.

Para cambiar la descripción de un rol vinculado a un servicio (API)

1. (Opcional) Para ver la descripción actual de un rol, utilice la operación de la API de IAM [GetRole](#).

Example

```
https://iam.amazonaws.com/  
?Action=GetRole  
&RoleName=AWSServiceRoleForElastiCache  
&Version=2010-05-08  
&AUTHPARAMS
```

2. Para actualizar la descripción de un rol, utilice la operación de la API de IAM [UpdateRoleDescription](#).

Example

```
https://iam.amazonaws.com/  
?Action=UpdateRoleDescription  
&RoleName=AWSServiceRoleForElastiCache  
&Version=2010-05-08  
&Description="New description"
```

Eliminar un rol vinculado a servicios para Amazon ElastiCache

Si ya no necesita utilizar una característica o servicio que requiere un rol vinculado a un servicio, recomendamos que elimine dicho rol. De esta forma no tiene una entidad no utilizada que no se monitoree ni mantenga de forma activa. Sin embargo, debe limpiar el rol vinculado al servicio antes de eliminarlo.

Amazon ElastiCache no elimina de forma automática el rol vinculado a servicios.

Limpiar un rol vinculado a un servicio

Antes de que pueda utilizar IAM para eliminar un rol vinculado a servicios, primero confirme que el rol no tiene recursos (clústeres o grupos de reproducción) asociados a él.

Para comprobar si el rol vinculado a un servicio tiene una sesión activa en la consola de IAM

1. Inicie sesión en la AWS Management Console y abra la consola de IAM en <https://console.aws.amazon.com/iam/>.
2. En el panel de navegación de la consola de IAM, elija Roles. A continuación, elija el nombre (no la casilla de verificación) del rol `AWSServiceRoleForElastiCache`.

3. En la página Resumen del rol seleccionado, elija la pestaña Asesor de acceso.
4. En la pestaña Asesor de acceso, revise la actividad reciente del rol vinculado a servicios.

Para eliminar recursos de Amazon ElastiCache que requieren AWSServiceRoleForelastiCache

- Para eliminar un clúster, consulte los siguientes temas:
 - [Usando el AWS Management Console](#)
 - [Usando el AWS CLI](#)
 - [Uso del ElastiCache API](#)
- Para eliminar un grupo de reproducción, consulte los siguientes temas:
 - [Eliminación de un grupo de reproducción \(consola\)](#)
 - [Eliminación de un grupo de reproducción de \(AWS CLI\)](#)
 - [Eliminar un grupo de replicación \(ElastiCache API\)](#)

Eliminación de un rol vinculado a servicios (consola de IAM)

Puede utilizar la consola de IAM para eliminar un rol vinculado a un servicio.

Para eliminar un rol vinculado a un servicio (consola)

1. Inicie sesión en la AWS Management Console y abra la consola de IAM en <https://console.aws.amazon.com/iam/>.
2. En el panel de navegación de la consola de IAM, elija Roles. A continuación, seleccione la casilla junto al nombre del rol que desea eliminar, no el nombre ni la fila.
3. En Role actions (Acciones de rol) en la parte superior de la página, elija Delete role (Eliminar rol).
4. En el cuadro de diálogo de confirmación, revise los datos del último acceso al servicio, que muestra cuándo cada uno de los roles seleccionados tuvo acceso a un servicio de AWS por última vez. Esto lo ayuda a confirmar si el rol está actualmente activo. Si desea continuar, seleccione Yes, Delete para enviar la solicitud de eliminación del rol vinculado al servicio.
5. Consulte las notificaciones de la consola de IAM para monitorear el progreso de la eliminación del rol vinculado al servicio. Como el proceso de eliminación del rol vinculado al servicio de IAM es asíncrono, dicha tarea puede realizarse correctamente o fallar después de que envía la solicitud de eliminación. Si la tarea no se realiza correctamente, puede seleccionar View

details (Ver detalles) o View Resources (Ver recursos) desde las notificaciones para obtener información sobre el motivo por el que no se pudo eliminar el rol.

Eliminación de un rol vinculado a servicios (CLI de IAM)

Puede utilizar las operaciones de IAM desde la AWS Command Line Interface para eliminar un rol vinculado a servicios.

Para eliminar un rol vinculado a un servicio (CLI)

1. Si no conoce el nombre del rol vinculado a servicios que desea eliminar, ingrese el siguiente comando. En este comando se enumeran los roles y los nombres de recursos de Amazon (ARN) de la cuenta.

```
$ aws iam get-role --role-name role-name
```

Utilice el nombre del rol, no el ARN, para hacer referencia a los roles con las operaciones de la CLI. Por ejemplo, si un rol tiene el ARN `arn:aws:iam::123456789012:role/myrole`, debe referirse a él como **myrole**.

2. Como los roles vinculados a servicios no se puede eliminar si están en uso o tienen recursos asociados, debe enviar una solicitud de eliminación. Esta solicitud puede denegarse si no se cumplen estas condiciones. Debe apuntar el valor `deletion-task-id` de la respuesta para comprobar el estado de la tarea de eliminación. Ingrese lo siguiente para enviar una solicitud de eliminación de un rol vinculado a servicios.

```
$ aws iam delete-service-linked-role --role-name role-name
```

3. Ingrese lo siguiente para verificar el estado de la tarea de eliminación.

```
$ aws iam get-service-linked-role-deletion-status --deletion-task-id deletion-task-id
```

El estado de la tarea de eliminación puede ser `NOT_STARTED`, `IN_PROGRESS`, `SUCCEEDED` o `FAILED`. Si ocurre un error durante la eliminación, la llamada devuelve el motivo del error para que pueda resolver el problema.

Eliminación de un rol vinculado a servicios (API de IAM)

Puede utilizar la API de IAM para eliminar un rol vinculado a un servicio.

Para eliminar un rol vinculado a un servicio (API)

1. Para enviar una solicitud de eliminación de un rol vinculado a un servicio, llame a [DeleteServiceLinkedRole](#). En la solicitud, especifique el nombre del rol.

Como los roles vinculados a servicios no se puede eliminar si están en uso o tienen recursos asociados, debe enviar una solicitud de eliminación. Esta solicitud puede denegarse si no se cumplen estas condiciones. Debe apuntar el valor `DeletionTaskId` de la respuesta para comprobar el estado de la tarea de eliminación.

2. Para comprobar el estado de la tarea de eliminación, realice una llamada a [GetServiceLinkedRoleDeletionStatus](#). En la solicitud, especifique el valor de `DeletionTaskId`.

El estado de la tarea de eliminación puede ser `NOT_STARTED`, `IN_PROGRESS`, `SUCCEEDED` o `FAILED`. Si ocurre un error durante la eliminación, la llamada devuelve el motivo del error para que pueda resolver el problema.

ElastiCache Permisos de API: referencia de acciones, recursos y condiciones

Cuando configure el [control de acceso](#) y escriba políticas de permisos para adjuntar a una política de IAM (políticas basadas en identidad o recurso), utilice la siguiente tabla como referencia. En la tabla se muestra cada operación de la ElastiCache API de Amazon y las acciones correspondientes para las que puedes conceder permisos para realizar la acción. Las acciones se especifican en el campo `Action` de la política y el valor de un recurso se especifica en el campo `Resource` de la política. A menos que se indique lo contrario, el recurso es necesario. Algunos campos incluyen recursos obligatorios y opcionales. Cuando no hay ARN de recurso, el recurso de la política es un comodín (*).

Puedes usar claves de condición en tus ElastiCache políticas para expresar las condiciones. Para ver una lista de claves ElastiCache de condición específicas, junto con las acciones y los tipos de recursos a los que se aplican, consulte [Uso de claves de condición](#). Para obtener una lista completa de las claves AWS generales, consulte las claves de [contexto de condición AWS globales en la Guía del usuario de IAM](#).

Note

Para especificar una acción, use el prefijo `elasticache:` seguido del nombre de operación de la API (por ejemplo, `elasticache:DescribeCacheClusters`).

Para ver una lista de ElastiCache acciones, consulta [Acciones definidas por Amazon ElastiCache](#) en la Referencia de autorización de servicio.

Validación de conformidad para Amazon ElastiCache

Los auditores externos evalúan la seguridad y el cumplimiento de AWS los servicios como parte de varios programas de AWS cumplimientoSOC, comoPCI, la Fed RAMP yHIPAA.

Para saber si un [programa de cumplimiento Servicio de AWS está dentro del ámbito de aplicación de programas de cumplimiento específicos](#), consulte [Servicios de AWS Alcance by Compliance Servicios de AWS](#) y elija el programa de cumplimiento que le interese. Para obtener información general, consulte Programas de [AWS cumplimiento > Programas AWS](#) .

Puede descargar informes de auditoría de terceros utilizando AWS Artifact. Para obtener más información, consulte [Descarga de informes en AWS Artifact](#) .

Su responsabilidad de cumplimiento al Servicios de AWS utilizarlos viene determinada por la confidencialidad de sus datos, los objetivos de cumplimiento de su empresa y las leyes y reglamentos aplicables. AWS proporciona los siguientes recursos para ayudar con el cumplimiento:

- [Guías de inicio rápido sobre seguridad y cumplimiento](#): estas guías de implementación analizan las consideraciones arquitectónicas y proporcionan los pasos para implementar entornos básicos centrados en AWS la seguridad y el cumplimiento.
- [Diseñando una arquitectura basada en la HIPAA seguridad y el cumplimiento en Amazon Web Services](#): en este documento técnico se describe cómo pueden utilizar las empresas AWS para crear HIPAA aplicaciones aptas.

 Note

No todos son aptos. Servicios de AWS HIPAA Para obtener más información, consulta la [Referencia de servicios HIPAA aptos](#).

- [AWS Recursos](#) de de cumplimiento: esta colección de libros de trabajo y guías puede aplicarse a su industria y ubicación.
- [AWS Guías de cumplimiento para clientes](#): comprenda el modelo de responsabilidad compartida desde el punto de vista del cumplimiento. En las guías se resumen las mejores prácticas para garantizar la seguridad Servicios de AWS y se orientan a los controles de seguridad en varios marcos (incluidos el Instituto Nacional de Estándares y Tecnología (NIST), el Consejo de Normas de Seguridad de la Industria de Tarjetas de Pago (PCI) y la Organización Internacional de Normalización (ISO)).
- [Evaluación de los recursos con reglas](#) en la guía para AWS Config desarrolladores: el AWS Config servicio evalúa en qué medida las configuraciones de los recursos cumplen con las prácticas internas, las directrices del sector y las normas.
- [AWS Security Hub](#)— Este Servicio de AWS proporciona una visión completa del estado de su seguridad interior AWS. Security Hub utiliza controles de seguridad para evaluar sus recursos de AWS y comprobar su cumplimiento con los estándares y las prácticas recomendadas del sector de la seguridad. Para obtener una lista de los servicios y controles compatibles, consulte la [Referencia de controles de Security Hub](#).
- [Amazon GuardDuty](#): Servicio de AWS detecta posibles amenazas para sus cargas de trabajo Cuentas de AWS, contenedores y datos mediante la supervisión de su entorno para detectar actividades sospechosas y maliciosas. GuardDuty puede ayudarlo a cumplir con varios requisitos

de conformidad, por ejemplo PCIDSS, cumpliendo con los requisitos de detección de intrusiones exigidos por ciertos marcos de cumplimiento.

- [AWS Audit Manager](#)— Esto le Servicio de AWS ayuda a auditar continuamente su AWS consumo para simplificar la gestión del riesgo y el cumplimiento de las normativas y los estándares del sector.

Más información

Para obtener información general sobre AWS la conformidad con la nube, consulte lo siguiente:

- [FIPSPuntos finales por servicio](#)
- [Actualizaciones de servicios en ElastiCache](#)
- [AWS Conformidad con la nube](#)
- [Modelo de responsabilidad compartida](#)
- [AWS PCIDSSPrograma de cumplimiento](#)

Resiliencia en Amazon ElastiCache

La infraestructura global de AWS se compone de regiones y zonas de disponibilidad de AWS. AWS Las regiones proporcionan varias zonas de disponibilidad físicamente independientes y aisladas que se encuentran conectadas mediante redes con un alto nivel de rendimiento y redundancia, además de baja latencia. Con las zonas de disponibilidad, puede diseñar y utilizar aplicaciones y bases de datos que realizan una conmutación por error automática entre zonas de disponibilidad sin interrupciones. Las zonas de disponibilidad tienen una mayor disponibilidad, tolerancia a errores y escalabilidad que las infraestructuras tradicionales de centros de datos únicos o múltiples.

Para obtener más información sobre las regiones y zonas de disponibilidad de AWS, consulte [Infraestructura global de AWS](#).

Además de la infraestructura global de AWS, Amazon ElastiCache ofrece varias características que ayudan con las necesidades de resiliencia y copia de seguridad de los datos.

Temas

- [Mitigación de errores](#)

Mitigación de errores

Cuando planifique ElastiCache la implementación de Amazon, debe planificar de manera que los errores tengan un impacto mínimo en su aplicación y sus datos. Los temas de esta sección abordan enfoques que puede aplicar para proteger la aplicación y los datos frente a errores.

Temas

- [Mitigar los errores al ejecutar Redis OSS](#)
- [Recomendaciones](#)

Mitigar los errores al ejecutar Redis OSS

Al ejecutar el motor OSS de Redis, dispone de las siguientes opciones para minimizar el impacto de un fallo en un nodo o en una zona de disponibilidad.

Mitigación de errores de nodos

Las cachés sin servidor mitigan automáticamente los errores de los nodos con una arquitectura de varias zonas de disponibilidad a fin de que los errores de los nodos sean transparentes para su aplicación. Los clústeres de autodiseño deben configurarse adecuadamente para mitigar el fallo de un nodo individual.

Para mitigar el impacto de los fallos de los nodos del OSS de Redis en los clústeres de diseño propio, dispone de las siguientes opciones:

Temas

- [Mitigación de errores: grupos de replicación de Redis OSS](#)

Mitigación de errores: grupos de replicación de Redis OSS

Un grupo de replicación de Redis OSS se compone de un único nodo principal desde el que la aplicación puede leer y escribir, y de 1 a 5 nodos de réplica de solo lectura. Cuando se escriben datos en el nodo principal, también se actualizan de forma asíncrona en los nodos de réplica de lectura.

Qué sucede en caso de error en una réplica de lectura

1. ElastiCache detecta la réplica de lectura fallida.
2. ElastiCache desconecta el nodo fallido.

3. ElastiCache lanza y aprovisiona un nodo de reemplazo en la misma zona de disponibilidad.
4. El nuevo nodo se sincroniza con el nodo principal.

Durante este tiempo, la aplicación podrá seguir realizando operaciones de lectura y escritura con los demás nodos.

Redis OSS Multi-AZ

Puede habilitar Multi-AZ en sus grupos de replicación de Redis OSS. Independientemente de que habilite Multi-AZ o no, se detectará y reemplazará automáticamente un error en el nodo principal. El modo en que esto tiene lugar varía en función de si Multi-AZ está habilitado o no.

Cuando Multi-AZ está habilitado

1. ElastiCache detecta la falla del nodo principal.
2. ElastiCache promueve el nodo de réplica de lectura con el menor retraso de replicación con respecto al nodo principal.
3. El resto de réplicas se sincronizarán con el nuevo nodo principal.
4. ElastiCache genera una réplica de lectura en la zona de disponibilidad del dispositivo primario que ha fallado.
5. El nuevo nodo se sincroniza con el nodo principal recién promovido.

La conmutación por error a un nodo de réplica suele ser más rápida que la creación y el aprovisionamiento de un nuevo nodo principal. Esto significa que la aplicación podrá reanudar la escritura en el nodo principal antes que si Multi-AZ no estuviera habilitado.

Para obtener más información, consulte [Minimizar el tiempo de inactividad en ElastiCache \(Redis OSS\) con Multi-AZ](#).

Cuando Multi-AZ está deshabilitado

1. ElastiCache detecta un fallo principal.
2. ElastiCache desconecta el principal.
3. ElastiCache crea y aprovisiona un nuevo nodo principal para reemplazar el nodo principal defectuoso.
4. ElastiCache sincroniza el nuevo primario con una de las réplicas existentes.
5. Cuando finaliza la sincronización, el nuevo nodo funciona como el nodo principal del clúster.

Durante los pasos 1 a 4 de este proceso, la aplicación no puede escribir en el nodo principal. Sin embargo, la aplicación podrá seguir leyendo datos de los nodos de réplica.

Para mejorar la protección, le recomendamos que lance los nodos del grupo de reproducción en distintas zonas de disponibilidad (AZ). De este modo, los errores en zonas de disponibilidad solo afectarán a los nodos de dichas zonas de disponibilidad, no a los demás.

Para obtener más información, consulte [Alta disponibilidad a través de grupos de reproducción](#).

Mitigación de errores de zona de disponibilidad

Las cachés sin servidor mitigan automáticamente los errores de las zonas de disponibilidad con una arquitectura replicada de varias zonas de disponibilidad a fin de que los errores de estas zonas sean transparentes para su aplicación.

Para mitigar el impacto de los errores de una zona de disponibilidad en un clúster de autodiseño, busque los nodos de cada partición en tantas zonas de disponibilidad como sea posible.

Independientemente de la cantidad de nodos que tenga en una partición, si se encuentran en la misma zona de disponibilidad, un error catastrófico en dicha zona tendría como resultado la pérdida de todos los datos de la partición. Sin embargo, si ubica los nodos en varias zonas de disponibilidad, un error en cualquiera de las zonas de disponibilidad tendría como consecuencia la pérdida solo de los nodos de dicha zona de disponibilidad.

Cada vez que se pierde un nodo, puede experimentar una reducción del rendimiento, ya que las operaciones de lectura son compartidas por menos nodos. Esta reducción del rendimiento continuará hasta que los nodos se reemplacen.

Para obtener información sobre cómo especificar las zonas de disponibilidad para los nodos OSS de Redis, consulte [Creación de un clúster de Redis OSS \(modo de clúster desactivado\) \(consola\)](#)

Para obtener más información acerca de las regiones y zonas de disponibilidad, consulte [Elección de regiones y zonas de disponibilidad](#).

Recomendaciones

Es recomendable crear cachés sin servidor en clústeres de autodiseño, ya que obtendrá automáticamente una mejor tolerancia a errores sin necesidad de configuración adicional. Sin embargo, al crear un clúster de autodiseño, hay dos tipos de errores para los que debe estar preparado: los errores de nodos individuales y los errores generalizados en zonas de disponibilidad. Los mejores planes de mitigación de errores abordarán ambos tipos de errores.

Minimización del impacto de los errores de nodos

Para minimizar el impacto de un error en un nodo, recomendamos que la implementación utilice varios nodos en cada partición y que distribuya los nodos en varias zonas de disponibilidad. Esto se hace automáticamente en las cachés sin servidor.

En el caso de los clústeres de diseño propio, le recomendamos que habilite la opción Multi-AZ en su grupo de replicación para que ElastiCache la conmutación por error automática a una réplica en caso de que se produzca un error en el nodo principal.

Minimización del impacto de los errores en la zona de disponibilidad

Para minimizar el impacto de un error en una zona de disponibilidad, recomendamos lanzar los nodos en tantas zonas de disponibilidad distintas como sea posible. Distribuir los nodos de manera regular en zonas de disponibilidad minimizará el impacto en caso de un improbable evento de error de zona de disponibilidad. Esto se hace automáticamente en las cachés sin servidor.

Otras precauciones

Si utiliza Redis OSS, además de lo anterior, le recomendamos que programe copias de seguridad periódicas del clúster. Las copias de seguridad (instantáneas) crean un archivo .rdb que podrá utilizar para restablecer la caché en caso de error o daño. Para obtener más información, consulte [Instantánea y restauración](#).

Seguridad de la infraestructura en ElastiCache de AWS

Como servicio administrado, ElastiCache de AWS se encuentra protegido por los procedimientos de seguridad de red globales de AWS que se describen en la sección de Seguridad y conformidad, en el [Centro de arquitectura de AWS](#).

Puede utilizar llamadas a la API publicadas en AWS para obtener acceso a ElastiCache a través de la red. Los clientes deben ser compatibles con Transport Layer Security (TLS) 1.2 o una versión posterior. Recomendamos TLS 1.3 o una versión posterior. Los clientes también deben ser compatibles con conjuntos de cifrado con confidencialidad directa total (PFS) tales como Ephemeral Diffie-Hellman (DHE) o Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). La mayoría de los sistemas modernos como Java 7 y posteriores son compatibles con estos modos.

Además, las solicitudes deben estar firmadas mediante un ID de clave de acceso y una clave de acceso secreta que esté asociada a una entidad principal de IAM. También puede utilizar [AWS](#)

[Security Token Service](#) (AWS STS) para generar credenciales de seguridad temporales para firmar solicitudes.

Actualizaciones de servicios en ElastiCache

ElastiCache supervisa automáticamente su flota de cachés, clústeres y nodos para aplicar las actualizaciones de servicio a medida que estén disponibles. Las actualizaciones de servicio para las cachés sin servidor se aplican de forma transparente y automática. En el caso de los clústeres de diseño propio, debe configurar un período de mantenimiento predefinido para ElastiCache poder aplicar estas actualizaciones. Sin embargo, en algunos casos es posible que este enfoque resulte demasiado rígido y que probablemente restrinja los flujos de negocio.

Con las actualizaciones de servicio, usted controla qué actualizaciones se implementan en sus clústeres de autodiseño y cuándo se aplican. También puede supervisar el progreso de estas actualizaciones en el ElastiCache clúster seleccionado en tiempo real.

Administrar las actualizaciones de los servicios

ElastiCache Las actualizaciones de servicio para los clústeres de diseño propio se publican de forma periódica. Si tiene uno o más clústeres autodiseñados aptos para esas actualizaciones del servicio, recibirá notificaciones por correo electrónico SNS, el Personal Health Dashboard (PHD) y CloudWatch los eventos de Amazon cuando se publiquen las actualizaciones. Las actualizaciones también se muestran en la página de actualizaciones del servicio de la ElastiCache consola. Al usar este panel, puede ver todas las actualizaciones del servicio y su estado para su ElastiCache flota. Las actualizaciones de servicio para las cachés sin servidor se aplican de forma transparente y no se pueden administrar mediante actualizaciones de servicio.

Puede controlar cuándo se debe aplicar una actualización antes de que se inicie una actualización automática. Le recomendamos encarecidamente que aplique cualquier actualización del tipo de actualización de seguridad lo antes posible para garantizar que sus ElastiCache clústeres cuenten siempre up-to-date con los parches de seguridad actuales.

En las siguientes secciones se describen detalladamente las opciones.

Temas

- [Aplicación de las actualizaciones de servicio](#)
- [Comprobar que se ha aplicado la última actualización de servicio mediante la consola AWS](#)
- [Detención de las actualizaciones de servicio](#)

Aplicación de las actualizaciones de servicio

Puede comenzar a aplicar las actualizaciones del servicio a la flota Redis desde el momento en que las actualizaciones tengan el estado available (disponible). Las actualizaciones del servicio son acumulativas. Es decir, todas las actualizaciones que no se hayan aplicado se incluirán con la última actualización.

Si una actualización de servicio tiene habilitada la actualización automática, puedes optar por no realizar ninguna acción cuando esté disponible. ElastiCache programará la aplicación de la actualización durante uno de los próximos períodos de mantenimiento de sus clústeres después de la fecha de inicio de la actualización automática. Recibirá notificaciones relacionadas con cada etapa de la actualización.

Note

Solo puede aplicar las actualizaciones de servicio que tengan un estado disponible o programado.

Para obtener más información sobre cómo revisar y aplicar cualquier actualización específica del servicio a los clústeres aplicables, consulte. ElastiCache [Aplicación de las actualizaciones de servicio con la consola](#)

Cuando haya una nueva actualización de servicio disponible para uno o más de sus ElastiCache clústeres, puede utilizar la ElastiCache consola o AWS CLI aplicar la actualización. API En las siguientes secciones se explican las opciones que puede utilizar para aplicar las actualizaciones.

Aplicación de las actualizaciones de servicio con la consola

Para consultar la lista de las distintas actualizaciones de servicio disponibles, junto con otra información, vaya a Service Updates (Actualizaciones de servicio) en la consola.

1. Inicia sesión en la ElastiCache consola de Amazon AWS Management Console y ábrela en <https://console.aws.amazon.com/elasticache/>.
2. En el panel de navegación, seleccione Service Updates (Actualizaciones de servicio).
3. En Service Updates (Actualizaciones de servicio) puede consultar lo siguiente:
 - Service update name (Nombre de actualización de servicio): el nombre único de la actualización de servicio

- **Update type (Tipo de actualización):** el tipo de actualización de servicio, que es o security-update o engine-update
 - **Update Severity (Gravedad de la actualización):** la prioridad de aplicación de la actualización:
 - **critical (crítica):** recomendamos que aplique esta actualización inmediatamente (en un plazo de 14 días o menos).
 - **important (importante):** recomendamos que aplique esta actualización tan pronto como el flujo de negocio lo permita (en un plazo de 30 días o menos).
 - **medium (media):** recomendamos que aplique esta actualización tan pronto como pueda (en un plazo de 60 días o menos).
 - **low (baja):** recomendamos que aplique esta actualización tan pronto como pueda (en un plazo de 90 días o menos).
 - **Engine version (Versión del motor):** si el tipo de actualización es actualización del motor, la versión del motor que se está actualizando.
 - **Fecha de lanzamiento:** cuándo se publica la versión y cuándo está disponible para su aplicación en los clústeres.
 - **Fecha de caducidad recomendada:** fecha ElastiCache orientativa para aplicar las actualizaciones.
 - **Status (Estado):** el estado de la actualización, que es uno de los siguientes:
 - **disponible:** la aplicación está lista para los clústeres correspondientes.
 - **complete (completa):** la actualización se ha aplicado correctamente.
 - **cancelled (cancelada):** la actualización se ha cancelado y ya no es necesaria.
 - **expired (caducada):** la actualización ya no está disponible para ser aplicada.
4. Elija una actualización individual (no el botón de la izquierda) para ver los detalles de la actualización del servicio.

En la sección Cluster update status (Estado de actualización del clúster), puede ver una lista de clústeres en los que la actualización del servicio no se ha aplicado o se ha aplicado recientemente. Para cada clúster, puede ver lo siguiente:

- **Cluster name (Nombre del clúster):** el nombre del clúster
- **Nodes Updated (Nodos actualizados):** la proporción de nodos en un clúster específico que se actualizaron o que permanecen disponibles para la actualización del servicio específica.
- **Update Type (Tipo de actualización):** el tipo de actualización de servicio, que es security-

- **Status (Estado):** el estado de la actualización de servicio en el clúster, que es uno de los siguientes:
 - **available (disponible):** la aplicación está lista para los clústeres Redis correspondientes.
 - **en progreso:** la actualización se está aplicando a este clúster.
 - **scheduled (programado):** se ha programado la fecha de actualización.
 - **complete (completa):** la actualización se ha aplicado correctamente. El clúster con el estado completo se mostrará durante 7 días después de su finalización.

Si ha elegido alguno o todos los clústeres con estado **available (disponible)** o **scheduled (programado)** y, luego, eligió **Apply now (Postúlese ahora)**, la actualización empezará a aplicarse en esos clústeres.

Aplicación de las actualizaciones de servicio con la AWS CLI

Tras recibir una notificación de que hay actualizaciones del servicio disponibles, puede inspeccionarlas y aplicarlas con AWS CLI:

- Para recuperar una descripción de las actualizaciones de servicio disponibles, ejecute el siguiente comando:

```
aws elasticache describe-service-updates --service-update-status
available
```

Para obtener más información, consulte [describe-service-updates](#).

- Para aplicar una actualización de servicio en una lista de clústeres, ejecute el siguiente comando:

```
aws elasticache batch-apply-update-action --service-update
ServiceUpdateNameToApply=sample-service-update --cluster-names cluster-1
cluster2
```

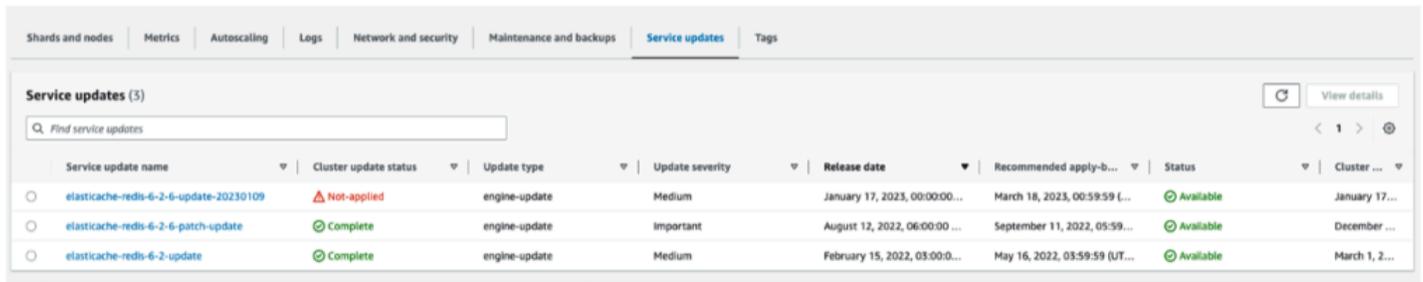
Para obtener más información, consulte [batch-apply-update-action](#).

Comprobar que se ha aplicado la última actualización de servicio mediante la consola AWS

Puede comprobar que sus clústeres ElastiCache (de RedisOSS) estén ejecutando la última actualización del servicio siguiendo estos pasos:

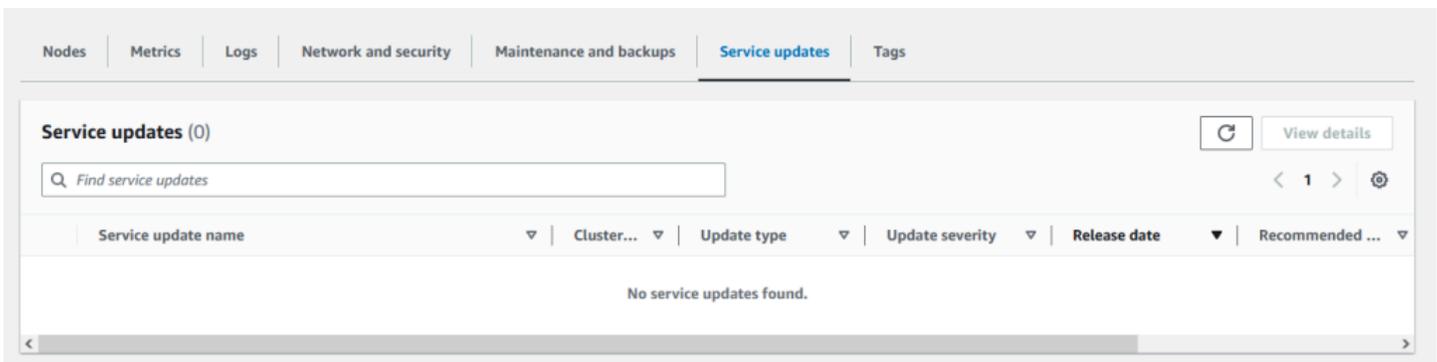
1. Elija un clúster aplicable en la página de clústeres de Redis OSS
2. Seleccione Actualizaciones de servicio en el panel de navegación para ver las actualizaciones de servicio aplicables a ese clúster, si las hubiera.

Si la consola muestra una lista de actualizaciones de servicio, puede seleccionar la actualización de servicio y elegir Aplicar ahora.



Service update name	Cluster update status	Update type	Update severity	Release date	Recommended apply-b...	Status	Cluster ...
elasticsearch-redis-6-2-6-update-20230109	Not-applied	engine-update	Medium	January 17, 2023, 00:00:00...	March 18, 2023, 00:59:59 (...)	Available	January 17...
elasticsearch-redis-6-2-6-patch-update	Complete	engine-update	Important	August 12, 2022, 06:00:00 ...	September 11, 2022, 05:59...	Available	December ...
elasticsearch-redis-6-2-update	Complete	engine-update	Medium	February 15, 2022, 03:00:0...	May 16, 2022, 05:59:59 (UT...	Available	March 1, 2...

Si la consola muestra el mensaje «No se ha encontrado ninguna actualización de servicio», significa que el clúster ElastiCache (RedisOSS) ya tiene aplicada la última actualización de servicio.



Service update name	Cluster...	Update type	Update severity	Release date	Recommended ...
No service updates found.					

Detención de las actualizaciones de servicio

Si es necesario, puede detener las actualizaciones de clústeres. Por ejemplo, es posible que desee detener las actualizaciones si tiene un aumento inesperado en los clústeres que se están actualizando. O es posible que desee detener las actualizaciones si tardan demasiado e interrumpen el flujo de un negocio en hora punta.

La operación [Stopping \(Deteniéndose\)](#) interrumpe inmediatamente todas las actualizaciones de esos clústeres y de todos los nodos que aún se deben actualizar. La operación continúa hasta completar todos los nodos que tienen el estado in progress (en curso). Sin embargo, detiene las actualizaciones de otros nodos del mismo clúster que tienen el estado update available (actualización disponible) y los cambia al estado Stopping (Deteniéndose).

Cuando se completa el flujo de trabajo Stopping (Deteniéndose), los nodos que tiene el estado Stopping (Deteniéndose) cambian al estado Stopped (Detenido). En función del flujo de trabajo de la actualización, algunos clústeres no tendrán todos los nodos actualizados. Otros clústeres podrían incluir algunos nodos que estén actualizados y otros nodos que aún tengan el estado update available (actualización disponible).

Puede terminar más adelante el proceso de actualización cuando el flujo de trabajo lo permita. En tal caso, elija los clústeres en los que desee completar las actualizaciones y, a continuación, elija Apply Now (Aplicar ahora). Para obtener más información, consulte [Aplicación de las actualizaciones de servicio](#).

Mediante la consola

Puede interrumpir una actualización de servicio mediante la ElastiCache consola. A continuación se indica cómo hacerlo:

- Una vez que se haya realizado una actualización de servicio en un clúster seleccionado, la ElastiCache consola mostrará la pestaña Ver/detener la actualización en la parte superior del panel de control. ElastiCache
- Para interrumpir la actualización, elija Stop Update (Detener actualización).
- Cuando se detenga la actualización, elija el clúster y examine el estado. Este cambia al estado Deteniéndose y al final al estado Detenido.

Mediante el AWS CLI

Puede interrumpir una actualización del servicio con AWS CLI. El siguiente ejemplo de código muestra cómo hacerlo.

Para un grupo de réplica, haga lo siguiente:

```
aws elasticache batch-stop-update-action --service-update-name sample-service-update --replication-group-ids my-replication-group-1 my-replication-group-2
```

Para un clúster de caché, haga lo siguiente:

```
aws elasticache batch-stop-update-action --service-update-name sample-service-update --cache-cluster-ids my-cache-cluster-1 my-cache-cluster-2
```

Para obtener más información, consulte [BatchStopUpdateAction](#).

Vulnerabilidades y exposiciones comunes (CVE): vulnerabilidades de seguridad abordadas en ElastiCache (Redis OSS)

Vulnerabilidades y exposiciones comunes (CVE) es una lista de entradas de vulnerabilidades de ciberseguridad conocidas públicamente. Cada entrada es un enlace que contiene un número de identificación, una descripción y al menos una referencia pública. En esta página encontrará una lista de las vulnerabilidades de seguridad que se han abordado en ElastiCache (Redis OSS).

Le recomendamos que actualice siempre a la versión más reciente ElastiCache (Redis OSS) para protegerse contra las vulnerabilidades conocidas. Al utilizar una caché ElastiCache sin servidor, las correcciones de CVE se aplican automáticamente a la memoria caché. Al operar clústeres de diseño propio, ElastiCache (Redis OSS) expone el componente PATCH. Por ejemplo, cuando se utiliza la versión 6.2.6 ElastiCache (Redis OSS), la versión principal es la 6, la versión secundaria es la 2 y la versión del parche es la 6. Las versiones PATCH son para correcciones de errores compatibles con versiones anteriores, correcciones de seguridad y cambios no funcionales.

Puede utilizar esta página para comprobar si una versión concreta de ElastiCache (Redis OSS) tiene una solución para una vulnerabilidad de seguridad específica. Si su clúster ElastiCache (Redis OSS) ejecuta una versión sin la corrección de seguridad, consulte la tabla siguiente y tome medidas. Puede actualizar a una versión más reciente ElastiCache (Redis OSS) que contenga la corrección o, si utiliza una versión ElastiCache (Redis OSS) que contenga la corrección, asegúrese de haber aplicado la última actualización de servicio consultando [Administrar las actualizaciones de los servicios](#). Para obtener más información sobre las versiones del motor ElastiCache (Redis OSS) compatibles y sobre cómo actualizarlas, consulte [Versiones del motor y actualizaciones](#).

Note

- Si un CVE está direccionado en una versión ElastiCache (Redis OSS), significa que también está direccionado en las versiones más recientes. Así, por ejemplo, si se corrige una vulnerabilidad en la versión 6.0.5 ElastiCache (Redis OSS), esta situación continúa en las versiones 6.2.6, 7.0.7 y 7.1.
- Un asterisco (*) en la siguiente tabla indica que debe haber aplicado la última actualización de servicio al clúster ElastiCache (Redis OSS) que ejecute la versión ElastiCache (Redis OSS) especificada para solucionar la vulnerabilidad de seguridad. Para obtener más información sobre cómo comprobar si se ha aplicado la última actualización de servicio a la

versión ElastiCache (Redis OSS) en la que se ejecuta el clúster, consulte. [Administrar las actualizaciones de los servicios](#)

ElastiCache Versión (Redis OSS)	CVE abordados
Redis OSS 6.0.5	CVE-202224735 *, CVE-202224736 *
Redis OSS 6.2.6	CVE-202224834 *, CVE-202235977 *, CVE-202236021 *, CVE-202224736
Redis OSS 7.0.7	CVE-2023-41056 *, CVE-202224834 *, CVE-202235977 , CVE-202236021 , CVE-202224736
Redis OSS 7.1.0	CVE-2023-41056 , CVE-202224834 , CVE-202235977 , CVE-202236021 , CVE-20224736

Registro y Supervisión en Amazon ElastiCache

Para administrar la memoria caché, es importante que conozca su rendimiento. ElastiCache genera métricas que se publican en Registros de Amazon CloudWatch para supervisar el rendimiento de la caché. Además, ElastiCache genera eventos cuando se producen cambios importantes en los recursos de la caché (por ejemplo, se crea o se elimina una caché).

Temas

- [Eventos y métricas sin servidor](#)
- [Eventos y métricas de clústeres de autodiseño](#)
- [Registro de llamadas a la API de Amazon ElastiCache con AWS CloudTrail](#)

Eventos y métricas sin servidor

En esta sección, se describen las métricas y los eventos que se deben supervisar al trabajar con cachés sin servidor.

Temas

- [Métricas de caché sin servidor](#)
- [Eventos de caché sin servidor](#)

Métricas de caché sin servidor

El espacio de AWS/ElastiCache nombres incluye las siguientes CloudWatch métricas para las cachés sin servidor de Redis OSS.

Métrica	Descripción	Unidad
BytesUsedForCache	El número total de bytes utilizados por los datos almacenados en la memoria caché.	Bytes
ElastiCacheProcessingUnits	El número total de ElastiCacheProcessingUnits (ECPUs)	Recuento

Métrica	Descripción	Unidad
	consumidas por las solicitudes ejecutadas en la memoria caché	
SuccessfulReadRequestLatency	Latencia de las solicitudes de lectura correctas.	Microsegundos
SuccessfulWriteRequestLatency	Latencia de las solicitudes de escritura correctas	Microsegundos
TotalCmdsCount	Recuento total de todos los comandos ejecutados en la memoria caché	Recuento
CacheHitRate	Indica la tasa de aciertos de la memoria caché. Esto se calcula utilizando estadísticas <code>cache_hits</code> y <code>cache_misses</code> de la siguiente manera: $\text{cache_hits} / (\text{cache_hits} + \text{cache_misses})$.	Porcentaje
CacheHits	El número de búsquedas de claves solo de lectura realizadas correctamente en la caché.	Recuento
CurrConnections	Número de conexiones de cliente en su caché.	Recuento
ThrottledCmds	El número de solicitudes que se vieron limitadas ElastiCache debido a que la carga de trabajo se ampliaba más rápido de lo que ElastiCache podía escalar.	Recuento

Métrica	Descripción	Unidad
NewConnections	El número total de conexiones que ha aceptado el servidor durante este periodo.	Recuento
Currltems	El número de elementos en la caché.	Recuento
CurrVolatileItems	El número de elementos en la caché con TTL.	Recuento
NetworkBytesIn	Total de bytes transferidos a la memoria caché	Bytes
NetworkBytesOut	Total de bytes transferidos a la memoria caché	Bytes
Evictions	El recuento de claves expulsadas por la caché	Recuento
IamAuthenticationExpirations	El número total de conexiones Redis OSS autenticadas por IAM que han caducado. Puede encontrar más información sobre Autenticación con IAM en la guía del usuario.	Recuento
IamAuthenticationThrottling	El número total de solicitudes de autenticación o HELLO de Redis OSS autenticadas por IAM restringidas. Puede encontrar más información sobre Autenticación con IAM en la guía del usuario.	Recuento

Métrica	Descripción	Unidad
KeyAuthorizationFailures	Número total de intentos fallidos de los usuarios de acceder a claves a las que no tienen permiso para acceder. Sugerimos configurar una alarma para detectar intentos de acceso sin autorización.	Recuento
AuthenticationFailures	El número total de intentos fallidos de autenticación en Redis OSS mediante el comando AUTH. Sugerimos configurar una alarma para detectar intentos de acceso sin autorización.	Recuento
CommandAuthorizationFailures	Número total de intentos fallidos de los usuarios de ejecutar comandos a los que no tienen permiso para llamar. Sugerimos configurar una alarma para detectar intentos de acceso sin autorización.	Recuento

Métricas en el nivel de comando

ElastiCache también emite las siguientes métricas a nivel de comando. Para cada tipo de comando, ElastiCache emite el recuento total de comandos y el número de ECPUs consumidas por ese tipo de comando.

Métrica	Descripción	Unidad
EvalBasedCmds	El número de comandos get recibidos por la caché.	Recuento

Métrica	Descripción	Unidad
EvalBasedCmdsECPUs	ECPU consumidas por comandos basados en eval.	Recuento
GeoSpatialBasedCmds	Número total de comandos para comandos basados en condiciones geoespaciales. Esto se deriva de la estadística de estadísticas de comandos de Redis OSS. Se obtiene al sumar todos los tipos de comandos geográficos: geoadd, geodist, geohash, geopos, georadius y georadius bymember.	Recuento
GeoSpatialBasedCmdsECPUs	ECPU consumidas por comandos geoespaciales.	Recuento
GetTypeCmds	El número total de comandos de solo lectura. Se deriva de la estadística commandstats de Redis OSS al sumar todos los comandos de tipo de solo lectura (get, hget, scard, lrange, etc.)	Recuento
GetTypeCmdsECPUs	Las ECPU consumidas por los comandos de lectura.	Recuento

Métrica	Descripción	Unidad
HashBasedCmds	El número total de comandos basados en hash. Se deriva de la estadística <code>commandstats</code> de Redis OSS al sumar todos los comandos que actúan sobre uno o más hashes (<code>hget</code> , <code>hkeys</code> , <code>hvals</code> , <code>hdel</code> , etc.).	Recuento
HashBasedCmdsECPUs	ECPU consumidas por comandos basados en hash.	Recuento
HyperLogLogBasedCmds	El número total de comandos HyperLogLog basados. Se deriva de la estadística de estadísticas de comandos de Redis OSS al sumar todos los comandos de tipo <code>pf</code> (<code>pfadd</code> , <code>pfcount</code> , <code>pfmerge</code> , etc.).	Recuento
HyperLogLogBasedCmdsECPUs	ECPUs consumidas por los comandos HyperLogLog basados en comandos.	Recuento
JsonBasedCmds	El número total de comandos JSON, incluidos los comandos de lectura y escritura. Esto se deriva de la estadística de estadísticas de comandos de Redis OSS al sumar todos los comandos JSON que actúan sobre las claves JSON.	Recuento

Métrica	Descripción	Unidad
JsonBasedCmdsECPUs	ECPU consumidas por todos los comandos JSON, incluidos los comandos de lectura y escritura.	Recuento
JsonBasedGetCmds	Número total de comandos de solo lectura de JSON. Se deriva de la estadística de estadísticas de comandos de Redis OSS al sumar todos los comandos de lectura de JSON que actúan sobre las claves de JSON.	Recuento
JsonBasedGetCmdsECPUs	ECPU consumidas por los comandos de solo lectura de JSON.	Recuento
JsonBasedSetCmds	Número total de comandos de escritura de JSON. Se deriva de la estadística de estadísticas de comandos de Redis OSS al sumar todos los comandos de escritura de JSON que actúan sobre las claves JSON.	Recuento
JsonBasedSetCmdsECPUs	ECPU consumidas por los comandos de escritura de JSON.	Recuento

Métrica	Descripción	Unidad
KeyBasedCmds	El número total de comandos basados en claves. Se deriva de la estadística <code>commandstats</code> de Redis OSS y suma todos los comandos que actúan sobre una o más claves en varias estructuras de datos (<code>del</code> , <code>expire</code> , <code>rename</code> , etc.).	Recuento
KeyBasedCmdsECPUs	ECPU consumidas por comandos basados en claves.	Recuento
ListBasedCmds	El número total de comandos basados en listas. Se deriva de la estadística de estadísticas de comandos de Redis OSS al sumar todos los comandos que actúan en una o más listas (<code>lindex</code> , <code>lrange</code> , <code>lpush</code> , <code>ltrim</code> , etc.).	Recuento
ListBasedCmdsECPUs	ECPU consumidas por comandos basados en listas.	Recuento
NonKeyTypeCmds	Número total de comandos no basados en claves. Se deriva de la estadística <code>commandstats</code> de Redis OSS al sumar todos los comandos que no actúan sobre una tecla, por ejemplo, <code>acl</code> , <code>dbsize</code> o <code>info</code> .	Recuento
NonKeyTypeCmdsECPUs	ECPUs consumidas por non-key-based los comandos.	Recuento

Métrica	Descripción	Unidad
PubSubBasedCmds	El número total de comandos para la funcionalidad publicación/suscripción. Se deriva de las estadísticas de comandos del OSS de Redis y suma todos los comandos utilizados para la funcionalidad pub/sub: psubscribe, publish, pubsub, punsubscribe, ssubscribe, sunsubscribe, spublish, subscribe y unsubscribe.	Recuento
PubSubBasedCmdsECPUs	ECPU consumidas por comandos basados en pub/sub.	Recuento
SetBasedCmds	El número total de comandos basados en instrucciones set. Se deriva de la estadística commandstats de Redis OSS al sumar todos los comandos que actúan en uno o más conjuntos (scard, sdiff, sadd, sunion, etc.).	Recuento
SetBasedCmdsECPUs	ECPU consumidas por comandos basados en conjuntos.	Recuento

Métrica	Descripción	Unidad
SetTypeCmds	El número total de tipos de comandos de escritura. Se deriva de la estadística de estadísticas de comandos de Redis OSS al sumar todos los tipos de comandos mutativos que funcionan con los datos (set, hset, sadd, lpop, etc.)	Recuento
SetTypeCmdsCPUs	ECPU consumidas por los comandos de escritura.	Recuento
SortedSetBasedCmds	El número total de comandos basados en instrucciones set ordenadas. Se deriva de la estadística de estadísticas de comandos de Redis OSS al sumar todos los comandos que actúan sobre uno o más conjuntos ordenados (zcount, zrange, zrank, zadd, etc.).	Recuento
SortedSetBasedCmdsECPUs	ECPU consumidas por comandos basados en conjuntos ordenados.	Recuento
StringBasedCmds	El número total de comandos basados en cadenas. Se deriva de la estadística commandstats de Redis OSS al sumar todos los comandos que actúan sobre una o más cadenas (strlen, setex, setrange, etc.).	Recuento

Métrica	Descripción	Unidad
StringBasedCmdsECPUs	ECPU consumidas por comandos basados en cadenas.	Recuento
StreamBasedCmds	El número total de comandos basados en secuencias. Se deriva de la estadística commandstats de Redis OSS al sumar todos los comandos que actúan sobre uno o más tipos de datos de flujos (xrange, xlen, xadd, xdel, etc.).	Recuento
StreamBasedCmdsECPUs	ECPU consumidas por comandos basados en transmisiones.	Recuento

Eventos de caché sin servidor

ElastiCache registra los eventos relacionados con la memoria caché sin servidor. Esta información incluye la fecha y la hora del evento, el nombre del origen y el tipo del origen del evento, así como una descripción del evento. Puede recuperar fácilmente los eventos del registro mediante la ElastiCache consola, el comando AWS CLI `describe-events` o la acción de la API `ElastiCache DescribeEvents`.

Puedes elegir monitorizar, asimilar, transformar y actuar en función de ElastiCache los eventos con Amazon EventBridge. Más información en Amazon EventBridge <https://docs.aws.amazon.com/eventbridge/latest/userguide/>.

Visualización de ElastiCache eventos (consola)

Para ver los eventos mediante la ElastiCache consola:

1. Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en <https://console.aws.amazon.com/elasticache/>

2. Para ver una lista de todos los eventos disponibles, elija Events (Eventos) en el panel de navegación.
3. En la pantalla Eventos, cada fila de la lista representa un evento y muestra el origen del evento, el tipo, la hora GMT y una descripción. Con la opción Filter podrá especificar si desea ver todos los eventos o simplemente los eventos de un tipo determinado de la lista de eventos.

Visualización de ElastiCache eventos (AWS CLI)

Para generar una lista de ElastiCache eventos mediante el AWS CLI, utilice el comando `describe-events`. Puede usar parámetros opcionales para controlar el tipo de eventos que se muestran en la lista, el marco temporal de los eventos de la lista, el número máximo de eventos que se incluirán en la lista y mucho más.

El código siguiente muestra hasta 40 eventos de caché sin servidor.

```
aws elasticache describe-events --source-type serverless-cache --max-items 40
```

El código siguiente muestra todos los eventos para cachés sin servidor de las últimas 24 horas (1440 minutos).

```
aws elasticache describe-events --source-type serverless-cache --duration 1440
```

Eventos sin servidor

En esta sección, se documentan los distintos tipos de eventos que puede recibir en sus cachés sin servidor.

Eventos de creación de caché sin servidor

Tipo-detalle	Descripción	Unidad	Origen	Mensaje
Caché creada	Arn de caché	creación	caché sin servidor	La caché <code><cache-name></code> se ha creado y está lista para usarse.
Caché creada	Arn de caché	creación	caché sin servidor	La caché <code><cache-name></code>

Tipo-detalle	Descripción	Unidad	Origen	Mensaje
	Arn de la instantánea			se ha creado y los datos se han restaurado a partir de la instantánea. La caché está lista para usarse.
Error durante la creación de caché	Arn de caché	failure	caché sin servidor	No se ha podido crear la memoria caché <cache-name>. No hay suficientes direcciones IP libres para crear un punto de conexión de VPC.
Error durante la creación de caché	Arn de caché	failure	caché sin servidor	No se ha podido crear la memoria caché <cache-name>. Se proporcionaron subredes no válidas en la solicitud.

Tipo-detalle	Descripción	Unidad	Origen	Mensaje
Error durante la creación de caché	Arn de caché	failure	caché sin servidor	No se ha podido crear la memoria caché <cache-name>. Se alcanzó el límite de cuota para crear un punto de conexión de VPC.
Error durante la creación de caché	Arn de caché	failure	caché sin servidor	No se ha podido crear la memoria caché <cache-name>. No tiene permiso para crear un punto de conexión de VPC.
Error durante la creación de caché	Arn de caché	failure	caché sin servidor	No se ha podido crear la memoria caché <cache-name>. Un usuario con una versión de Redis OSS incompatible está presente en el grupo de usuarios < >. user-group-name

Tipo-detalle	Descripción	Unidad	Origen	Mensaje
Error durante la creación de caché	Arn de caché Arn de instantáneas de caché	failure	caché sin servidor	No se ha podido crear la memoria caché <cache-name>. El grupo de usuarios <user-group-name > proporcionado no existe.
Error durante la creación de caché	Arn de caché	failure	caché sin servidor	No se ha podido crear la memoria caché <cache-name>. La restauración de los datos a partir de la instantánea falló debido a <reason>. Motivos de los errores: <ul style="list-style-type: none"> error al recuperar el archivo de S3. el md5 esperado no coincide con el md5 real. el archivo RDB proporcionado tiene una versión no compatible.

Eventos de actualización de caché sin servidor

Tipo-detalle	Lista de recursos	Categoría	Origen	Mensaje
Caché actualizada	Arn de caché	configuration change	caché sin servidor	SecurityGroups actualizado para la memoria caché<cache-name>.
Caché actualizada	Arn de caché	configuration change	caché sin servidor	Etiquetas actualizadas para la caché <cache-name>.
Error durante la actualización de caché	Arn de caché	configuration change	caché sin servidor	Error en la actualización de la caché <cache-name>. Un usuario con una versión de Redis OSS incompatible está presente en el grupo de usuarios < user-group-name >.
Error durante la actualización de caché	Arn de caché	configuration change	caché sin servidor	Se produjo un <cache-name>error al actualizar la memoria caché. SecurityGroups error en la actualización.

Tipo-detalle	Lista de recursos	Categoría	Origen	Mensaje
Error durante la actualización de caché	Arn de caché	configuration change	caché sin servidor	<cache-name>Falló una actualización de la memoria caché. SecurityGroups se produjo un error en la actualización porque los permisos eran insuficientes.
Error durante la actualización de caché	Arn de caché	configuration change	caché sin servidor	Error en la actualización de la caché <cache-name>. SecurityGroups la actualización ha fallado porque no SecurityGroups son válidas.

Eventos de eliminación de caché sin servidor

Tipo-detalle	Lista de recursos	Categoría	Origen	Mensaje
Caché eliminada	Arn de caché	deletion	caché sin servidor	Se ha eliminado la caché <cache-name>.

Eventos de límite de uso de caché sin servidor

Tipo-detalle	Descripción	Unidad	Origen	Mensaje
Caché actualizada	Arn de caché	configuration change	caché sin servidor	Límites actualizados para la caché <cache-name>.
Se acerca el límite de caché	Arn de caché	notification	caché sin servidor	El slot <X> utiliza más del <Y> % del límite por slot de 32 GB. Por ejemplo, el slot 10 utiliza más del 90 % del límite por slot de 32 GB.
Error durante la actualización de caché	Arn de caché	failure	caché sin servidor	Error en la actualización del límite en la caché <cache-name>, ya que la caché ha sido eliminada.
Error durante la actualización de caché	Arn de caché	failure	caché sin servidor	Error en la actualización del límite en la caché <cache-name>, debido a una configuración no válida.
Error durante la actualización de caché	Arn de caché	failure	caché sin servidor	No se pudo actualizar el límite de la caché <cache-na

Tipo-detalle	Descripción	Unidad	Origen	Mensaje
				me> porque los datos actualmente almacenados en caché superan los nuevos límites. Vacíe algunos datos antes de aplicar los límites.

Eventos de instantáneas de caché sin servidor

Tipo-detalle	Lista-recursos	Categoría	Origen	Mensaje
Instantánea creada	Arn de caché Arn de la instantánea	creación	serverless-cache-snapshot	Instantánea <snapshot-name> creada para la caché <cache-name>.
Error en la creación de la instantánea	Arn de caché Arn de la instantánea	failure	serverless-cache-snapshot	No se ha podido crear la instantánea para la caché <cache-name>. No se ha podido crear la instantánea <snapshot-name> con la clave <key-id> administrada por el cliente <reason>.

Tipo-detalle	Lista-recursos	Categoría	Origen	Mensaje
				<p>Mensajes sobre el motivo del error:</p> <ul style="list-style-type: none">• porque la clave administrada por el cliente está desactivada• porque no se ha podido encontrar la clave administrada por el cliente• porque se ha agotado el tiempo de espera de la solicitud

Tipo-detalle	Lista-recursos	Categoría	Origen	Mensaje
Error en la creación de la instantánea	Arn de caché Arn de la instantánea	failure	serverless-cache-snapshot	<p>No se ha podido crear la instantánea para la caché <cache-name>. La creación de la instantánea <snapshot-name> ha fallado debido a <reason>.</p> <p>Motivo predeterminado:</p> <ul style="list-style-type: none"> debido a un error interno
Error en la exportación de instantánea	Arn de la instantánea	failure	serverless-cache-snapshot	<p>No se ha podido exportar la instantánea para la caché <cache-name>. No se pudo exportar la instantánea al depósito %s porque ElastiCache no tiene permisos para acceder al depósito.</p>

Tipo-detalle	Lista-recursos	Categoría	Origen	Mensaje
Error en la exportación de instantánea	Arn de la instantánea	failure	serverless-cache-snapshot	No se ha podido exportar la instantánea para la caché <cache-name>. No se ha podido exportar la instantánea al bucket '%s' porque ya hay un objeto con el mismo nombre en el bucket.
Error en la exportación de instantánea	Arn de la instantánea	failure	serverless-cache-snapshot	No se ha podido exportar la instantánea para la caché <cache-name>. No se ha podido exportar la instantánea al bucket '%s' porque el identificador de la cuenta del propietario del bucket ha cambiado.

Tipo-detalle	Lista-recursos	Categoría	Origen	Mensaje
Error en la exportación de instantánea	Arn de la instantánea	failure	serverless-cache-snapshot	No se ha podido exportar la instantánea para la caché <cache-name>. No se ha podido exportar la instantánea al bucket '%s' porque no se puede acceder al bucket de S3.
Error en la exportación de instantánea	Arn de la instantánea	failure	serverless-cache-snapshot	No se ha podido exportar la instantánea para la caché <cache-name>. No se ha podido exportar la instantánea al bucket '%s' porque no se puede acceder al bucket.
Error en la exportación de instantánea	Arn de la instantánea	failure	serverless-cache-snapshot	No se ha podido exportar la instantánea para la caché <cache-name>. No se ha podido exportar la instantánea al bucket '%s' porque el bucket no existe.

Tipo-detalle	Lista-recursos	Categoría	Origen	Mensaje
Error en la exportación de instantánea	Arn de la instantánea	failure	serverless-cache-snapshot	No se ha podido exportar la instantánea para la caché <cache-name>. No se ha podido exportar la instantánea al bucket '%s' con la clave administrada por el cliente %s de instantánea de origen <reason>.
Error en la exportación de instantánea	Arn de la instantánea	failure	serverless-cache-snapshot	No se ha podido exportar la instantánea para la caché <cache-name>. No se ha podido exportar la instantánea al bucket '%s'.

Tipo-detalle	Lista-recursos	Categoría	Origen	Mensaje
Error en la copia de instantánea	Arn 1 de la instantánea Arn 2 de la instantánea	failure	serverless-cache-snapshot	Error al copiar la instantánea <snapshot-name>. No se ha podido copiar la instantánea '%s' a la instantánea '%s' con la clave administrada por el cliente <key-id> de la instantánea de origen <reason-name>.
Error en la copia de instantánea	Arn 1 de la instantánea Arn 2 de la instantánea	failure	serverless-cache-snapshot	Error al copiar la instantánea <snapshot-name>. No se ha podido copiar la instantánea '%s' a la instantánea '%s' con la clave administrada por el cliente '%s' '%s' de la instantánea de destino.

Eventos y métricas de clústeres de autodiseño

En esta sección, se describen las métricas, los eventos y los registros que verá cuando trabaje con clústeres de autodiseño.

Temas

- [Métricas para clústeres de autodiseño](#)
- [Eventos de clústeres de autodiseño](#)
- [Entrega de registro](#)
- [Monitoreo del uso con métricas de CloudWatch](#)
- [SNSMonitorización de ElastiCache eventos por parte de Amazon](#)

Métricas para clústeres de autodiseño

Cuando usted diseña los clústeres, ElastiCache emite métricas en cada nivel de nodo, como las métricas en el nivel de host y las métricas de caché.

Para obtener más información sobre las métricas en el nivel de host, consulte [Métricas de nivel de host](#).

Para obtener más información sobre las métricas en el nivel de nodo, consulte [Métricas para Redis OSS](#).

Eventos de clústeres de autodiseño

ElastiCache registra los eventos relacionados con sus cachés de diseño propio. Al trabajar con clústeres de diseño propio, puede ver los eventos de los clústeres en la ElastiCache consola, mediante Amazon Simple Notification Service (AWS CLI SNS) o mediante Amazon Simple Notification Service (SNS). Los eventos de clúster de diseño propio no se publican en Amazon EventBridge.

La información de eventos de clúster de autodiseño incluye la fecha y la hora del evento, el nombre del origen y el tipo del origen del evento, así como una descripción del mismo. Puede recuperar fácilmente los eventos del registro mediante la ElastiCache consola, el comando AWS CLI `describe-events` o la acción de la API `ElastiCache DescribeEvents`.

Visualización de ElastiCache eventos (consola)

El siguiente procedimiento muestra los eventos mediante la ElastiCache consola.

Para ver los eventos mediante la ElastiCache consola

1. Inicie sesión en la ElastiCache consola AWS Management Console y ábrala en <https://console.aws.amazon.com/elasticache/>

2. Para ver una lista de todos los eventos disponibles, elija Events (Eventos) en el panel de navegación.
3. En la pantalla Eventos, cada fila de la lista representa un evento y muestra el origen del evento, el tipo, la hora GMT y una descripción. Con la opción Filter podrá especificar si desea ver todos los eventos o simplemente los eventos de un tipo determinado de la lista de eventos.

Visualización de ElastiCache eventos (AWS CLI)

Para generar una lista de ElastiCache eventos mediante el AWS CLI, utilice el comando `describe-events`. Puede usar parámetros opcionales para controlar el tipo de eventos que se muestran en la lista, el marco temporal de los eventos de la lista, el número máximo de eventos que se incluirán en la lista y mucho más.

El código siguiente muestra hasta 40 eventos de clúster de autodiseño.

```
aws elasticache describe-events --source-type cache-cluster --max-items 40
```

El código siguiente muestra todos los eventos para cachés de autodiseño de las últimas 24 horas (1440 minutos).

```
aws elasticache describe-events --source-type cache-cluster --duration 1440
```

Eventos de clústeres de autodiseño

Esta sección contiene la lista de eventos que puede esperar recibir para sus clústeres de autodiseño.

Los siguientes ElastiCache eventos activan las notificaciones de Amazon SNS. Para obtener más información sobre los eventos, consulte [Visualización de eventos de ElastiCache](#).

Nombre del evento	Mensaje	Descripción
ElastiCache:AddCacheNodeComplete	ElastiCache:AddCacheNodeComplete : <i>cache-cluster</i>	Se ha agregado un nodo de caché al clúster de caché y está listo para su uso.
ElastiCache: AddCacheNodeFailed debido a la insuficie	ElastiCache:AddCacheNodeFailed : <i>cluster-name</i>	No se pudo agregar un nodo de caché porque no hay

Nombre del evento	Mensaje	Descripción
Adición de direcciones IP gratuitas		suficientes direcciones IP disponibles.
ElastiCache:CacheClusterParametersChanged	ElastiCache:CacheClusterParametersChanged : <i>cluster-name</i>	Se han cambiado uno o varios parámetros del clúster de caché.
ElastiCache:CacheClusterProvisioningComplete	ElastiCache:CacheClusterProvisioningComplete <i>cluster-name-0001-005</i>	El aprovisionamiento de un clúster de caché se ha completado y los nodos de caché del clúster de caché están listos para el uso.
ElastiCache:CacheClusterProvisioningFailed debido a un estado de red incompatible	ElastiCache:CacheClusterProvisioningFailed : <i>cluster-name</i>	Se ha intentado lanzar un nuevo clúster de caché en una nube privada virtual (VPC) no existente.
ElastiCache:CacheClusterScalingComplete	CacheClusterScalingComplete : <i>cluster-name</i>	El escalado del clúster de caché se ha completado con éxito.
ElastiCache:CacheClusterScalingFailed	ElastiCache:CacheClusterScalingFailed : nombre del <i>clúster</i>	Se ha producido un error en la operación de ampliación del clúster de caché.

Nombre del evento	Mensaje	Descripción
ElastiCache:CacheClusterSecurityGroupModified	ElastiCache:CacheClusterSecurityGroupModified : <i>cluster-name</i>	<p>Se ha producido uno de los eventos siguientes:</p> <ul style="list-style-type: none">• La lista de los grupos de seguridad de caché autorizados para el clúster de caché se ha modificado.• Se han autorizado uno o varios de los nuevos grupos de seguridad de EC2 en alguno de los grupos de seguridad de caché asociados al clúster de caché.• Se han rechazado uno o varios de los nuevos grupos de seguridad de EC2 de alguno de los grupos de seguridad de caché asociados al clúster de caché.

Nombre del evento	Mensaje	Descripción
ElastiCache:CacheNodeReplaceStarted	ElastiCache:CacheNodeReplaceStarted : <i>cluster-name</i>	<p>ElastiCache ha detectado que el host que ejecuta un nodo de caché está degradado o es inalcanzable y ha empezado a reemplazar el nodo de caché.</p> <div data-bbox="1068 495 1507 810"><p> Note</p><p>La entrada de DNS del nodo de caché de reemplazo no se ha cambiado.</p></div> <p>En la mayoría de los casos, cuando se produce este evento, no es necesario actualizar la lista de servidores de sus clientes. Sin embargo, algunas bibliotecas cliente de caché pueden dejar de usar el nodo de caché incluso después de haber ElastiCache reemplazado el nodo de caché; en este caso, la aplicación debería actualizar la lista de servidores cuando se produzca este evento.</p>

Nombre del evento	Mensaje	Descripción
ElastiCache:CacheNodeReplaceComplete	ElastiCache:CacheNodeReplaceComplete : <i>cluster-name</i>	<p>ElastiCache ha detectado que el host que ejecuta un nodo de caché está degradado o es inalcanzable y ha terminado de reemplazar el nodo de caché.</p> <div data-bbox="1068 541 1507 856"><p> Note</p><p>La entrada de DNS del nodo de caché de reemplazo no se ha cambiado.</p></div> <p>En la mayoría de los casos, cuando se produce este evento, no es necesario actualizar la lista de servidores de sus clientes. Sin embargo, algunas bibliotecas cliente de caché pueden dejar de usar el nodo de caché incluso después de haber ElastiCache reemplazado el nodo de caché; en este caso, la aplicación debería actualizar la lista de servidores cuando se produzca este evento.</p>

Nombre del evento	Mensaje	Descripción
ElastiCache:CacheNodesRebooted	ElastiCache:CacheNodesRebooted : <i>cluster-name</i>	Uno o varios nodos de caché se han reiniciado. Aparece el mensaje (Memcached): "Cache node %s shutdown" A continuación, aparece un segundo mensaje: "Cache node %s restarted"
ElastiCache: CertificateRenewalComplete (solo Redis OSS)	ElastiCache:CertificateRenewalComplete	El certificado de entidad de certificación de Amazon se renovó con éxito.
ElastiCache:CreateReplicationGroupComplete	ElastiCache:CreateReplicationGroupComplete : <i>cluster-name</i>	El grupo de reproducción se ha creado correctamente.
ElastiCache>DeleteCacheClusterComplete	ElastiCache>DeleteCacheClusterComplete : <i>cluster-name</i>	Se ha completado la eliminación de un clúster de caché y de todos los nodos de caché asociados.
ElastiCache: FailoverComplete (Solo Redis OSS)	ElastiCache:FailoverComplete : <i>mycluster</i>	La conmutación por error a un nodo de réplica se ha realizado correctamente.
ElastiCache:ReplicationGroupIncreaseReplicaCountFinished	ElastiCache:ReplicationGroupIncreaseReplicaCountFinished : <i>cluster-name-0001-005</i>	Aumentó el número de réplicas en el clúster.

Nombre del evento	Mensaje	Descripción
ElastiCache:ReplicationGroupIncreaseReplicaCountStarted	ElastiCache:ReplicationGroupIncreaseReplicaCountStarted : <i>cluster-name-0003-004</i>	Ha comenzado el proceso de agregar réplicas al clúster.
ElastiCache:NodeReplacementCanceled	ElastiCache:NodeReplacementCanceled : <i>cluster-name</i>	Un nodo del clúster que cuyo reemplazo estaba programado ya no está programado para el reemplazo.
ElastiCache:NodeReplacementRescheduled	ElastiCache:NodeReplacementRescheduled : <i>cluster-name</i>	<p>Un nodo de su clúster que estaba programado para el reemplazo se ha vuelto a programar para el reemplazo durante el nuevo periodo descrito en la notificación.</p> <p>Para obtener información acerca de las acciones que puede emprender, consulte Sustitución de nodos.</p>
ElastiCache:NodeReplacementScheduled	ElastiCache:NodeReplacementScheduled : <i>cluster-name</i>	<p>Un nodo de su clúster se ha programado para el reemplazo durante el periodo que se describe en la notificación.</p> <p>Para obtener información acerca de las acciones que puede emprender, consulte Sustitución de nodos.</p>
ElastiCache:RemoveCacheNodeComplete	ElastiCache:RemoveCacheNodeComplete : <i>cluster-name</i>	Un nodo de caché se ha eliminado del clúster de caché.

Nombre del evento	Mensaje	Descripción
ElastiCache:ReplicationGroupScalingComplete	ElastiCache:ReplicationGroupScalingComplete : <i>cluster-name</i>	La operación de ampliación del grupo de reproducción se ha completado correctamente.
ElastiCache:ReplicationGroupScalingFailed	"Failed applying modification to cache node type to %s."	Se ha producido un error en la operación de ampliación del grupo de reproducción.
ElastiCache:ServiceUpdateAvailableForNode	"Service update is available for cache node %s."	Hay una actualización de autoservicio disponible para el nodo.
ElastiCache: SnapshotComplete (Solo Redis OSS)	ElastiCache:SnapshotComplete : <i>cluster-name</i>	Una instantánea de caché se ha completado correctamente.
ElastiCache: SnapshotFailed (Solo Redis OSS)	SnapshotFailed : <i>cluster-name</i>	<p>Se ha producido un error en la instantánea de caché. Consulte los eventos de caché del clúster para obtener más detalles acerca de la causa.</p> <p>Si describe la instantánea (consulte DescribeSnapshots), el estado será failed.</p>

Entrega de registro

Note

Redis OSS Slow Log es compatible con los clústeres de caché y grupos de replicación de Redis OSS que utilicen el motor a partir de la versión 6.0.

El registro del motor OSS de Redis es compatible con los clústeres de caché y los grupos de replicación de Redis OSS que utilicen el motor a partir de la versión 6.2.

La entrega de registros le permite transmitir Redis OSS [SLOWLOG o Redis OSS Engine Log](#) a uno de estos dos destinos:

- Amazon Data Firehose
- Amazon CloudWatch Logs

La entrega de registros se habilita y configura al crear o modificar un clúster mediante ElastiCache las API. Cada entrada de registro se entregará al destino especificado en uno de los dos formatos: JSON o TEXT.

Periódicamente, se recupera un número fijo de entradas de registro lento del motor OSS de Redis. En función del valor especificado para el parámetro `slowlog-max-len` del motor, es posible que las entradas de registro lento adicionales no se entreguen en el destino.

Puede optar por cambiar las configuraciones de entrega o deshabilitar la entrega de registros en cualquier momento mediante la AWS consola o una de las API de modificación, [modify-cache-cluster](#) o [modify-replication-group](#) bien.

Debe establecer el parámetro `apply-immediately` para todas las modificaciones de entrega de registros.

Note

CloudWatch Los cargos de Amazon Logs se aplican cuando la entrega de registros está habilitada, incluso cuando los registros se entregan directamente a Amazon Data Firehose. Para obtener más información, consulta la sección Vended Logs en [Amazon CloudWatch Pricing](#).

Contenido de una entrada de registro lento

El registro lento ElastiCache (Redis OSS) contiene la siguiente información:

- `CacheClusterId`— El ID del clúster de caché
- `CacheNodeId`— El ID del nodo de caché
- `Id`: una identificación progresiva única para cada entrada de registro lento
- `Marca temporal`: la marca de tiempo de Unix en la que se procesó el comando registrado
- `Duración`: la cantidad de tiempo necesario para la ejecución, en microsegundos
- `Comando`: el comando utilizado por el cliente. Por ejemplo, `set foo bar` donde `foo` está la clave y donde `bar` está el valor. ElastiCache (Redis OSS) reemplaza el nombre y el valor reales de la clave por `(2 more arguments)` para evitar la exposición de datos confidenciales.
- `ClientAddress`— Dirección IP y puerto del cliente
- `ClientName`— Nombre del cliente si se establece mediante el `CLIENT SETNAME` comando

Contenido de una entrada de registro del motor

El registro del motor ElastiCache (Redis OSS) contiene la siguiente información:

- `CacheClusterId`— El ID del clúster de caché
- `CacheNodeId`— El ID del nodo de caché
- `Nivel de registro`: `LogLevel` puede ser una de las siguientes opciones: `VERBOSE("-")`, `NOTICE("*")`, `WARNING("#")`.
- `Tiempo`: la hora UTC del mensaje registrado. La hora tiene el siguiente formato: `"DD MMM YYYY hh:mm:ss.ms UTC"`
- `Rol`: rol del nodo desde el que se emite el registro. Puede ser uno de los siguientes: «M» para principal, «S» para réplica, «C» para proceso secundario de escritor que trabaja en RDB/AOF o «X» para centinela.
- `Mensaje`: mensaje de registro del motor OSS de Redis.

Permisos para configurar el registro

Debe incluir los siguientes permisos de IAM en la política de usuario/rol de IAM:

- `logs:CreateLogDelivery`

- logs:UpdateLogDelivery
- logs>DeleteLogDelivery
- logs:GetLogDelivery
- logs>ListLogDeliveries

Para obtener más información, consulte [Información general sobre los permisos y las políticas de la administración del acceso](#).

Especificaciones del tipo y formato de registro

Registro lentos

El registro lento admite JSON y TEXT

A continuación se muestra un ejemplo de formato JSON:

```
{
  "CacheClusterId": "logslowxxxxmsxj",
  "CacheNodeId": "0001",
  "Id": 296,
  "Timestamp": 1605631822,
  "Duration (us)": 0,
  "Command": "GET ... (1 more arguments)",
  "ClientAddress": "192.168.12.104:55452",
  "ClientName": "logslowxxxxmsxj##"
}
```

A continuación se muestra un ejemplo de formato TEXT:

```
logslowxxxxmsxj,0001,1605631822,30,GET ... (1 more
arguments),192.168.12.104:55452,logslowxxxxmsxj##
```

Registro del motor

El registro del motor admite JSON y TEXT

A continuación se muestra un ejemplo de formato JSON:

```
{
```

```
"CacheClusterId": "xxxxxxxxxzy-engine-log-test",
"CacheNodeId": "0001",
"LogLevel": "VERBOSE",
"Role": "M",
"Time": "12 Nov 2020 01:28:57.994 UTC",
"Message": "Replica is waiting for next BGSAVE before synchronizing with the primary.
Check back later"
}
```

A continuación se muestra un ejemplo de formato TEXT:

```
xxxxxxxxxzy-engine-log-test/0001:M 29 Oct 2020 20:12:20.499 UTC * A slow-running Lua
script detected that is still in execution after 10000 milliseconds.
```

ElastiCache destinos de registro

En esta sección se describen los destinos de registro que puede elegir para sus ElastiCache registros. Cada sección proporciona instrucciones a fin de configurar el registro para el tipo de destino e información sobre cualquier comportamiento específico del tipo de destino. Después de configurar el destino de registro, puede proporcionar sus especificaciones a la configuración de ElastiCache registro para empezar a iniciar sesión en él.

Temas

- [Amazon CloudWatch Logs](#)
- [Amazon Data Firehose](#)

Amazon CloudWatch Logs

- Usted especifica un grupo de CloudWatch registros al que se entregarán los registros.
- Los registros de varios clústeres y grupos de replicación de Redis OSS se pueden enviar al mismo grupo de registros.
- Se creará una secuencia de registro nueva para cada nodo dentro de un clúster de caché o grupo de reproducción y los registros se entregarán a las secuencias de registro respectivas. El nombre de la secuencia de registro utilizará el siguiente formato: `elasticache/${engine-name}/${cache-cluster-id}/${cache-node-id}/${log-type}`

Permisos para publicar registros en Logs CloudWatch

Debe tener los siguientes ajustes de permisos para configurar ElastiCache (Redis OSS) el envío de registros a un grupo de CloudWatch registros:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "logs:CreateLogDelivery",
        "logs:GetLogDelivery",
        "logs:UpdateLogDelivery",
        "logs>DeleteLogDelivery",
        "logs:ListLogDeliveries"
      ],
      "Resource": [
        "*"
      ],
      "Effect": "Allow",
      "Sid": "ElastiCacheLogging"
    },
    {
      "Sid": "ElastiCacheLoggingCWL",
      "Action": [
        "logs:PutResourcePolicy",
        "logs:DescribeResourcePolicies",
        "logs:DescribeLogGroups"
      ],
      "Resource": [
        "*"
      ],
      "Effect": "Allow"
    }
  ]
}
```

Para obtener más información, consulte [Registros enviados a CloudWatch registros](#).

Amazon Data Firehose

- Especificas un flujo de entrega de Firehose al que se entregarán los troncos.
- Los registros de varios grupos de replicación y clústeres de Redis OSS se pueden enviar al mismo flujo de entrega.

- Los registros de cada nodo dentro de un clúster de caché o grupo de reproducción se entregarán al mismo flujo de entrega. Puede distinguir los mensajes de registro de diferentes nodos de caché en función de los `cache-cluster-id` y `cache-node-id` incluidos en cada mensaje de registro.
- La entrega de troncos a Firehose no está disponible actualmente en la región de Asia Pacífico (Osaka).

Permisos para publicar registros en Firehose

Debe tener los siguientes permisos para configurar ElastiCache (Redis OSS) el envío de registros a una transmisión de entrega de Amazon Kinesis Data Firehose.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "logs:CreateLogDelivery",
        "logs:GetLogDelivery",
        "logs:UpdateLogDelivery",
        "logs>DeleteLogDelivery",
        "logs:ListLogDeliveries"
      ],
      "Resource": [
        "*"
      ],
      "Effect": "Allow",
      "Sid": "ElastiCacheLogging"
    },
    {
      "Sid": "ElastiCacheLoggingFHSLR",
      "Action": [
        "iam:CreateServiceLinkedRole"
      ],
      "Resource": "*",
      "Effect": "Allow"
    },
    {
      "Sid": "ElastiCacheLoggingFH",
      "Action": [
        "firehose:TagDeliveryStream"
      ],
    }
  ]
}
```

```
        "Resource": "Amazon Kinesis Data Firehose delivery stream ARN",
        "Effect": "Allow"
    }
]
}
```

Especificación de la entrega de registros mediante la consola

Con el, AWS Management Console puede crear un clúster de Redis OSS (modo de clúster deshabilitado) siguiendo los pasos descritos en [Creación de un clúster de Redis OSS \(modo de clúster desactivado\) \(consola\)](#) o crear un clúster de Redis OSS (modo de clúster habilitado) siguiendo los pasos de. [Creación de un clúster de Redis OSS \(modo de clúster activado\) \(consola\)](#) En cualquier caso, puede configurar la entrega de registros al realizar lo siguiente:

1. En Configuración avanzada de Redis OSS, seleccione Registros y, a continuación, seleccione Registros lentos o Registros del motor.
2. En Log format (Formato de registro), elija Text o JSON.
3. En Tipo de destino, elija CloudWatch Logs o Kinesis Firehose.
4. En Destino del registro, elija Crear nuevo e introduzca el nombre del bucket de Amazon S3, el nombre del grupo de CloudWatchLogs registros o el nombre de la transmisión de Kinesis Data Firehose, o CloudWatch bien elija Seleccionar existente y, a continuación, elija el nombre del grupo de registros o el nombre de la transmisión de Kinesis Data Firehose.

Cuándo modificar un clúster:

Puede elegir entre habilitar o deshabilitar la entrega de registros o cambiar el tipo de destino, el formato o el destino:

1. [Inicie sesión en la consola y abra la consola en https://console.aws.amazon.com/elasticache/.](https://console.aws.amazon.com/elasticache/)
[ElastiCache](#)
2. En el panel de navegación, elija los clústeres de Redis OSS.
3. En la lista de clústeres, elija el clúster que desea modificar. Elija el Cluster name (Nombre del clúster) y no la casilla de verificación al lado.
4. En la página de Cluster name (Nombre del clúster), seleccione la pestaña de Logs (Registros).
5. Para habilitar/deshabilitar los registros lentos, elija Enable slow logs (Habilitar registros lentos) o Disable slow logs (Deshabilitar registros lentos).

6. Para habilitar/deshabilitar los registros del motor, elija **Enable engine logs** (Habilitar registros del motor) o **Disable engine logs** (Deshabilitar registros del motor).
7. Para cambiar la configuración, elija **Modify slow logs** (Modificar registros lentos) o **Modify engine logs** (Modificar registros del motor):
 - En Tipo de destino, elija **CloudWatch Logs** o **Kinesis Firehose**.
 - En Destino del registro, elija **Crear nuevo** e introduzca el nombre del grupo de **CloudWatchLogs** registros o el nombre de la transmisión de **Kinesis Data Firehose**. O bien, elija **Seleccionar existente** y, a continuación, elija el nombre del grupo de **CloudWatchLogs** registros o el nombre de la transmisión de **Kinesis Data Firehose**.

Especificar la entrega de registros mediante el AWS CLI

Registro lento

Cree un grupo de replicación con una entrega lenta de CloudWatch registros a Logs.

Para Linux, macOS o Unix:

```
aws elasticache create-replication-group \  
  --replication-group-id test-slow-log \  
  --replication-group-description test-slow-log \  
  --engine redis \  
  --cache-node-type cache.r5.large \  
  --num-cache-clusters 2 \  
  --log-delivery-configurations '{  
    "LogType":"slow-log",  
    "DestinationType":"cloudwatch-logs",  
    "DestinationDetails":{  
      "CloudWatchLogsDetails":{  
        "LogGroup":"my-log-group"  
      }  
    },  
    "LogFormat":"json"  
  }'
```

Para Windows:

```
aws elasticache create-replication-group ^  
  --replication-group-id test-slow-log ^
```

```

--replication-group-description test-slow-log ^
--engine redis ^
--cache-node-type cache.r5.large ^
--num-cache-clusters 2 ^
--log-delivery-configurations '{
  "LogType":"slow-log",
  "DestinationType":"cloudwatch-logs",
  "DestinationDetails":{
    "CloudWatchLogsDetails":{
      "LogGroup":"my-log-group"
    }
  },
  "LogFormat":"json"
}'

```

Modifique un grupo de replicación para entregar registros lentos a CloudWatch Logs

Para Linux, macOS o Unix:

```

aws elasticache modify-replication-group \
  --replication-group-id test-slow-log \
  --apply-immediately \
  --log-delivery-configurations '
{
  "LogType":"slow-log",
  "DestinationType":"cloudwatch-logs",
  "DestinationDetails":{
    "CloudWatchLogsDetails":{

      "LogGroup":"my-log-group"
    }
  },
  "LogFormat":"json"
}'

```

Para Windows:

```

aws elasticache modify-replication-group ^
  --replication-group-id test-slow-log ^
  --apply-immediately ^
  --log-delivery-configurations '
{
  "LogType":"slow-log",

```

```
"DestinationType":"cloudwatch-logs",
"DestinationDetails":{
  "CloudWatchLogsDetails":{
    "LogGroup":"my-log-group"
  }
},
"LogFormat":"json"
}'
```

Modificar un grupo de reproducción para deshabilitar la entrega de registro lento

Para Linux, macOS o Unix:

```
aws elasticache modify-replication-group \
  --replication-group-id test-slow-log \
  --apply-immediately \
  --log-delivery-configurations '
{
  "LogType":"slow-log",
  "Enabled":false
}'
```

Para Windows:

```
aws elasticache modify-replication-group ^
  --replication-group-id test-slow-log ^
  --apply-immediately ^
  --log-delivery-configurations '
{
  "LogType":"slow-log",
  "Enabled":false
}'
```

Registro del motor

Cree un grupo de replicación con la entrega de registros del motor a CloudWatch Logs.

Para Linux, macOS o Unix:

```
aws elasticache create-replication-group \
  --replication-group-id test-slow-log \
  --replication-group-description test-slow-log \
```

```

--engine redis \
--cache-node-type cache.r5.large \
--num-cache-clusters 2 \
--log-delivery-configurations '{
  "LogType":"engine-log",
  "DestinationType":"cloudwatch-logs",
  "DestinationDetails":{
    "CloudWatchLogsDetails":{
      "LogGroup":"my-log-group"
    }
  },
  "LogFormat":"json"
}'

```

Para Windows:

```

aws elasticache create-replication-group ^
--replication-group-id test-slow-log ^
--replication-group-description test-slow-log ^
--engine redis ^
--cache-node-type cache.r5.large ^
--num-cache-clusters 2 ^
--log-delivery-configurations '{
  "LogType":"engine-log",
  "DestinationType":"cloudwatch-logs",
  "DestinationDetails":{
    "CloudWatchLogsDetails":{
      "LogGroup":"my-log-group"
    }
  },
  "LogFormat":"json"
}'

```

Modifique un grupo de replicación para entregar el registro del motor a Firehose

Para Linux, macOS o Unix:

```

aws elasticache modify-replication-group \
--replication-group-id test-slow-log \
--apply-immediately \
--log-delivery-configurations '
{
  "LogType":"engine-log",

```

```

    "DestinationType":"kinesis-firehose",
    "DestinationDetails":{
      "KinesisFirehoseDetails":{
        "DeliveryStream":"test"
      }
    },
    "LogFormat":"json"
  }'

```

Para Windows:

```

aws elasticache modify-replication-group ^
  --replication-group-id test-slow-log ^
  --apply-immediately ^
  --log-delivery-configurations '
{
  "LogType":"engine-log",
  "DestinationType":"kinesis-firehose",
  "DestinationDetails":{
    "KinesisFirehoseDetails":{
      "DeliveryStream":"test"
    }
  },
  "LogFormat":"json"
}'

```

Modifique un grupo de reproducción para cambiar al formato del motor

Para Linux, macOS o Unix:

```

aws elasticache modify-replication-group \
  --replication-group-id test-slow-log \
  --apply-immediately \
  --log-delivery-configurations '
{
  "LogType":"engine-log",
  "LogFormat":"json"
}'

```

Para Windows:

```

aws elasticache modify-replication-group ^

```

```
--replication-group-id test-slow-log ^
--apply-immediately ^
--log-delivery-configurations '
{
  "LogType":"engine-log",
  "LogFormat":"json"
}'
```

Modifique un grupo de reproducción para deshabilitar la entrega de registro de motor

Para Linux, macOS o Unix:

```
aws elasticache modify-replication-group \
  --replication-group-id test-slow-log \
  --apply-immediately \
  --log-delivery-configurations '
{
  "LogType":"engine-log",
  "Enabled":false
}'
```

Para Windows:

```
aws elasticache modify-replication-group ^
  --replication-group-id test-slow-log ^
  --apply-immediately ^
  --log-delivery-configurations '
{
  "LogType":"engine-log",
  "Enabled":false
}'
```

Monitoreo del uso con métricas de CloudWatch

ElastiCache proporciona métricas que permiten monitorear los clústeres. Puede acceder a dichas métricas a través de CloudWatch. Para obtener más información sobre CloudWatch, consulte la [documentación de CloudWatch](#).

ElastiCache proporciona métricas de nivel de anfitrión (por ejemplo, sobre el uso de la CPU) y métricas específicas del software del motor de caché (por ejemplo, las tasas de caché y los errores de caché). Estas métricas se miden y publican para cada nodo de caché en intervalos de 60 segundos.

Important

Considere la posibilidad de configurar alarmas de CloudWatch con determinadas métricas clave para, de este modo, recibir una notificación si el rendimiento del clúster de caché comienza a disminuir. Para obtener más información, consulte la sección [¿Qué métricas debo monitorear?](#) de esta guía.

Temas

- [Métricas de nivel de host](#)
- [Métricas para Redis OSS](#)
- [¿Qué métricas debo monitorear?](#)
- [Elección de periodos y estadísticas de métricas](#)
- [Monitoreo de métricas de nodo y clúster de CloudWatch](#)

Métricas de nivel de host

El espacio de nombres AWS/ElastiCache incluye las siguientes métricas de nivel de host para los distintos nodos de caché. Estas métricas se miden y publican para cada nodo de caché en intervalos de 60 segundos.

Véase también

- [Métricas para Redis OSS](#)

Métrica	Descripción	Unidad
CPUUtilization	El porcentaje de CPU utilización de todo el host. Como Redis OSS es de un solo subproceso, le recomendamos que supervise las EngineCPUUtilization métricas de los nodos con 4 o más nodos. vCPUs	Porcentaje
CPUCreditBalance	<p>El número de CPU créditos acumulados por una instancia desde su lanzamiento o inicio. En el caso de la versión T2 Standard, CPUCredit Balance también incluye la cantidad de créditos de lanzamiento acumulados.</p> <p>Los créditos se acumulan en el saldo de créditos una vez obtenidos y se eliminan del saldo de créditos cuando se gastan. El saldo de créditos tiene un límite máximo, determinado por el tamaño de la instancia. Una vez que se ha alcanzado el límite, los nuevos créditos obtenidos se descartarán. Para T2 Standard, los créditos de inicialización no cuentan para el límite.</p> <p>Los créditos incluidos en CPUCreditBalance están disponibles para que la instancia los gaste y supere su uso habitual. CPU</p> <p>CPU las métricas crediticias solo están disponibles con una frecuencia de cinco minutos.</p> <p>Estas métricas no se encuentran disponibles para las instancias de rendimiento ampliable T2.</p>	Créditos (v CPU - minutos)

Métrica	Descripción	Unidad
CPUCreditUsage	<p>El número de CPU créditos que la instancia ha gastado en su CPU utilización. Un CPU crédito equivale a una v CPU ejecutándose al 100% durante un minuto o a una combinación equivalente de vCPUs utilización y tiempo (por ejemplo, una v CPU funcionando al 50% durante dos minutos o dos vCPUs ejecutándose al 25% durante dos minutos).</p> <p>CPULas métricas crediticias solo están disponibles con una frecuencia de cinco minutos. Si se especifica un periodo superior a cinco minutos, utilice la estadística Sum en lugar de Average.</p> <p>Estas métricas no se encuentran disponibles para las instancias de rendimiento ampliable T2.</p>	Créditos (v CPU - minutos)
FreeableMemory	La cantidad de memoria libre disponible en el host. Esto se deriva de los RAM búferes y la memoria caché que, según el sistema operativo, se pueden liberar.	Bytes
NetworkBytesIn	El número de bytes que el host ha leído de la red.	Bytes
NetworkBytesOut	El número de bytes enviados en todas las interfaces de red por la instancia.	Bytes
NetworkPacketsIn	El número de paquetes recibidos en todas las interfaces de red por la instancia. Esta métrica identifica el volumen de tráfico de red entrante en cuanto al número de paquetes de una sola instancia.	Recuento

Métrica	Descripción	Unidad
NetworkPacketsOut	El número de paquetes enviados en todas las interfaces de red por la instancia. Esta métrica identifica el volumen de tráfico de red saliente en cuanto al número de paquetes de una sola instancia.	Recuento
NetworkBandwidthInAllowanceExceeded	El número de paquetes formados en cola o eliminados el ancho de banda agregado entrante superó el máximo de la instancia.	Recuento
NetworkConntrackAllowanceExceeded	El número de paquetes eliminados porque el seguimiento de conexiones superó el máximo de la instancia y no se pudieron establecer nuevas conexiones. Esto puede provocar la pérdida de paquetes para el tráfico hacia o desde la instancia.	Recuento
NetworkBandwidthOutAllowanceExceeded	El número de paquetes en cola o eliminados porque el ancho de banda agregado saliente superó el máximo de la instancia.	Recuento
NetworkPacketsPerSecondAllowanceExceeded	El número de paquetes en cola o eliminados debido a que los paquetes bidireccional superaron el máximo de la instancia.	Recuento
NetworkMaxBytesIn	La ráfaga máxima por segundo de bytes recibidos en cada minuto.	Bytes
NetworkMaxBytesOut	La ráfaga máxima por segundo de bytes transmitidos en cada minuto.	Bytes
NetworkMaxPacketsIn	La ráfaga máxima por segundo de paquetes recibidos en cada minuto.	Recuento
NetworkMaxPacketsOut	La ráfaga máxima por segundo de paquetes transmitidos en cada minuto.	Recuento

Métrica	Descripción	Unidad
SwapUsage	La cantidad de espacio de intercambio utilizado en el host.	Bytes

Métricas para Redis OSS

El espacio de AWS/ElastiCache nombres incluye las siguientes métricas de Redis OSS.

Con la excepción de `ReplicationLag` y `EngineCPUUtilization`, estas métricas se derivan del comando OSS de Redis. `info` Cada métrica se calcula en el nivel de nodo de caché.

[Para obtener la documentación completa del `info` comando OSS de Redis, consulte <http://redis.io/commands/info>.](http://redis.io/commands/info)

Véase también

- [Métricas de nivel de host](#)

Métrica	Descripción	Unidad
ActiveDefragHits	El número de reasignaciones de valor por minuto que ha realizado el proceso de desfragmentación activo. Esto se deriva de las <code>active_defrag_hits</code> estadísticas de Redis OSS INFO.	Número
AuthenticationFailures	El número total de intentos fallidos de autenticarse en Redis OSS mediante el comando AUTH. Puede encontrar más información sobre los errores de autenticación individuales mediante el comando ACL LOG . Sugerimos configurar una alarma para detectar intentos de acceso sin autorización.	Recuento

Métrica	Descripción	Unidad
	El número total de bytes asignados por Redis OSS para todos los fines, incluidos el conjunto de datos, los búferes, etc.	Bytes
BytesUsedForCache	Dimension: Tier=Memory para los clústeres de OSS de Redis, utilizando Organización de datos en niveles : el número total de bytes que la memoria utiliza como caché. Este es el valor de la used_memory estadística en Redis OSS INFO.	Bytes
	Dimension: Tier=SSD para los clústeres OSS de Redis que utilizan Organización de datos en niveles : el número total de bytes utilizados en la memoria caché por el SSD.	Bytes
BytesReadFromDisk	Número total de bytes leídos del disco por minuto. Compatible solo con clústeres que utilizan Organización de datos en niveles .	Bytes
BytesWrittenToDisk	Número total de bytes escritos en el disco por minuto. Compatible solo con clústeres que utilizan Organización de datos en niveles .	Bytes
CacheHits	El número de búsquedas de claves solo de lectura realizadas correctamente en el diccionario principal. Esto se deriva de las keypace_hits estadísticas de Redis OSS INFO.	Recuento
CacheMisses	El número de búsquedas de claves solo de lectura que no se realizaron correctamente en el diccionario principal. Esto se deriva de las keypace_misses estadísticas de Redis OSS INFO.	Recuento

Métrica	Descripción	Unidad
CommandAuthorizationFailures	Número total de intentos fallidos de los usuarios de ejecutar comandos a los que no tienen permiso para llamar. Puede encontrar más información sobre los errores de autenticación individuales mediante el comando ACL LOG . Sugerimos configurar una alarma para detectar intentos de acceso sin autorización.	Recuento
CacheHitRate	Indica la eficiencia de uso de la instancia OSS de Redis. Si la relación de caché es inferior a aproximadamente 0,8, significa que una cantidad significativa de claves se han expulsado, han caducado o no existen. Esto se calcula utilizando estadísticas <code>cache_hits</code> y <code>cache_misses</code> de la siguiente manera: $\text{cache_hits} / (\text{cache_hits} + \text{cache_misses})$.	Porcentaje
ChannelAuthorizationFailures	El número total de intentos con error de los usuarios de acceder a canales a los que no tienen permiso para acceder. Puede encontrar más información sobre los errores de autenticación individuales mediante el comando ACL LOG . Sugerimos configurar una alarma en esta métrica para detectar intentos de acceso sin autorización.	Recuento
CurrConnections	El número de conexiones de cliente, excluidas las conexiones de las réplicas de lectura. ElastiCache utiliza de dos a cuatro de las conexiones para supervisar el clúster en cada caso. Esto se deriva de la <code>connected_clients</code> estadística de Redis OSS INFO .	Recuento

Métrica	Descripción	Unidad
	El número de elementos en la caché. Se deriva de la keyspace estadística OSS de Redis, que suma todas las claves de todo el espacio de claves.	Recuento
CurrItems	Dimension: Tier=Memory para clústeres que utilizan Organización de datos en niveles . Número de elementos en la memoria.	Recuento
	Dimension: Tier=SSD (unidades de estado sólido) para clústeres que utilizan Organización de datos en niveles . Número de elementos en SSD.	Recuento
CurrVolatileItems	Número total de claves en todas las bases de datos que tienen un conjunto ttl. Se deriva de la expires estadística OSS de Redis, que suma todas las claves con un ttl establecido en todo el espacio de claves.	Recuento
DatabaseCapacityUsagePercentage	<p>El porcentaje de la capacidad de datos total para el clúster que está en uso.</p> <p>En las instancias con niveles de datos, la métrica se calcula como $(\text{used_memory} - \text{mem_not_counted_for_evict} + \text{SSD used}) / (\text{maxmemory} + \text{SSD total capacity})$, donde <code>used_memory</code> y <code>maxmemory</code> se toma de Redis OSS INFO.</p> <p>En todos los demás casos, la métrica se calcula utilizando <code>used_memory/maxmemory</code></p>	Porcentaje

Métrica	Descripción	Unidad
DatabaseCapacityUsageCountedForEvictPercentage	<p>El porcentaje de la capacidad de datos total para el clúster que está en uso, excluida la memoria utilizada para gastos generales y COB. Esta métrica se calcula de la siguiente manera:</p> $\frac{\text{used_memory} - \text{mem_not_counted_for_evict}}{\text{maxmemory}}$ <p>En las instancias de datos por niveles, la métrica se calcula de la siguiente manera:</p> $\frac{(\text{used_memory} + \text{SSD used})}{(\text{maxmemory} + \text{SSD total capacity})}$ <p>donde <code>used_memory</code> y <code>maxmemory</code> se toman de Redis OSS INFO</p>	Porcentaje
DatabaseMemoryUsagePercentage	<p>El porcentaje de la memoria para el clúster que está en uso. Se calcula utilizando datos <code>used_memory/maxmemory</code> de Redis OSS INFO.</p>	Porcentaje
DatabaseMemoryUsageCountedForEvictPercentage	<p>El porcentaje de la memoria para el clúster que está en uso, excluida la memoria utilizada para gastos generales y COB. Esto se calcula a partir <code>used_memory-mem_not_counted_for_evict/maxmemory</code> de Redis OSS INFO.</p>	Porcentaje

Métrica	Descripción	Unidad
DB0AverageTTL	<p>Expone el DBO a partir avg_ttl de la keypace estadística del comando OSS INFO de Redis. Las réplicas no hacen que venzan las claves, sino que esperan a que los nodos principales hagan que venzan las claves. Cuando un nodo principal hace que una clave venza (o la expulsa debido a LRU), sintetiza un comando DEL, que se transmite a todas las réplicas. Por lo tanto, DB0AverageTTL es 0 para los nodos de réplica, debido al hecho de que no vencen las claves y, por lo tanto, no realizan un seguimiento de TTL.</p>	Milisegundos

Métrica	Descripción	Unidad
EngineCPUUtilization	<p>Proporciona el uso de la CPU del subproceso del motor Redis OSS. Como Redis OSS es de un solo subproceso, puede utilizar esta métrica para analizar la carga del propio proceso de Redis OSS. La EngineCPUUtilization métrica proporciona una visibilidad más precisa del proceso OSS de Redis. Puede utilizarla junto con la métrica CPUUtilization . CPUUtilization muestra el uso de la CPU para la instancia de servidor como un conjunto, lo que incluye otro sistema operativo y los procesos de administración. En los tipos de nodos grandes que tienen cuatro o más vCPU, utilice la métrica EngineCPUUtilization para monitorear y establecer umbrales de escalado.</p>	Porcentaje

 **Note**

En un ElastiCache host, los procesos en segundo plano supervisan el host para proporcionar una experiencia de base de datos gestionada. Estos procesos en segundo plano pueden ocupar una parte importante de la carga de trabajo de la CPU. Esto no es significativo en host más grandes con más de 2 vCPU. Pero puede afectar a hosts más pequeños con 2 vCPU o menos. Si solo supervisa la EngineCPUUtilization métrica, no se dará cuenta de situaciones en las que el host esté sobrecargado debido a un uso elevado de la CPU por parte del OSS de Redis y a un uso elevado

Métrica	Descripción	Unidad
	de la CPU debido a los procesos de supervisión en segundo plano. Por lo tanto, recomendamos monitorear la métrica <code>CPUUtilization</code> en los hosts con 2 vCPU o menos.	
Evictions	El número de claves que se han desalojado debido al límite <code>maxmemory</code> . Esto se deriva de la <code>evicted_keys</code> estadística de Redis OSS INFO .	Recuento
GlobalDatastoreReplicationLag	Es el retraso entre el nodo primario de la región secundaria y el nodo primario de la región principal. En el caso de Redis OSS activado en modo clúster, el retardo indica el retardo máximo entre los fragmentos.	Segundos
IamAuthenticationExpirations	El número total de conexiones Redis OSS autenticadas por IAM que han caducado. Puede encontrar más información sobre Autenticación con IAM en la guía del usuario.	Recuento
IamAuthenticationThrottling	El número total de solicitudes de autenticación o HELLO de Redis OSS autenticadas por IAM restringidas. Puede encontrar más información sobre Autenticación con IAM en la guía del usuario.	Recuento
IsMaster	Indica si el nodo es el nodo primario de la partición o clúster actual. La métrica puede ser 0 (no principal) o 1 (principal).	Recuento

Métrica	Descripción	Unidad
KeyAuthorizationFailures	Número total de intentos fallidos de los usuarios de acceder a claves a las que no tienen permiso para acceder. Puede encontrar más información sobre los errores de autenticación individuales mediante el comando ACL LOG . Sugerimos configurar una alarma para detectar intentos de acceso sin autorización.	Recuento
KeysTracked	El número de claves rastreadas por Redis OSS, expresado en porcentaje. <code>tracking-table-max-keys</code> El seguimiento de claves se utiliza para ayudar al almacenamiento en caché del lado del cliente y notifica a los clientes cuando se modifican las claves.	Recuento
MemoryFragmentationRatio	Indica la eficiencia en la asignación de memoria del motor OSS de Redis. Determina dos umbrales supondrán comportamientos diferentes. El valor recomendado es tener fragmentación por encima de 1,0. Esto se calcula a partir de la información OSS mem_fragmentation_ratio statistic de Redis .	Número

Métrica	Descripción	Unidad
NewConnections	<p>El número total de conexiones que ha aceptado el servidor durante este periodo. Esto se deriva de la <code>total_connections_received</code> estadística de Redis OSS INFO.</p> <div style="border: 1px solid #00a0e3; border-radius: 10px; padding: 10px; margin-top: 10px;"> <p> Note</p> <p>Si utiliza la versión 5 o inferior ElastiCache (Redis OSS), se utilizan entre dos y cuatro de las conexiones indicadas por esta métrica ElastiCache para supervisar el clúster. Sin embargo, si se utiliza la versión 6 o superior ElastiCache (Redis OSS), las conexiones que utiliza ElastiCache para supervisar el clúster no se incluyen en esta métrica.</p> </div>	Recuento
NumItemsReadFromDisk	El número total de elementos recuperados del disco por minuto. Compatible solo con clústeres que utilizan Organización de datos en niveles .	Recuento
NumItemsWrittenToDisk	El número total de elementos escritos en disco por minuto. Compatible solo con clústeres que utilizan Organización de datos en niveles .	Recuento
MasterLinkHealthStatus	Este estado tiene dos valores: 0 o 1. El valor 0 indica que los datos del nodo ElastiCache principal no están sincronizados con Redis OSS en EC2. El valor 1 indica que los datos están sincronizados. Para completar la migración, utilice la operación de CompleteMigrationAPI .	Booleano

Métrica	Descripción	Unidad
Reclaimed	El número total de eventos de vencimiento de clave. Esto se deriva de la <code>expired_keys</code> estadística de Redis OSS INFO .	Recuento
ReplicationBytes	Para los nodos en una configuración que se replica, <code>ReplicationBytes</code> indica el número de bytes que el nodo primario envía a todas las réplicas. Esta métrica representa la carga de escritura del grupo de reproducción. Esto se deriva de la <code>master_repl_offset</code> estadística de Redis OSS INFO .	Bytes
ReplicationLag	Esta métrica solo se aplica a un nodo que se ejecuta como una réplica de lectura. Representa a lo que tarda la réplica en aplicar los cambios del nodo principal, en segundos. Para el motor OSS de Redis a partir de la versión 5.0.6, el retraso se puede medir en milisegundos.	Segundos
SaveInProgress	Esta métrica binaria devuelve 1 cuando hay una operación de guardado en segundo plano en curso (con ramificación o sin ella) y 0 en caso contrario. El proceso de guardado en segundo plano se usa normalmente durante las instantáneas y las sincronizaciones. Estas operaciones pueden mermar el rendimiento. Con la métrica <code>SaveInProgress</code> , puede diagnosticar si un proceso de guardado en segundo plano causó o no una merma del rendimiento. Esto se deriva de la <code>rdb_bgsave_in_progress</code> estadística de Redis OSS INFO .	Booleano

Métrica	Descripción	Unidad
TrafficManagementActive	Indica si ElastiCache (Redis OSS) gestiona activamente el tráfico mediante el ajuste del tráfico asignado a los comandos entrantes, la supervisión o la replicación. El tráfico se gestiona cuando se envían al nodo más comandos de los que puede procesar Redis OSS y se utiliza para mantener la estabilidad y el funcionamiento óptimo del motor. Cualquier punto de datos de 1 puede indicar que ha disminuido la escala del nodo para la carga de trabajo que se proporciona.	Booleano

 **Note**

Si esta métrica permanece activa, evalúe el clúster para decidir si es necesario escalar verticalmente o escalar horizontalmente. Las métricas relacionadas incluyen `NetworkBandwidthOutAllowanceExceeded` y `EngineCPUUtilization`.

Disponibilidad de EngineCPUUtilization

AWS Las regiones que se enumeran a continuación están disponibles en todos los tipos de nodos compatibles.

Región	Nombres de las regiones
us-east-2	US East (Ohio)
us-east-1	Este de EE. UU. (Norte de Virginia)

Región	Nombres de las regiones
us-west-1	Oeste de EE. UU. (Norte de California)
us-west-2	Oeste de EE. UU. (Oregón)
ap-northeast-1	Asia-Pacífico (Tokio)
ap-northeast-2	Asia-Pacífico (Seúl)
ap-northeast-3	Asia-Pacífico (Osaka)
ap-east-1	Asia-Pacífico (Hong Kong)
ap-south-1	Asia-Pacífico (Mumbai)
ap-southeast-1	Asia-Pacífico (Singapur)
ap-southeast-2	Asia-Pacífico (Sídney)
ap-southeast-3	Asia-Pacífico (Yakarta)
ca-central-1	Canadá (centro)
cn-north-1	China (Pekín)
cn-northwest-2	China (Ningxia)
me-south-1	Medio Oriente (Baréin)
eu-central-1	Europa (Fráncfort)
eu-west-1	Europa (Irlanda)
eu-west-2	Europa (Londres)
eu-west-3	UE (París)
eu-south-1	Europa (Milán)
af-south-1	África (Ciudad del Cabo)

Región	Nombres de las regiones
eu-north-1	Europa (Estocolmo)
sa-east-1	América del Sur (São Paulo)
us-gov-west-1	AWS GovCloud (EE. UU.-Oeste)
us-gov-east-1	AWS GovCloud (EE. UU.-Este)

A continuación se muestran agrupaciones de determinados tipos de comandos, que se obtienen de `info commandstats`: La sección `commandstats` proporciona estadísticas basadas en el tipo de comando, incluido el número de llamadas, el tiempo total de CPU consumido por estos comandos y el promedio de CPU consumido por ejecución de comando. Para cada tipo de comando, se agrega la siguiente línea: `cmdstat_XXX: calls=XXX,usec=XXX,usec_per_call=XXX`.

Las métricas de latencia que se enumeran a continuación se calculan mediante la estadística `commandstats` de [Redis OSS INFO](#). Se calculan de la siguiente manera: $\text{delta}(\text{usec}) / \text{delta}(\text{calls})$. `delta` se calcula como la diferencia en un minuto. La latencia se define como el tiempo de CPU que tarda la CPU en ElastiCache procesar el comando. Tenga en cuenta que para los clústeres que utilizan la organización de datos en niveles, el tiempo que tarda en obtener elementos de SSD no se incluye en estas mediciones.

Para obtener una lista completa de los comandos disponibles, consulte los [comandos de redis](#) en la documentación de Redis OSS.

Métrica	Descripción	Unidad
<code>ClusterBasedCmds</code>	El número total de comandos basados en clústeres. Se deriva de la <code>commandstats</code> estadística OSS de Redis al sumar todos los comandos que actúan en un clúster (<code>cluster slot,cluster info</code> , etc.).	Recuento
<code>ClusterBasedCmdsLatency</code>	Latencia de comandos basados en clústeres.	Microsegundos

Métrica	Descripción	Unidad
EvalBasedCmds	El número total de comandos para los comandos basados en eval. Se deriva de la <code>commandstats</code> estadística OSS de Redis sumando, <code>eval</code> <code>evalsha</code>	Recuento
EvalBasedCmdsLatency	Latencia de comandos basados en eval.	Microsegundos
GeoSpatialBasedCmds	Número total de comandos para comandos basados en condiciones geoespaciales. Se deriva de la estadística OSS de Redis. <code>commandstats</code> Esto se obtiene al sumar todos los tipos de comandos geográficos: <code>geoadd</code> , <code>geodist</code> , <code>geohash</code> , <code>geopos</code> , <code>georadius</code> y <code>georadiusbymember</code> .	Recuento
GeoSpatialBasedCmdsLatency	Latencia de comandos basados en condiciones geoespaciales.	Microsegundos
GetTypeCmds	El número total de comandos de escritura de read-only. Se deriva de la <code>commandstats</code> estadística OSS de Redis al sumar todos los comandos de read-only tipo (<code>get</code> , <code>hget</code> , <code>scardlrange</code> , etc.)	Recuento
GetTypeCmdsLatency	Latencia de los comandos de lectura.	Microsegundos
HashBasedCmds	El número total de comandos basados en hash. Se deriva de la <code>commandstats</code> estadística OSS de Redis al sumar todos los comandos que actúan sobre uno o más hashes (<code>hget</code> , <code>hkeys</code> , <code>hvals</code> etc.). <code>hdel</code>	Recuento
HashBasedCmdsLatency	Latencia de comandos basados en hash.	Microsegundos

Métrica	Descripción	Unidad
HyperLogLogBasedCmds	El número total de comandos basados en HyperLogLog . Se deriva de la <code>commandstats</code> estadística OSS de Redis al sumar todos los pf tipos de comandos (<code>pfadd</code> , <code>pfcount</code> , <code>pfmerge</code> etc.).	Recuento
HyperLogLogBasedCmdsLatency	Latencia de los comandos basados HyperLogLog.	Microsegundos
JsonBasedCmds	El número total de comandos JSON, incluidos los comandos de lectura y escritura. Se deriva de la <code>commandstats</code> estadística OSS de Redis al sumar todos los comandos JSON que actúan sobre las claves JSON.	Recuento
JsonBasedCmdsLatency	Latencia de todos los comandos JSON, incluidos los comandos de lectura y escritura.	Microsegundos
JsonBasedGetCmds	Número total de comandos de solo lectura de JSON. Se deriva de la <code>commandstats</code> estadística OSS de Redis al sumar todos los comandos de lectura de JSON que actúan sobre las claves de JSON.	Recuento
JsonBasedGetCmdsLatency	Latencia de los comandos de solo lectura de JSON.	Microsegundos
JsonBasedSetCmds	Número total de comandos de escritura de JSON. Se deriva de la <code>commandstats</code> estadística de Redis OSS al sumar todos los comandos de escritura de JSON que actúan sobre las claves de JSON.	Recuento
JsonBasedSetCmdsLatency	Latencia de los comandos de escritura de JSON.	Microsegundos

Métrica	Descripción	Unidad
KeyBasedCmds	El número total de comandos basados en claves. Se deriva de la <code>commandstats</code> estadística OSS de Redis al sumar todos los comandos que actúan sobre una o más claves en varias estructuras de datos (<code>del</code> , <code>expire</code> , <code>rename</code> , etc.).	Recuento
KeyBasedCmdsLatency	Latencia de comandos basados en claves.	Microsegundos
ListBasedCmds	El número total de comandos basados en listas. Se deriva de la <code>commandstats</code> estadística OSS de Redis al sumar todos los comandos que actúan en una o más listas (<code>lindex</code> , <code>lrange</code> , <code>lpush</code> , <code>ltrim</code> etc.).	Recuento
ListBasedCmdsLatency	Latencia de comandos basados en listas.	Microsegundos
NonKeyTypeCmds	Número total de comandos no basados en claves. Se deriva de la <code>commandstats</code> estadística OSS de Redis al sumar todos los comandos que no actúan sobre una tecla, por ejemplo, <code>acl dbsize info</code>	Recuento
NonKeyTypeCmdsLatency	Latencia de los comandos non-key-based	Microsegundos
PubSubBasedCmds	El número total de comandos para la funcionalidad publicación/suscripción. Se deriva de las <code>commandstats</code> estadísticas del OSS de Redis al sumar todos los comandos utilizados para la funcionalidad pub/sub: <code>psubscribe</code> , <code>publish</code> , <code>pubsub</code> , <code>punsubscribe</code> , <code>ssubscribe</code> , <code>esunsubscribe</code> , <code>spublish</code> y <code>unsubscribe</code>	Recuento

Métrica	Descripción	Unidad
PubSubBasedCmdsLatency	Latencia de comandos basados en publicación/suscripción.	Microsegundos
SetBasedCmds	El número total de comandos basados en instrucciones set. Se deriva de la <code>commandstats</code> estadística OSS de Redis al sumar todos los comandos que actúan en uno o más conjuntos (<code>scard</code> , <code>sdiff</code> , <code>sadd</code> etc.). <code>union</code>	Recuento
SetBasedCmdsLatency	Latencia de comandos basados en conjuntos.	Microsegundos
SetTypeCmds	El número total de tipos de comandos de write. Se deriva de la <code>commandstats</code> estadística OSS de Redis al sumar todos los mutative tipos de comandos que funcionan con los datos (<code>set</code> , <code>hsetsadd</code> , <code>lpop</code> etc.)	Recuento
SetTypeCmdsLatency	Latencia de comandos de escritura.	Microsegundos
SortedSetBasedCmds	El número total de comandos basados en instrucciones set ordenadas. Se deriva de la <code>commandstats</code> estadística OSS de Redis al sumar todos los comandos que actúan sobre uno o más conjuntos ordenados (<code>zcount</code> , <code>zrangezrank</code> , <code>zadd</code> etc.).	Recuento
SortedSetBasedCmdsLatency	Latencia de comandos basados en orden.	Microsegundos
StringBasedCmds	El número total de comandos basados en cadenas. Se deriva de la <code>commandstats</code> estadística OSS de Redis al sumar todos los comandos que actúan sobre una o más cadenas (<code>strlen</code> , <code>setex</code> , <code>setrange</code> etc.).	Recuento

Métrica	Descripción	Unidad
StringBasedCmdsLatency	Latencia de comandos basados en cadenas	Microsegundos
StreamBasedCmds	El número total de comandos basados en secuencias. Se deriva de la <code>commandstats</code> estadística OSS de Redis al sumar todos los comandos que actúan sobre uno o más tipos de datos de flujos (<code>xrange</code> ,,,, <code>xlenxadd</code> , <code>xdel</code> etc.).	Recuento
StreamBasedCmdsLatency	Latencia de comandos basados en secuencias.	Microsegundos

¿Qué métricas debo monitorear?

Las siguientes CloudWatch métricas ofrecen una buena visión ElastiCache del rendimiento. En la mayoría de los casos, le recomendamos que configure CloudWatch alarmas para estas métricas, de modo que pueda tomar medidas correctivas antes de que se produzcan problemas de rendimiento.

Métricas que se van a monitorear

- [CPUUtilization](#)
- [EngineCPUUtilization](#)
- [SwapUsage](#)
- [Evictions](#)
- [CurrConnections](#)
- [Memoria](#)
- [Network](#)
- [Latencia](#)
- [Replicación](#)
- [Administración del tráfico](#)

CPUUtilization

Se trata de una métrica de nivel de host que muestra un valor como un porcentaje. Para obtener más información, consulte [Métricas de nivel de host](#).

En los tipos de nodos pequeños que tienen dos CPU virtuales o menos, utilice la métrica `CPUUtilization` para monitorear la carga de trabajo.

En general, sugerimos que establezca el umbral en el 90 % del ancho de banda de la CPU disponible. Como Redis OSS es de un solo subproceso, el valor umbral real debe calcularse como una fracción de la capacidad total del nodo. Por ejemplo, supongamos que está usando un tipo de nodo con dos núcleos. En este caso, el umbral de `CPUUtilization` sería de $90/2$, es decir, el 45 %.

Deberá determinar su propio umbral en función del número de núcleos del nodo de caché que use. Si supera este umbral y su carga de trabajo principal es de solicitudes de lectura, escale el clúster de caché de forma ascendente agregando réplicas de lectura. Si la carga de trabajo principal es de solicitudes de escritura, en función de la configuración del clúster, recomendamos que:

- Clústeres de Redis OSS (modo de clúster desactivado): se amplían mediante un tipo de instancia de caché más grande.
- Clústeres de Redis OSS (modo de clúster activado): añade más fragmentos para distribuir la carga de trabajo de escritura entre más nodos principales.

Tip

En lugar de utilizar la métrica a nivel de `hostCPUUtilization`, los usuarios de Redis OSS podrían utilizar la métrica `Redis OSSEngineCPUUtilization`, que indica el porcentaje de uso en el núcleo del motor de Redis OSS. [Para comprobar si esta métrica está disponible en sus nodos y obtener más información, consulte Métricas de Redis OSS.](#)

Para los tipos de nodos más grandes con 4 vCPU o más, puede utilizar la `EngineCPUUtilization` métrica, que indica el porcentaje de uso en el núcleo del motor OSS de Redis. Para comprobar si esta métrica está disponible en sus nodos y obtener más información, consulte [Métricas](#) de Redis OSS.

EngineCPUUtilization

Para los tipos de nodos más grandes con 4 vCPU o más, puede utilizar la `EngineCPUUtilization` métrica, que indica el porcentaje de uso en el núcleo del motor OSS de Redis. Para comprobar si esta métrica está disponible en sus nodos y obtener más información, consulte [Métricas](#) de Redis OSS.

Para obtener más información, consulte la sección CPU en [Supervisión de las mejores prácticas con Amazon ElastiCache \(Redis OSS\) mediante Amazon CloudWatch](#).

SwapUsage

Se trata de una métrica de nivel de host que muestra un valor en bytes. Para obtener más información, consulte [Métricas de nivel de host](#).

Una `FreeableMemory` CloudWatch métrica cercana a 0 (es decir, inferior a 100 MB) o una `SwapUsage` métrica superior a la `FreeableMemory` métrica indica que un nodo está bajo presión de memoria. Si esto sucede, consulte los siguientes temas:

- [Asegurarse de tener suficiente memoria para crear una instantánea de Redis OSS](#)

- [Administración de la memoria reservada](#)

Evictions

Es una métrica del motor de la caché Recomendamos que determine su propio umbral de alarma para esta métrica en función de las necesidades de su aplicación.

CurrConnections

Es una métrica del motor de la caché Recomendamos que determine su propio umbral de alarma para esta métrica en función de las necesidades de su aplicación.

Un número creciente de CurrConnections podría indicar un problema con la aplicación; tendrá que investigar el comportamiento de la aplicación para solucionar este problema.

Para obtener más información, consulte la sección Conexiones en [Supervisión de las mejores prácticas con Amazon ElastiCache \(Redis OSS\) mediante Amazon CloudWatch](#).

Memoria

La memoria es un aspecto fundamental de Redis OSS. Es necesario comprender la utilización de la memoria de un clúster para evitar la pérdida de datos y adaptarse al crecimiento futuro del conjunto de datos. Las estadísticas sobre la utilización de la memoria de un nodo están disponibles en la sección de memoria del comando [INFO](#) OSS de Redis.

Para obtener más información, consulte la sección Memoria en [Supervisión de las mejores prácticas con Amazon ElastiCache \(Redis OSS\) mediante Amazon CloudWatch](#).

Network

Uno de los factores determinantes de la capacidad de la banda ancha de red del clúster es el tipo de nodo seleccionado. Para obtener más información sobre la capacidad de red de tu nodo, consulta los [ElastiCache precios de Amazon](#).

Para obtener más información, consulte la sección Red en [Supervisión de las mejores prácticas con Amazon ElastiCache \(Redis OSS\) mediante Amazon CloudWatch](#).

Latencia

Puede medir la latencia de un comando con un conjunto de CloudWatch métricas que proporcionan latencias agregadas por estructura de datos. [Estas métricas de latencia se calculan mediante la commandstats estadística del comando OSS INFO de Redis](#).

Para obtener más información, consulte la sección Latencia en [Supervisión de las mejores prácticas con Amazon ElastiCache \(Redis OSS\) mediante Amazon CloudWatch](#).

Replicación

El volumen de datos que se replican es visible a través de la métrica `ReplicationBytes`. Aunque esta métrica representa la carga de escritura del grupo de replicación, no proporciona información acerca del estado de replicación. Para este propósito, puede utilizar la métrica `ReplicationLag`.

Para obtener más información, consulte la sección Replicación en [Supervisión de las mejores prácticas con Amazon ElastiCache \(Redis OSS\) mediante Amazon CloudWatch](#).

Administración del tráfico

ElastiCache (Redis OSS) gestiona automáticamente el tráfico de un nodo cuando se envían al nodo más comandos entrantes de los que puede procesar Redis OSS. Esto se hace para mantener un funcionamiento y una estabilidad óptimos del motor.

Cuando el tráfico se administra activamente en un nodo, la métrica `TrafficManagementActive` emite puntos de datos de 1. Esto indica que el nodo ha disminuido la escala para la carga de trabajo que se proporciona. Si esta métrica sigue siendo 1 durante largos periodos de tiempo, evalúe el clúster para decidir si es necesario escalar verticalmente o escalar horizontalmente.

Para obtener más información, consulte la métrica `TrafficManagementActive` en la página [Métricas](#).

Elección de periodos y estadísticas de métricas

Aunque CloudWatch permitirá elegir cualquier periodo y estadística para cada métrica, tenga en cuenta que no todas las combinaciones serán útiles. Por ejemplo, las estadísticas Average, Minimum y Maximum son útiles para CPUUtilization; sin embargo, la estadística Sum no lo es.

Todas las muestras de ElastiCache se publican por un periodo de 60 segundos para cada nodo de caché individual. La métrica de nodo de caché solo contendrá una única muestra para cualquier periodo de 60 segundos.

Para obtener más información acerca de cómo recuperar las métricas para los nodos de caché, consulte [Monitoreo de métricas de nodo y clúster de CloudWatch](#).

Monitoreo de métricas de nodo y clúster de CloudWatch

ElastiCache y CloudWatch se encuentran integrados para ofrecer una amplia variedad de métricas. Estas métricas se pueden monitorear mediante CloudWatch.

Note

El siguiente ejemplo requiere el uso de las herramientas de línea de comandos de CloudWatch. A fin de obtener más información sobre CloudWatch y cómo descargar las herramientas para desarrolladores, consulte la [página de producto de CloudWatch](#).

Los siguientes procedimientos muestran cómo utilizar CloudWatch a fin de recopilar estadísticas de espacio de almacenamiento para un clúster de caché en la última hora.

Note

Los valores de StartTime y EndTime proporcionados en los ejemplos siguientes se proporcionan con fines ilustrativos. Deberá sustituir los valores de hora de inicio y finalización para sus nodos de caché.

Para obtener más información sobre los límites de ElastiCache, consulte [AWS Service Limits](#) de ElastiCache.

Monitoreo de métricas de nodo y clúster de CloudWatch (consola)

Para recopilar estadísticas de uso de la CPU de un clúster de caché

1. Inicie sesión en la AWS Management Console y abra la consola de ElastiCache en <https://console.aws.amazon.com/elasticache/>.
2. Seleccione los nodos de caché de los que desea ver métricas.

 Note

La selección de más de 20 nodos deshabilita la visualización de métricas en la consola.

- a. En la página de Cache Clusters (Clústeres de caché) de la AWS Management Console, haga clic en el nombre de uno o varios clústeres de caché.

Aparecerá la página de detalles del clúster de caché.
- b. Haga clic en la pestaña Nodes (Nodos) situada en la parte superior de la ventana.
- c. En la pestaña Nodes de la ventana de detalles, seleccione los nodos de caché para los que desea ver métricas.

Aparecerá una lista de las métricas de CloudWatch en la parte inferior de la ventana de la consola.

- d. Haga clic en la métrica CPU Utilization.

Se abrirá la consola de CloudWatch y se mostrarán las métricas seleccionadas. Puede usar los cuadros de lista desplegable Statistic y Period y la pestaña Time Range para cambiar las métricas mostradas.

Monitoreo de las métricas de nodo y clúster de CloudWatch mediante la CLI de CloudWatch

Para recopilar estadísticas de uso de la CPU de un clúster de caché

- Para Linux, macOS o Unix:

```
aws cloudwatch get-metric-statistics \  
  --namespace AWS/ElastiCache \  
  --metric-name CPUUtilization \  
  --dimensions='[{"Name":"CacheClusterId","Value":"test"},  
{"Name":"CacheNodeId","Value":"0001"}]' \  
  --statistics=Average \  
  --start-time 2018-07-05T00:00:00 \  
  --end-time 2018-07-06T00:00:00 \  

```

```
--period=3600
```

Para Windows:

```
aws cloudwatch get-metric-statistics ^
  --namespace AWS/ElastiCache ^
  --metric-name CPUUtilization ^
  --dimensions='[{"Name":"CacheClusterId","Value":"test"},
{"Name":"CacheNodeId","Value":"0001"}]' ^
  --statistics=Average ^
  --start-time 2018-07-05T00:00:00 ^
  --end-time 2018-07-06T00:00:00 ^
  --period=3600
```

Monitoreo de las métricas de clúster y nodo de CloudWatch mediante la API de CloudWatch

Para recopilar estadísticas de uso de la CPU de un clúster de caché

- Llame a la API de CloudWatch `GetMetricStatistics` con los siguientes parámetros (tenga en cuenta que las horas de inicio y finalización solo se muestran como ejemplos; tendrá que sustituir estos valores por sus propias horas de inicio y finalización):
 - `Statistics.member.1=Average`
 - `Namespace=AWS/ElastiCache`
 - `StartTime=2013-07-05T00:00:00`
 - `EndTime=2013-07-06T00:00:00`
 - `Period=60`
 - `MeasureName=CPUUtilization`
 - `Dimensions=CacheClusterId=mycachecluster,CacheNodeId=0002`

Example

```
http://monitoring.amazonaws.com/
  ?Action=GetMetricStatistics
  &SignatureVersion=4
  &Version=2014-12-01
  &StartTime=2018-07-05T00:00:00
```

```
&EndTime=2018-07-06T23:59:00
&Period=3600
&Statistics.member.1=Average
&Dimensions.member.1="CacheClusterId=mycachecluster"
&Dimensions.member.2="CacheNodeId=0002"
&Namespace=&AWS;/ElastiCache
&MeasureName=CPUUtilization
&Timestamp=2018-07-07T17%3A48%3A21.746Z
&AWS;AccessKeyId=<&AWS; Access Key ID>
&Signature=<Signature>
```

SNSMonitorización de ElastiCache eventos por parte de Amazon

Cuando se producen eventos importantes para un clúster, ElastiCache envía una notificación a un SNS tema específico de Amazon. Por ejemplo, errores al agregar un nodo, adiciones de nodos correctas, cambios en un grupo de seguridad, etc. Al monitorear los eventos clave, podrá conocer el estado actual de los clústeres y, dependiendo del evento, adoptar medidas correctivas.

Temas

- [Gestión de SNS las notificaciones de ElastiCache Amazon](#)
- [Visualización de eventos de ElastiCache](#)
- [Notificaciones de eventos y Amazon SNS](#)

Gestión de SNS las notificaciones de ElastiCache Amazon

Puede ElastiCache configurar el envío de notificaciones de eventos importantes del clúster mediante Amazon Simple Notification Service (AmazonSNS). En estos ejemplos, configurará un clúster con el nombre de recurso de Amazon (ARN) de un SNS tema de Amazon para recibir notificaciones.

Note

- En este tema se presupone que te has registrado en Amazon SNS y que has configurado y suscrito un SNS tema de Amazon. Para obtener más información sobre cómo realizar esto, consulte la [Guía para desarrolladores de Amazon Simple Notification](#)
- De forma predeterminada, API `modify-replication-group` afecta a todos los grupos de una región y no solo al grupo especificado actualmente. Si desea configurar un grupo

específico de una región de forma diferente a los demás grupos, puede utilizar la `--notification-topic-arn` opción para crear un tema independiente para ese grupo.

Añadir un SNS tema de Amazon

En las siguientes secciones se muestra cómo añadir un SNS tema de Amazon mediante la AWS consola AWS CLI, la o la ElastiCache API.

Añadir un SNS tema de Amazon (consola)

El siguiente procedimiento muestra cómo añadir un SNS tema de Amazon a un clúster. Para añadir un SNS tema de Amazon para un grupo de replicación, en el paso 2, en lugar de elegir un clúster, elija un grupo de replicación y, a continuación, siga los mismos pasos restantes.

Note

Este proceso también se puede utilizar para modificar el SNS tema de Amazon.

Para añadir o modificar un SNS tema de Amazon para un clúster (consola)

1. Inicie sesión en AWS Management Console y abra la ElastiCache consola en <https://console.aws.amazon.com/elasticache/>.
2. En Clústeres, elige el clúster para el que quieres añadir o modificar un SNS tema de AmazonARN.
3. Elija Modificar.
4. En Modificar clúster, en Tema para SNS notificación, elige el SNS tema que quieres añadir o selecciona ARNEntrada manual y escribe el SNS tema ARN de Amazon.
5. Elija Modificar.

Añadir un SNS tema de Amazon (AWS CLI)

Para añadir o modificar un SNS tema de Amazon para un clúster, usa el AWS CLI comandomodify-cache-cluster.

El siguiente ejemplo de código añade un SNS tema de Amazon arn a my-cluster.

Para Linux, macOS o Unix:

```
aws elasticache modify-cache-cluster \  
  --cache-cluster-id my-cluster \  
  --notification-topic-arn arn:aws:sns:us-  
west-2:123456789xxx:ElastiCacheNotifications
```

Para Windows:

```
aws elasticache modify-cache-cluster ^  
  --cache-cluster-id my-cluster ^  
  --notification-topic-arn arn:aws:sns:us-west-2:123456789xx:ElastiCacheNotifications
```

Para obtener más información, consulte. [modify-cache-cluster](#)

Añadir un SNS tema de Amazon (ElastiCache API)

Para añadir o modificar un SNS tema de Amazon para un clúster, llama a la `ModifyCacheCluster` acción con los siguientes parámetros:

- `CacheClusterId=my-cluster`
- `TopicArn=arn%3Aaws%3Asns%3Aus-west-2%3A565419523791%3AElastiCacheNotifications`

Example

```
https://elasticache.amazon.com/  
  ?Action=ModifyCacheCluster  
  &ApplyImmediately=false  
  &CacheClusterId=my-cluster  
  &NotificationTopicArn=arn%3Aaws%3Asns%3Aus-  
west-2%3A565419523791%3AElastiCacheNotifications  
  &Version=2014-12-01  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Timestamp=20141201T220302Z  
  &X-Amz-Algorithm=&AWS;4-HMAC-SHA256  
  &X-Amz-Date=20141201T220302Z  
  &X-Amz-SignedHeaders=Host  
  &X-Amz-Expires=20141201T220302Z  
  &X-Amz-Credential=<credential>
```

```
&X-Amz-Signature=<signature>
```

Para obtener más información, consulte [ModifyCacheCluster](#).

Activación y desactivación de las notificaciones de Amazon SNS

Puede habilitar o deshabilitar las notificaciones para un clúster. Los siguientes procedimientos te muestran cómo deshabilitar SNS las notificaciones de Amazon.

Activación y desactivación de SNS las notificaciones de Amazon (consola)

Para deshabilitar las SNS notificaciones de Amazon mediante el AWS Management Console

1. Inicia sesión en AWS Management Console y abre la ElastiCache consola en <https://console.aws.amazon.com/elasticache/>.
2. Para ver una lista de los clústeres que ejecutan RedisOSS, en el panel de navegación, elija OSSRedis.
3. Elija la casilla situada a la izquierda del nombre del clúster cuya notificación desee modificar.
4. Elija Modificar.
5. En Modificar clúster, en Tema de SNS notificación, seleccione Desactivar notificaciones.
6. Elija Modificar.

Activación y desactivación de SNS las notificaciones de Amazon (AWS CLI)

Para deshabilitar SNS las notificaciones de Amazon, usa el comando `modify-cache-cluster` con los siguientes parámetros:

Para Linux, macOS o Unix:

```
aws elasticache modify-cache-cluster \  
  --cache-cluster-id my-cluster \  
  --notification-topic-status inactive
```

Para Windows:

```
aws elasticache modify-cache-cluster ^  
  --cache-cluster-id my-cluster ^  
  --notification-topic-status inactive
```

Activación y desactivación de SNS las notificaciones de Amazon () ElastiCache API

Para deshabilitar SNS las notificaciones de Amazon, llama a la `ModifyCacheCluster` acción con los siguientes parámetros:

- `CacheClusterId=my-cluster`
- `NotificationTopicStatus=inactive`

Esta llamada devuelve un resultado similar al siguiente:

Example

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=ModifyCacheCluster  
&ApplyImmediately=false  
&CacheClusterId=my-cluster  
&NotificationTopicStatus=inactive  
&Version=2014-12-01  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20141201T220302Z  
&X-Amz-Algorithm=&AWS;4-HMAC-SHA256  
&X-Amz-Date=20141201T220302Z  
&X-Amz-SignedHeaders=Host  
&X-Amz-Expires=20141201T220302Z  
&X-Amz-Credential=<credential>  
&X-Amz-Signature=<signature>
```

Visualización de eventos de ElastiCache

ElastiCache registra eventos relacionados con instancias del clúster, grupos de seguridad y grupos de parámetros. Esta información incluye la fecha y la hora del evento, el nombre del origen y el tipo del origen del evento, así como una descripción del evento. Puede recuperar con facilidad eventos del registro mediante la consola de ElastiCache, el comando de la AWS CLI `describe-events` o la acción de la API de ElastiCache `DescribeEvents`.

Los siguientes procedimientos muestran cómo visualizar todos los eventos de ElastiCache de las últimas 24 horas (1440 minutos).

Visualización de eventos de ElastiCache (Consola)

El siguiente procedimiento muestra eventos mediante la consola de ElastiCache.

Para ver eventos mediante la consola de ElastiCache

1. Inicie sesión en la AWS Management Console y abra la consola de ElastiCache en <https://console.aws.amazon.com/elasticache/>.
2. Para ver una lista de todos los eventos disponibles, elija Events (Eventos) en el panel de navegación.

En la pantalla Events (Eventos), cada fila de la lista representa un evento y muestra el origen del evento, el tipo de evento (`cache-cluster`, `cache-parameter-group`, `cache-security-group` o `cache-subnet-group`), la hora GMT del evento y la descripción de este último.

Con la opción `Filter` podrá especificar si desea ver todos los eventos o simplemente los eventos de un tipo determinado de la lista de eventos.

Visualización de eventos de ElastiCache (AWS CLI)

Para generar una lista de eventos de ElastiCache mediante la AWS CLI, utilice el comando `describe-events`. Puede usar parámetros opcionales para controlar el tipo de eventos que se muestran en la lista, el marco temporal de los eventos de la lista, el número máximo de eventos que se incluirán en la lista y mucho más.

El código siguiente muestra hasta 40 eventos de clúster de caché.

```
aws elasticache describe-events --source-type cache-cluster --max-items 40
```

El código siguiente muestra todos los eventos de las últimas 24 horas (1440 minutos).

```
aws elasticache describe-events --source-type cache-cluster --duration 1440
```

La salida del comando `describe-events` es similar a la siguiente.

```
aws elasticache describe-events --source-type cache-cluster --max-items 40
{
  "Events": [
    {
      "SourceIdentifier": "my-mem-cluster",
      "SourceType": "cache-cluster",
      "Message": "Finished modifying number of nodes from 1 to 3",
      "Date": "2020-06-09T02:01:21.772Z"
    },
    {
      "SourceIdentifier": "my-mem-cluster",
      "SourceType": "cache-cluster",
      "Message": "Added cache node 0002 in availability zone us-west-2a",
      "Date": "2020-06-09T02:01:21.716Z"
    },
    {
      "SourceIdentifier": "my-mem-cluster",
      "SourceType": "cache-cluster",
      "Message": "Added cache node 0003 in availability zone us-west-2a",
      "Date": "2020-06-09T02:01:21.706Z"
    },
    {
      "SourceIdentifier": "my-mem-cluster",
      "SourceType": "cache-cluster",
      "Message": "Increasing number of requested nodes",
      "Date": "2020-06-09T01:58:34.178Z"
    },
    {
      "SourceIdentifier": "mycluster-0003-004",
      "SourceType": "cache-cluster",
      "Message": "Added cache node 0001 in availability zone us-west-2c",
      "Date": "2020-06-09T01:51:14.120Z"
    },
    {
      "SourceIdentifier": "mycluster-0003-004",
      "SourceType": "cache-cluster",

```

```

    "Message": "This cache cluster does not support persistence (ex:
'appendonly'). Please use a different instance type to enable persistence.",
    "Date": "2020-06-09T01:51:14.095Z"
  },
  {
    "SourceIdentifier": "mycluster-0003-004",
    "SourceType": "cache-cluster",
    "Message": "Cache cluster created",
    "Date": "2020-06-09T01:51:14.094Z"
  },
  {
    "SourceIdentifier": "mycluster-0001-005",
    "SourceType": "cache-cluster",
    "Message": "Added cache node 0001 in availability zone us-west-2b",
    "Date": "2020-06-09T01:42:55.603Z"
  },
  {
    "SourceIdentifier": "mycluster-0001-005",
    "SourceType": "cache-cluster",
    "Message": "This cache cluster does not support persistence (ex:
'appendonly'). Please use a different instance type to enable persistence.",
    "Date": "2020-06-09T01:42:55.576Z"
  },
  {
    "SourceIdentifier": "mycluster-0001-005",
    "SourceType": "cache-cluster",
    "Message": "Cache cluster created",
    "Date": "2020-06-09T01:42:55.574Z"
  },
  {
    "SourceIdentifier": "mycluster-0001-004",
    "SourceType": "cache-cluster",
    "Message": "Added cache node 0001 in availability zone us-west-2b",
    "Date": "2020-06-09T01:28:40.798Z"
  },
  {
    "SourceIdentifier": "mycluster-0001-004",
    "SourceType": "cache-cluster",
    "Message": "This cache cluster does not support persistence (ex:
'appendonly'). Please use a different instance type to enable persistence.",
    "Date": "2020-06-09T01:28:40.775Z"
  },
  {
    "SourceIdentifier": "mycluster-0001-004",

```

```
        "SourceType": "cache-cluster",
        "Message": "Cache cluster created",
        "Date": "2020-06-09T01:28:40.773Z"
    }
]
}
```

Para obtener más información como, por ejemplo, los parámetros disponibles y los valores de parámetros permitidos, consulte [describe-events](#).

Visualización de eventos de ElastiCache (API de ElastiCache)

Para generar una lista de eventos de ElastiCache mediante la API de ElastiCache, utilice la acción `DescribeEvents`. Puede usar parámetros opcionales para controlar el tipo de eventos que se muestran en la lista, el marco temporal de los eventos de la lista, el número máximo de eventos que se incluirán en la lista y mucho más.

El código siguiente muestra los 40 eventos de clúster de caché más recientes.

```
https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeEvents
&MaxRecords=40
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&SourceType=cache-cluster
&Timestamp=20150202T192317Z
&Version=2015-02-02
&X-Amz-Credential=<credential>
```

El código siguiente muestra los eventos de clúster de caché de las últimas 24 horas (1440 minutos).

```
https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeEvents
&Duration=1440
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&SourceType=cache-cluster
&Timestamp=20150202T192317Z
&Version=2015-02-02
&X-Amz-Credential=<credential>
```

Las acciones anteriores deberían producir una salida similar a la siguiente.

```
<DescribeEventsResponse xmlns="http://elasticache.amazonaws.com/doc/2015-02-02/">
  <DescribeEventsResult>
    <Events>
      <Event>
        <Message>Cache cluster created</Message>
        <SourceType>cache-cluster</SourceType>
        <Date>2015-02-02T18:22:18.202Z</Date>
        <SourceIdentifier>mem01</SourceIdentifier>
      </Event>

      (...output omitted...)

    </Events>
  </DescribeEventsResult>
  <ResponseMetadata>
    <RequestId>e21c81b4-b9cd-11e3-8a16-7978bb24ffdf</RequestId>
  </ResponseMetadata>
</DescribeEventsResponse>
```

Para obtener más información como, por ejemplo, los parámetros disponibles y los valores de parámetros permitidos, consulte [DescribeEvents](#).

Notificaciones de eventos y Amazon SNS

ElastiCache puede publicar mensajes mediante Amazon Simple Notification Service (SNS) cuando se producen eventos importantes en un clúster de caché. Esta característica se puede usar para actualizar las listas de servidor de las máquinas cliente conectadas a puntos de conexión de nodo de caché individuales de un clúster de caché.

Note

Para obtener más información sobre Amazon Simple Notification Service (SNS), incluido la información sobre los precios y enlaces a los documentos de Amazon SNS, consulte la [Página de producto de Amazon SNS](#).

Las notificaciones se publican en un tema específico de Amazon SNS. A continuación se describen los requisitos para las notificaciones:

- Solo se puede configurar un tema para ElastiCache las notificaciones.

- La AWS cuenta propietaria del tema de Amazon SNS debe ser la misma cuenta propietaria del clúster de caché en el que están habilitadas las notificaciones.
- El tema de Amazon SNS en el que publica no se puede cifrar.

Note

Es posible adjuntar un tema de Amazon SNS cifrado (en reposo) al clúster. Sin embargo, el estado del tema en la ElastiCache consola aparecerá como inactivo, lo que desvincula de forma efectiva el tema del clúster cuando se ElastiCache envían mensajes al tema.

- El tema Amazon SNS debe estar en la misma región que el ElastiCache clúster.

ElastiCache Eventos

Los siguientes ElastiCache eventos activan las notificaciones de Amazon SNS. Para obtener más información sobre los eventos, consulte [Visualización de eventos de ElastiCache](#).

Nombre del evento	Mensaje	Descripción
ElastiCache:AddCacheNodeComplete	ElastiCache:AddCacheNodeComplete : <i>cache-cluster</i>	Se ha agregado un nodo de caché al clúster de caché y está listo para su uso.
ElastiCache: AddCacheNodeFailed debido a la insuficiencia de direcciones IP gratuitas	ElastiCache:AddCacheNodeFailed : <i>cluster-name</i>	No se pudo agregar un nodo de caché porque no hay suficientes direcciones IP disponibles.
ElastiCache:CacheClusterParametersChanged	ElastiCache:CacheClusterParametersChanged : <i>cluster-name</i>	Se han cambiado uno o varios parámetros del clúster de caché.
ElastiCache:CacheClusterProvisioningComplete	ElastiCache:CacheClusterProvisioningComplete <i>cluster-name-0001-005</i>	El aprovisionamiento de un clúster de caché se ha completado y los nodos de caché del clúster de caché están listos para el uso.

Nombre del evento	Mensaje	Descripción
ElastiCache: CacheClusterProvisioningFailed debido a un estado de red incompatible	ElastiCache:CacheClusterProvisioningFailed : <i>cluster-name</i>	Se ha intentado lanzar un nuevo clúster de caché en una nube privada virtual (VPC) no existente.
ElastiCache:CacheClusterScalingComplete	CacheClusterScalingComplete : <i>cluster-name</i>	El escalado del clúster de caché se ha completado con éxito.
ElastiCache:CacheClusterScalingFailed	ElastiCache:CacheClusterScalingFailed : nombre del <i>clúster</i>	Se ha producido un error en la operación de ampliación del clúster de caché.

Nombre del evento	Mensaje	Descripción
ElastiCache:CacheClusterSecurityGroupModified	ElastiCache:CacheClusterSecurityGroupModified : <i>cluster-name</i>	<p>Se ha producido uno de los eventos siguientes:</p> <ul style="list-style-type: none">• La lista de los grupos de seguridad de caché autorizados para el clúster de caché se ha modificado.• Se han autorizado uno o varios de los nuevos grupos de seguridad de EC2 en alguno de los grupos de seguridad de caché asociados al clúster de caché.• Se han rechazado uno o varios de los nuevos grupos de seguridad de EC2 de alguno de los grupos de seguridad de caché asociados al clúster de caché.

Nombre del evento	Mensaje	Descripción
ElastiCache:CacheNodeReplaceStarted	ElastiCache:CacheNodeReplaceStarted : <i>cluster-name</i>	<p>ElastiCache ha detectado que el host que ejecuta un nodo de caché está degradado o es inalcanzable y ha empezado a reemplazar el nodo de caché.</p> <div data-bbox="1068 495 1507 810"><p> Note</p><p>La entrada de DNS del nodo de caché de reemplazo no se ha cambiado.</p></div> <p>En la mayoría de los casos, cuando se produce este evento, no es necesario actualizar la lista de servidores de sus clientes. Sin embargo, algunas bibliotecas cliente de caché pueden dejar de usar el nodo de caché incluso después de haber ElastiCache reemplazado el nodo de caché; en este caso, la aplicación debería actualizar la lista de servidores cuando se produzca este evento.</p>

Nombre del evento	Mensaje	Descripción
ElastiCache:CacheNodeReplaceComplete	ElastiCache:CacheNodeReplaceComplete : <i>cluster-name</i>	<p>ElastiCache ha detectado que el host que ejecuta un nodo de caché está degradado o es inalcanzable y ha terminado de reemplazar el nodo de caché.</p> <div data-bbox="1068 541 1507 856"><p> Note</p><p>La entrada de DNS del nodo de caché de reemplazo no se ha cambiado.</p></div> <p>En la mayoría de los casos, cuando se produce este evento, no es necesario actualizar la lista de servidores de sus clientes. Sin embargo, algunas bibliotecas cliente de caché pueden dejar de usar el nodo de caché incluso después de haber ElastiCache reemplazado el nodo de caché; en este caso, la aplicación debería actualizar la lista de servidores cuando se produzca este evento.</p>

Nombre del evento	Mensaje	Descripción
ElastiCache:CacheNodesRebooted	ElastiCache:CacheNodesRebooted : <i>cluster-name</i>	Uno o varios nodos de caché se han reiniciado. Aparece el mensaje (Memcached): "Cache node %s shutdown" A continuación, aparece un segundo mensaje: "Cache node %s restarted"
ElastiCache: CertificateRenewalComplete (solo Redis OSS)	ElastiCache:CertificateRenewalComplete	El certificado de entidad de certificación de Amazon se renovó con éxito.
ElastiCache:CreateReplicationGroupComplete	ElastiCache:CreateReplicationGroupComplete : <i>cluster-name</i>	El grupo de reproducción se ha creado correctamente.
ElastiCache>DeleteCacheClusterComplete	ElastiCache>DeleteCacheClusterComplete : <i>cluster-name</i>	Se ha completado la eliminación de un clúster de caché y de todos los nodos de caché asociados.
ElastiCache: FailoverComplete (Solo Redis OSS)	ElastiCache:FailoverComplete : <i>mycluster</i>	La conmutación por error a un nodo de réplica se ha realizado correctamente.
ElastiCache:ReplicationGroupIncreaseReplicaCountFinished	ElastiCache:ReplicationGroupIncreaseReplicaCountFinished : <i>cluster-name-0001-005</i>	Aumentó el número de réplicas en el clúster.

Nombre del evento	Mensaje	Descripción
ElastiCache:ReplicationGroupIncreaseReplicaCountStarted	ElastiCache:ReplicationGroupIncreaseReplicaCountStarted : <i>cluster-name-0003-004</i>	Ha comenzado el proceso de agregar réplicas al clúster.
ElastiCache:NodeReplacementCanceled	ElastiCache:NodeReplacementCanceled : <i>cluster-name</i>	Un nodo del clúster que cuyo reemplazo estaba programado ya no está programado para el reemplazo.
ElastiCache:NodeReplacementRescheduled	ElastiCache:NodeReplacementRescheduled : <i>cluster-name</i>	<p>Un nodo de su clúster que estaba programado para el reemplazo se ha vuelto a programar para el reemplazo durante el nuevo periodo descrito en la notificación.</p> <p>Para obtener información acerca de las acciones que puede emprender, consulte Sustitución de nodos.</p>
ElastiCache:NodeReplacementScheduled	ElastiCache:NodeReplacementScheduled : <i>cluster-name</i>	<p>Un nodo de su clúster se ha programado para el reemplazo durante el periodo que se describe en la notificación.</p> <p>Para obtener información acerca de las acciones que puede emprender, consulte Sustitución de nodos.</p>
ElastiCache:RemoveCacheNodeComplete	ElastiCache:RemoveCacheNodeComplete : <i>cluster-name</i>	Un nodo de caché se ha eliminado del clúster de caché.

Nombre del evento	Mensaje	Descripción
ElastiCache:ReplicationGroupScalingComplete	ElastiCache:ReplicationGroupScalingComplete : <i>cluster-name</i>	La operación de ampliación del grupo de reproducción se ha completado correctamente.
ElastiCache:ReplicationGroupScalingFailed	"Failed applying modification to cache node type to %s."	Se ha producido un error en la operación de ampliación del grupo de reproducción.
ElastiCache:ServiceUpdateAvailableForNode	"Service update is available for cache node %s."	Hay una actualización de autoservicio disponible para el nodo.
ElastiCache: SnapshotComplete (Solo Redis OSS)	ElastiCache:SnapshotComplete : <i>cluster-name</i>	Una instantánea de caché se ha completado correctamente.
ElastiCache: SnapshotFailed (Solo Redis OSS)	SnapshotFailed : <i>cluster-name</i>	<p>Se ha producido un error en la instantánea de caché. Consulte los eventos de caché del clúster para obtener más detalles acerca de la causa.</p> <p>Si describe la instantánea (consulte DescribeSnapshots), el estado será failed.</p>

Temas relacionados de

- [Visualización de eventos de ElastiCache](#)

Registro de llamadas a la API de Amazon ElastiCache con AWS CloudTrail

Amazon ElastiCache se integra a AWS CloudTrail, un servicio que proporciona un registro de las acciones realizadas por un usuario, un rol o un servicio de AWS en Amazon ElastiCache. CloudTrail obtiene todas las llamadas a la API para Amazon ElastiCache como eventos, incluido las llamadas procedentes de la consola de Amazon ElastiCache y de las llamadas de código a las operaciones de la API de Amazon ElastiCache. Si crea un registro de seguimiento, puede habilitar la entrega continua de eventos de CloudTrail a un bucket de Amazon S3, incluido los eventos de Amazon ElastiCache. Si no configura un registro de seguimiento, puede ver los eventos más recientes de la consola de CloudTrail en el Event history (Historial de eventos). Mediante la información recopilada por CloudTrail, puede determinar la solicitud que se realizó a Amazon ElastiCache, la dirección IP de origen desde la que se realizó, quién la realizó y cuándo se realizó y otros detalles adicionales.

Para obtener más información acerca de CloudTrail, consulte la [Guía del usuario de AWS CloudTrail](#).

Información sobre Amazon ElastiCache en CloudTrail

CloudTrail se habilita en su cuenta de AWS cuando la crea. Cuando se produce una activación en Amazon ElastiCache, se registra en un evento de CloudTrail junto con los eventos de los demás servicios de AWS en Event history (Historial de eventos). Puede ver, buscar y descargar los últimos eventos de la cuenta de AWS. Para obtener más información, consulte [Ver eventos con el historial de eventos de CloudTrail](#).

Para mantener un registro continuo de los eventos de la cuenta de AWS, incluido los eventos de Amazon ElastiCache, cree un registro de seguimiento. Un registro de seguimiento permite a CloudTrail enviar archivos de registro a un bucket de Amazon S3. De forma predeterminada, cuando se crea un registro de seguimiento en la consola, el registro de seguimiento se aplica a todas las regiones. El registro de seguimiento registra los eventos de todas las regiones de la partición de AWS y envía los archivos de registro al bucket de Amazon S3 especificado. También es posible configurar otros servicios de AWS para analizar en profundidad y actuar en función de los datos de eventos recopilados en los registros de CloudTrail. Para obtener más información, consulte los siguientes temas:

- [Introducción a la creación de registros de seguimiento](#)
- [Servicios e integraciones compatibles con CloudTrail](#)
- [Configuración de notificaciones de Amazon SNS para CloudTrail](#)

- [Recibir archivos de registro de CloudTrail de varias regiones](#) y [Recibir archivos de registro de CloudTrail de varias cuentas](#)

CloudTrail registra todas las acciones de Amazon ElastiCache y se encuentran documentadas en la [Referencia de la API de ElastiCache](#). Por ejemplo, las llamadas a las acciones `CreateCacheCluster`, `DescribeCacheCluster` y `ModifyCacheCluster` generan entradas en los archivos de registros de CloudTrail.

Cada entrada de registro o evento contiene información sobre quién generó la solicitud. La información de identidad del usuario le ayuda a determinar lo siguiente:

- Si la solicitud se realizó con las credenciales raíz o del usuario de IAM de .
- Si la solicitud se realizó con credenciales de seguridad temporales de un rol o fue un usuario federado.
- Si la solicitud la realizó otro servicio de AWS.

Para obtener más información, consulte el [Elemento `userIdentity` de CloudTrail](#).

Descripción de las entradas de archivo de registros de Amazon ElastiCache

Un registro de seguimiento es una configuración que permite la entrega de eventos como archivos de registros en un bucket de Amazon S3 que especifique. Los archivos log de CloudTrail pueden contener una o varias entradas de log. Un evento representa una única solicitud de cualquier origen e incluye información acerca de la acción solicitada, la fecha y la hora de la acción, los parámetros de la solicitud, etcétera. Los archivos de registro de CloudTrail no son un rastro de pila ordenado de las llamadas a las API públicas, por lo que no aparecen en ningún orden específico.

En el ejemplo siguiente, se muestra una entrada de registro de CloudTrail que ilustra la acción `CreateCacheCluster`.

```
{
  "eventVersion": "1.01",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "EXAMPLEEXAMPLEEXAMPLE",
    "arn": "arn:aws:iam::123456789012:user/elasticache-allow",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "userName": "elasticache-allow"
  }
}
```

```

},
"eventTime":"2014-12-01T22:00:35Z",
"eventSource":"elasticache.amazonaws.com",
"eventName":"CreateCacheCluster",
"awsRegion":"us-west-2",
"sourceIPAddress":"192.0.2.01",
"userAgent":"AWS CLI/ElastiCache 1.10 API 2014-12-01",
"requestParameters":{
  "numCacheNodes":2,
  "cacheClusterId":"test-memcached",
  "engine":"memcached",
  "aZMode":"cross-az",
  "cacheNodeType":"cache.m1.small",
},
"responseElements":{
  "engine":"memcached",
  "clientDownloadLandingPage":"https://console.aws.amazon.com/elasticache/home#client-download:",
  "cacheParameterGroup":{
    "cacheParameterGroupName":"default.memcached1.4",
    "cacheNodeIdsToReboot":{
    },
    "parameterApplyStatus":"in-sync"
  },
  "preferredAvailabilityZone":"Multiple",
  "numCacheNodes":2,
  "cacheNodeType":"cache.m1.small",

  "cacheClusterStatus":"creating",
  "autoMinorVersionUpgrade":true,
  "preferredMaintenanceWindow":"thu:05:00-thu:06:00",
  "cacheClusterId":"test-memcached",
  "engineVersion":"1.4.14",
  "cacheSecurityGroups":[
    {
      "status":"active",
      "cacheSecurityGroupName":"default"
    }
  ],
  "pendingModifiedValues":{
  }
},
"requestID":"104f30b3-3548-11e4-b7b8-6d79ffe84edd",

```

```
"eventID":"92762127-7a68-42ce-8787-927d2174cde1"
}
```

En el siguiente ejemplo, se muestra una entrada de registro de CloudTrail que ilustra la acción `DescribeCacheCluster`. Tenga en cuenta que para todas las llamadas de Amazon ElastiCache Describe (Describe*), la sección `ResponseElements` se elimina y aparece como `null`.

```
{
  "eventVersion":"1.01",
  "userIdentity":{
    "type":"IAMUser",
    "principalId":"EXAMPLEEXAMPLEEXAMPLE",
    "arn":"arn:aws:iam::123456789012:user/elasticache-allow",
    "accountId":"123456789012",
    "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
    "userName":"elasticache-allow"
  },
  "eventTime":"2014-12-01T22:01:00Z",
  "eventSource":"elasticache.amazonaws.com",
  "eventName":"DescribeCacheClusters",
  "awsRegion":"us-west-2",
  "sourceIPAddress":"192.0.2.01",
  "userAgent":"AWS CLI/ElastiCache 1.10 API 2014-12-01",
  "requestParameters":{
    "showCacheNodeInfo":false,
    "maxRecords":100
  },
  "responseElements":null,
  "requestID":"1f0b5031-3548-11e4-9376-c1d979ba565a",
  "eventID":"a58572a8-e81b-4100-8e00-1797ed19d172"
}
```

En el ejemplo siguiente se muestra una entrada de log de CloudTrail que registra una acción `ModifyCacheCluster`.

```
{
  "eventVersion":"1.01",
  "userIdentity":{
    "type":"IAMUser",
    "principalId":"EXAMPLEEXAMPLEEXAMPLE",
    "arn":"arn:aws:iam::123456789012:user/elasticache-allow",
    "accountId":"123456789012",
```

```
    "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
    "userName":"elasticache-allow"
  },
  "eventTime":"2014-12-01T22:32:21Z",
  "eventSource":"elasticache.amazonaws.com",
  "eventName":"ModifyCacheCluster",
  "awsRegion":"us-west-2",
  "sourceIPAddress":"192.0.2.01",
  "userAgent":"AWS CLI/ElastiCache 1.10 API 2014-12-01",
  "requestParameters":{"
    "applyImmediately":true,
    "numCacheNodes":3,
    "cacheClusterId":"test-memcached"
  },
  "responseElements":{"
    "engine":"memcached",
    "clientDownloadLandingPage":"https://console.aws.amazon.com/elasticache/home#client-download:",
    "cacheParameterGroup":{"
      "cacheParameterGroupName":"default.memcached1.4",
      "cacheNodeIdsToReboot":{"
        },
      "parameterApplyStatus":"in-sync"
    },
    "cacheClusterCreateTime":"Dec 1, 2014 10:16:06 PM",
    "preferredAvailabilityZone":"Multiple",
    "numCacheNodes":2,
    "cacheNodeType":"cache.m1.small",
    "cacheClusterStatus":"modifying",
    "autoMinorVersionUpgrade":true,
    "preferredMaintenanceWindow":"thu:05:00-thu:06:00",
    "cacheClusterId":"test-memcached",
    "engineVersion":"1.4.14",
    "cacheSecurityGroups":[
      {
        "status":"active",
        "cacheSecurityGroupName":"default"
      }
    ],
    "configurationEndpoint":{"
      "address":"test-memcached.example.cfg.use1prod.cache.amazonaws.com",
      "port":11211
    },
    "pendingModifiedValues":{"
```

```
        "numCacheNodes":3
    }
},
"requestID":"807f4bc3-354c-11e4-9376-c1d979ba565a",
"eventID":"e9163565-376f-4223-96e9-9f50528da645"
}
```

Cuotas para ElastiCache

Tu AWS cuenta tiene cuotas predeterminadas, antes denominadas límites, para cada AWS servicio. A menos que se indique lo contrario, cada cuota es específica de la región de . Puede solicitar el aumento de algunas cuotas, pero otras no se pueden aumentar.

Para ver las cuotas ElastiCache, abra la [consola Service Quotas](#). En el panel de navegación, elija AWS servicios y seleccione ElastiCache.

Para solicitar un aumento de cuota, consulte [Solicitud de aumento de cuota](#) en la Guía del usuario de Service Quotas. Si la cuota aún no se encuentra disponible en Service Quotas, utilice el [formulario de aumento del límite](#).

Su AWS cuenta tiene las siguientes cuotas relacionadas con ElastiCache.

Recurso	Predeterminado
Cachés sin servidor por región	40
Instantáneas sin servidor por día y por caché	24
Nodos por región	300
Nodos por clúster por tipo de instancia (modo de clúster de Redis OSS activado)	90
Nodos por fragmento (modo de clúster de Redis OSS desactivado)	6
Grupos de parámetros por región	300
Grupos de seguridad por región	50
Grupos de subredes por región	300
Subredes por grupo de subredes	20
Usuarios por grupo de usuarios	100
Número máximo de usuarios	1 000

Recurso	Predeterminado
Número máximo de grupos de usuarios	100

Referencia

En los temas de esta sección se abordan el trabajo con la API de Amazon ElastiCache y la sección ElastiCache de la AWS CLI. También se describen mensajes de error y notificaciones de servicio comunes.

- [Uso de la API de ElastiCache](#)
- [Referencia de la API de ElastiCache](#)
- [Sección ElastiCache de la referencia de la AWS CLI](#)
- [Mensajes de ElastiCache error de Amazon](#)
- [Notificaciones](#)

Uso de la API de ElastiCache

Esta sección proporciona descripciones orientadas a tareas sobre el uso y la implementación de operaciones de ElastiCache. Para obtener una descripción completa de dichas operaciones, consulte la [Referencia de la API de Amazon ElastiCache](#)

Temas

- [Uso de la API de consultas](#)
- [Bibliotecas disponibles](#)
- [Solución de problemas de aplicaciones](#)

Uso de la API de consultas

Parámetros de consulta

Las solicitudes basadas en consultas HTTP son solicitudes HTTP que utilizan el verbo HTTP GET o POST y un parámetro de consulta denominado `Action`.

Cada solicitud de consulta debe incluir algunos parámetros comunes para realizar la autenticación y la selección de una acción.

Algunas operaciones toman listas de parámetros. Estas listas se especifican utilizando la notación `param.n`. Los valores de `n` son números enteros a partir de 1.

Autenticación de solicitudes de consulta

Solo se pueden enviar solicitudes de consulta a través de HTTPS y cada una de ellas debe incluir una firma. En esta sección se describe cómo crear la firma. El método que se describe en el procedimiento siguiente se conoce como firma versión 4.

A continuación se indican los pasos básicos que se utilizan para autenticar las solicitudes en AWS. En este proceso se presupone que se ha registrado en AWS y que dispone de un ID de clave de acceso y una clave de acceso secreta.

Proceso de autenticación de consulta

1. El remitente crea una solicitud para AWS.
2. El remitente calcula la firma de la solicitud, una operación hash para código de autenticación de mensajes (HMAC) basado en hash mediante una función hash SHA-1, tal y como se define en la siguiente sección de este tema.
3. El remitente de la solicitud envía a AWS los datos de la misma, la firma y el ID de clave de acceso (el identificador de clave de la clave de acceso secreta utilizada).
4. AWS utiliza el ID de clave de acceso para buscar la clave de acceso secreta.
5. AWS genera una firma a partir de los datos de la solicitud y la clave de acceso secreta con el mismo algoritmo que se utilizó para calcular la firma de la solicitud.
6. Si las firmas coinciden, se considera que la solicitud es auténtica. Si la comparación falla, se descarta la solicitud y AWS devuelve una respuesta de error.

Note

Si una solicitud contiene un parámetro `Timestamp`, la firma calculada para la solicitud caduca 15 minutos después de su valor.

Si una solicitud contiene un parámetro `Expires`, la firma caduca en el momento especificado por el parámetro `Expires`.

Para calcular la firma de la solicitud

1. Cree la cadena de consulta canónica que necesitará más adelante en este procedimiento:

- a. Ordene los componentes UTF-8 de la cadena de consulta por nombre de parámetro con el orden de bytes natural. Los parámetros pueden proceder de GET URI o del cuerpo de la solicitud POST (cuando Content-Type es application/x-www-form-urlencoded).
 - b. Codifique como dirección URL el nombre y los valores del parámetro, aplicando las reglas siguientes:
 - i. No incluya en la codificación de la dirección URL ninguno de los caracteres no reservados definidos en la norma RFC 3986. Estos caracteres no reservados son A–Z, a–z, 0–9, guion (-), carácter de subrayado (_), punto (.) y tilde (~).
 - ii. Codifique con signos de porcentaje el resto de los caracteres con %XY, donde X e Y son caracteres hexadecimales (0-9 y A-F mayúsculas).
 - iii. Codifique con signos de porcentaje los caracteres extendidos UTF-8 con el formato %XY%ZA...
 - iv. Codifique con el signo de porcentaje el carácter de espacio como %20 y no como + (lo que se hace en las codificaciones comunes).
 - c. Separe los nombres de los parámetros codificados de sus valores codificados con el signo igual (=) (carácter ASCII 61), aunque el valor del parámetro esté vacío.
 - d. Separe los pares de nombre-valor con el carácter ampersand (&) (código ASCII 38).
2. Cree la cadena para firmar según la siguiente pseudogramática ("\n" representa un carácter de nueva línea ASCII).

```
StringToSign = HTTPVerb + "\n" +  
ValueOfHostHeaderInLowercase + "\n" +  
HTTPRequestURI + "\n" +  
CanonicalizedQueryString <from the preceding step>
```

El componente HTTPRequestURI es el componente de la ruta absoluta HTTP del URI hasta la cadena de consulta, pero sin incluirla. Si HTTPRequestURI está vacío, utilice una barra diagonal (/).

3. Calcule una HMAC conforme con RFC 2104 con la cadena que acaba de crear, su clave de acceso secreta como la clave y SHA256 o SHA1 como algoritmo de hash.

Para obtener más información, consulte <https://www.ietf.org/rfc/rfc2104.txt>.

4. Convierta el valor resultante en base 64.

5. Incluya el valor como valor del parámetro Signature de la solicitud.

A continuación se muestra una solicitud de muestra (se han agregado saltos de línea para facilitar la lectura).

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=DescribeCacheClusters  
  &CacheClusterIdentifier=myCacheCluster  
  &SignatureMethod=HmacSHA256  
  &SignatureVersion=4  
  &Version=2014-12-01
```

Para la cadena de consulta anterior, debería calcular la firma HMAC de la siguiente cadena.

```
GET\n  
  elasticache.amazonaws.com\n  
  Action=DescribeCacheClusters  
  &CacheClusterIdentifier=myCacheCluster  
  &SignatureMethod=HmacSHA256  
  &SignatureVersion=4  
  &Version=2014-12-01  
  &X-Amz-Algorithm=&AWS;4-HMAC-SHA256  
  &X-Amz-Credential=AKIADQKE4SARGYLE%2F20140523%2Fus-west-2%2Felasticache  
%2Faws4_request  
  &X-Amz-Date=20141201T223649Z  
  &X-Amz-SignedHeaders=content-type%3Bhost%3Buser-agent%3Bx-amz-content-sha256%3Bx-  
amz-date  
    content-type:  
    host:elasticache.us-west-2.amazonaws.com  
    user-agent:CacheServicesAPICommand_Client  
  x-amz-content-sha256:  
  x-amz-date:
```

El resultado es la siguiente solicitud firmada.

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=DescribeCacheClusters  
  &CacheClusterIdentifier=myCacheCluster  
  &SignatureMethod=HmacSHA256  
  &SignatureVersion=4  
  &Version=2014-12-01
```

```
&X-Amz-Algorithm=&AWS;4-HMAC-SHA256
&X-Amz-Credential=AKIADQKE4SARGYLE/20141201/us-west-2/elasticache/aws4_request
&X-Amz-Date=20141201T223649Z
&X-Amz-SignedHeaders=content-type;host;user-agent;x-amz-content-sha256;x-amz-date
&X-Amz-Signature=2877960fced9040b41b4feaca835fd5cfeb9264f768e6a0236c9143f915ffa56
```

Para obtener información detallada acerca del proceso de firma y el cálculo de la firma de la solicitud, consulte el tema [Proceso de firma Signature Version 4](#) y sus subtemas.

Bibliotecas disponibles

AWS ofrece kits de desarrollo de software (SDK) para los desarrolladores de software que prefieran crear aplicaciones mediante las API de lenguaje específico en lugar de la API de consulta. Estos SDK proporcionan funciones básicas (que no se incluyen en las API), como la autenticación de solicitudes, los reintentos de solicitudes y la gestión de errores para que se pueda comenzar más fácilmente. Encontrará SDK y recursos adicionales para los siguientes lenguajes de programación:

- [Java](#)
- [Windows y .NET](#)
- [PHP](#)
- [Python](#)
- [Ruby](#)

Para obtener información acerca de otros lenguajes, consulte las [Código de muestra y bibliotecas](#).

Solución de problemas de aplicaciones

ElastiCache proporciona errores específicos y descriptivos para ayudarlo a solucionar problemas durante la interacción con la API de ElastiCache.

Recuperación de errores

Normalmente, conviene que una aplicación compruebe si una solicitud generó un error antes de emplear tiempo en procesar los resultados. La forma más fácil de averiguar si se ha producido un error, consiste en buscar un nodo de `ERROR` en la respuesta de la API de ElastiCache.

La sintaxis XPath permite comprobar fácilmente si hay un nodo `ERROR`, y ofrece un método sencillo de recuperar el mensaje de error y su código. La partición de código siguiente utiliza Perl y el módulo

XML::XPath para determinar si se ha producido un error durante una solicitud. Si es así, el código imprime el primer mensaje de error y su código en la respuesta.

```
use XML::XPath;
my $xp = XML::XPath->new(xml =>$response);
if ( $xp->find("//Error") )
{print "There was an error processing your request:\n", " Error code: ",
$xp->findvalue("//Error[1]/Code"), "\n", " ",
$xp->findvalue("//Error[1]/Message"), "\n\n"; }
```

Consejos para la solución de problemas

Recomendamos los siguientes procesos para diagnosticar y solucionar problemas con la API de ElastiCache.

- Verifique si ElastiCache se ejecuta correctamente.

Para ello, solo tiene que abrir una ventana del navegador y enviar una solicitud de consulta al servicio de ElastiCache (como <https://elasticache.amazonaws.com>). El mensaje 500 Error de servidor interno o `MissingAuthenticationTokenException` confirmará que el servicio está disponible y que responde a las solicitudes.

- Comprobar la estructura de la solicitud.

Cada operación de ElastiCache tiene una página de referencia en la Referencia de la API de ElastiCache. Compruebe que utiliza los parámetros correctamente. Para obtener ideas sobre lo que podría estar mal, examine las solicitudes de muestra o los escenarios de usuario para ver si esos ejemplos realizan operaciones similares.

- Visite el foro.

Existe un foro de discusión de ElastiCache donde puede buscar soluciones a los problemas que otras personas han experimentado al utilizar este servicio. Para ver el foro, consulte

<https://forums.aws.amazon.com/>

Configuración de la interfaz de línea de comandos de ElastiCache

En esta sección, se describen los requisitos previos para ejecutar las herramientas de línea de comandos, se explica dónde se pueden obtener y cómo configurar las herramientas y su entorno, y se incluyen una serie de ejemplos sobre los usos habituales de las herramientas.

Siga las instrucciones de este tema únicamente si va a utilizar la AWS CLI de ElastiCache.

Important

La interfaz de línea de comandos (CLI) de Amazon ElastiCache no admite mejoras de ElastiCache después de la versión de la API del 30/09/2014. Para utilizar la funcionalidad de ElastiCache más reciente desde la línea de comandos, utilice la [interfaz de la línea de comandos de AWS](#).

Temas

- [Requisitos previos](#)
- [Obtención de las herramientas de línea de comandos](#)
- [Configuración de las herramientas](#)
- [Suministro de credenciales para las herramientas](#)
- [Variables de entorno](#)

Requisitos previos

En este documento, se presupone que se trabaja en un entorno de Linux/UNIX o Windows. Las herramientas de línea de comandos de Amazon ElastiCache también funcionan en Mac OS X, que es un entorno basado en UNIX; sin embargo, en esta guía no se incluyen instrucciones específicas para Mac OS X.

Por convención, todo el texto de las líneas de comando va precedido de un símbolo de línea de comando genérico: **PROMPT>** . Es probable que el símbolo de línea de comando real que aparezca en su equipo sea distinto. También utilizamos **\$** para representar comandos específicos de Linux/UNIX y **C:\>** para representar comandos específicos de Windows. La salida de ejemplo que se obtiene como resultado del comando se muestra inmediatamente después sin ningún prefijo.

El entorno en tiempo de ejecución de Java

Las herramientas de línea de comandos que se utilizan en esta guía requieren que se ejecute Java versión 5 o posterior. Se acepta una instalación de JRE o de JDK. Para ver y descargar los JRE para las distintas plataformas, como Linux/UNIX y Windows, consulte [Descargas de Java SE](#).

Configuración de la variable Java Home

Las herramientas de línea de comandos dependen de una variable de entorno (JAVA_HOME) para localizar Java Runtime. Esta variable de entorno se debe establecer en la ruta completa del directorio que contiene un subdirectorio denominado `bin` que, a su vez, contiene el ejecutable `java` (en Linux y UNIX) o `java.exe` (en Windows).

Para establecer la variable `Java_Home`

1. Establezca la variable `Java_Home`.

- En Linux y UNIX, escriba el comando siguiente:

```
$ export JAVA_HOME=<PATH>
```

- En Windows, escriba el siguiente comando:

```
C:\> set JAVA_HOME=<PATH>
```

2. Confirme la configuración de la ruta ejecutando `$JAVA_HOME/bin/java -version` comprobando la salida.

- En Linux/UNIX, verá una salida similar a la siguiente:

```
$ $JAVA_HOME/bin/java -version
java version "1.6.0_23"
Java(TM) SE Runtime Environment (build 1.6.0_23-b05)
Java HotSpot(TM) Client VM (build 19.0-b09, mixed mode, sharing)
```

- En Windows, verá una salida similar a la siguiente:

```
C:\> %JAVA_HOME%\bin\java -version
java version "1.6.0_23"
Java(TM) SE Runtime Environment (build 1.6.0_23-b05)
Java HotSpot(TM) Client VM (build 19.0-b09, mixed mode, sharing)
```

Obtención de las herramientas de línea de comandos

Las herramientas de línea de comandos se encuentran disponibles como archivo ZIP en el [sitio web de herramientas para desarrolladores de ElastiCache](#). Estas herramientas están escritas en Java, e incluyen scripts de shell para Windows 2000/XP/Vista/Windows 7, Linux/UNIX y Mac OSX. El archivo ZIP es autónomo y no requiere instalación; basta con descargarlo y descomprimirlo en un directorio del equipo local.

Configuración de las herramientas

Las herramientas de línea de comandos dependen de una variable de entorno (AWS_ELASTICACHE_HOME) para localizar las bibliotecas que necesitan. Es necesario configurar esta variable de entorno para poder utilizar las herramientas. Establézcala en la ruta del directorio en el que descomprimió las herramientas de línea de comandos. Este directorio se denomina ElastiCacheCli-A.B.nnnn (A, B y n son números de versión) y contiene subdirectorios denominados bin y lib.

Para configurar la variable de entorno AWS_ELASTICACHE_HOME

- Abra una ventana de línea de comandos e ingrese uno de los siguientes comandos para establecer la variable de entorno AWS_ELASTICACHE_HOME.
 - En Linux y UNIX, escriba el comando siguiente:

```
$ export &AWS;_ELASTICACHE_HOME=<path-to-tools>
```

- En Windows, escriba el siguiente comando:

```
C:\> set &AWS;_ELASTICACHE_HOME=<path-to-tools>
```

Para facilitar el uso de las herramientas, recomendamos que agregue el directorio BIN de herramientas a la ruta del sistema. En esta guía se presupone que el directorio BIN está en la ruta del sistema.

Para agregar el directorio BIN de las herramientas a la ruta del sistema

- Introduzca los siguientes comandos para agregar el directorio BIN de las herramientas a la ruta del sistema.

- En Linux y UNIX, escriba el comando siguiente:

```
$ export PATH=$PATH:$&AWS;_ELASTICACHE_HOME/bin
```

- En Windows, escriba el siguiente comando:

```
C:\> set PATH=%PATH%;%&AWS;_ELASTICACHE_HOME%\bin
```

Note

Las variables de entorno de Windows se restablecen cuando se cierra la ventana de comandos. Es posible que desee configurarlas de forma permanente. Para obtener más información, consulte la documentación de su versión de Windows.

Note

Las rutas que contienen espacios deben encerrarse entre comillas; por ejemplo:
"C:\Program Files\Java"

Suministro de credenciales para las herramientas

Las herramientas de línea de comandos necesitan la clave de acceso y la clave de acceso secreta de AWS que se proporcionan con la cuenta de AWS. Puede obtenerlas desde la línea de comandos o desde un archivo de credenciales situado en el sistema local.

La implementación incluye un archivo de plantilla `${AWS_ELASTICACHE_HOME}/credential-file-path.template` que se debe editar con información personalizada. A continuación se muestra el contenido del archivo de plantilla:

```
AWSAccessKeyId=<Write your AWS access ID>  
AWSSecretKey=<Write your AWS secret key>
```

⚠ Important

En UNIX, limite los permisos al propietario del archivo de credenciales:

```
$ chmod 600 <the file created above>
```

Una vez que el archivo de credenciales se encuentre configurado, tendrá que configurar la variable de entorno `AWS_CREDENTIAL_FILE` para que las herramientas de ElastiCache puedan encontrar la información.

Para configurar la variable de entorno `AWS_CREDENTIAL_FILE`

1. Establezca la variable de entorno :

- En Linux y UNIX, actualice la variable con el comando siguiente:

```
$ export &AWS;_CREDENTIAL_FILE=<the file created above>
```

- En Windows, establezca la variable con el comando siguiente:

```
C:\> set &AWS;_CREDENTIAL_FILE=<the file created above>
```

2. Compruebe que la configuración funciona correctamente; para ello, ejecute el comando siguiente:

```
elasticache --help
```

Debería ver la página de uso de todos los comandos de ElastiCache.

Variables de entorno

Las variables de entorno pueden resultar útiles para crear scripts, configurar valores predeterminados o anularlos temporalmente.

Además de la variable de entorno `AWS_CREDENTIAL_FILE`, la mayoría de las herramientas de la API incluidas en la interfaz de línea de comandos de ElastiCache admiten las siguientes variables:

- `EC2_REGION`: la región de AWS que se utilizará.

- `AWS_ELASTICACHE_URL`: la URL que se utilizará para la llamada de servicio. No es necesario especificar otro punto de enlace regional si se especifica `EC2_REGION` o se pasa el parámetro `--region`.

En los siguientes ejemplos se muestra cómo establecer la variable de entorno `EC2_REGION` para configurar la región utilizada por las herramientas de la API:

Linux, OS X o Unix

```
$ export EC2_REGION=us-west-1
```

Windows

```
$ set EC2_REGION=us-west-1
```

Mensajes de ElastiCache error de Amazon

Amazon devuelve los siguientes mensajes de error ElastiCache. Es posible que reciba otros mensajes de error devueltos por ElastiCache otros AWS servicios o por Redis OSS. Para obtener descripciones de los mensajes de error de fuentes distintas ElastiCache, consulte la documentación de la fuente que genera el mensaje de error.

- [Cluster node quota exceeded](#)
- [Customer's node quota exceeded](#)
- [Manual snapshot quota exceeded](#)
- [Insufficient cache cluster capacity](#)

Mensaje de error: se ha superado la cuota de nodos del clúster. Cada clúster puede tener como máximo %n nodos en esta región.

Causa: ha intentado crear o modificar un clúster y el resultado de dicha modificación daría lugar a un clúster con más de %n nodos.

Solución: cambie la solicitud para que el clúster no tenga más de %n nodos. O bien, si necesitas más de %n nodos, realiza tu solicitud mediante el [formulario de solicitud de Amazon ElastiCache Node](#).

Para obtener más información, consulta [Amazon ElastiCache Limits](#) en Referencia general de Amazon Web Services.

Mensajes de error: se ha superado la cuota de nodos de clientes. Puede tener como máximo %n nodos en esta región O Ya ha alcanzado su cuota de %s nodos en esta región.

Causa: ha intentado crear o modificar un clúster y el resultado de dicha modificación es que su cuenta tendría más de %n nodos en todos los clústeres de esta región.

Solución: cambie la solicitud de forma que el total de nodos en la región de todos los clústeres para esta cuenta no supere %n nodos. O bien, si necesitas más de %n nodos, realiza tu solicitud mediante el [formulario de solicitud de Amazon ElastiCache Node](#).

Para obtener más información, consulta [Amazon ElastiCache Limits](#) en Referencia general de Amazon Web Services.

Mensajes de error: The maximum number of manual snapshots for this cluster taken within 24 hours has been reached or The maximum number of manual snapshots for this node taken within 24 hours has been reached its quota of %n

Causa: ha intentado tomar una instantánea manual de un clúster cuando ya ha tomado el máximo permitido de instantáneas manuales en un periodo de 24 horas.

Solución: espere 24 horas para intentar tomar otra instantánea manual del clúster. O bien, si necesita tomar una instantánea manual ahora, tome la instantánea de otro nodo que tenga los mismos datos como, por ejemplo, un nodo diferente de un clúster.

Mensajes de error: InsufficientCacheClusterCapacity

Causa: AWS no tiene suficiente capacidad bajo demanda disponible para brindar servicio a la solicitud.

Solución:

- Espere unos minutos y después envíe la solicitud de nuevo; la capacidad puede cambiar frecuentemente.

- Envíe una nueva solicitud con una cantidad reducida de nodos o fragmentos (grupos de nodos). Por ejemplo, si hace una única solicitud para lanzar 15 nodos, intente hacer 3 solicitudes para 5 nodos o 15 solicitudes de 1 nodo en su lugar.
- Si está lanzando un clúster, envíe una solicitud nueva sin especificar ninguna zona de disponibilidad.
- Si está lanzando un clúster, envíe una solicitud nueva usando un tipo de nodo distinto (que puede ampliar en una fase posterior). Para obtener más información, consulte [Escalado ElastiCache \(Redis OSS\)](#).

Notificaciones

En este tema se tratan ElastiCache las notificaciones que podrían interesarle. Una notificación es una situación o un evento que, en la mayoría de los casos, es temporal y solo se mantiene hasta que se encuentra y se implementa una solución. Las notificaciones suelen tener una fecha de inicio y una fecha de resolución. Pasada esta fecha, la notificación deja de ser relevante. Las notificaciones pueden ser relevantes o irrelevantes para usted. Las directrices de implementación que recomendamos permiten mejorar el desempeño de su clúster, siempre que decida seguirlas.

Las notificaciones no anuncian características o ElastiCache funciones nuevas o mejoradas.

ElastiCache Notificaciones generales

Actualmente, no hay ElastiCache notificaciones pendientes que no sean específicas del motor.

ElastiCache (Redis OSS) notificaciones específicas

Actualmente no hay notificaciones pendientes ElastiCache (Redis OSS).

ElastiCache Historial de documentación de (RedisOSS)

- API versión: 02/02/2015
- Última actualización de la documentación: 27 de noviembre de 2023

La siguiente tabla describe los cambios importantes introducidos en cada versión de la a partir de marzo de 2018. Para recibir notificaciones sobre las actualizaciones de esta documentación, puede suscribirse al feed. RSS

Actualizaciones recientes ElastiCache (RedisOSS)

Cambio	Descripción	Fecha
ElastiCache (RedisOSS) agregó soporte para tamaños de nodos C7gn adicionales	ElastiCache (RedisOSS) agregó soporte para tamaños de nodos C7gN adicionales.	10 de enero de 2024
ElastiCache (RedisOSS) ahora admite la creación de cachés sin servidor	Ahora puede crear cachés sin servidor, lo que simplifica la administración de la caché y permite el escalado al instante para dar soporte a las aplicaciones más exigentes . Para obtener más información, consulte Cómo elegir una opción de implementación . Como parte de esta función, se agregaron nuevos permisos para permitir la asociación de <code>ElastiCacheServiceRolePolicy</code> <code>AmazonElastiCacheFullAccess</code> cachés sin servidor con puntos de conexión gestionados. VPC Además, se agregaron permisos para revisar la	27 de noviembre de 2023

experiencia de uso de la consola mediante la política AmazonElastiCacheFullAccess .

[ElastiCache \(RedisOSS\) ahora admite la modificación del modo de clúster](#)

Ahora puede migrar clústeres del modo de clúster desactivado (CMD) al modo de clúster activado (CME). Para obtener más información, consulte [Modificación del modo de clúster](#).

11 de mayo de 2023

[ElastiCache \(RedisOSS\) ahora admite la modificación de la configuración de cifrado en tránsito](#)

Ahora puede cambiar la TLS configuración de los clústeres de Redis sin necesidad de volver a crear o aprovisionar los OSS clústeres ni afectar a la disponibilidad de las aplicaciones. Para obtener más información, consulte [Habilitación del cifrado en tránsito en un clúster existente](#).

28 de diciembre de 2022

[ElastiCache \(RedisOSS\) ahora admite la autenticación de usuarios mediante IAM](#)

IAM La autenticación le permite autenticar una conexión a ElastiCache (RedisOSS) mediante identidades. AWS IAM Esto le permite reforzar el modelo de seguridad y simplificar muchas tareas de seguridad administrativa. Para obtener más información, consulte [Autenticarse](#) con IAM

16 de noviembre de 2022

[ElastiCache \(RedisOSS\) ahora es compatible con Redis 7 OSS](#)

Esta versión incorpora varias funciones nuevas a Amazon ElastiCache (RedisOSS): OSS funciones de Redis, ACL mejoras y Sharded Pub/Sub. Para obtener más información, consulte [ElastiCache \(Redis\) versión 7.0. OSS](#)

8 de noviembre de 2022

[ElastiCache \(RedisOSS\) ahora es compatible IPV6](#)

ElastiCache es compatible con las versiones 4 y 6 (IPv4yIPv6) del Protocolo de Internet, lo que le permite configurar el clúster para que solo acepte IPv4 conexiones, solo IPv6 conexiones o ambas, IPv4 y IPv6 conexiones (de doble pila). IPv6 [es compatible con las cargas de trabajo que utilizan el OSS motor Redis a partir de la versión 6.2 en adelante en todas las instancias integradas en el sistema Nitro](#). No hay cargos adicionales por acceder a más de. ElastiCache IPv6 Para obtener más información, consulte [Choosing a network type](#) (Elección del tipo de red).

7 de noviembre de 2022

[ElastiCache \(RedisOSS\) ahora admite el formato nativo de notación de JavaScript objetos \(JSON\)](#)

El formato nativo de notación de JavaScript objetos (JSON) es una forma sencilla y sin esquemas de codificar conjuntos de datos complejos dentro de los clústeres de Redis. OSS Puede almacenar y acceder a los datos de forma nativa mediante el formato JavaScript Object Notation (JSON) dentro de los clústeres de Redis y actualizar los JSON datos almacenados en esos OSS clústeres, sin necesidad de administrar código personalizado para serializarlos y deserializarlos. [Para obtener más información, consulte Primeros pasos con JSON](#)

25 de mayo de 2022

[ElastiCache ahora es compatible PrivateLink](#)

AWS PrivateLink le permite acceder de forma privada a ElastiCache API las operacion es sin una puerta de enlace a Internet, NAT dispositivo, VPN conexión o conexión AWS Direct Connect. Para obtener más información, consulte [Amazon ElastiCache API y VPC puntos de enlace de interfaz \(AWS PrivateLink\)](#) para Redis o OSS [Amazon ElastiCache API y puntos de VPC enlace de interfaz \(AWS PrivateLink\)](#) para Memcached.

24 de enero de 2022

[ElastiCache \(RedisOSS\) ahora es compatible con Redis 6.2 y Data Tiering OSS](#)

Amazon ElastiCache (RedisOSS) presenta la próxima versión del OSS motor Redis compatible con Amazon. ElastiCache ElastiCache (RedisOSS) 6.2 incluye mejoras de rendimiento para los clústeres TLS habilitados que utilizan tipos de nodos x86 con 8 vCPUs o más o tipos de nodos Graviton2 con 4 o más. vCPUs ElastiCache (RedisOSS) también introduce la estratificación de los datos. Puede utilizar la organización de datos en niveles como una forma más económica de escalar los clústeres hasta cientos de terabytes de capacidad. [Para obtener más información, consulte ElastiCache \(RedisOSS\) versión 6.2 \(mejorada\) y Organización de datos por niveles.](#)

23 de noviembre de 2021

[Respaldo para Auto Scaling](#)

ElastiCache (RedisOSS) ahora es compatible con Auto Scaling. ElastiCache El escalado automático (RedisOSS) es la capacidad de aumentar o disminuir automáticamente los fragmentos o réplicas deseados en su servicio ElastiCache (RedisOSS). ElastiCache aprovecha el servicio Application Auto Scaling para proporcionar esta funcionalidad. Para obtener más información, consulte [Clústeres de Auto Scaling ElastiCache \(RedisOSS\)](#).

19 de agosto de 2021

[Support para la entrega de registros OSS lentos de Redis](#)

ElastiCache ahora te permite transmitir Redis OSS SLOWLOG a uno de dos destinos: Amazon Data Firehose o Amazon CloudWatch Logs. Para obtener más información, consulte [Entrega de registros](#).

22 de abril de 2021

[Respaldo para etiquetado de recursos y claves de condición](#)

ElastiCache ahora admite el etiquetado para ayudarle a gestionar sus clústeres y otros ElastiCache recursos. Para obtener más información, consulte [Etiquetar ElastiCache](#) los recursos. ElastiCache también introduce la compatibilidad con las claves de condición. Puede especificar las condiciones que determinan cómo surtirá efecto una IAM política. Para obtener más información, consulte [Uso de las claves de condición](#).

7 de abril de 2021

[ElastiCache ya está disponible en AWS Outposts](#)

[AWS Outposts ofrece](#) AWS servicios, infraestructura y modelos operativos nativos a prácticamente cualquier centro de datos, espacio de ubicación conjunta o instalación local. Puedes implementarlo ElastiCache en Outposts para configurar, operar y usar la memoria caché de forma local, tal como lo harías en la nube. Para obtener más información, consulte [Uso de Outposts para Redis o OSS](#) [Uso de Outposts](#) para Memcached.

8 de octubre de 2020

[ElastiCache ahora OSS es compatible con Redis 6](#)

Amazon ElastiCache (RedisOSS) presenta la próxima versión del OSS motor Redis compatible con Amazon. ElastiCache Esta versión incluye [autenticación de usuarios con control de acceso basado en roles](#), respaldo sin versiones, almacenamiento en caché del lado del cliente y mejoras operativas significativas. Para obtener más información, consulte [ElastiCache \(RedisOSS\) la versión 6.0 \(mejorada\)](#).

7 de octubre de 2020

[ElastiCache ahora es compatible con Zonas Locales](#)

Una zona local es una extensión de una AWS región que está geográficamente cerca de sus usuarios. Puede extender cualquier nube privada virtual (VPC) de una AWS región principal a las Zonas Locales creando una nueva subred y asignándola a una Zona Local. Para obtener más información, consulte [Uso de Local Zones](#).

25 de septiembre de 2020

[ElastiCache \(RedisOSS\) ahora permite escalar su entorno de OSS clústeres de Redis hasta 500 nodos o 500 fragmentos](#)

El modo OSS Clúster de Redis permite realizar configuraciones que puede utilizar para particionar los datos en varios fragmentos y ofrece una mejor escalabilidad, rendimiento y disponibilidad. Esta función está disponible en Amazon ElastiCache (RedisOSS) a partir de la versión 5.0.6 en todas AWS las regiones y para todos los entornos de clústeres ElastiCache (RedisOSS) existentes y nuevos. Para obtener más información, consulte Nodos y fragmentos de [Redis OSS](#).

13 de agosto de 2020

[ElastiCache ahora admite permisos a nivel de recursos](#)

Ahora puede restringir el alcance de los permisos de un usuario especificando los ElastiCache recursos en una política AWS Identity and Access Management (IAM). Para obtener más información, consulte los [permisos de nivel de recursos](#).

12 de agosto de 2020

[ElastiCache \(RedisOSS\) añade métricas adicionales de Amazon CloudWatch](#)

ElastiCache (RedisOSS) ahora admite nuevas CloudWatch métricas, incluidas PubSubCmds y HyperLogLogBasedCmds. Para obtener una lista completa, consulte [Metrics for OSS Redis](#).

10 de junio de 2020

[ElastiCache ahora admite la actualización automática de clústeres ElastiCache](#)

Amazon ElastiCache ahora admite la actualización automática de ElastiCache los clústeres una vez pasada la «fecha de caducidad recomendada» de la actualización del servicio. ElastiCache utilizará su ventana de mantenimiento para programar la actualización automática de los clústeres aplicables. Para obtener más información, consulte [Actualizaciones de autoservicio](#).

13 de mayo de 2020

[ElastiCache \(RedisOSS\) ahora es compatible con Global Datastore for Redis OSS](#)

La OSS función Global Datastore for Redis ofrece una replicación totalmente gestionada, rápida, fiable y segura en todas las regiones. AWS Con esta función, puede crear clústeres de réplicas de lectura entre regiones para ElastiCache (RedisOSS) a fin de permitir las lecturas de baja latencia y la recuperación ante desastres en todas las regiones. AWS Puede crear, modificar y describir un almacén de datos global. También puede añadir o eliminar AWS regiones de su almacén de datos global y promover una AWS región como principal dentro de un almacén de datos global. Para obtener más información, consulte [Replicación entre AWS regiones](#) mediante un almacén de datos global.

16 de marzo de 2020

[ElastiCache \(RedisOSS\) ahora es compatible con la versión 5.0.6 de Redis OSS](#)

Para obtener más información, consulte [ElastiCache \(RedisOSS\) versión 5.0.6 \(mejorada\)](#).

18 de diciembre de 2019

[Amazon ElastiCache ahora admite nodos de caché estándar T3](#)

Ahora puede lanzar la próxima generación de nodos de caché estándar T3 con ráfagas de uso general en Amazon.

Las instancias estándar T3 EC2 de Amazon proporcionan un nivel de CPU rendimiento básico con la capacidad de interrumpir el CPU uso en cualquier momento hasta que se agoten los créditos acumulados.

Para obtener más información, consulte [Tipos de nodos compatibles](#).

12 de noviembre de 2019

[Amazon ElastiCache ahora admite la modificación del AUTH token en un servidor ElastiCache \(RedisOSS\) existente](#)

ElastiCache (RedisOSS) 5.0.6 30 de octubre de 2019

ahora le permite modificar los tokens de autenticación configurando y rotando nuevos tokens. Puede modificar los tokens activos mientras están en uso. También puede añadir tokens nuevos a los clústeres habilitados con cifrado en tránsito que se configuraron anteriormente sin tokens de autenticación. Se trata de un proceso de dos pasos mediante el cual puede configurar y rotar el token sin interrumpir las solicitudes de los clientes. Actualmente, esta función no es compatible con AWS CloudFormation. Para obtener más información, consulte [Autenticación de usuarios con el comando Redis OSS AUTH](#).

[Amazon ElastiCache ahora admite la migración de datos en línea desde Redis OSS a Amazon EC2](#)

Ahora puede usar Online Migration para migrar sus datos de Redis autohospedado en OSS Amazon a EC2 Amazon. ElastiCache Para obtener más información, consulte [Migración en línea](#) a ElastiCache

28 de octubre de 2019

[ElastiCache \(RedisOSS\) introduce el escalado vertical en línea para el modo Clúster de Redis. OSS](#)

Ahora puede ampliar o reducir su clúster de Redis OSS fragmentado bajo demanda. ElastiCache (RedisOSS) cambia el tamaño del clúster cambiando el tipo de nodo, mientras el clúster permanece en línea y atiende las solicitudes entrantes. Para obtener más información, consulte [Escalado vertical en línea mediante la modificación del tipo de nodo.](#)

20 de agosto de 2019

[ElastiCache \(RedisOSS\) ahora permite a los usuarios utilizar un único punto final de lectura para su clúster de Amazon ElastiCache \(RedisOSS\).](#)

Esta función le permite dirigir todo el tráfico de lectura a su clúster ElastiCache (RedisOSS) a través de un único punto final a nivel de clúster para aprovechar el equilibrio de carga y la mayor disponibilidad. Para obtener más información, consulte [Búsqueda de puntos de enlace de conexión.](#)

13 de junio de 2019

[ElastiCache \(RedisOSS\) ahora permite a los usuarios aplicar las actualizaciones del servicio según su propio horario](#)

Con esta característica, puede elegir aplicar las actualizaciones del servicio en el momento que lo desee y no solo durante los periodos de mantenimiento. Esto minimizará las interrupciones del servicio, especialmente durante los picos de actividad empresarial, y ayudará a garantizar el cumplimiento si su clúster cuenta con programas de cumplimiento ElastiCache compatibles. Para obtener más información, consulta [Actualizaciones de autoservicio en Amazon ElastiCache](#) y [Validación de conformidad para Amazon ElastiCache](#).

4 de junio de 2019

[ElastiCache Ofertas de instancias reservadas estándar: pago inicial parcial, pago anticipado total y sin pago inicial.](#)

Las instancias reservadas le ofrecen la flexibilidad de reservar una ElastiCache instancia de Amazon por un período de uno o tres años en función del tipo de instancia y AWS la región. Para obtener más información, consulte [Administración de costos con los nodos reservados](#).

18 de enero de 2019

[ElastiCache \(RedisOSS\) admiten hasta 250 nodos por clúster de OSS Redis](#)

El límite de nodos o fragmentos se puede aumentar hasta un máximo de 250 por clúster ElastiCache (OSSRedis). Para obtener más información, consulte [Particiones](#).

19 de noviembre de 2018

[ElastiCache \(RedisOSS\) admite la conmutación por error automática y las copias de seguridad y restauración en todos los nodos T2](#)

ElastiCache (RedisOSS) incorpora la compatibilidad con la conmutación automática por error, la creación de instantáneas y la creación de copias de seguridad y restauración en todos los nodos T2. Para obtener más información, consulte [Backup and Restore and Snapshot ElastiCache \(RedisOSS\)](#).

19 de noviembre de 2018

[ElastiCache \(RedisOSS\) es compatible con los nodos M5 y R5](#)

ElastiCache (RedisOSS) ahora es compatible con los nodos M5 y R5, tipos de instancias de uso general y optimizadas para la memoria basados en el sistema Nitro. AWS Para obtener más información, consulte [Tipos de nodos compatibles](#).

23 de octubre de 2018

[Posibilidad de cambiar dinámicamente el número de réplicas de lectura](#)

ElastiCache (RedisOSS) ha agregado soporte para agregar y eliminar réplicas de lectura de cualquier clúster sin tiempo de inactividad del clúster. Para obtener más información sobre estos y otros cambios de esta versión, consulte [Cambiar el número de réplicas en la Guía del usuario de ElastiCache \(Redis OSS\)](#). Consulte también [DecreaseReplicaCounty](#) [IncreaseReplicaCounten](#) en la ElastiCache API Referencia.

17 de septiembre de 2018

[Certificación de RAMP conformidad de la Reserva Federal](#)

ElastiCache (RedisOSS) ahora cuenta con la certificación de RAMP conformidad con la Reserva Federal. Para obtener más información, consulta [Validación de conformidad para Amazon ElastiCache](#).

30 de agosto de 2018

[Actualizaciones del motor Redis OSS \(modo de clúster activado\)](#)

Amazon ElastiCache (RedisOSS) ha agregado soporte para actualizar las versiones del motor Redis OSS (habilitado para el modo de clúster). Para obtener más información, consulte [Actualización de las versiones del motor](#).

20 de agosto de 2018

[PCIDSScertificación de conformidad](#)

ElastiCache (RedisOSS) ahora cuenta con la certificación de PCI DSS conformidad. Para obtener más información, consulta [Validación de conformidad para Amazon ElastiCache](#).

5 de julio de 2018

[Support para ElastiCache \(RedisOSS\) 4.0.10](#)

ElastiCache (RedisOSS) ahora es compatible con Redis OSS 4.0.10, lo que incluye tanto el cifrado como el cambio de tamaño de los clústeres en línea en una sola versión. Para obtener más información, consulte [ElastiCache \(RedisOSS\) versión 4.0.10 \(mejorada\)](#).

14 de junio de 2018

[Estructura de la guía del usuario](#)

[La guía de ElastiCache usuario única se ha reestructurado ahora de forma que haya guías de usuario independientes para Redis OSS \(Guía del usuario de ElastiCache \(RedisOSS\)\) y para Memcached \(Guía del usuario \(Memcached\)\). ElastiCache](#) La estructura de la documentación de la sección [AWS CLI Command Reference: elasticache](#) y [Amazon ElastiCache API Reference](#) permanece inalterada.

20 de abril de 2018

[Support para EngineCPU Utilization metric](#)

ElastiCache (RedisOSS) agregó una nueva métrica EngineCPU Utilization, que indica el porcentaje de su CPU capacidad que se está utilizando actualmente. Para obtener más información, consulte [Metrics for OSS Redis](#).

9 de abril de 2018

En la siguiente tabla se describen los cambios importantes introducidos en la Guía del usuario antes de marzo de 2018.

Cambio	Descripción	Fecha de modificación
Soporte para la región Asia-Pacífico (Osaka-Local).	<p>ElastiCache se agregó soporte para la región Asia-Pacífico (local de Osaka). La región Asia-Pacífico (Osaka) admite actualmente una única zona de disponibilidad y solo es por invitación. Para más información, consulte los siguientes temas:</p> <ul style="list-style-type: none"> • Regiones admitidas • Tipos de nodos de caché compatibles 	12 de febrero de 2018
Soporte para EU (París)	<p>ElastiCache se agregó soporte para la región de la UE (París). Para más información, consulte los siguientes temas:</p> <ul style="list-style-type: none"> • Regiones admitidas • Tipos de nodos de caché compatibles 	18 de diciembre de 2017

Cambio	Descripción	Fecha de modificación
Soporte para la región China (Ningxia)	<p>Amazon ElastiCache agregó soporte para la región de China (Ningxia). Para más información, consulte los siguientes temas:</p> <ul style="list-style-type: none">• Regiones admitidas• Tipos de nodos de caché compatibles	11 de diciembre de 2017
Respaldo con para roles vinculados a servicios	<p>Esta versión de ElastiCache agregó soporte para Service Linked Roles (SLR). Para más información, consulte los siguientes temas:</p> <ul style="list-style-type: none">• Uso de roles vinculados a servicios para Amazon ElastiCache• Configure sus permisos (solo para ElastiCache usuarios nuevos)	7 de diciembre de 2017
Respaldo con los tipos de nodo R4	<p>Esta versión ElastiCache agregó soporte para los tipos de nodos R4 en todas AWS las regiones compatibles ElastiCache con. Puede adquirir nodos del tipo R4 como nodos de caché reservados o en diferido. Para más información, consulte los siguientes temas:</p> <ul style="list-style-type: none">• Tipos de nodos de caché compatibles• Parámetros específicos del tipo de nodo de Redis OSS	20 de noviembre de 2017

Cambio	Descripción	Fecha de modificación
ElastiCache (RedisOSS) 3.2.10 y compatibilidad con la refragmentación en línea	<p>Amazon ElastiCache (RedisOSS) añade soporte para ElastiCache (RedisOSS) 3.2.10. ElastiCache (RedisOSS) también introduce el cambio de tamaño del clúster en línea para añadir o eliminar fragmentos del clúster mientras se siguen atendiendo las solicitudes de E/S entrantes. Para más información, consulte los siguientes temas:</p> <ul style="list-style-type: none">• Cambio de tamaño de clústeres online• Refragmentación y reequilibrio de particiones en línea para Redis OSS (modo de clúster habilitado)	9 de noviembre de 2017
HIPAAaptitud	<p>ElastiCache La versión 3.2.6 (RedisOSS) ahora está certificada como apta HIPAA cuando el cifrado está habilitado en el clúster. Para más información, consulte los siguientes temas:</p> <ul style="list-style-type: none">• Validación de conformidad para Amazon ElastiCache• Seguridad de datos en Amazon ElastiCache	2 de noviembre de 2017

Cambio	Descripción	Fecha de modificación
<p>ElastiCache (RedisOSS) 3.2.6 y compatibilidad con el cifrado</p>	<p>ElastiCache añade compatibilidad con ElastiCache (RedisOSS) 3.2.6, que incluye dos funciones de cifrado:</p> <ul style="list-style-type: none"> • El cifrado en tránsito cifra los datos mientras están en tránsito; por ejemplo, entre los nodos de un clúster o entre un clúster y la aplicación. • El cifrado en reposo cifra los datos en el disco durante las operaciones de sincronización y copia de seguridad. <p>Para más información, consulte los siguientes temas:</p> <ul style="list-style-type: none"> • Seguridad de datos en Amazon ElastiCache • Versiones compatibles ElastiCache (Redis OSS) 	<p>25 de octubre de 2017</p>
<p>Tema de patrones de conexión</p>	<p>ElastiCache la documentación agrega un tema que cubre varios patrones para acceder a un ElastiCache clúster en AmazonVPC.</p> <p>Para obtener más información, consulte Patrones de acceso para acceder a una ElastiCache caché en una Amazon VPC la Guía ElastiCache del usuario.</p>	<p>24 de abril de 2017</p>

Cambio	Descripción	Fecha de modificación
Posibilidad de realizar pruebas con la conmutación por error automática	<p>ElastiCache añade soporte para probar la conmutación por error automática en los clústeres de Redis que admiten la replicación. OSS Para más información, consulte los siguientes temas:</p> <ul style="list-style-type: none"> • Prueba de la conmutación por error automática en la Guía del usuario de ElastiCache . • TestFailover en la Referencia. ElastiCache API • test-failover en la Referencia de la AWS CLI . 	4 de abril de 2017
Restauración mejorada de Redis OSS	<p>ElastiCache añade OSS copias de seguridad y restauración de Redis mejoradas con el cambio de tamaño del clúster. Esta característica permite restaurar una copia de seguridad en otro clúster con un número de particiones distinto del que tenía el clúster que se utilizó para crear la copia de seguridad. (En el caso de API yCLI, esta función puede restaurar un número diferente de grupos de nodos en lugar de un número diferente de fragmentos). Esta actualización también es compatible con diferentes configuraciones de OSS ranuras Redis. Para obtener más información, consulte Restauración desde una copia de seguridad a una nueva caché.</p>	15 de marzo de 2017

Cambio	Descripción	Fecha de modificación
Nuevo parámetro de administración de OSS memoria de Redis	<p>ElastiCache añade un nuevo OSS parámetro de <code>Redisreserved-memory-percent</code>, que facilita la gestión de la memoria reservada. Este parámetro está disponible en todas las versiones de ElastiCache (RedisOSS). Para más información, consulte los siguientes temas:</p> <ul style="list-style-type: none"> • Administración de la memoria reservada • Nuevos parámetros para Redis OSS 3.2.4 	15 de marzo de 2017
Respaldo con la región UE Oeste (Londres)	<p>ElastiCache añade soporte para la región de la UE (Londres). Actualmente, solo se admiten los tipos de nodos T2 y M4. Para más información, consulte los siguientes temas:</p> <ul style="list-style-type: none"> • Regiones admitidas • Tipos de nodos de caché compatibles 	13 de diciembre de 2016
Respaldo con la región Canadá (Montreal)	<p>ElastiCache añade soporte para la región de Canadá (Montreal). Actualmente, solo se admiten los nodos de tipo M4 y T2 en esta AWS región. Para más información, consulte los siguientes temas:</p> <ul style="list-style-type: none"> • Regiones admitidas • Tipos de nodos de caché compatibles 	8 de diciembre de 2016
Respaldo con los tipos de nodo M4 y R3	<p>ElastiCache añade compatibilidad con los tipos de nodos R3 y M4 en la región de Sudamérica (São Paulo) y con los tipos de nodos M4 en la región de China (Pekín). Para más información, consulte los siguientes temas:</p> <ul style="list-style-type: none"> • Regiones admitidas • Tipos de nodos de caché compatibles 	1 de noviembre de 2016

Cambio	Descripción	Fecha de modificación
Respaldo con la región EE. UU Este 2 (Ohio)	<p>ElastiCache añade compatibilidad con la región EE.UU. Este (Ohio) (us-east-2) con los tipos de nodos M4, T2 y R3. Para más información, consulte los siguientes temas:</p> <ul style="list-style-type: none"> • Regiones admitidas • Tipos de nodos de caché compatibles 	17 de octubre de 2016
Support para Redis Cluster OSS	<p>ElastiCache añade soporte para Redis OSS Cluster (mejorado). Los clientes que utilizan Redis OSS Cluster pueden particionar sus datos en hasta 15 fragmentos (grupos de nodos). Cada partición admite la reproducción con hasta 5 réplicas de lectura por partición. Los tiempos de conmutación por error automática de Redis OSS Cluster son aproximadamente un cuarto más largos que los de las versiones anteriores.</p> <p>Esta versión incluye una consola de administración rediseñada que utiliza terminología en consonancia con el uso del sector.</p> <p>Para más información, consulte los siguientes temas:</p> <ul style="list-style-type: none"> • Comparación de Memcached y Redis OSS • ElastiCache componentes y características: tenga en cuenta las secciones sobre los nodos, las particiones, los clústeres y la reproducción. • ElastiCache Terminología (Redis OSS) 	12 de octubre de 2016

Cambio	Descripción	Fecha de modificación
Respaldo con el tipo de nodo M4	<p>ElastiCache añade compatibilidad con la familia M4 de tipos de nodos en la mayoría AWS de las regiones compatibles con. ElastiCache Puede adquirir nodos del tipo M4 como nodos de caché reservados o en diferido. Para más información, consulte los siguientes temas:</p> <ul style="list-style-type: none"> • Tipos de nodos de caché compatibles • Parámetros específicos del tipo de nodo de Redis OSS 	3 de agosto de 2016
Respaldo con la región Mumbai	<p>ElastiCache añade soporte para la región de Asia Pacífico (Bombay). Para más información, consulte los siguientes temas:</p> <ul style="list-style-type: none"> • Tipos de nodos de caché compatibles • Parámetros específicos del tipo de nodo de Redis OSS 	27 de junio de 2016
Exportación de instantáneas	<p>ElastiCache añade la posibilidad de exportar una OSS instantánea de Redis para que pueda acceder a ella desde fueraElastiCache. Para más información, consulte los siguientes temas:</p> <ul style="list-style-type: none"> • Exportación de una copia de seguridad en la Guía del ElastiCache usuario de Amazon • CopySnapshot en Amazon ElastiCache API Reference 	26 de mayo de 2016
Escalado vertical del tipo de nodo	<p>ElastiCache añade la posibilidad de ampliar el tipo de OSS nodo de Redis. Para obtener más información, consulte Escalado ElastiCache (Redis OSS) .</p>	24 de marzo de 2016

Cambio	Descripción	Fecha de modificación
Simplificación de la actualización del motor	ElastiCache añade la posibilidad de actualizar fácilmente el motor de OSS caché de Redis. Para obtener más información, consulte Versiones del motor y actualizaciones .	22 de marzo de 2016
Respaldo con los tipos de nodo R3	ElastiCache añade compatibilidad con los tipos de nodos R3 en la región de China (Pekín) y en la región de Sudamérica (São Paulo). Para obtener más información, consulte Tipos de nodos de caché compatibles .	16 de marzo de 2016
Acceso ElastiCache mediante una función Lambda	Se ha añadido un tutorial sobre cómo configurar una función Lambda para acceder a ella ElastiCache en Amazon. VPC Para obtener más información, consulte ElastiCache tutoriales y vídeos .	12 de febrero de 2016
Support para Redis 2.8.24 OSS	ElastiCache añade compatibilidad con la OSS versión 2.8.24 de Redis con mejoras añadidas desde Redis 2.8.23. OSS Entre las mejoras se incluyen varias correcciones de errores y respaldo con el registro de direcciones de acceso a la memoria incorrectas. Para más información, consulte los siguientes temas: <ul style="list-style-type: none"> • ElastiCache (Redis OSS) versión 2.8.24 (mejorada) • OSSNotas de la versión 2.8 de Redis 	20 de enero de 2016
Soporte para la región Asia-Pacífico (Seúl).	ElastiCache añade compatibilidad con la región Asia Pacífico (Seúl) (ap-northeast-2) con los tipos de nodos t2, m3 y r3.	6 de enero de 2016

Cambio	Descripción	Fecha de modificación
Cambio de ElastiCache consola de Amazon.	Como las OSS versiones más recientes de Redis ofrecen una experiencia de usuario mejor y más estable, OSS las versiones 2.6.13, 2.8.6 y 2.8.19 de Redis ya no aparecen en la consola de administración. ElastiCache Para obtener más opciones y más información, consulte Versiones compatibles ElastiCache (Redis OSS) .	15 de diciembre de 2015
Support para Redis OSS 2.8.23.	ElastiCache añade compatibilidad con la OSS versión 2.8.23 de Redis con mejoras añadidas desde Redis 2.8.22. OSS Entre las mejoras se incluye corrección de errores y respaldo con el nuevo parámetro <code>close-on-slave-write</code> que, si está habilitado, desconecta a los clientes que intentan escribir en una réplica de solo lectura. Para obtener más información, consulte ElastiCache (Redis OSS) versión 2.8.23 (mejorada) .	13 de noviembre de 2015

Cambio	Descripción	Fecha de modificación
Support para Redis OSS 2.8.22.	<p>ElastiCache añade compatibilidad con la OSS versión 2.8.22 de Redis con mejoras y mejoras ElastiCache añadidas desde la versión 2.8.21. Entre las mejoras, se incluyen las características siguientes:</p> <ul style="list-style-type: none"><li data-bbox="402 506 1109 800">• Implementación de un proceso de almacenamiento sin ramificaciones que permite almacenar contenido correctamente cuando una situación de escasa memoria disponible podría provocar errores en la operación de almacenamiento con ramificaciones.<li data-bbox="402 835 1109 940">• Métricas adicionales, y. CloudWatch SaveInProgressReplicationBytes<li data-bbox="402 976 1109 1123">• Para habilitar las sincronizaciones parciales, el OSS parámetro Redis <code>repl-backlog-size</code> ahora se aplica a todos los clústeres. <p>Para obtener una lista completa de los cambios, así como información adicional, consulte ElastiCache (Redis OSS) versión 2.8.22 (mejorada).</p> <p>Esta versión de la documentación incluye una reorganización de la documentación y la eliminación de la documentación de la interfaz de línea de ElastiCache comandos (CLI). Para el uso de la línea de comandos, consulte la línea de AWS comandos de ElastiCache.</p>	28 de septiembre de 2015

Cambio	Descripción	Fecha de modificación
Support para Redis 2.8.21 OSS	ElastiCache añade compatibilidad con la OSS versión 2.8.21 de Redis y las mejoras de Redis desde la versión 2.8.19. OSS Esta versión de Redis incluye varias correcciones de erroresOSS. Para obtener más información, consulte las notas de la versión OSS 2.8 de Redis .	29 de julio de 2015
Tema nuevo: Acceder ElastiCache desde el exterior AWS	Se agregó un nuevo tema sobre cómo acceder a ElastiCache los recursos desde el exterior AWS. Para obtener más información, consulte Acceder ElastiCache desde el exterior AWS .	9 de julio de 2015
Incorporación de mensajes de reemplazo de nodos	ElastiCache agrega tres mensajes relacionados con el reemplazo programado de nodos ElastiCache: NodeReplacementScheduledNodeReplacementRescheduled, yElastiCache:NodeReplacementCanceled. Para obtener más información y las medidas que puede tomar cuando se programa el reemplazo ElastiCache de un nodo, consulte Notificaciones de eventos y Amazon SNS .	11 de junio de 2015

Cambio	Descripción	Fecha de modificación
Support for Redis OSS v. 2.8.19.	<p>ElastiCache añade compatibilidad con la OSS versión 2.8.19 de Redis y mejoras en Redis desde la versión 2.8.6. OSS Este apoyo incluye respaldo con:</p> <ul style="list-style-type: none"> • La estructura de HyperLogLog datos, con los comandos de OSS Redis, y. PFADD PFCOUNT PFMERGE • Consultas de rangos lexicográficos con los nuevos comandos ZRANGEBYLEX, y ZLEXCOUNT. ZREMRANGEBYLEX • Se introdujeron varias correcciones de errores, como impedir que un nodo principal enviara datos obsoletos a los nodos de réplica al fallar el nodo principal SYNC cuando un proceso secundario de guardado en segundo plano (bgsave) finalizaba inesperadamente. <p>Para obtener más información HyperLogLog, consulte la OSS nueva estructura de datos de Redis: la. HyperLogLog</p> <p>Para obtener más información sobre PFADD, y PFCOUNT PFMERGE, consulte la OSS documentación de Redis y haga clic en. HyperLogLog</p>	11 de marzo de 2015
Respaldo con las etiquetas de asignación de costos	<p>ElastiCache añade compatibilidad con las etiquetas de asignación de costes. Para obtener más información, consulte Monitoreo de costos con etiquetas de asignación de costos.</p>	9 de febrero de 2015

Cambio	Descripción	Fecha de modificación
Support for AWS GovCloud (US-West) Region	ElastiCache añade soporte para la región AWS GovCloud (EEUU-Oeste) (us-gov-west-1).	29 de enero de 2015
Soporte para la región de Europa (Fráncfort)	ElastiCache añade soporte para la región de Europa (Fráncfort) (eu-central-1).	19 de enero de 2015
Soporte Multi-AZ para grupos de replicación de Redis OSS	ElastiCache añade compatibilidad con Multi-AZ desde el nodo principal a una réplica de lectura en un grupo de replicación de OSS Redis. ElastiCache supervisa el estado del grupo de replicación. Si el principal falla, convierte ElastiCache automáticamente una réplica en principal y, a continuación, la reemplaza. Para obtener más información, consulte Minimizar el tiempo de inactividad en ElastiCache (Redis OSS) con Multi-AZ .	24 de octubre de 2014
AWS CloudTrail Se admite el registro de API llamadas	ElastiCache agrega soporte para su uso AWS CloudTrail para registrar todas las ElastiCache API llamadas. Para obtener más información, consulte Registro de llamadas a la API de Amazon ElastiCache con AWS CloudTrail .	15 de septiembre de 2014
Respaldo con nuevos tamaños de instancias	ElastiCache añade compatibilidad con instancias adicionales de uso general (T2). Para obtener más información, consulte Configuración de los parámetros de motor mediante los grupos de parámetros .	11 de septiembre de 2014
Respaldo con nuevos tamaños de instancias	ElastiCache añade soporte para instancias adicionales de uso general (M3) e instancias optimizadas para memoria (R3). Para obtener más información, consulte Configuración de los parámetros de motor mediante los grupos de parámetros .	1 de julio de 2014

Cambio	Descripción	Fecha de modificación
Backup y restauración para clústeres de Redis OSS	En esta versión, ElastiCache permite a los clientes crear instantáneas de sus clústeres de Redis y crear nuevos OSS clústeres con estas instantáneas. Una copia de seguridad es una copia del clúster en un momento específico y consta de los metadatos del clúster y de todos los datos de la memoria caché de Redis. OSS Las copias de seguridad se almacenan en Amazon S3 y los clientes pueden restaurar los datos a partir de una instantánea en un nuevo clúster en cualquier momento. Para obtener más información, consulte Instantánea y restauración .	24 de abril de 2014
Redis 2.8.6 OSS	ElastiCache es compatible con Redis OSS 2.8.6, además de Redis 2.6.13. OSS Con Redis OSS 2.8.6, los clientes pueden mejorar la resiliencia y la tolerancia a los errores de las réplicas de lectura, ya que admite la resincronización parcial y un número mínimo de réplicas de lectura definido por el usuario que debe estar disponible en todo momento. Redis OSS 2.8.6 también ofrece soporte completo para que los clientes puedan recibir notificaciones de los eventos que se publish-and-subscribe produzcan en el servidor.	13 de marzo de 2014

Cambio	Descripción	Fecha de modificación
Motor de caché Redis OSS	<p>ElastiCache ofrece el software de motor de OSS caché Redis, además de Memcached. Los clientes que utilizan Redis actualmente OSS pueden «sembrar» un nuevo clúster de OSS caché de ElastiCache Redis con los datos existentes de un archivo de OSS instantáneas de Redis, lo que facilita la migración a un entorno gestionado. ElastiCache</p> <p>Para respaldar las capacidades de OSS replicación de Redis, ahora admite grupos de ElastiCache API replicación. Los clientes pueden crear un grupo de replicación con un nodo de OSS caché de Redis principal y agregar uno o más nodos de réplica de lectura que permanecen sincronizados automáticamente con los datos de la caché del nodo principal. Las aplicaciones que realizan un uso intensivo de las operaciones de lectura pueden descargarse a una réplica de lectura, lo que reduce la carga en el nodo primario. Las réplicas de lectura también protegen frente a la pérdida de datos en caso de un posible error de nodo de caché primario.</p>	3 de septiembre de 2013
Support para Amazon Virtual Private Cloud predeterminado (VPC)	<p>En esta versión, ElastiCache está totalmente integrado con Amazon Virtual Private Cloud (VPC). Para los nuevos clientes, los clústeres de caché se crean en Amazon de forma VPC predeterminada. Para obtener más información, consulte Seguridad de ElastiCache y Amazon VPC.</p>	8 de enero de 2013

Cambio	Descripción	Fecha de modificación
Support para Amazon Virtual Private Cloud (VPC)	En esta versión, ElastiCache los clústeres se pueden lanzar en Amazon Virtual Private Cloud (VPC). De forma predeterminada, los clústeres de caché de los nuevos clientes se crean VPC automáticamente en Amazon; los clientes actuales pueden migrar a Amazon VPC a su propio ritmo. Para obtener más información, consulte Seguridad de ElastiCache y Amazon VPC .	20 de diciembre de 2012
Nuevos tipos de nodo de caché	Esta versión ofrece cuatro tipos de nodos de caché adicionales.	13 de noviembre de 2012
Nodos de caché reservados	Esta versión ofrece respaldo con los nodos de caché reservados.	5 de abril de 2012
Nueva guía	Esta es la primera versión de la Guía del ElastiCache usuario de Amazon.	22 de agosto de 2011

AWS Glosario

Para obtener la AWS terminología más reciente, consulte el [AWS glosario](#) de la Glosario de AWS Referencia.

Las traducciones son generadas a través de traducción automática. En caso de conflicto entre la traducción y la versión original de inglés, prevalecerá la versión en inglés.