使用者指南

Amazon Elastic Compute Cloud

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

Amazon Elastic Compute Cloud: 使用者指南

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

Amazon 的商標和商業外觀不得用於任何非 Amazon 的產品或服務，也不能以任何可能造成客戶混淆、任何貶低或使 Amazon 名譽受損的方式使用 Amazon 的商標和商業外觀。所有其他非 Amazon 擁有的商標均為其各自擁有者的財產，這些擁有者可能附屬於 Amazon，或與 Amazon 有合作關係，亦或受到 Amazon 贊助。

Table of Contents

什麼是 Amazon EC2？ ... 1
功能 ... 1
相關服務 ... 2
存取 EC2 ... 3
定價 ... 4
估算、計費和成本最佳化 ... 5
資源 ... 6
入門教學課程 ... 7
步驟 1：啟動執行個體 ... 8
步驟 2：連線到您的執行個體 ... 9
步驟 3：清理您的執行個體 ... 12
後續步驟 ... 13
最佳實務 ... 14
Amazon Machine Image ... 16
AMI特徵 ... 17
啟動許可 ... 17
Root device type (根設備類型) ... 18
確定AMI根設備類型 ... 19
虛擬化類型 ... 20
尋找一個 AMI ... 21
AMIs在支付 AWS Marketplace ... 29
出售您的 AMI AWS Marketplace ... 30
尋找付費 AMI ... 30
購買付費 AMI ... 32
擷取產品代碼 ... 32
使用已支付支援 ... 33
已付款和支援帳單 AMIs ... 33
管理您的 訂閱 ... 34
AMI生命週 ... 34
創建一個 AMI ... 35
建立執行個體存放區支援 AMI ... 42
創建一個AMI使用視窗系統 ... 79
複製 AMI ... 94
存儲和恢復 AMI ... 103

檢查上AMI次使用的時間 112
棄用 AMI 113
禁用 AMI 120
取消註冊 AMI 126
開機模式 133
UEFI開機模式的需求 134
AMI開機模式參數 136
執行個體類型開機模式 138
執行個體開機模式 142
作業系統開機模式 144
設定AMI開機模式 146
UEFI變數 151
UEFI安全開機 151
AMI 加密 165
執行個體啟動案例 165
映像複製案例 168
瞭解共用 AMIs 170
已驗證的供應商 170
尋找共用 AMIs 171
準備使用 Linux AMIs 的共用功能 174
讓您的AMI公開 175
瞭解封鎖公開存取 178
與組織共同AMI使用 OUs 188
AMI與特定 AWS 帳戶共用 197
取消與您的帳戶AMI共享 201
建立共用 Linux 的建議 AMIs 203
監控AMI事件 208
事件詳細資訊 209
available 事件 210
failed 事件 210
deregistered 事件 211
disabled 事件 212
了解 AMI 帳單 212
AMI 帳單欄位 213
尋找 AMI 帳單資訊 215
驗證帳單上的 AMI 費用 217

AMI配額 218
要求增加配額 AMIs 218
執行個體 220
執行個體類型 221
可用的執行個體類型 222
硬體規格 223
AMI虛擬化類型 225
尋找執行個體類型 225
EC2實體類型尋找工 230
Compute Optimizer 建議 232
實例類型變更 235
爆量效能執行個體 242
GPU 執行個體 288
Mac 執行個體 302
EBS最佳化 326
CPU選項 398
AMD SEV-SNP 539
處理器狀態控制 545
帳單和購買選項 547
隨需執行個體 547
預留執行個體 550
Spot 執行個體 607
專用執行個體 691
專用執行個體 740
Capacity Reservations 746
啟動範本 820
限制 821
許可 822
控制啟動執行個體 828
建立 830
修改 (管理版本) 844
Delete 848
啟動執行個體 850
例證參數參考 852
使用啟動執行個體精靈進行啟動 864
使用啟動範本啟動 867

從現有執行個體啟動 873
從啟動 AWS Marketplace AMI 875
連接至您的執行個體 879
獲取所需的實例詳細信息 879
找出私有金鑰並設定許可 881
(選用) 取得執行個體指紋 882
使用 Connect 到您的 Linux 執行個體 SSH 884
使用 Connect 至您的 Windows 執行個體 RDP 898
使用工作階段管理員進行連接 907
使 Connect「執EC2行個體 Connect 908
使用EC2執行個體連線端點連 941
實例狀態變更 966
依執行個體狀態計費 966
待處理執行 968
停止的執行 968
休眠執行個體 968
將執行個體 969
終止的實例 969
實例狀態之間的差異 970
停止和啟動 971
休眠 979
重新開機 1006
終止 1008
淘汰 1017
執行個體恢復 1021
執行個體中繼資料 1030
執行個體中繼資料分類 1031
動態資料分類 1041
訪問實例元數據 1042
設定IMDS選項 1076
啟動時執行命令 1100
範例：AMI啟動索引值 1123
偵測主機是否為EC2執行個體 1127
檢查 執行個體身分文件 1127
檢查系統 UUID 1128
檢查系統虛擬機器產生識別符 1129

執行個體身分文件 1134
檢索實例身份證明文件 1135
驗證執行個體身份證 1137
公開憑證 1147
時鐘同步 1200
閏秒 1201
使用本地 Amazon 時間同步服務 1202
使用公共的 Amazon 時間同步服務 1213
比較 Linux 執行個體的時間戳記 1215
變更執行個體的時區 1216
管理裝置驅動 1218
網絡驅動 1219
顯示卡驅動 1219
儲存裝置驅動 1219
AMD驅動 1220
NVIDIA司機 1226
在視窗上安裝ENA驅動程式 1260
窗戶光伏驅動器 1275
AWS 視窗NVMe驅動 1304
設定執行個體 1311
視窗特定的系統設定 1312
啟動代理程式 1313
EC2快速啟動 1457
變更視窗管理員密碼 1477
新增視窗系統元件 1478
在 Windows 上安裝 WSL 1483
升級 Windows 執行個體 1484
執行就地升級 1485
執行自動升級 1489
移轉至目前一代的執行個體類型 1499
故障診斷升級 1506
教學課程：將EC2執行處理連線至 RDS 1507
教學課程目標 1507
Context 1507
架構 1508
考量事項 1509

教學課程所需時間 1510
成本 1510
選項 1：使用EC2主控台自動連線 1510
選項 2：使用RDS主控台自動連線 1521
選項 3：手動連接 1530
機群 1540
功能和優勢 1540
使用哪種艦隊方法？ 1541
組態選項 1542
請求類型 1543
消費限額 1570
屬性型執行個體類型選項 1572
執行個體權重 1604
分配策略 1606
容量重新平衡 1612
Capacity Reservations 1616
與EC2艦隊合作 1617
EC2艦隊請求狀態 1617
建立EC2艦隊 1618
標記EC2艦隊 1631
描述EC2艦隊 1633
修改EC2叢集 1637
刪除EC2叢集 1638
使用現貨艦隊 1643
Spot Fleet 請求狀態 1643
建立 Spot 機群 1644
標記 Spot Fleet 1660
描述一個現貨艦隊 1669
修改 Spot Fleet 請求 1670
取消 (刪除) Spot 叢集請求 1671
Spot Fleet 的自動擴展 1673
監控您的車隊 1682
使用以下方式監控您 CloudWatch 1682
使用以下方式監控您 EventBridge 1685
教學課程 1702
教學課程：將EC2叢集設定為使用執行個體加權 1703

教學課程：將EC2叢集設定為使用隨需執行個體做為主要容量 1706
教學課程：設定EC2叢集以使用目標容量保留啟動隨需執行個 1708
教學課程：設定EC2叢集以將執行個體啟動至容量區塊 1714
EC2叢集的範例CLI組態 1716
範例 1：啟動 競價型執行個體 做為預設購買選項 1717
範例 2：啟動 隨需執行個體 做為預設購買選項 1717
範例 3：啟動 隨需執行個體 做為主要容量 1718
範例 4：使用多個容量保留啟動隨需執行個體 1718
範例 5：當目標容量總計超過未使用的容量保留數時，使用容量保留啟動隨需執行個體 1722
範例 6：使用目標容量保留啟動隨需執行個體 1725
範例 7：設定容量重新平衡以啟動替代 Spot 執行個體 1728
範例 8：在容量最佳化叢集中啟動 Spot 執行個體 1730
範例 9：在具有優先順序的容量最佳化叢集中啟動 Spot 執行個體 1731
範例 10：在 price-capacity-optimized 叢集中啟動 Spot 執行個體 1732
範例 11：設定以屬性為基礎的執行個體類型選取 1733
範例CLI組態競價型叢集 1734
範例 1：使用區域中最低價格的可用區域或子網來啟動 競價型執行個體 1735
範例 2：使用指定清單中最低價格的可用區域或子網來啟動 競價型執行個體 1736
範例 3：使用指定清單中最低價格的執行個體類型來啟動 競價型執行個體 1738
範例 4。覆寫請求的價格 1739
範例 5：使用多樣化分配策略來啟動 Spot Fleet 1741
範例 6：使用執行個體權重來啟動 Spot Fleet 1744
範例 7：使用隨需容量來啟動 Spot Fleet 1745
範例 8：設定容量重新平衡以啟動替代 競價型執行個體 1746
範例 9：在容量最佳化機群中啟動 Spot 執行個體 1748
範例 10：使用優先順序在容量最佳化機群中啟動 Spot 執行個體 1749
範例 11：在 priceCapacityOptimized 叢集中啟動 Spot 執行個體 1750
範例 12：設定屬性型執行個體類型選取 1751
機群配額 1752
請求增加目標容量的配額 1753
聯網 1755
區域 (Region) 和區域 (Zone) 1756
區域 1757
可用區域 1759
本機區域 1762
Wavelength 區域 1764

AWS Outposts 1765
執行個體 IP 定址 1766
私人IPv4地址 1767
公共IPv4地址 1768
公共IPv4位址最佳化 1769
IPv6地址 1770
EC2實例主機名稱 1771
Link-local 地址 1771
IPv4地址 1772
IPv6地址 1774
多個 IP 地址 1777
視窗上的多個IPv4位址 1786
執行個體主機名稱類型 1793
EC2主機名稱的類型 1793
在哪裡可以找到資源名稱和 IP 名稱 1795
在資源名稱和 IP 名稱之間進行選擇 1796
變更基於資源的命名選項 1797
使用自有 IP 地址 1798
BYOIP定義 1799
需求與配額 1800
區域可用性 1800
Local Zone 可用性 1801
必要條件 1801
建立您的地址範圍 1809
使用您的地址範圍 1817
彈性 IP 地址 1818
彈性 IP 地址定價 1819
彈性 IP 地址基本概念 1819
彈性 IP 地址配額 1820
建立彈性 IP 地址的關聯 1820
傳輸彈性 IP 位址 1825
釋出彈性 IP 地址 1829
使用反向DNS的電子郵件應用 1831
網路介面 1833
網路介面概念 1835
網路卡 1836

每個網路介面的 IP 位址 1838
建立網路介面 1840
管理 IP 地址 1844
修改網路介面屬性 1846
多個網路介面 1848
申請者受管的網路介面 1851
前綴委派 1853
刪除網路介面 1860
網路頻寬 1861
可用執行個體頻寬 1861
監控執行個體頻寬 1864
增強型網路 1864
彈性網路介面卡 (ENA) 1865
ENA快遞 1879
Intel 82599 VF 1900
監控網路效能 1911
疑難排解 ENA 1920
視窗ENA的疑難排解 1933
改善 Linux 上的網路延遲 1949
硝基性能考量 1952
在視窗上最佳化網路效能 1957
Elastic Fabric Adapter 1958
EFA基礎 1959
支援的介面和程式庫 1960
支援的執行個體類型 1961
支援的作業系統 1962
EFA限制 1963
EFA定價 1963
EFA加速執行個體 1963
開始使用EFA和 MPI 1968
開始使用EFA和 NCCL 1984
建立並附加 EFA 2006
分離並刪除 EFA 2008
監控 EFA 2008
驗證EFA安裝程式 2009
執行個體拓撲 2021

運作方式 2022
必要條件 2025
範例 2027
置放群組 2038
置放策略 2039
建立置放群組 2044
變更例證位置 2045
刪除置放群組 2047
共用置放群組 2048
放置群組 AWS Outposts 2052
網絡 MTU 2053
巨型框MTU架 2054
路徑MTU探索 2055
MTU為您的執行個體設定 2056
疑難排解 2061
Virtual Private Cloud (VPC) 2061
您的預設 VPCs 2062
非預設 VPCs 2063
網際網路存取 2063
共用子網路 2063
IPv6只有子網 2064
安全 2065
資料保護 2066
Amazon EBS 數據安全 2067
靜態加密 2067
傳輸中加密 2068
基礎架構安全 2069
網路隔離 2070
實體主機上的隔離 2070
控制網路流量 2070
恢復能力 2072
法規遵循驗證 2073
身分與存取管理 2074
身分型政策 2075
政策範例 API 2084
主控台的範例政策 2124

AWS 受管政策 2135
IAM角色 2138
更新管理 2149
Windows 執行個體的最佳做法 2149
高階安全性最佳做法 2150
更新管理 2150
組態管理 2152
變更管理 2153
Amazon EC2 視窗執行個體的稽核與責任 2153
金鑰對 2154
建立金鑰對 2155
標記金鑰對 2163
描述您的金鑰配對 2165
刪除您的金鑰對 2172
在 Linux 執行個體上新增或取代公開金鑰 2173
驗證指紋 2175
安全群組 2177
概要 2178
建立安全群組 2179
變更執行個體的安全性群組 2181
刪除安全群組 2183
連線追蹤 2184
不同使用案例的安全群組規則 2189
硝基 TPM 2194
要求 2195
AMI為硝基啟用 Linux 系統 TPM 2197
驗證AMI已啟用硝基 TPM 2198
啟用或停止使用硝基 TPM 2199
確認執行個體已啟用 Nitro TPM 2199
擷取公開認可金鑰 2201
Windows 執行個體的認證保護 2202
必要條件 2202
啟動支援的執行個體 2203
禁用內存完整性 2204
開啟認證保全 2205
確認認證保全正在執行 2206

AWS PrivateLink 2207
建立介面VPC端點 2207
建立端點政策 2208
儲存 2210
AWS 儲存定價 2211
Amazon EBS 2211
EBS音量限制 2212
Amazon EC2 實例商店 2215
資料持久性 2217
例項存放區限制 2218
SSD執行個體儲存體 2221
新增執行個體存放區磁碟區 2224
為 M1 和 C1 執行個體啟用交換磁碟區 2229
初始化實例存儲卷 2232
根磁碟區 2234
Amazon EBS 支持的實例 2234
執行個體商店支援執行個體 (僅限 Linux 執行個體) 2236
執行個體終止後保留根磁碟區 2237
取代根磁碟區 2240
磁碟區的裝置名稱 2249
可用裝置名稱 2249
裝置名稱考量 2251
區塊型儲存設備對應 2252
區塊型儲存設備映射概念 2253
將區塊裝置對應新增至 AMI 2256
將塊設備映射添加到實例 2259
將磁碟映射至磁碟區 2266
將NVME磁碟對映至磁碟區 2267
將非磁碟對映至NVME磁碟區 2272
預防未完成的寫入 2281
支援的區塊大小 2282
要求 2282
檢查實例支持 2283
設定工作量 2284
視窗VSSEBS快照 2286
什麼是VSS？ 2286

VSS基於 Amazon EBS 快照解決方案的運作方式 2287
VSS前提 2288
建立VSS快照 2298
疑難排解VSS快 2306
還原EBS磁碟區 2310
版本歷史記錄 2311
物件儲存、檔案儲存和檔案快取 2314
Amazon S3 2315
Amazon EFS 2317
Amazon FSx 2320
Amazon 文件緩存 2326
管理資源 2327
為您的資源選擇一個地區 2327
尋找您的資源 2328
主控台步驟 2328
CLI和API步驟 2334
全域檢視 (跨區域) 2337
Amazon EC2 全球視圖 2337
標記您的 資源 2340
標籤基本概念 2340
標記您的 資源 2341
標籤限制 2342
標籤與存取管理 2343
標記您的資源以便計費 2343
標記資源權限 2344
新增和移除標籤 2347
按標籤過濾資源 2350
使用執行個體中繼資料檢 2351
Service Quotas 2356
檢視您目前的配額 2356
請求增加 2357
使用通訊埠 25 傳送的電子郵件限制 2357
監控 資源 2358
監控您的執行個體狀態 2359
狀態檢查 2359
狀態變更事件 2367

排程事件 ... 2369
使用監控執行個體 CloudWatch ... 2399
實例警報 ... 2400
管理詳細監控 ... 2401
CloudWatch 度量 ... 2404
安裝和設定 CloudWatch 代理程式 ... 2419
指標的統計資料 ... 2423
檢視監控圖 ... 2431
建立警示 ... 2431
建立警示以停止、終止、重新啟動或復原執行個體 ... 2432
使用自動化 EventBridge ... 2444
Amazon EC2 事件類型 ... 2444
使用記錄API呼叫 CloudTrail ... 2445
Amazon EC2 API 管理事件 CloudTrail ... 2446
Amazon EC2 API 活動示例 ... 2446
使用EC2執行個體 Connect 線建立的稽核 ... 2448
監視器。 NET以及SQL伺服器應用 ... 2449
追蹤您的免費方案用量 ... 2450
疑難排解 ... 2453
實例啟動問題 ... 2453
無效裝置名稱 ... 2454
超過執行個體限制 ... 2454
執行個體容量不足 ... 2455
目前並不支援要求的組態。請查看文件以了解支援的組態。 ... 2455
立刻終止執行個體 ... 2456
許可不足 ... 2457
Windows 啟動後不久CPU使用率高 (僅適用於 Windows 執行個體) ... 2458
實例停止問題 ... 2459
強制停止執行個體 ... 2459
(選擇性) 建立取代執行個體 ... 2460
實例終止問題 ... 2462
立刻終止執行個體 ... 2462
延遲的執行個體終止 ... 2463
仍顯示已終止的執行個體 ... 2463
錯誤：執行個體可能無法終止。修改其 'disableApiTermination' 實例屬性 ... 2463
執行個體自動啟動或終止 ... 2463

無法訪問的實例 2464
重新啟動執行個體 2464
執行個體主控台輸出 2464
擷取無法連線執行個體的螢幕擷取畫面 2465
Windows 執行個體的常見螢幕擷 2467
當主機電腦故障時的執行個體復原 2476
執行個體SSH問題 2476
連線問題的常見原因 2477
連線到執行個體時發生錯誤：連線時間逾時 2479
錯誤：無法載入金鑰 ... 期待：ANYPRIVATEKEY 2481
錯誤：伺服器無法辨識使用者金鑰 2482
錯誤：許可遭拒或 [執行個體] 連接埠 22 已關閉連線 2484
錯誤：未受保護的私有金鑰檔案 2486
錯誤:私密金鑰必須以「-------」開頭，並以「BEGINRSAPRIVATEKEY----------」結尾 END RSA PRIVATE KEY 2488
錯誤：伺服器拒絕了我們的金鑰，或是沒有可用的支援驗證方法 2488
無法對執行個體執行 Ping 動作 2489
錯誤：伺服器意外關閉網路連線 2489
錯誤：EC2執行個體 Connect 的主機金鑰驗證失敗 2489
無法使用執行個體 Connect 至 Ubuntu EC2 執行個體 2491
我的私有金鑰遺失。如何連線到我的執行個體？ 2492
Linux 執行個體狀態檢查失敗 2498
檢閱狀態檢查資訊 2499
擷取系統日誌 2499
解決 Linux 執行個體的系統記錄錯誤 2500
記憶體不足：終止程序 2501
ERROR: mmu_update 失敗 (記憶體管理更新失敗) 2502
I/O 錯誤 (區塊型儲存設備故障) 2503
I/OERROR：既不是本地磁盤也不是遠程磁盤（破碎的分佈式塊設備） 2504
request_module：失控迴圈 modprobe (在舊版 Linux 上建立舊核心 modprobe 的迴圈) 2505
「FATAL: 內核太舊了」和「fsck：嘗試打開 /dev 時沒有這樣的文件或目錄」（內核和不匹配）AMI 2506
「FATAL: 無法載入 /lib/ 模組" 或 "BusyBox" (遺失核心模組) 2507
ERROR無效的內核（EC2不兼容的內核） 2509
fsck：嘗試開啟時找不到此等檔案或目錄… (找不到檔案系統) 2510
掛載檔案作業系統的一般錯誤 (掛載失敗) 2512

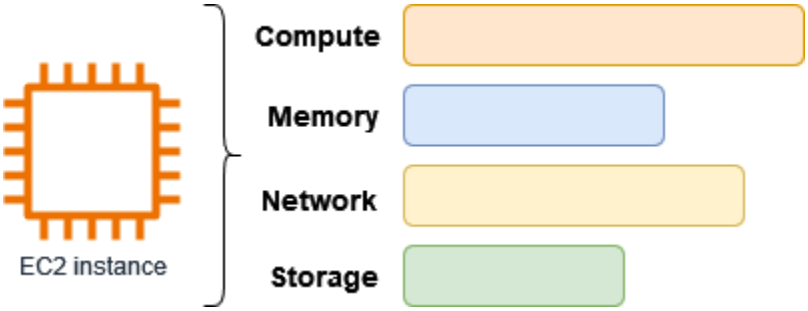
VFS: 無法在未知的區塊上掛載根 fs (根檔案系統不符) 2514
錯誤：無法判定根設備的主/次編號… (根檔案系統/裝置不相符) 2516
XENBUS：沒有驅動程序的設備... 2517
…未進行檢查的天數，強制進行檢查 (需要進行檔案系統檢查) 2518
fsck 凍結於結束狀態… (缺少裝置) 2519
GRUB提示符（灰塵 >） 2520
啟動介面 eth0：裝置 eth0 具有與預期不同的MAC位址，而忽略。(硬式編碼MAC位址) 2522
無法載入SELinux策略。機器正處於強制執行模式。現在停止中。（SELinux配置錯誤） 2524
XENBUS：連接到設備的超時（Xenbus 超時） 2525
Linux 實例從錯誤的卷啟動 2526
視窗實例RDP問題 2528
遠端桌面無法連線到遠端電腦 2528
使用 macOS 用RDP戶端時發生錯誤 2531
RDP顯示黑屏而不是桌面 2531
無法使用非管理員的使用者從遠端登入至執行個體 2532
使用的遠程桌面問題故障 AWS Systems Manager 2532
在具有遠端登錄的EC2執行個體上啟用遠端桌面 2536
我的私有金鑰遺失。如何連接到我的 Windows 執行個體？ 2537
視窗實例啟動問題 2537
「未提供密碼」 2538
「尚未提供密碼」 2539
「無法擷取 Windows 密碼」 2539
等待中繼資料服務 2539
無法啟動 Windows 2543
Windows 不是正版 (0x80070005) 2545
無終端機伺服器的授權伺服器可提供授權 2545
「部分設定是由您的組織所管理」 2545
視窗實例問題 2546
EBS磁碟區不會在視窗伺服器 2016 和 2019 上初始化 2546
將 EC2 Windows 執行個體啟動至目錄服務還原模式 (DSRM) 2548
執行個體遺失網路連線能力或排程任務未如預期執行 2550
無法取得主控台輸出 2550
網路不提供 Windows Server 2012 R2 2551
磁碟簽章衝突 2551
重設管理員密碼 2552
使用 EC2Launch v2 重置密碼 2553

使用重設密碼 EC2Launch 2558
使用重設密碼 EC2Config 2563
疑難排解系統問題 2568
EC2Rescue針對 Linux 執行個體 2570
安裝 EC2Rescue 2570
執行EC2Rescue命令 2574
開發EC2Rescue模組 2576
EC2Rescue視窗執行個體 2582
疑難排解使 EC2Rescue GUI 2583
疑難排解使 EC2Rescue CLI 2588
疑難排解使用EC2Rescue和 Systems Manager 2596
EC2序列主控台 2599
必要條件 2600
設定EC2序列主控台的存取權 2607
Connect 至EC2序列主控台 2615
中斷與EC2序列主控台的連線 2623
使用EC2序列主控台疑難排解執行個體 2624
傳送診斷中斷 2632
支援的執行個體類型 2633
必要條件 2633
傳送診斷中斷 2637
文件歷史紀錄 2638
2018 年及更早版本的歷史 2657
..... mmdclxxvi

什麼是 Amazon EC2？

Amazon 彈性運算雲（AmazonEC2）在 Amazon Web Services（AWS）雲中提供按需，可擴展的計算容量。使用 Amazon 可EC2降低硬體成本，因此您可以更快地開發和部署應用程式。您可以使用 Amazon 根據需EC2要啟動任意數量或少量的虛擬伺服器、設定安全性和聯網，以及管理儲存。您可以新增容量 (擴充) 來處理需要大量運算的工作，例如每月或每年的例行程序，或是網站流量暴增。當使用量降低時，您可以再次減少容量 (縮減)。

EC2執行個體是 AWS 雲端中的虛擬伺服器。啟動EC2執行個體時，您指定的執行個體類型會決定執行個體可用的硬體。每種執行個體類型都提供不同的運算、記憶體、網路和儲存資源平衡。如需詳細資訊，請參閱 Amazon EC2 執行個體類型指南。



Amazon 的特點 EC2

Amazon EC2 提供以下高級功能：

執行個體

虛擬伺服器。

Amazon 機器圖像（AMIs）

供執行個體使用的預先設定範本，在其中封裝伺服器所需的元件 (包括作業系統和其他軟體)。

執行個體類型

適用於執行個體的各種組態CPU、記憶體、儲存、網路容量和圖形硬體。

Amazon EBS 卷

使用 Amazon 彈性區塊存放區 (AmazonEBS) 為您的資料提供持久性儲存磁碟區。

執行個體儲存體磁碟區

儲存磁碟區可供您停止、休眠或終止執行個體時所刪除的暫時性資料使用。

金鑰對

保護執行個體的登入資訊。 AWS 存儲公鑰，並將私鑰存儲在安全的地方。

安全群組

虛擬防火牆可讓您指定存取執行個體的協定、連接埠和來源 IP 範圍，以及執行個體可連線的目的地 IP 範圍。

Amazon EC2 支援商家或服務供應商處理、儲存和傳輸信用卡資料，並且已通過驗證符合支付卡產業 (PCI) 資料安全標準 (DSS)。如需有關 PCI DSS (包括如何要求 AWS PCI符合性 Package 件副本) 的詳細資訊，請參閱PCIDSS層級 1。

相關服務

與 Amazon 一起使用的服務 EC2

您可以在使用 Amazon 部署 AWS 服務 的執行個體搭配使用其他執行個體EC2。

Amazon EC2 Auto Scaling

協助確保您擁有正確數量的 Amazon EC2 執行個體可用來處理應用程式的負載。

AWS Backup

自動備份您的 Amazon EC2 執行個體和連接到它們的 Amazon EBS 磁碟區。

Amazon CloudWatch

監控您的執行個體和 Amazon EBS 磁碟區。

Elastic Load Balancing

自動將傳入應用程式流量分配至多個執行個體。

Amazon GuardDuty

偵測潛在未經授權或惡意使用您的EC2執行個體。

EC2 Image Builder

自動建立、管理和部署自訂、安全和up-to-date 伺服器影像。

AWS Launch Wizard

調整協力廠商應用程式的 AWS 資源大小、設定和部署資源，而不必手動識別和佈建個別 AWS 資源。

AWS Systems Manager

使用此安全 end-to-end管理解決方案在執行EC2個體上大規模執行作業。

其他運算服務

您可以使用其他 AWS 運算服務而不是使用 Amazon 來啟動執行個體EC2。

Amazon Lightsail

使用 Amazon Lightsail 建置網站或 Web 應用程式，這是一個雲端平台，可提供快速部署專案所需的資源，而且每月價格低廉且可預測。要比較 Amazon EC2 和 Lightsail，請參閱 Amazon Lightsail 或 Amazon。EC2

Amazon 彈性容器服務（AmazonECS）

在執行個體叢集上部署、管理和擴充容器化應用程式。EC2如需詳細資訊，請參閱選擇 AWS 容器服務。

Amazon Elastic Kubernetes Service（Amazon）EKS

在 AWS上執行您的 Kubernetes 應用程式。如需詳細資訊，請參閱選擇 AWS 容器服務。

訪問 Amazon EC2

您可以使用下列界面建立和管理 Amazon EC2 執行個體：

Amazon EC2 遊戲

用於建立和管理 Amazon EC2 執行個體和資源的簡單 Web 介面。如果您已註冊 AWS 帳戶，則可以登入 AWS Management Console 並從EC2主控台首頁選取EC2來存取 Amazon 主控台。

AWS Command Line Interface

可讓您使用命令列殼層中的命令與 AWS 服務互動。Windows、Mac 和 Linux 系統皆提供支援。如需有關 AWS CLI 的詳細資訊，請參閱 《AWS Command Line Interface 使用者指南》。您可以在命EC2令參考中找到 Amazon AWS CLI 命令。

AWS CloudFormation

Amazon EC2 支持使用 AWS CloudFormation. 您可以使用JSON或YAML格式建立描述 AWS 資源的範本，並為您 AWS CloudFormation 佈建和設定這些資源。您可以重複使用 CloudFormation

範本來多次佈建相同的資源，無論是在相同的區域和帳戶中，還是在多個區域和帳戶中。如需 Amazon 支援的資源類型和屬性的詳細資訊EC2，請參閱AWS CloudFormation 使用者指南中的EC2資源類型參考。

AWS SDKs

如果您偏好使用特定語言來建置應用程式，APIs而不是透過HTTP或提交要求HTTPS，請為軟體開發人員 AWS 提供程式庫、範例程式碼、教學課程和其他資源。這些程式庫提供的基本功能可自動化任務，例如密碼編譯簽署要求、重試要求，以及處理錯誤回應，讓您更容易入門。如需詳細資訊，請參閱 在 AWS上建置的工具。

AWS Tools for PowerShell

一 PowerShell 組以公開的功能為基礎建置的模組 AWS SDK for .NET。「工具」可 PowerShell 讓您從命令列對資 AWS 源執行作業的 PowerShell 指令碼。若要開始使用，請參閱《 使用者指南AWS Tools for Windows PowerShell》。您可以在指令程式參考中找到 Amazon EC2 的AWS Tools for PowerShell 指令程式。

查詢 API

Amazon EC2 提供了一個查詢API。這些請求是HTTP或使用HTTP動詞GET或名為POST的 Query 參數的HTTPS請求`Action`。如需有關 Amazon API 動作的詳細資訊EC2，請參閱 Amazon EC2 API 參考中的動作。

Amazon 定價 EC2

Amazon EC2 提供以下定價選項：

免費方案

您可以免費開始使EC2用 Amazon。若要探索免費方案選項，請參閱AWS 免費方案。

隨需執行個體

您只需要按秒支付使用執行個體的費用 (最少 60 秒)，無需長期購買或預付款。

Savings Plans

您可以承諾以USD每小時為單位的一致使用量，為期 1 或 3 年，以降低 Amazon EC2 成本。

預留執行個體

您可以承諾特定執行個體組態 (包括執行個體類型和區域)，期限為 1 或 3 年，以降低 Amazon EC2 成本。

Spot 執行個體

請求未使用的EC2執行個體，這可大幅EC2降低 Amazon 成本。

專用執行個體

使用完全專供您使用的實體EC2伺服器 (無論是隨選或作為 Savings Plan 的一部分)，降低成本。您可以使用現有繫結伺服器的軟體授權，並取得符合合規要求的協助。

On-Demand Capacity Reservations

在任何時間內為特定可用區域中的EC2執行個體保留運算容量。

以秒計費

從帳單中移除未使用分鐘和秒數的成本。

如需 Amazon 的完整費用和價格清單，以EC2及有關購買模式的詳細資訊，請參閱 Amazon EC2 定價。

估算、計費和成本最佳化

若要為您的 AWS 使用案例建立估算值，請使用 AWS Pricing Calculator.

若要估算將 Microsoft 工作負載轉換為使用部署開放原始碼和雲端原生服務的現代化架構的成本 AWS，請使用適用於 Microsoft 工作負載的現AWS 代化計算器。

若要查看您的帳單，請前往 AWS Billing and Cost Management 主控台中的帳單與成本管理儀表板。您的帳單內含用量報告的連結，可提供帳單的詳細資訊。若要深入瞭解 AWS 帳戶帳單，請參閱 B AWS illing and Cost Management 使用者指南。

如果您對帳 AWS 單、帳戶和事件有任何疑問，請聯絡 Sup AWS port 部門。

若要計算範本佈建環境的成本，請參閱雲端成本中心。計算佈建環境的成本時，請記得包括附帶成本，例如EBS磁碟區的快照儲存。

您可以使用最佳化 AWS 環境的成本、安全性和效能AWS Trusted Advisor。

您可以用 AWS Cost Explorer 來分析EC2執行個體的成本和用量。您可以檢視最近 13 個月的資料，並預測未來 12 個月的可能花費金額。有關更多信息，請參閱AWS Cost Management 用戶指南 AWS Cost Explorer中的分析成本。

資源

- Amazon EC2 功能
- AWS Re: 文章
- AWS 技能生成器
- AWS Support
- 動手教程
- 網頁寄存
- 視窗開啟 AWS

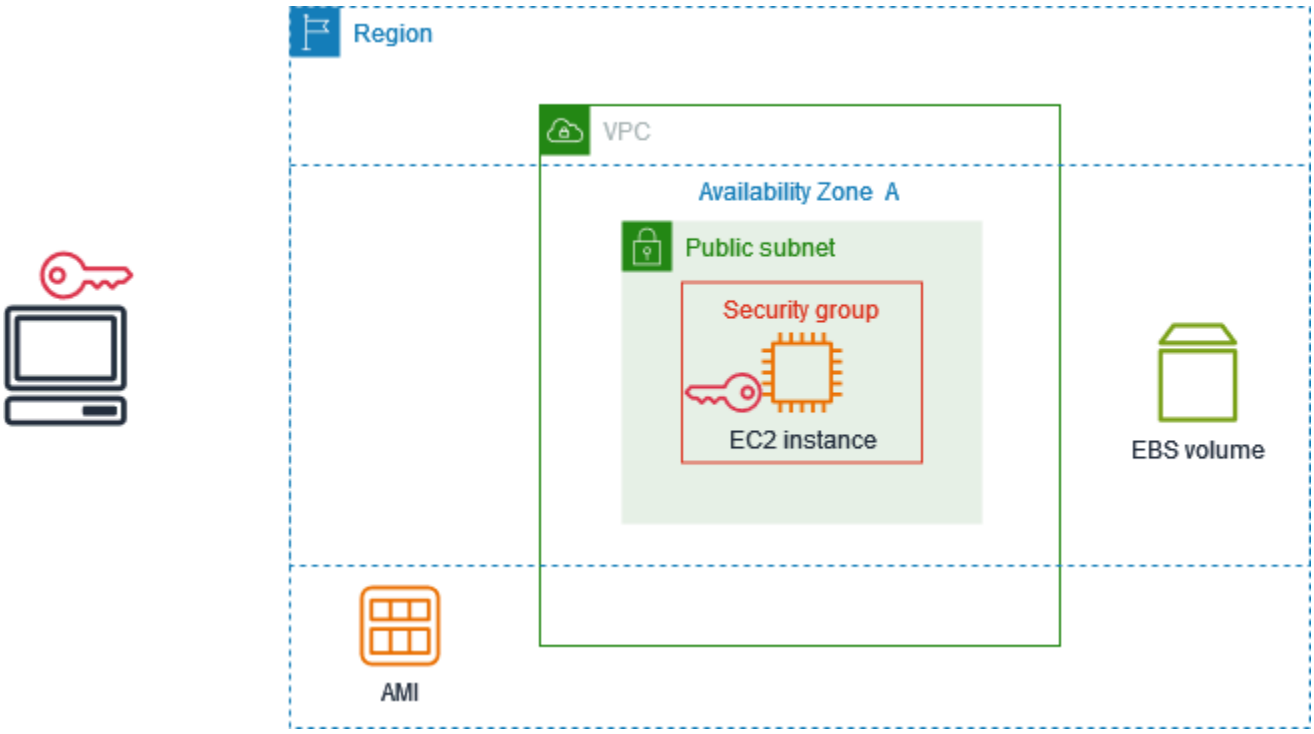
開始使用 Amazon EC2

使用本教學開始使用 Amazon 彈性運算雲端 (AmazonEC2)。您將學習如何啟動並連線至EC2執行個體。執行個體是 AWS 雲端中的虛擬伺服器。使用 AmazonEC2，您可以設定和設定執行個體上執行的作業系統和應用程式。

概觀

下圖顯示您將在本教學課程中使用的關鍵元件：

- 映像檔 — 包含要在執行個體 (例如作業系統) 上執行之軟體的範本。
- key pair 組 — 連線至執行個體時，用來證明身分的一組安全認證。公開金鑰位於您的執行個體上，私密金鑰位於您的電腦上。
- 網路 — 虛擬私有雲 (VPC) 是專用於您的 AWS 帳戶. 為了協助您快速開始使用，您的帳戶在每個帳戶VPC中都有預設值 AWS 區域，且每個預設值在每個可用區域中都VPC有一個預設子網路。
- 安全群組 — 充當虛擬防火牆，以控制輸入和輸出流量。
- EBS卷 — 我們需要一個根卷的映像。您可以選擇性地新增資料磁碟區。



本教學課程的費用

當您註冊 AWS, 你可以開始使用 Amazon EC2 使用 AWS 免費方案. 如果您 AWS 帳戶 在 12 個月前建立的產品，但尚未超過 Amazon 的免費方案權益EC2，則完成本教學課程不會花費任何費用，因為我們會協助您選擇免費方案權益範圍內的選項。否則，從啟動執行個體到終止執行個體 (這是本教學的最後一項工作)，您仍需支付標準 Amazon EC2 使用費，即使該執行個體仍處於閒置狀態。

如需判斷您是否符合免費方案資格的指示，請參閱the section called “追蹤您的免費方案用量”。

任務

- 步驟 1：啟動執行個體
- 步驟 2：連線到您的執行個體
- 步驟 3：清理您的執行個體
- 後續步驟

步驟 1：啟動執行個體

您可以使用下列程序所 AWS Management Console 述來啟動EC2執行個體。本教學旨在協助您快速啟動您的第一個執行個體，所以不涵蓋全部的可能選項。

啟動執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在屏幕頂部的導航欄中，我們顯示當前的 AWS 區域 -例如，俄亥俄州。您可以使用選取的「區域」，或選擇性地選取距離您較近的「區域」。
3. 從EC2主控台儀表板的 Launch 執行個體窗格中，選擇 Launch 執行個體。
4. 在 Name and tags (名稱和標籤) 下的 Name (名稱) 欄位中，輸入執行個體的描述性名稱。
5. 在 Application and OS Images (Amazon Machine Image) (應用程式和作業系統映像 (Amazon Machine Image)) 下，執行下列動作：
   a. 選擇 [快速入門]，然後為您的執行個體選擇作業系統 (OS)。對於您的第一個 Linux 執行個體，我們建議您選擇 Amazon Linux。
   b. 從 Amazon 機器映像 (AMI) 中，選取標示為符AMI合免費方案資格的項目。
6. 在 [執行個體類型] 下方，針對 [執行個體類型] 選擇`t2.micro`符合免費方案的資格。在無法使用`t2.micro`的區域中，符`t3.micro`合免費方案的資格。
7. 在 key pair (登入) 下，對於 key pair 名稱，選擇現有的 key pair，或選擇建立新的金鑰配對來建立您的第一個金鑰配對。

> **Warning**
>
> 如果您選擇不使用 key pair 繼續 (不建議使用)，您將無法使用本教學課程中所述的方法連線至執行個體。

8. 在 [網路設定] 底下，請注意我們已選取您的預設值VPC，選取了在我們為您選擇的可用區域中使用預設子網路的選項，並設定安全性群組，其中包含允許從任何地方連線至執行個體的規則。對於您的第一個執行個體，我們建議您使用預設設定。否則，您可以按照以下方式更新網絡設置：
   - (選擇性) 若要使用特定的預設子網路，請選擇 [編輯]，然後選擇子網路。
   - (選擇性) 若要使用其他項目VPC，請選擇 「編輯」，然後選擇現有的VPC。如果VPC未設定公用網際網路存取，您將無法連線至執行個體。
   - (選擇性) 若要限制特定網路的輸入連線流量，請選擇 [自訂] 而非 [任何地方]，然後輸入網路的CIDR區塊。
   - (選擇性) 若要使用不同的安全性群組，請選擇選取現有的安全性群組，然後選擇現有的安全性群組。如果安全性群組沒有允許來自網路的連線流量的規則，您將無法連線至執行個體。對於 Linux 執行個體，您必須允許SSH流量。對於 Windows 執行個體，您必須允許RDP流量。
9. 在 [設定儲存空間] 下，請注意我們已設定根磁碟區，但沒有資料磁碟區。這足以用於測試目的。
10. 檢閱 Summary (摘要) 面板中執行個體組態的摘要，並在準備就緒時選擇 Launch instance (啟動執行個體)。
11. 如果啟動成功，請從「成功」通知中選擇執行個體的 ID，以開啟「執行個體」頁面並監控啟動狀態。
12. 選取該執行個體的核取方塊。初始執行個體狀態為`pending`。在執行個體啟動後，它的生命週期狀態會變為 `running`。選擇 [狀態與警示] 索引標籤。執行個體通過其狀態檢查後，就可以接收連線要求。

步驟 2：連線到您的執行個體

您使用的程序視執行個體的作業系統而定。如果您無法連線至您的執行個體，請參閱解決連線到 Amazon EC2 Linux 執行個體的問題取得協助。

Linux 執行個體

您可以使用任何SSH客戶端連接到 Linux 實例。如果您在電腦上執行 Windows，請開啟終端機並執行`ssh`命令以確認您已安裝SSH用戶端。如果找不到該命令，請安裝「打開 SSH Windows 版」。

若要使用連線至執行個體 SSH

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後選取 Connect (連線)。
4. 在 [Connect 至執行個體] 頁面上，選擇用SSH戶端索引標籤。
5. (選擇性) 如果您在啟動執行個體時建立 key pair 組，並將私密金鑰 (.pem 檔案) 下載到執行 Linux 或 macOS 的電腦，請執行範例chmod命令來設定私密金鑰的權限。
6. 複製範例SSH指令。下面是一個例子，其中 *key-pair-name*.pem 是您的私鑰文件的名稱，*ec2-user* 是與影像相關聯的使用者名稱，@符號後面的字串是實體的公開名DNS稱。

```
ssh -i key-pair-name.pem ec2-user@ec2-198-51-100-1.us-east-2.compute.amazonaws.com
```

7. 在電腦的終端機視窗中，執行您在上一個步驟中儲存的ssh命令。如果私密金鑰檔案不在目前目錄中，您必須在此指令中指定金鑰檔案的完整路徑。

   以下是回應範例：

```
The authenticity of host 'ec2-198-51-100-1.us-east-2.compute.amazonaws.com
 (198-51-100-1)' can't be established.
ECDSA key fingerprint is l4UB/neBad9tvkgJf1QZWxheQmR59WgrgzEimCG6kZY.
Are you sure you want to continue connecting (yes/no)?
```

8. (選擇性) 在您第一次啟動執行個體時，確認安全性警示中的指紋是否符合主控台輸出中包含的執行個體指紋。若要取得主控台輸出，請選擇 [動作]、[監視和疑難排解]、[取得系統記錄 如果指紋不相符，可能是有人試圖 man-in-the-middle 攻擊。如果它們符合，請繼續下一個步驟。
9. 輸入 **yes**。

   以下是回應範例：

```
Warning: Permanently added 'ec2-198-51-100-1.us-
east-2.compute.amazonaws.com' (ECDSA) to the list of known hosts.
```

Windows 執行個體

若要使用連線至 Windows 執行個體RDP，您必須擷取初始管理員密碼，然後在連線至執行個體時輸入此密碼。在執行個體啟動後，需要幾分鐘的時間處理才能提供密碼。

管理員帳戶的預設使用者名稱取決於中包含的作業系統 (OS) 的語言AMI。若要確定正確的使用者名稱，請識別作業系統AMI的語言，然後選擇對應的使用者名稱。例如，對於英語操作系統，用戶名是`Administrator`，對於法國操作系統來說`Administrateur`，對於葡萄牙語操作系統而言，它是`Administrador`。如果作業系統的語言版本沒有相同語言的使用者名稱，請選擇使用者名稱`Administrator (Other)`。如需詳細資訊，請參閱 Microsoft TechNet Wiki 中的 Windows 系統管理員帳戶的本地化名稱。

擷取初始管理員密碼

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後選取 Connect (連線)。
4. 在 [Connect 至執行個體] 頁面上，選擇用RDP戶端索引標籤。
5. 在使用者名稱中，選擇管理員帳戶的預設使用者名稱。您選擇的使用者名稱必須與您用來啟動執行個體的作業系統 (OS) 語言相符。AMI如果沒有與您的作業系統相同語言的使用者名稱，請選擇「管理員 (其他)」。
6. 選擇 [取得密碼]。
7. 在 [取得 Windows 密碼] 頁面上，執行下列動作：
   a. 選擇「上傳私密金鑰檔案」，然後瀏覽至您在啟動執行個體時指定的私密金鑰 (`.pem`) 檔案。選取檔案並選取 Open (開啟)，將檔案的完整內容複製至此視窗。
   b. 選擇「解密密碼」。[取得 Windows 密碼] 頁面會關閉，且執行個體的預設管理員密碼會顯示在 [密碼] 底下，取代先前顯示的 [取得密碼] 連結。
   c. 複製密碼並將其保存在安全的地方。必須使用此密碼來連接至執行個體。

下列程序會使用適用於 Windows 的遠端桌面連線用戶端 (MSTSC)。如果您使用不同的用RDP戶端，請下載RDP檔案，然後參閱用RDP戶端的說明文件，瞭解建立RDP連線的步驟。

使用用RDP戶端連線至 Windows 執行個體

1. 在 [Connect 至執行個體] 頁面上，選擇 [下載遠端桌面檔案]。檔案下載完成時，請選擇「取消」返回「執行處理」頁面。RDP檔案即會下載至您的`Downloads`資料夾。
2. 執行`mstsc.exe`以開啟用RDP戶端。
3. 展開 [顯示選項]，選擇 [開啟]，然後從`Downloads`資料夾中選取 .rdp 檔案。

4. 根據預設，電腦是執行個體的公用IPv4DNS名稱，使用者名稱是系統管理員帳戶。若要使用來連線至執行個體，IPv6請將執行個體的公用IPv4DNS名稱取代為其IPv6位址。檢閱預設設定並視需要進行變更。
5. 選擇連線。如果您收到遠端連線發行者未知的警告，請選擇「Connect 線」以繼續。
6. 輸入您先前儲存的密碼，然後選擇 [確定]。
7. 由於自我簽署憑證的性質，您可能會收到安全憑證無法驗證的警告。執行以下任意一項：
   - 如果您信任憑證，請選擇 [是] 連線至您的執行個體。
   - [Windows] 在繼續之前，請將證書的指紋與系統日誌中的值進行比較，以確認遠端電腦的身份。選擇檢視憑證，然後從詳細資料索引標籤中選擇指紋。將此值與 [動作]、[監視及疑難排解]、[取得系統記錄] `RDPCERTIFICATE-THUMBPRINT` 中的值進行比較。
   - [Mac OS X] 繼續之前，請先將憑證的指紋與系統記錄檔中的值進行比較，以確認遠端電腦的身分識別。選擇顯示憑證，展開詳細資料，然後選擇SHA1指紋。將此值與 [動作]、[監視及疑難排解]、[取得系統記錄] `RDPCERTIFICATE-THUMBPRINT` 中的值進行比較。
8. 如果RDP連線成功，RDP用戶端會顯示 Windows 登入畫面，然後顯示 Windows 桌面平台。如果您收到錯誤訊息，請參閱the section called “遠端桌面無法連線到遠端電腦”。當您完成RDP連線時，您可以關閉用RDP戶端。

步驟 3：清理您的執行個體

在您完成為本教學建立的執行個體後，您應該終止此執行個體完成清理。如果想在清理前使用此執行個體執行更多作業，請參閱後續步驟。

> **Important**
>
> 終止執行個體可有效的將其刪除，無法在終止執行個體後重新連接到執行個體。

一旦執行個體狀態變更為或，您就不會再為該執行個體產生費用，`shutting down`或`terminated`計入免費方案限制的用量。若要保留執行個體供日後使用，但不會產生計入免費方案限制的費用或用量，您可以立即停止執行個體，稍後再重新啟動。如需詳細資訊，請參閱停止並啟動 Amazon EC2 執行個體。

終止您的執行個體

1. 在導覽窗格中，選擇 Instances (執行個體)。在執行個體清單中，選取執行個體。
2. 選擇 Instance state (執行個體狀態)、Terminate instance (終止執行個體)。

3. 出現確認提示時，請選擇 Terminate (終止)。

   Amazon EC2 關閉並終止您的實例。在您的執行個體終止後，它仍會短暫顯示於主控台，然後項目才會自動刪除。您無法自行從主控台顯示畫面中移除已終止的執行個體。

後續步驟

啟動執行個體之後，您可能想要探索下列後續步驟：

- 了解如何使用主控台追蹤 Amazon EC2 免費方案的使用情況。如需詳細資訊，請參閱the section called “追蹤您的免費方案用量”。
- 設定 CloudWatch 警示，以在您的使用量超過免費方案時通知您。如需詳細資訊，請參閱「AWS 免費方案 使用指南」AWS Billing 中的「追蹤您的使用情況
- 新增EBS磁碟區。如需詳細資訊，請參閱 Amazon EBS使用者指南中的建立 Amazon EBS 磁碟區。
- 了解如何使用Run命令遠端管理EC2執行個體。如需詳細資訊，請參閱 AWS Systems Manager 使用者指南中的 AWS Systems Manager 執行命令。
- 了解執行個體購買選項。如需詳細資訊，請參閱Amazon EC2 帳單和購買選項。
- 取得有關執行個體類型的建議。如需詳細資訊，請參閱從EC2執行個體類型尋找工具取得。

Amazon 的最佳實踐 EC2

為了確保 Amazon 獲得最大利益EC2，我們建議您執行下列最佳實務。

安全

- 管理對 AWS 資源的存取，並儘可能APIs使用與身分識別提供者和IAM角色聯合的身分識別。如需詳細資訊，請參閱《使用指南》中的IAM〈建立IAM策略〉。
- 為您的安全群組實作最嚴苛的規則。
- 定期修補、更新和保護您執行個體上的作業系統與應用程式。如需詳細資訊，請參閱 更新管理。如需 Windows 作業系統的特定指導方針，請參閱Windows 執行個體的安全性最佳做法。
- 使用 Amazon Inspector 自動探索和掃描 Amazon EC2 執行個體，找出軟體漏洞和意外的網路暴露。如需更多資訊，請參閱 Amazon Inspector 使用者指南。
- 使用 AWS Security Hub 控制項根據安全最佳實務和安全標準監控 Amazon EC2 資源。如需有關使用 Security Hub 的詳細資訊，請參閱《AWS Security Hub 使用者指南》中的 Amazon Elastic Compute Cloud 控制項。

儲存

- 了解根裝置類型對資料持久性、備份和復原的意涵。如需詳細資訊，請參閱 Root device type (根設備類型)。
- 針對作業系統與資料使用個別的 Amazon EBS 磁碟區。確保執行個體終止後能持續儲存您資料的磁碟區。如需詳細資訊，請參閱 在執行個體終止時保留資料。
- 使用您執行個體可用的執行個體存放區存放暫存資料。請記住，當您讓您的執行個體停止、休眠或終止時，會刪除存放在執行個體存放區的資料。如果您將執行個體存放區用為資料庫儲存體，請確保您的叢集有能確保容錯能力的複寫因素。
- 加密EBS磁碟區和快照。如需詳細資訊，請參閱 Amazon EBS使用者指南中的 Amazon EBS 加密。

資源管理

- 使用執行個體中繼資料和自訂資源標籤追踪並找出您的 AWS 資源。如需詳細資訊，請參閱使用執行個體中繼資料管理EC2執行個和標記您的 Amazon EC2 資源。
- 查看您目前的 Amazon 限制EC2。計劃在您需要的時候，先行請求提高限制。如需詳細資訊，請參閱 Amazon EC2 服務配額。

- 用於檢 AWS Trusted Advisor 查您的 AWS 環境，然後在存在機會時提出建議，以節省金錢、改善系統可用性和效能，或協助縮小安全性漏洞。如需詳細資訊，請參閱《AWS Support 使用者指南》中的 AWS Trusted Advisor。

備份與復原

- 使用 Amazon EBS 快照定期備份EBS磁碟區，並從執行個體建立 Amazon Machine Image (AMI)，將組態儲存為範本，以便 future 啟動執行個體。如需有助於實現此使用案例的 AWS 服務的詳細資訊，請參閱AWS Backup和 Amazon Data Lifecycle Manager。
- 跨多個可用區域部署您應用程式的關鍵元件，並適當複寫您的資料。
- 設計您的應用程式在執行個體重新啟動時處理動態 IP 定址。如需詳細資訊，請參閱 Amazon EC2 實例 IP 定址。
- 監控與回應事件。如需詳細資訊，請參閱 監控 Amazon EC2 資源。
- 請確定您已準備好可處理容錯移轉。如需基本解決方案，您可將網路界面或彈性 IP 地址手動連接到替代執行個體。如需詳細資訊，請參閱 彈性網絡介面。對於自動化解決方案，您可以使用 Amazon EC2 Auto Scaling。如需詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南。
- 定期測試復原執行個體和 Amazon EBS 磁碟區的程序，以確保成功還原資料和服務。

聯網

- 將應用程式的 time-to-live (TTL) 值設定為 255，用於IPv4和IPv6。如果您使用較小的值，則可能TTL會在應用程式流量傳輸過程中過期，造成執行個體可達性問題。

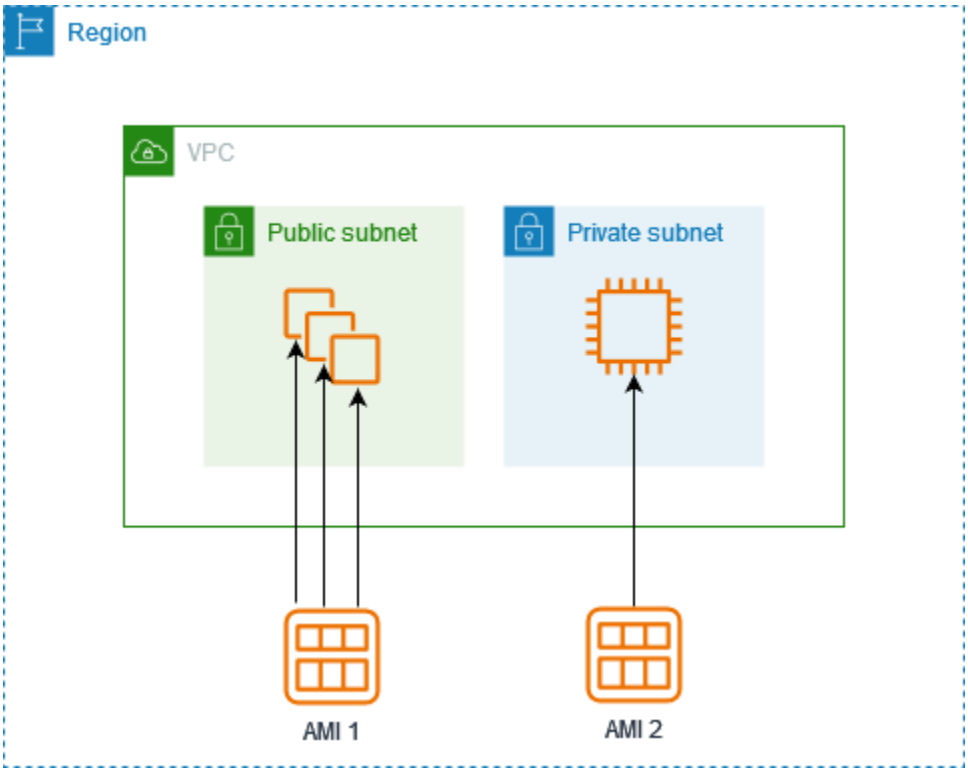
Amazon 的 Amazon 機器映像 EC2

Amazon 機器映像 (AMI) 是提供設定和啟動 Amazon EC2 執行個體所需軟體的映像檔。每個都AMI包含一個區塊裝置對應，指定要連接至您啟動的執行個體的區塊裝置。您必須在啟動執行個體AMI時指定。AMI必須與您為執行個體選擇的執行個體類型相容。您可以使用AMI提供者 AWS、公眾AMI、與您分享的其他人，或AMI您從中購買的 AWS Marketplace。AMI

一AMI個特定於以下內容：

- 區域
- 作業系統
- 處理器架構
- Root device type (根設備類型)
- 虛擬化類型

當您需要具有相同組態的多個執行個體AMI時，可以從一個執行個體啟動多個執行個體。當您需要具有不同組態的執行個體時，可以使用不同AMIs來啟動執行個體，如下圖所示。



您可以AMI從 Amazon EC2 執行個體建立，然後使用它來啟動具有相同組態的執行個體。您可以將某個區域複製AMI到另一個「 AWS 區域」，然後使用它在該區域中啟動執行個體。您也可以與其他帳戶

共AMI用您建立的帳戶，以便他們可以使用相同的設定啟動執行個體。您可以AMI使用出售您的 AWS Marketplace.

目錄

- AMIAmazon 的類型和特徵 EC2
- 尋找符AMI合執行個EC2體需求的
- AMIs在 Amazon EC2 實例中支付 AWS Marketplace
- Amazon EC2 AMI 週期
- 使用 Amazon EC2 開機模式執行個體啟動行為
- 對 EBS 後端 AMI 使用加密
- 了解 Amazon 中的共享AMI用量 EC2
- 使用 Amazon 監控AMI事件 EventBridge
- 了解 AMI 帳單資訊
- AMIAmazon 的配額 EC2

AMIAmazon 的類型和特徵 EC2

啟動執行個體時，您選擇的AMI執行個體必須與您選擇的執行個體類型相容。您可以根據下列特性選取AMI要使用的：

- 區域
- 作業系統
- 處理器架構
- 啟動許可
- Root device type (根設備類型)
- 虛擬化類型

啟動許可

的擁有者透過指定啟動權限來AMI決定其可用性。啟動許可分為以下類別。

啟動許可	描述
public	擁有者會授與所有 AWS 帳戶的啟動權限。
explicit	擁有者將啟動權限授與特定 AWS 帳戶、組織或組織單位 (OUs)。
implicit	擁有者具有的隱含啟動權限AMI。

Amazon 和 Amazon EC2 社區提供了大量的公眾選擇AMIs。如需詳細資訊，請參閱了解 Amazon 中的共享AMI用量 EC2。開發人員可以收取他們的AMIs. 如需詳細資訊，請參閱AMIs在 Amazon EC2 實例中支付 AWS Marketplace。

Root device type (根設備類型)

所有AMIs這些都歸類為由 Amazon 支持EBS或由實例存儲支持。

- Amazon EBS 支援 AMI — 從啟動的執行個體的根裝置AMI是從 Amazon EBS 快照建立的 Amazon 彈性區塊存放區 (AmazonEBS) 磁碟區。同時支援 Linux 和視窗AMIs。
- Amazon 執行個體存放區支援 AMI — 從啟動的執行個體的根裝置AMI是從 Amazon S3 中存放的範本建立的執行個體存放區磁碟區。AMIs僅支援 Linux 系統。Windows AMIs 不支援根裝置的執行個體存放區。

如需詳細資訊，請參閱Amazon EC2 執行個體的根磁碟區。

下表摘要說明使用這兩種類型的重要差異AMIs。

特性	Amazon 支EBS持 AMI	Amazon 實例商店支持 AMI
根設備磁碟區	EBS體積	執行個體存放區磁碟區
執行個體的開機時間	通常少於 1 分鐘	通常少於 5 分鐘
資料持久性	根據預設，當執行個體終止時，系統會刪除根磁碟區。* 依預設，執行個體終止後，任何其他EBS磁碟區上的資料仍然存在。	只有在執行個體的生命週期，才會持續保存執行個體存放區磁碟區上的資料。

特性	Amazon 支EBS持 AMI	Amazon 實例商店支持 AMI
停止狀態	可以處於停止狀態。即使執行個體停止且未執行，根磁碟區仍會保留在 Amazon 中 EBS	無法進入停止狀態；執行個體正在執行或已終止
修改	執行個體類型、核心、RAM磁碟和使用者資料可在執行個體停止時變更。	在執行個體的生命週期，執行個體屬性是固定的。
費用	您需支付執行個體使用EBS量、磁碟區使用量以及以EBS快照形AMI式儲存的費用。	您需要支付執行個體用量和存放AMI在 Amazon S3 中的費用。
AMI建立 / 捆綁	使用單一命令/呼叫	需要安裝和使用工AMI具

* EBS 根磁碟區預設會將`DeleteOnTermination`旗標設定為`true`。如需如何變更此標記，以便在終止之後持續保存磁碟區的資訊，請參閱 在 Amazon 實EC2例終止後保留 Amazon EBS 根卷。

** 僅支援「`io2`EBS區塊快速」。如需詳細資訊，請參閱 Amazon EBS 使用者指南中的佈建區IOPSSSD塊快速磁碟區。

確定您的根設備類型 AMI

您AMI用來啟動EC2執行個體的決定了根磁碟區的類型。EC2執行個體的根磁碟區是EBS磁碟區或執行個體儲存磁碟區。目前一代的執行個體類型僅支援EBS根磁碟區。唯一支援執行個體儲存根磁碟區的執行個體類型是 C1、C3、D2、I2、M1、M2、M3、R3 和 X1。

使用控制台確定的根AMI設備類型

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 AMIs，然後選取AMI。
3. 在 Details (詳細資訊) 索引標籤中，檢查 Root device type (根設備類型) 的值，如下所示：
   - `ebs`-這是一個 EBS-支持AMI。
   - `instance store`— 這是商店支援的執行個體AMI。

若要AMI使用指令列判斷的根裝置類型

您可以使用下列其中一個命令。

- describe-images (AWS CLI)
- Get-EC2Image (AWS Tools for Windows PowerShell)

虛擬化類型

Amazon 機器映像使用兩種虛擬化類型之一：半虛擬 (PV) 或硬體虛擬機器 (HVM)。PV 和HVMAMIs之間的主要差異在於它們的啟動方式以及它們是否可以利用特殊的硬件擴展（CPU，網絡和存儲）來獲得更好的性能。窗戶AMIs是HVMAMIs。

下表比較HVM和 PV AMIs。

特性	HVM	PV
描述	HVMAMIs透過執行映像的根區塊裝置的主開機記錄，會顯示一組完全虛擬化的硬體和開機。這種虛擬化類型提供在沒有任何修改的情況下直接在虛擬機器上執行作業系統之能力，就像執行在裸機硬體上一樣。Amazon EC2 主機系統會模擬提供給客座端的部分或全部基礎硬體。	使用名為 PV AMIs-的特殊開機載入程式進行 PV 開機GRUB，它會啟動開機週期，然後鏈結載入映像`menu.lst`檔中指定的核心。全虛擬化訪客可以在不明確支援虛擬化的主機硬體上執行。如需 PV-GRUB 及其在 Amazon 中使用的詳細資訊EC2，請參閱使用者提供的核心。
支援的執行個體類型	目前所有的執行個體類型都支援HVMAMIs。	下列前一代執行個體類型支援PVAMIs：C1、C3、M1、M3、M2 和 T1。目前一代的執行個體類型不支援 PV AMIs。
硬體延伸支援	HVM客座端可利用硬體延伸功能，快速存取主機系統上的基礎硬體。他們必須使用增強型網路和GPU處理。要通過指	否，他們無法利用特殊的硬體延伸功能，例如增強型網路或GPU處理。

特性	HVM	PV
	令傳遞到專門的網絡和GPU設備，操作系統必須能夠訪問本地硬件平台，並且HVM虛擬化提供了這種訪問。如需詳細資訊，請參閱Amazon EC2 執行個體上的增強聯網。	
如何找到	使用主控台或描述映像檔指AMI令hvm，確認已將的虛擬化類型設定為。	使用主控台或描述映像檔指AMI令paravirtual ，確認已將的虛擬化類型設定為。

光伏開啟 HVM

Paravirtual 來實傳統上在儲存和網路操作方面的表現比來HVM實更好，因為他們可以利用特殊的 I/O 驅動程式來避免模擬網路和磁碟硬體的額外負荷，而HVM客座端則必須將這些指令轉換為模擬硬體。現在HVM客座端可以使用 PV 驅動程式，因此無法移植為在半虛擬化環境中執行的作業系統仍然可以透過使用它們來看到儲存和網路 I/O 的效能優勢。在HVM驅動程式上使用這些 PV，HVM客座端可以獲得與半虛擬客座端相同或更好的效能。

尋找符AMI合執行個EC2體需求的

AMI包括啟動執行個體所需的元件和應用程式，例如作業系統和根磁碟區類型。若要啟動執行個體，您必須找到符合您需求的執行個AMI體。

選取時AMI，請考慮下列要啟動之執行個體的需求：

- 的「 AWS 區域」對於每個「AMI區域」而言都AMIIDs是獨一無二的。
- 作業系統 (例如 Linux 或視窗)。
- 架構 (例如 32 位元、64 位元或 64 位元ARM)。
- 根裝置類型 (例如，Amazon EBS 或執行個體存放區)。
- 供應商（例如，Amazon Web Services）。
- 其他軟體 (例如，SQL伺服器)。

要查找 Amazon Linux 2023AMI，請參閱 Amazon 20 AL223 用戶指南EC2中的 Amazon 網站上的 0 23。

要查找 UbuntuAMI，請參閱規範 Ubuntu 網站上的 Amazon EC2 AMI 定位器。

若要尋找 RHELAMI，請參閱 Red Hat 網站上的 RHEL 映像檔 (AMI AWS) 可在 Amazon Web Services 上取得 ()。

有多種方法可以找AMI到滿足您需求的。你可以找到一個AMI使用 Amazon EC2 控制台 AWS CLI AWS Tools for Windows PowerShell，，和 AWS Systems Manager。

查找使AMI用 Amazon EC2 控制台

您可以AMIs使用 Amazon EC2 控制台找到。當您使用啟動執行處理精靈啟動執行處理AMIs時，您可以從清單中選取，或使用「映像檔」頁面搜尋所有可AMIs用的項目。

若要尋找AMI使用啟動執行處理精靈

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 從導覽列選取要在其中啟動執行個體的區域。無論您的位置為何，皆可選取任何可用的區域。AMIIDs每個 AWS 區域都是獨一無二的。
3. 從主控台儀表板選擇 Launch Instance (啟動執行個體)。
4. (新主控台) 在應用程式和作業系統映像 (Amazon Machine Image) 下，選擇快速入門，為您的執行個體選擇作業系統 (OS)，然後從 Amazon Machine Image (AMI) 中選取清單AMIs中常用的作業系統之一。如果沒有看到您要使用的AMI項目，請選擇 [瀏覽更多] AMIs 以瀏覽完整AMI目錄。如需詳細資訊，請參閱應用程式和作業系統映像 (Amazon Machine Image)。

   (舊主控台) 在 [快速入門] 索引標籤上，從清單中常用的其中一個選項AMIs中選取。如果您沒有看到要使用的AMI項目，請選擇 [我的] AMIs AWS Marketplace、或 AMIs [社群] 索引標籤來尋找其他項目AMIs。

若要尋找AMI使用AMIs頁面

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 從導覽列選取要在其中啟動執行個體的區域。無論您的位置為何，皆可選取任何可用的區域。AMIIDs每個 AWS 區域都是獨一無二的。
3. 在導覽窗格中，選擇AMIs。

4. (選擇性) 使用篩選條件和搜尋選項，將顯示清單的範圍限定AMIs為僅查看符AMIs合您準則的項目。

   例如，若要列出所有AMIs提供者 AWS，請選擇 [公開影像]。然後使用搜尋選項進一步限定顯示清單的範圍AMIs。選擇 Search (搜尋) 列，然後從選單中依序選擇 Owner alias(擁有者別名)、= 運算子和值 amazon。若要尋找AMIs符合特定平台 (例如 Linux 或 Windows) 的平台，請再次選擇 [搜尋] 列以選擇 [平台]，然後選擇 = 運算子，然後從提供的清單中選擇作業系統。
5. (選用) 選擇偏好設定圖示，以選取要顯示的映像屬性，例如根裝置類型。或者，您可以AMI從清單中選取，然後在「詳細資訊」(Details) 標籤上檢視其屬性。
6. 在選擇之前AMI，請務必檢查它是由實例存儲還是由 Amazon 支持，EBS並且知道這種差異的影響非常重要。如需詳細資訊，請參閱Root device type (根設備類型)。
7. 若要從此啟動執行個體AMI，請選取該執行個體，然後從映像檔選擇啟動執行個體。如需有關使用主控台啟動執行個體的詳細資訊，請參閱使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體。如果您尚未準備好啟動執行個體，請記下 AMI ID 以供稍後使用。

找到一個AMI使用 AWS CLI

您可以使用「描述影像」 AWS CLI 指令AMIs，僅列出符合您需求的項目。找到符合您需求的 ID 後，請記下其 ID，以便您可以使用它來啟動執行個AMI體。如需詳細資訊，請參閱《AWS Command Line Interface 使用者指南》中的啟動您的執行個體

describe-images 指令支援篩選參數。例如，使用`--owners`參數來顯示 Amazon AMIs 擁有的公眾。

```
aws ec2 describe-images --owners amazon
```

您可以將以下過濾器添加到上一個命令中，以僅顯示 Windows AMIs。

```
--filters "Name=platform,Values=windows"
```

您可以將以下過濾器添加到以前的命令中，以僅顯示由 Amazon AMIs 支持的命令EBS。

```
--filters "Name=root-device-type,Values=ebs"
```

> **Important**
>
> 如果您省略`describe-images`指令中的`--owners`參數，則會傳回您擁有啟動權限的所有影像，而不論擁有權為何。

找到一個AMI使用 AWS Tools for Windows PowerShell

您可以使用 PowerShell 指令程式只列出符合AMIs您需求的 Windows。如需相關資訊和範例，請參閱使用AWS Tools for Windows PowerShell 者指南 PowerShell中的使用 Windows 尋找 Amazon 電腦映像。

找到符合您需求的 ID 後，請記下其 ID，以便您可以使用它來啟動執行個AMI體。如需詳細資訊，請參閱使用AWS Tools for Windows PowerShell 者指南PowerShell中的使用 Windows 啟動 Amazon EC2 執行個體。

AMI使用「Systems Manager」參數尋找

使用 Amazon EC2 主控台中的EC2啟動執行個體精靈啟動執行個體時，可以AMI從清單中選取 (如中所述查找使AMI用 Amazon EC2 控制台)，也可以選取指向 AMI ID 的 AWS Systems Manager 參數 (如本節所述)。如果您使用自動化程式碼來啟動執行個體，您可以指定 Systems Manager 參數，而不是 AMI ID。

Systems Manager 參數是您可以在 Systems Manager 參數存放區中建立的客戶定義索引鍵/值組。參數存放區提供一個中央存放區，可將您的應用程式組態值外部化。如需詳細資訊，請參閱 AWS Systems Manager 使用者指南中的 AWS Systems Manager 參數存放區。

當您建立指向 AMI ID 的參數時，請務必將資料類型指定為`aws:ec2:image`。指定此資料類型可確保在建立或修改參數時，會將參數值驗證為 AMI ID。如需詳細資訊，請參閱AWS Systems Manager 使用者指南中的 Amazon 機器映像IDs的原生參數支援。

主題

- 使用案例
- 許可
- 限制
- 使用 Systems Manager 參數啟動執行個體

使用案例

當您使用「Systems Manager」參數來指向時 AMIIDs，您的使用者可以更輕鬆地在啟動例證AMI時選取正確的參數。Systems Manager 參數還可以簡化自動化程式碼的維護。

Easier for users (對使用者更簡單)

如果您需要使用特定的執行個體啟動AMI，且會定期更新，建議您要求您的使用者選取「Systems Manager」參數來尋找AMI。AMI要求您的使用者選取「Systems Manager」參數，可確保使用最新AMI的執行個體來啟動執行個體。

例如，您組織中的每個月都可以建立具有最新作業系統和應用程式修補程式的新版本。AMI您也會要求使用者使用您的最新版本啟動執行個體AMI。為了確保您的使用者使用最新版本，您可以建立指向正確 AMI ID 的 Systems Manager 參數 (例如`golden-ami`)。每次建立新版本的時候，AMI都會更新參數中的 AMI ID 值，使其永遠指向最新版本AMI。您的使用者不必瞭解定期更新，AMI因為他們每次都會繼續選取相同的 Systems Manager 參數。為您的使用 Systems Manager 參數可AMI讓他們更輕鬆地為執行個體啟動選AMI取正確的參數。

Simplify automation code maintenance (簡化自動化程式碼維護)

如果您使用自動化程式碼來啟動執行個體，您可以指定 Systems Manager 參數，而不是 AMI ID。如果建立了新版本AMI的，您可以變更參數中的 AMI ID 值，使其指向最新版本AMI。參考參數的自動化程式碼不需要在每次建立新版本AMI時修改。如此可簡化自動化維護作業，並有助於降低部署成本。

> **Note**
>
> 當您變更「Systems Manager」參數指向的 AMI ID 時，執行中的例證不會受到影響。

許可

如果您使用在啟動執行個體精靈AMIIDs中指向的 Systems Manager 參數，則必須將下列權限新增至您的IAM原則：

- `ssm:DescribeParameters`— 授予檢視和選取 Systems Manager 參數的權限。
- `ssm:GetParameters`— 授與擷取 Systems Manager 參數值的權限。

您也可以限制對特定 Systems Manager 參數的存取。如需詳細資訊和IAM原則範例，請參閱範例：使用EC2啟動執行個體精靈。

限制

AMIs和「Systems Manager 參數是區域特有的。若要跨區域使用相同的 Systems Manager 參數名稱，請在每個區域中建立具有相同名稱的 Systems Manager 參數 (例如 `golden-ami`)。在每個「區域」中，將「Systems Manager」參數指向該區域AMI中的參數。

使用 Systems Manager 參數啟動執行個體

您可以使用主控台或 AWS CLI來啟動執行個體。您可以指定指向 AMI ID 的 AWS Systems Manager 參數，而不是指定 AMI ID。

New console

若要尋找AMI使用 Systems Manager 參數 (主控台)

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 從導覽列選取要在其中啟動執行個體的區域。無論您的位置為何，皆可選取任何可用的區域。
3. 從主控台儀表板選擇 Launch Instance (啟動執行個體)。
4. 在應用程式和作業系統映像 (Amazon 機器映像) 下，選擇瀏覽更多AMIs。
5. 選擇搜尋列右側的箭頭按鈕，然後選擇 Search by Systems Manager parameter (依 Systems Manager 參數搜尋)。
6. 對於 Systems Manager parameter (Systems Manager 參數)，請選取一個參數。對應的 AMI ID 會顯示在 [目前解析為] 下方。
7. 選擇 Search (搜尋)。與 AMI ID AMIs 相符的會出現在清單中。
8. 從清單AMI中選取，然後選擇「選取」。

如需有關使用啟動執行個體精靈啟動執行個體的詳細資訊，請參閱 使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體。

Old console

若要尋找AMI使用 Systems Manager 參數 (主控台)

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 從導覽列選取要在其中啟動執行個體的區域。無論您的位置為何，皆可選取任何可用的區域。
3. 從主控台儀表板選擇 Launch Instance (啟動執行個體)。
4. 選擇 Search by Systems Manager parameter (依 Systems Manager 參數搜尋) (位於右上角)。
5. 對於 Systems Manager parameter (Systems Manager 參數)，請選取一個參數。對應的 AMI ID 會出現在 [目前解析為] 旁邊。
6. 選擇 Search (搜尋)。與 AMI ID AMIs 相符的會出現在清單中。
7. 從清單AMI中選取，然後選擇「選取」。

使用 AWS Systems Manager 參數而不是 AMI ID (AWS CLI) 啟動例證

下列範例會使用 Systems Manager 參數 `golden-ami` 來啟動 `m5.xlarge` 執行個體。參數指向一個 AMI ID。

若要在命令中指定參數，請使用下列語法：`resolve:ssm:/parameter-name`，其中 `resolve:ssm` 是標準字首，`parameter-name` 是唯一的參數名稱。請注意，參數名稱區分大小寫。參數名稱的反斜線只有在參數是階層的一部分時才需要，例如 `/amis/production/golden-ami`。如果參數不是階層架構的一部分，則可以省略反斜線。

在此範例中，`--count` 和 `--security-group` 參數不包含在內。對於 `--count`，預設值為 1。如果您有預設VPC和預設安全性群組，則會使用這些群組。

```
aws ec2 run-instances
    --image-id resolve:ssm:/golden-ami
    --instance-type m5.xlarge
    ...
```

若要使用特定版本的 AWS Systems Manager 參數啟動執行個體 (AWS CLI)

Systems Manager 參數擁有版本支援。參數的每個反覆項目都會指定一個唯一的版本號碼。您可以參考參數的版本，如下所示 `resolve:ssm:parameter-name:version`，其中 `version` 是唯一的版本號碼。根據預設，在未指定任何版本時，會使用最新版本的參數。

下列範例會使用參數的版本 2。

在此範例中，`--count` 和 `--security-group` 參數不包含在內。對於`--count`，預設值為1如果您有預設VPC和預設安全性群組，則會使用它們。

```
aws ec2 run-instances
    --image-id resolve:ssm:/golden-ami:2
    --instance-type m5.xlarge
    ...
```

若要使用提供的 public 參數啟動執行個體 AWS

Systems Manager 提供公共參數由公眾AMIs提供 AWS。您可以在啟動執行個體時使用 public 參數，以確保您使用的是最新的AMIs。

如需詳細資訊，請參閱使用 Systems Manager 公AMIs用參數尋找最新資訊。

使用 Systems Manager 公AMIs用參數尋找最新資訊

AWS Systems Manager 為公眾提供公共參數由AMIs維護 AWS. 您可以在啟動執行個體時使用 public 參數，以確保您使用的是最新的AMIs。例如，公共參數`/aws/service/ami-amazon-linux-latest/al2023-ami-kernel-default-arm64`在所有區域都可以使用，並且永遠指向指定區域中 Arm64 架構的最新版本。AMI

公用參數可從下列路徑取得：

- Linux – `/aws/service/ami-amazon-linux-latest`
- Windows – `/aws/service/ami-windows-latest`

若要檢視目前 AWS 區域中所有 Linux 或AMIs視窗的清單

使用以下get-parameters-by-path AWS CLI 命令來查看當前 AWS 區域AMIs中所有 Linux 或 Windows 的列表。對於 Linux 和視窗而言，`--path`參數的值不同。

對於 Linux：

```
aws ssm get-parameters-by-path \
    --path /aws/service/ami-amazon-linux-latest \
    --query "Parameters[].Name"
```

針對 Windows：

```
aws ssm get-parameters-by-path \
    --path /aws/service/ami-windows-latest \
    --query "Parameters[].Name"
```

使用公有參數啟動執行個體

下列範例會為映像 ID 指定一個 Systems Manager 公用參數，以便使用最新的 Amazon Linux 2023 AMI 啟動執行個體。

若要在命令中指定參數，請使用下列語法：`resolve:ssm:`*`public-parameter`*，其中 `resolve:ssm` 是標準前綴，而 *`public-parameter`* 是公有參數的路徑和名稱。

在此範例中，`--count` 和 `--security-group` 參數不包含在內。對於 `--count`，預設值為 1。如果您有預設VPC和預設安全性群組，則會使用這些群組。

```
aws ec2 run-instances \
```

```
    --image-id resolve:ssm:/aws/service/ami-amazon-linux-latest/al2023-ami-kernel-default-x86_64 \
    --instance-type m5.xlarge \
    --key-name MyKeyPair
```

如需詳細資訊，請參閱 AWS Systems Manager 使用者指南中的使用公有參數。

如需使用 Systems Manager 參數的範例，請參閱使用參數存放區查詢最新的 Amazon Linux 和AMIIDs使 AWS Systems Manager用 AWS Systems Manager 參數存放區查詢最新 Windows AMI。

AMIs在 Amazon EC2 實例中支付 AWS Marketplace

「已付款」AMI AMI 是在中列出待售的 AWS Marketplace。這 AWS Marketplace 是一個在線商店，您可以在其中購買可運行的軟件 AWS AMIs，包括可用於啟動EC2實例的軟件。 AWS Marketplace AMIs它們分為類別，例如「開發人員工具」，以便您找到符合您需求的產品。如需詳細資訊 AWS Marketplace，請參閱AWS Marketplace網站。

您可以 AWS Marketplace 從AMIs第三方購買，包括AMIs來自 Red Hat 等組織的服務合約。您還可以創建一個AMI並將其出售 AWS Marketplace 給其他 Amazon EC2 用戶。如果您遵循一些簡單的準則，建立一個安全，可用AMI於公共消費的過程是一個相當簡單的過程。若要取得有關如何建立和使用共用的資訊AMIs，請參閱了解 Amazon 中的共享AMI用量 EC2。

從付費AMI執行個體啟動執行個體與從任何其他執行個體啟動執行個體相同AMI。不需額外的參數。執行個體會根據擁有者設定的費率以及相關 Web 服務的標準使用費 (例如，在 Amazon 中執行 m5.small 執行個體類型的小時費率) 收費。AMI EC2可能也需支付額外稅金。付費的擁有者AMI可以確認是否使用該付費執行個體啟動了特定執行個體AMI。

> **Important**
>
> Amazon DevPay 不再接受新的賣家或產品。 AWS Marketplace 現在是通過銷售軟件和服務的單一統一電子商務平台 AWS。如需如何部署及銷售軟體的詳細資訊 AWS Marketplace，請參閱在 AWS Marketplace 中銷售。 AWS Marketplace 支持AMIs由 Amazon 支持EBS。

目錄

- 出售您的 AMI AWS Marketplace
- 尋找付費 AMI
- 購買已付款 AMI AWS Marketplace

- 從您的執行個體擷取 AWS Marketplace 產品代碼
- 針對支援的 AWS Marketplace 產品使用付費支援
- 已付款和支援帳單 AMIs
- 管理您的 AWS Marketplace 訂閱

出售您的 AMI AWS Marketplace

你可以賣你AMI使用 AWS Marketplace. AWS Marketplace 提供有組織的購物體驗。此外， AWS Marketplace AWS 還支援 Amazon EBS 支援的預留執行AMIs個體和 Spot 執行個體等功能。

如需如何在「」AMI 上銷售您的產品的詳細資訊 AWS Marketplace，請參閱在 AWS Marketplace 中銷售。

尋找付費 AMI

有幾種方法，你可AMIs以找到，可供您購買. 例如，您可以使用 AWS MarketplaceAmazon EC2 主控台或命令列。或者，開發人員可能會讓您知道有關付費AMI自己的信息。

AMI使用主控台尋找付費

AMI使用控制台查找付費

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇AMIs。
3. 中，選擇 Public images (公有映像) 作為第一個篩選條件。
4. 在搜尋列中，依序選擇 Owner alias (擁有者別名)、= 和 aws-marketplace。
5. 如果您知道產品代碼，請選擇 Product code (產品代碼) 和 =，然後輸入產品代碼。

查找付費AMI使用 AWS Marketplace

若要尋找付費AMI使用 AWS Marketplace

1. 打開 AWS Marketplace.
2. 在搜尋欄位中輸入作業系統的名稱，然後選擇搜尋按鈕 (放大鏡)。
3. 若要進一步限制搜尋結果的範圍，請使用其中一種分類或篩選條件。
4. 每一種產品都帶有其產品類型的標籤：`AMI` 或 `Software as a Service`

使用搜尋AMI付費 AWS CLI

您可以AMI使用以下描述圖像命令（ ）找到付費。AWS CLI

```
aws ec2 describe-images
    --owners aws-marketplace
```

此命令會傳回描述每個詳細資料的眾多詳細資料AMI，包括付費的產品代碼AMI。`describe-images` 的輸出包含產品代碼的項目，如下所示：

```
"ProductCodes": [
    {
        "ProductCodeId": "product_code",
        "ProductCodeType": "marketplace"
    }
],
```

如果您知道產品代碼，您可以依產品代碼篩選結果。此範例會傳回最新AMI的指定產品代碼。

```
aws ec2 describe-images
    --owners aws-marketplace \
    --filters "Name=product-code,Values=product_code" \
    --query "sort_by(Images, &CreationDate)[-1].[ImageId]"
```

AMI使用視窗工具尋找付費 PowerShell

您可以AMI使用以下Get-EC2Image命令找到付費。

```
PS C:\> Get-EC2Image -Owner aws-marketplace
```

付費的輸出AMI包括產品代碼。

```
ProductCodeId           ProductCodeType
-------------           ---------------
product_code            marketplace
```

如果您知道產品代碼，您可以依產品代碼篩選結果。此範例會傳回最新AMI的指定產品代碼。

```
PS C:\> (Get-EC2Image -Owner aws-marketplace -Filter @{"Name"="product-
code";"Value"="product_code"} | sort CreationDate -Descending | Select-Object -First
 1).ImageId
```

購買已付款 AMI AWS Marketplace

您必須AMI先註冊 (購買) 付費，然後 EC2 Amazon 能使用AMI.

通常，付費賣家會向您AMI提供有關的信息AMI，包括其價格以及您可以在其中購買的鏈接。當您按一下連結時，系統會先要求您登入 AWS，然後您可以購買AMI.

AMI使用主機購買付費

您可以AMI通過使用 Amazon EC2 啟動嚮導購買付費。如需詳細資訊，請參閱從一個啟動 Amazon EC2 實例 AWS Marketplace AMI。

使用訂閱產品 AWS Marketplace

若要使用 AWS Marketplace，您必須有一個 AWS 帳戶. 若要從產 AWS Marketplace 品啟動執行個體，您必須註冊才能使用 Amazon EC2 服務，而且您必須訂閱要從中啟動執行個體的產品。您可以使用下列其中一種方法來訂閱中的產品 AWS Marketplace：

- AWS Marketplace 網站：您可以使用一鍵式部署功能快速啟動預先配置的軟件。如需詳細資訊，請參閱中的AMI基礎產品 AWS Marketplace。
- Amazon EC2 啟動精靈：您可以直接從精靈搜尋執行個體AMI並啟動執行個體。如需詳細資訊，請參閱從一個啟動 Amazon EC2 實例 AWS Marketplace AMI。

從您的執行個體擷取 AWS Marketplace 產品代碼

您可以使用執行個體中繼資料擷取執行個體的 AWS Marketplace 產品代碼。如果執行個體具有產品代碼，Amazon 會EC2傳回該代碼。如需擷取中繼資料的詳細資訊，請參閱 訪問實例的實EC2例元數據。

若要擷取產品代碼，請使用執行個體作業系統的指令。

Linux

IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-
aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-
data/product-codes
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/product-codes
```

Windows

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/product-codes
```

針對支援的 AWS Marketplace 產品使用付費支援

Amazon EC2 也讓開發人員能夠提供軟體 (或衍生AMIs) 的支援。開發人員能建立可讓您註冊使用的支援產品。在註冊支援產品期間，開發人員會為您提供產品代碼，然後您必須與自己AMI的產品代碼建立關聯。這可讓開發人員確認您的執行個體符合取得支援的資格。它也能確認在您執行產品執行個體的時候，您確實有根據開發人員指定的產品規定支付費用。

> **Important**
>
> 您無法搭配預留執行個體使用支援產品。您一律必須支付支援產品賣方指定的價格。

若要將產品代碼與您AMI的產品代碼相關聯，請使用下列其中一個指令，其中 ami_id 是產品代碼的識別碼：AMI

- modify-image-attribute (AWS CLI)

```
aws ec2 modify-image-attribute --image-id ami_id --product-codes "product_code"
```

- Edit-EC2ImageAttribute (AWS Tools for Windows PowerShell)

```
PS C:\> Edit-EC2ImageAttribute -ImageId ami_id -ProductCode product_code
```

在您設定產品代碼屬性後，您便無法變更或移除它。

已付款和支援帳單 AMIs

在每個月底，您會收到一封電子郵件，其中包含您的信用卡在該月使用任何付款或支AMIs持的金額。此賬單與您的常規 Amazon EC2 賬單分開。如需詳細資訊，請參閱 AWS Marketplace 買方指南中的產品付款。

管理您的 AWS Marketplace 訂閱

在 AWS Marketplace 網站上，您可以檢查訂閱詳細資料、檢視廠商的使用說明、管理您的訂閱等。

檢查您的訂閱詳細資訊

1. 登入 AWS Marketplace。
2. 選擇 Your Marketplace Account (您的 Marketplace 帳戶)。
3. 選擇 Manage your software subscriptions (管理您的軟體訂閱)。
4. 將會列出所有您目前的訂閱。選擇「使用指示」以檢視使用產品的特定指示，例如連線至執行中執行個體的使用者名稱。

若要取消訂 AWS Marketplace 閱

1. 請確認您已終止任何從該訂閱執行的執行個體。
   a. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
   b. 在導覽窗格中，選擇 Instances (執行個體)。
   c. 選取執行個體，然後選取 Instance state (執行個體狀態)、Terminate instance (終止執行個體)。
   d. 出現確認提示時，請選擇 Terminate (終止)。
2. 登入 AWS Marketplace，然後依序選擇 Your Marketplace Account (您的 Marketplace 帳戶) 及 Manage your software subscriptions (管理您的軟體訂閱)。
3. 選擇 Cancel subscription (取消訂閱)。系統會提示您確認取消。

> **Note**
> 取消訂閱後，您將無法再從中啟動任何執行個體AMI。若要AMI再次使用，您需要在 AWS Marketplace 網站上或透過 Amazon EC2 主控台中的啟動精靈重新訂閱。

Amazon EC2 AMI 週期

Amazon 機器映像檔 (AMI) 是提供設定和啟動執行個體所需軟體的映像檔。您必須在啟動執行個體AMI時指定。

Amazon 提供您AMIs可以用來啟動執行個體，也可以建立自己的執行個體AMIs。例如，您可以從現有的執行個體啟動執行個體AMI、自訂執行個體 (例如，安裝軟體和設定作業系統設定)，然後將此更新的環境另存為新的環境AMI。所有執行個體自訂都會儲存到AMI，以便您從新執行個體啟動的執行個體AMI包含這些自訂項目。

您AMI只能在其 AWS 區域 中使用建立它的。如果您需要在多個區域中啟動具有相同組態的執行個體，您可以在一個區域AMI中建立一個，然後將您的執行個體複製AMI到其他區域。

若要避免暫時使用執行個體，您可以停用AMI。停用之後AMI，就無法使用它來啟動新的執行個體。啟用之後AMI，您可以使用它再次啟動執行個體。請注意，取消註冊AMI並不會影響您已從中啟動的AMI實例。

當您不再需要時AMI，可以將其取消註冊。取消註冊之後AMI，您無法使用它來啟動新的執行個體。請注意，取消註冊AMI並不會影響您已從中啟動的AMI實例。

您可以使用 Amazon Data Lifecycle Manager 自動化 Amazon EBS 支援的快照及其支援快照的建立、保留、複製、棄用AMIs和取消註冊。如需詳細資訊，請參閱 Amazon Data Lifecycle Manager。

目錄

- 創建一個 Amazon 支EBS持 AMI
- 建立執行個體存放區支援 AMI
- EC2AMI使用視窗系統創建一個 Amazon
- 複製 Amazon EC2 AMI
- AMI使用 S3 存放和恢復
- 檢查 Amazon 上EC2AMI次使用的時間
- 棄用 Amazon EC2 AMI
- 禁用 Amazon EC2 AMI
- 註銷 Amazon EC2 AMI

創建一個 Amazon 支EBS持 AMI

您可以AMI從 Amazon 執行個EC2體或 Amazon 執行個體根裝置的快照建立自己的 EC2 Amazon EBS 支援。

若要AMI從執行個體建立 Amazon EBS 支援，請先使用現有 Amazon 支援的執行個體啟動執行個EBS體AMI。這AMI可以是您從中取得 AWS Marketplace、使用虛擬機器匯入/匯出建立的，或您可以存取

AMI的任何其他項目。自訂執行個體以符合您的特定需求後，請建立並註冊新的執行個體AMI。然後，您可以使用 new AMI 來啟動包含自訂項目的新執行個體。

以下描述的程序適用於由加密 EC2 Amazon 彈性區塊存放區 (AmazonEBS) 磁碟區 (包括根磁碟區) 支援的 Amazon 執行個體，以及未加密磁碟區。

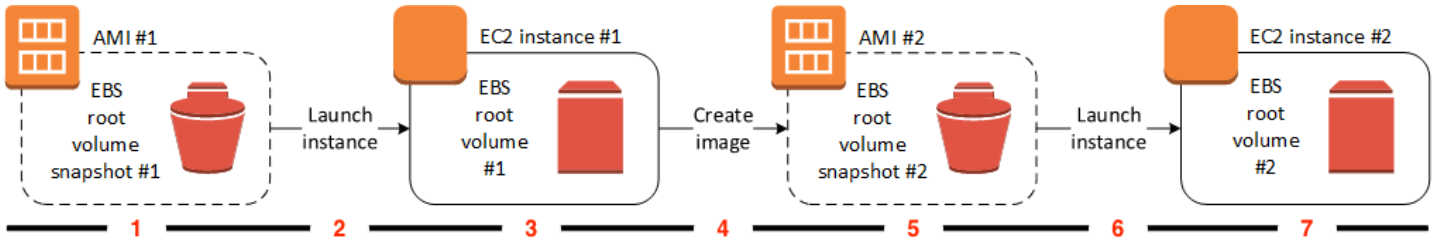
實例存儲AMIs支持的AMI創建過程是不同的。如需詳細資訊，請參閱建立執行個體存放區支援 AMI。

目錄

- 從執行個體AMI建立的概觀
- 從執行AMI個體建立
- AMI從快照建立

從執行個體AMI建立的概觀

下圖總結了從執行中執行個體建立 Amazon EBS-backed 的程序：AMI從現有EC2執行個體開始AMI、啟動執行個體、自訂執行個體、AMI從中建立新執行個體，最後啟動新AMI執行個體。該圖表中的數字與隨後描述中的數字相符。



1 — AMI #1：從現有的開始 AMI

找到與您要創建的AMI類似的現有AMI內容。這可以是AMI您從中取得的 AWS Marketplace、您使用虛擬機器匯入/匯出建立的，或AMI是您可以存取的任何其他項目。AMI您將根據您的需求AMI進行自定義。

在圖表中，EBS根磁碟區快照 #1 表示AMI為 Amazon EBS 支援，AMI且根磁碟區的相關資訊儲存在此快照中。

2 — 從現有啟動執行個體 AMI

設定執行個體的方法AMI是從您想要建立新執行個體的基礎AMI上啟動執行個體AMI，然後自訂執行個體 (在圖表中的 3 表示)。然後，您將創建一個包AMI含自定義的新內容（在圖表中的 4 表示）。

3 — EC2 執行個體 #1: 自訂執行個體

連線到您的執行個體並根據需要自訂。您的新功能AMI將包括這些自訂項目。

您可以在執行個體上執行下列任一個動作以自訂：

- 安裝軟體和應用程式
- 複製資料
- 刪除暫存檔案並重組您的硬碟，來減少啟動時間。
- 連接額外的 EBS 磁碟區

4 – 建立映像

當您AMI從執行個體建立執行個體時，Amazon 會先關EC2閉執行個體的電源，AMI以確保在建立程序期間停止執行個體上的所有內容並保持一致狀態。如果您確信執行個體處於適合AMI建立的一致狀態，可以告訴 Amazon EC2 不要關閉執行個體電源並重新啟動。某些檔案系統 (例如) 可以凍結和解凍活動，讓您在不重新啟動執行個體的情況下建立映像檔是安全的。XFS

在建AMI立過程中，Amazon EC2 會建立執行個體根磁碟區以及連接到執行個體的任何其他EBS磁碟區的快照。您需要支付快照費用，直到您取消註冊AMI並刪除快照為止。如果連接到執行個體的任何磁碟區已加密，則AMI只有在支援 Amazon 加EBS密的執行個體上才能成功啟動新磁碟區。

視磁碟區的大小而定，AMI建立程序可能需要幾分鐘的時間才能完成 (有時最長可達 24 小時)。在建立磁碟區之前，您可能會發現建立磁碟區快照更有效率AMI。如此一來，建立時只需要建立小型的AMI增量快照，而且程序會更快地完成 (快照建立的總時間保持不變)。

5 — AMI #2: 全新 AMI

程序完成後，您會從執行個AMI體的根磁碟區建立新的快照 (快照 #2)。如果您已將執行個體儲存磁碟區或EBS磁碟區新增至執行個體，除了根裝置磁碟區之外，新磁碟區的區塊裝置對應會AMI包含這些磁碟區的資訊。

Amazon EC2 會自動AMI為您註冊。

6 — 從新的啟動實例 AMI

您可以使用 new AMI 來啟動執行個體。

7 — EC2 執行個體 #2: 新執行個體

當您使用新的執行個體啟動執行個體時AMI，Amazon EC2 會使用快照為執行個體的根磁碟區建立新EBS磁碟區。如果您在自訂執行個體時新增了執行個體儲存EBS磁碟區或磁碟區，則新磁碟區的區塊裝置對映會AMI包含這些磁碟區的資訊，而從新啟動的執行個體的區塊裝置對應AMI會自動包含這些磁碟區的資訊。在新執行個體的區塊裝置對應中指定的執行個體儲存磁碟區是新的，不包含

您用來建立執行個體的執行個體儲存磁碟區的任何資料。AMIEBS磁碟區上的資料仍然存在。如需詳細資訊，請參閱Amazon EC2 執行個體上磁碟區的區塊裝置對映。

當您從 EBS-backed 建立新的執行個體時AMI，您應該先初始化其根磁碟區和任何其他EBS儲存，然後再將其投入生產環境。如需詳細資訊，請參閱 Amazon EBS使用者指南中的初始化 Amazon EBS 磁碟區。

從執行AMI個體建立

如果您有現有的執行個體，您可以AMI從這個執行個體建立。

Console

若要建立 AMI

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取要從中建立的執行個體AMI，然後選擇 [動作]、[映像和範本] > [建立映像]。

> **Tip**
> 如果停用此選項，表示您的執行個體不是 Amazon EBS 支援的執行個體。

4. 在 Create image (建立映像) 頁面上，指定下列資訊：
   a. 對於 Image name (映像名稱)，輸入映像的唯一名稱，最多 127 個字元。
   b. 對於 Image description (映像描述)，輸入映像的選擇性描述，最多 255 個字元。
   c. 對於 Reboot 執行個體，請保持核取方塊保持選取狀態 (預設值)，或清除核取方塊。
      - 如果選取了 Re boot 執行個體，Amazon EC2 建立新執行個體時會重新啟動執行個體AMI，以便在資料處於靜態狀態時拍攝連接磁碟區的快照，以確保一致的狀態。
      - 如果清除重新啟動執行個體，Amazon EC2 建立新執行個體時AMI，不會關閉執行個體並重新啟動執行個體。

> **Warning**
> 如果您清除 Reboot 執行個體，我們無法保證建立映像檔的檔案系統完整性。

d. 執行個體磁碟區 — 您可以修改根磁碟區，以及新增其他 Amazon EBS 和執行個體存放區磁碟區，如下所示：

   i. 根磁碟區在第一列中定義。

      - 若要變更根磁碟區的大小，請在 Size (大小) 中輸入所需的值。
      - 如果您選取「終止時刪除」(Delete)，當您終止從此建立的例證時AMI，EBS磁碟區會被刪除。如果您在終止時清除「刪除」(Delete)，則當您終止執行個體時，EBS磁碟區不會被刪除。如需詳細資訊，請參閱在執行個體終止時保留資料。

   ii. 若要新增EBS磁碟區，請選擇 [新增磁碟區] (這會新增一列)。對於「儲存空間類型」EBS，請選擇並填寫列中的欄位。當您從新執行個體啟動執行個體時AMI，其他磁碟區會自動附加至執行個體。空白磁碟區必須進行格式化及掛載。以快照為基礎的磁碟區必須掛載。

   iii. 若要新增執行個體存放區磁碟區，請參閱將實例存儲卷添加到 Amazon EC2 AMI。當您從新的執行個體啟動執行個體時AMI，系統會自動初始化並掛接其他磁碟區。這些磁碟區不包含您所依據的執行個體執行個體的執行個體儲存磁碟區中的資料AMI。

e. 標籤 — 您可以使用相同的標籤來標記AMI和快照，也可以使用不同的標籤來標記快照。

   - 若要使用相同標籤標記AMI和快照，請選擇「將影像和快照標記在一起」。相同的標籤會套用AMI至所建立的每個快照。
   - 若要使用不同標籤來標記AMI和快照，請分別選擇「標記影像和快照」。不同的標籤會套用至所建立的AMI和快照。但是所有快照都會取得相同的標籤；您無法使用不同的標籤標記每個快照。

   若要新增標籤，請選擇 Add tag (新增標籤)，然後輸入標籤的鍵和值。針對每個標籤，重複執行上述動作。

f. 當您準備好建立影像時AMI，請選擇 [建立映像檔]。

5. 若要檢視您正在建立AMI時的狀態，請執行下列步驟：

   a. 在導覽窗格中，選擇AMIs。

   b. 將過濾器設置為「我擁有」，然後在列表AMI中找到您的。

      一開始，狀態為 `pending`，但在幾分鐘後應該會變更為 `available`。

6. (選擇性) 若要檢視為新建立的快照，請執行下列動作AMI：

   a. 請記下您AMI在上一個步驟中找到的 ID。

b. 在導覽窗格中，選擇 Snapshots (快照)。

c. 將篩選器設定為 [我擁有]，然後在 [說明] 欄中尋找含有新 AMI ID 的快照。

當您從此啟動執行個體時AMI，Amazon EC2 會使用此快照建立其根裝置磁碟區。

AWS CLI

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱訪問 Amazon EC2。

- create-image (AWS CLI)
- New-EC2Image (AWS Tools for Windows PowerShell)

AMI從快照建立

如果您有執行個體根裝置磁碟區的快照，您可以AMI從此快照建立快照。

> **Note**
>
> 在大多數情況下，AMIs對於 Windows、 RedHatSUSE、和SQL伺服器而言，需要正確的授權資訊才能顯示在AMI. 如需詳細資訊，請參閱了解 AMI 帳單資訊。AMI從快照建立時，`RegisterImage`作業會從快照的中繼資料衍生正確的帳單資訊，但這需要有適當的中繼資料。要驗證是否應用了正確的帳單信息，請檢查新的平台詳細信息字段AMI。如果欄位空白或與預期的作業系統程式碼 (例如 Windows、、RedHatSUSE、或SQL) 不相符，表示AMI建立失敗，您應該捨棄AMI並遵循中的指示從執行AMI個體建立。

Console

若要AMI從快照建立

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Snapshots (快照)。
3. 選取要從中建立的快照AMI，然後選擇動作 > 從快照建立映像。
4. 在 [從快照建立映像] 頁面上，指定下列資訊：

   a. 對於 Image name (映像名稱)，輸入映像的描述性名稱。

   b. 對於 Description (描述)，輸入映像的簡短描述。

c. 對於 Architecture (架構)，選擇映像架構。如果是 32 位元，請選擇「i386」(64 位元)、64 位元選擇「arm64」(64 位元)，或選擇 64 位ARM元 (64 位元 macOS)。

d. 對於 Root device name (根設備名稱)，輸入要用於根設備磁碟區的設備名稱。如需詳細資訊，請參閱Amazon EC2 執行個體上磁碟區的裝置名稱。

e. 針對虛擬化類型，請選擇從此啟動的執行個體要使用的虛擬化類型AMI。如需詳細資訊，請參閱虛擬化類型。

f. (僅適用於半虛擬化) 對於 Kernel ID (核心 ID)，選取映像的作業系統核心。如果您是使用執行個體的根設備磁碟區快照，請選取與原始執行個體相同的核心 ID。如果不確定，請使用預設核心。

g. (僅適用於半虛擬化) 針對RAM磁碟 ID，請選取映像檔的RAM磁碟。如果您選取特定的核心，您可能需要選取含有驅動程式的特定RAM磁碟來支援該核心。

h. 對於「開機」模式，請選擇映像檔的開機模式，或選擇 [使用預設值]，如此當執行個體啟動時AMI，執行個體會以執行個體類型支援的開機模式啟動。如需詳細資訊，請參閱設置 Amazon 的啟動模式 EC2 AMI。

i. (選擇性) 在封鎖裝置對應下，自訂根磁碟區並新增其他資料磁碟區。

   針對每個磁碟區，您可以指定大小、類型、效能特性、終止時的刪除行為，以及加密狀態。若為根磁碟區，大小不可小於快照的大小。對於磁碟區類型，「一般用途」SSD gp3 是預設選項。

j. (選擇性) 在「標籤」下，您可以將一或多個標籤新增至新標籤AMI。若要新增標籤，請選擇 Add tag (新增標籤)，然後輸入標籤的鍵和值。針對每個標籤，重複執行上述動作。

k. 當您準備好建立影像時AMI，請選擇 [建立映像檔]。

5. (僅適用於 Windows RedHat、SUSE、和SQL伺服器) 若要確認是否套用了正確的帳單資訊，請檢查新的平台詳細資料欄位AMI。如果欄位空白或與預期的作業系統程式碼 (例如 Windows 或 RedHat) 不相符，表示AMI建立失敗，您應該捨棄AMI並遵循中的指示從執行AMI個體建立。

AWS CLI

使用指令行AMI從快照建立

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱訪問 Amazon EC2。

- register-image (AWS CLI)
- Register-EC2Image (AWS Tools for Windows PowerShell)

建立執行個體存放區支援 AMI

您AMI在啟動執行個體時指定的決定了根裝置磁碟區的類型。

若要建立以商店為基礎的 Linux 執行個體AMI，請從您從現有執行個體商店支援的 Linux 啟動的執行個體開始。AMI自訂執行個體以滿足您的需求後，請捆綁磁碟區並註冊新的執行個體AMI，您可以使用這些自訂項啟動新的執行個體。

您無法建立執行個體存放區支援的 Windows，AMI因為 Windows AMIs 不支援根裝置的執行個體存放區。

> **Important**
>
> 只有下列執行個體類型支援執行個體儲存區磁碟區做為根裝置：
> C1、C3、D2、I2、M2、M2、M3、R3 和 X1。

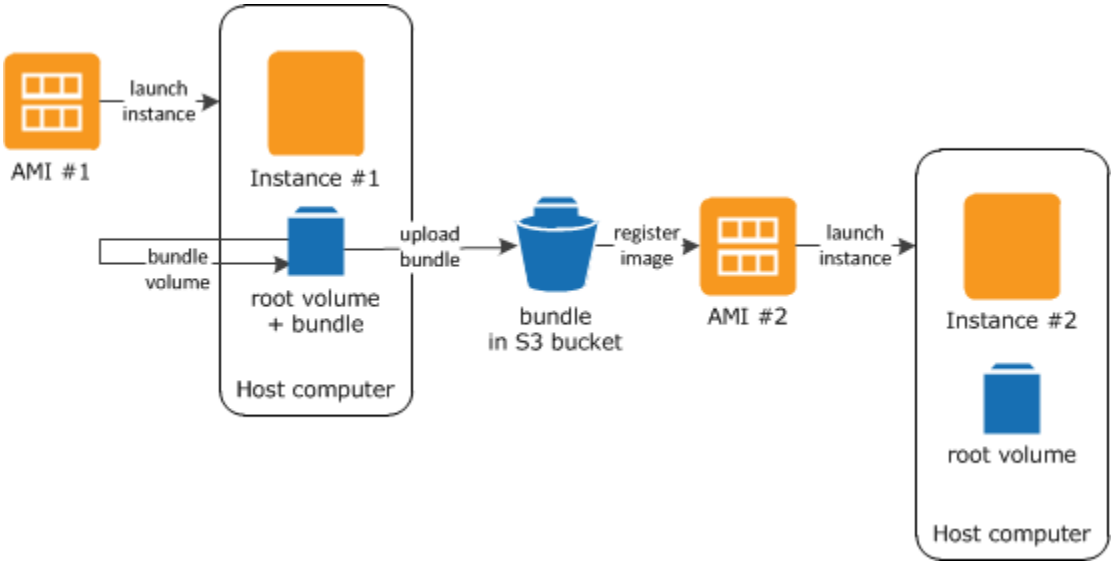
Amazon EBS 支持的AMI創建過程是不同的AMIs。如需詳細資訊，請參閱創建一個 Amazon 支EBS持AMI。

目錄

- AMI創作概述
- 必要條件
- AMI從 Amazon Linux 實例創建一個
- 設置 Amazon EC2 AMI 工具
- Amazon EC2 AMI 工具參考
- 將您的執行個體存放區轉換AMI為支援 EBS AMI

AMI創作概述

下圖摘要說明AMI從執行處理商店支援執行處理建立的程序。



首先，從類似於您要建立的執行個AMI體啟動執行個體。AMI您可以連線到您的執行個體並自訂。設定好所要的執行個體後，您可以組合執行個體。完成綁定程序需要幾分鐘。程序完成後，您將擁有 bundle，其中包含映像工作資訊清單檔案 (`image.manifest.xml`) 和含有根磁碟區範本的檔案 (`image.part.xx`)。接下來，您將捆綁包上傳到您的 Amazon S3 存儲桶，然後註冊您的AMI.

> **Note**
>
> 若要將物件上傳至執行個體商店支援 Linux 的 S3 儲存貯體AMI，ACLs必須為儲存貯體啟用。否則，Amazon EC2 將無法在要上傳的對象ACLs上進行設置。如果您的目標儲存貯體使用儲存貯體擁有者強制執行 S3 物件擁有權的設定，這將無法運作，因ACLs為已停用。如需詳細資訊，請參閱使用 S3 物件擁有權控制上傳物件的擁有權。

當您使用新的執行個體啟動執行個體時AMI，我們會使用您上傳到 Amazon S3 的服務包為執行個體建立根磁碟區。Amazon S3 中 bundle 所用的儲存空間會使您的帳戶產生費用，直到刪除為止。如需詳細資訊，請參閱註銷 Amazon EC2 AMI。

如果除了根裝置磁碟區之外，將執行個體存放區磁碟區新增至執行個體，則新磁碟區的區塊裝置對應會AMI包含這些磁碟區的資訊，而從新磁碟區啟動的執行個體的區塊裝置對應AMI會自動包含這些磁碟區的資訊。如需詳細資訊，請參閱Amazon EC2 執行個體上磁碟區的區塊裝置對映。

必要條件

您必須先完成下列工作AMI，才能建立：

- 安裝工AMI具。如需詳細資訊，請參閱設置 Amazon EC2 AMI 工具。

- 安裝 AWS CLI. 如需詳細資訊，請參閱開始設定 AWS Command Line Interface。
- 請確定您有適用於套件的 S3 儲存貯體，並且儲存貯體已ACLs啟用。如需有關設定的詳細資訊 ACLs，請參閱設定ACLs。
  - 若要使用建立 S3 儲存貯體 AWS Management Console，請在的開啟 Amazon S3 主控台，https://console.aws.amazon.com/s3/然後選擇建立儲存貯體。
  - 若要使用建立 S3 儲存貯體 AWS CLI，您可以使用 mb 命令。如果您安裝的AMI工具版本為 1.5.18 或更新版本，您也可以使用`ec2-upload-bundle`命令建立 S3 儲存貯體。如需詳細資訊，請參閱ec2-upload-bundle。
- 確保您服務包中的檔案未在 S3 儲存貯體中加密。如果您需要加密AMI，則可以使用 EBS-backed 來AMI代替。如需詳細資訊，請參閱對 EBS 後端 AMI 使用加密。
- 確保您有您的 AWS 帳戶 ID。如需詳細資訊，請參閱《AWS 帳戶管理參考指南》中的檢視 AWS 帳戶 識別碼。
- 請確定您擁有可使用 AWS CLI的憑證。如需詳細資訊，請參閱《AWS Account Management 參考指南》中的AWS 帳戶最佳做法。
- 請確定您擁有 X.509 憑證和對應的私有金鑰。
  - 如果需要建立 X.509 憑證，請參閱管理簽署憑證。X.509 憑證和私密金鑰可用來加密和解密您的 AMI.
  - [中國 (北京)] 使用 `$EC2_AMITOOL_HOME/etc/ec2/amitools/cert-ec2-cn-north-1.pem` 憑證。
  - [AWS GovCloud (美國西部)] 使用`$EC2_AMITOOL_HOME/etc/ec2/amitools/cert-ec2-gov.pem`憑證。
- 連線到您的執行個體並加以自訂。例如，您可以安裝軟體和應用程式、複製資料、刪除暫存檔，以及修改 Linux 組態。

AMI從 Amazon Linux 實例創建一個

下列程序說明如何AMI從執行 Amazon Linux 1 的執行個體商店支援執行個體建立。它們可能不適用於運行其他 Linux 發行版的實例。

準備使用工AMI具（僅HVM實例）

1. 這些AMI工具需要GRUB傳統才能正常啟動。使用以下命令進行安裝GRUB：

```
[ec2-user ~]$ sudo yum install -y grub
```

2. 使用下列命令安裝分割區管理套件：

```
[ec2-user ~]$ sudo yum install -y gdisk kpartx parted
```

若要AMI從執行個體以商店支援的 Amazon Linux 執行個體建立

此程序假設您已滿足 必要條件 中的先決條件。

在以下命令中，替換每個命令 *user input placeholder* 使用您自己的信息。

1. 將您的登入資料上傳到執行個體。我們使用這些登入資料來確保只有您和 Amazon 才EC2能存取您的AMI.

   a. 在執行個體上為登入資料建立如下的暫時目錄：

   ```
   [ec2-user ~]$ mkdir /tmp/cert
   ```

   如此您就能將登入資料從建立的映像中排除。

   b. 使用 scp 等安全複製工具從電腦上將 X.509 憑證和對應的私有金鑰複製到執行個體上的 `/tmp/cert` 目錄內。下列scp命令中的`-i` *`my-private-key`*`.pem`選項是您用來連線至執行個體的私密金鑰SSH，而非 X.509 私密金鑰。例如：

   ```
   you@your_computer:~ $ scp -i my-private-key.pem /
   path/to/pk-HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem /
   path/to/cert-HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem ec2-
   user@ec2-203-0-113-25.compute-1.amazonaws.com:/tmp/cert/
   pk-HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem  100%  717     0.7KB/s   00:00
   cert-HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem  100%  685     0.7KB/s   00:00
   ```

   或者，由於這些檔案為純文字檔案，因此您可用文字編輯器開啟憑證和金鑰，然後將內容複製到 `/tmp/cert` 中的新檔案內。

2. 從執行個體內執行 ec2-bundle-vol 命令，準備要上傳至 Amazon S3 的 bundle。務必指定 `-e` 選項，以排除憑證存放位置的目錄。根據預設，bundle 程序不包含可能帶有敏感資訊的檔案。這些檔案包含 `*.sw`、`*.swo`、`*.swp`、`*.pem`、`*.priv`、`*id_rsa*`、`*id_dsa*` `*.gpg`、`*.jks`、`*/.ssh/authorized_keys` 和 `*/.bash_history`。若要包含所有這些檔案，請使用 `--no-filter` 選項。若要包含這些檔案中的一部分，請使用 `--include` 選項。

> **Important**
>
> 依預設，AMI捆綁程序會在代表您的根磁碟區的`/tmp`目錄中建立壓縮、加密的檔案集合。如果 `/tmp` 內沒有足夠的可用磁碟空間可存放 bundle，您需要用 `-d` *`/path/to/bundle/storage`* 選項指定其他的 bundle 存放位置。某些執行個體會掛載在`/mnt`或您可以使用`/media/ephemeral0`的暫時儲存，或者您也可以建立、連接和掛接新的AmazonEBS) 磁碟區來存放組合。如需詳細資訊，請參閱 Amazon EBS使用者指南中的建立 Amazon EBS 磁碟區。

a. 您必須以 root 執行 ec2-bundle-vol 命令。針對多數命令，您皆能用 sudo 來取得更高級的許可，但在此情況下您應執行 sudo -E su 以保持環境變數。

```
[ec2-user ~]$ sudo -E su
```

請注意，bash 提示現在將您識別為根使用者，且貨幣符號已換成雜湊標籤，表示您正在根 shell 下：

```
[root ec2-user]#
```

b. 若要建立AMI套裝軟體，請執行下列ec2-bundle-vol命令：

```
[root ec2-user]# ec2-bundle-vol -k /tmp/cert/pk-
HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem -c /tmp/cert/cert-
HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem -u 123456789012 -r x86_64 -e /tmp/cert --
partition gpt
```

> **Note**
>
> 對於中國 (北京) 和 AWS GovCloud (美國西部) 區域，請使用`--ec2cert`參數並根據必要條件指定憑證。

建立映像可能需要幾分鐘。完成此指令後，您的 `/tmp` (或非預設) 目錄會包含套件 (`image.manifest.xml`加上多個) `image.part.`*`xx`* 文件）。

c. 退出根 shell。

```
[root ec2-user]# exit
```

3. (選擇性) 若要新增更多執行個體儲存磁碟區，請在`image.manifest.xml`檔案中為您的AMI. 如需詳細資訊，請參閱Amazon EC2 執行個體上磁碟區的區塊裝置對映。

   a. 為您的 `image.manifest.xml` 檔案建立備份。

   ```
   [ec2-user ~]$ sudo cp /tmp/image.manifest.xml /tmp/image.manifest.xml.bak
   ```

   b. 重新格式化 `image.manifest.xml` 檔案，以方便讀取和編輯。

   ```
   [ec2-user ~]$ sudo xmllint --format /tmp/image.manifest.xml.bak > /tmp/
   image.manifest.xml
   ```

   c. 使用文字編輯器編輯 `image.manifest.xml` 中的區塊型裝置映射。下列範例顯示 `ephemeral1` 執行個體存放磁碟區的新項目。

   > **Note**
   > 如需排除檔案清單，請參閱 ec2-bundle-vol。

   ```
   <block_device_mapping>
     <mapping>
       <virtual>ami</virtual>
       <device>sda</device>
     </mapping>
     <mapping>
       <virtual>ephemeral0</virtual>
       <device>sdb</device>
     </mapping>
     <mapping>
       <virtual>ephemeral1</virtual>
       <device>sdc</device>
     </mapping>
     <mapping>
       <virtual>root</virtual>
       <device>/dev/sda1</device>
     </mapping>
   </block_device_mapping>
   ```

d. 儲存 `image.manifest.xml` 檔案並結束文字編輯器。

4. 若要上傳 bundle 至 Amazon S3，請照下列所示執行 ec2-upload-bundle 命令。

```
[ec2-user ~]$ ec2-upload-bundle -b amzn-s3-demo-bucket/bundle_folder/bundle_name -
m /tmp/image.manifest.xml -a your_access_key_id -s your_secret_access_key
```

> **Important**
>
> 若要AMI在美國東部 (維吉尼亞北部) 以外的區域註冊，您必須同時指定目標區域與`--region`選項，以及已存在於目標區域中的值區路徑，或指定可在目標區域建立的唯一值區路徑。

5. (選用) bundle 上傳至 Amazon S3 之後，您可使用下列 `/tmp` 命令將 bundle 從執行個體的 rm 目錄中移除：

```
[ec2-user ~]$ sudo rm /tmp/image.manifest.xml /tmp/image.part.* /tmp/image
```

> **Important**
>
> 如果用 `-d` *`/path/to/bundle/storage`* 選項指定 Step 2 中的路徑，請使用該路徑，而非 `/tmp`。

6. 要註冊您的AMI，請運行寄存器映像命令，如下所示。

```
[ec2-user ~]$ aws ec2 register-image --image-location amzn-s3-demo-
bucket/bundle_folder/bundle_name/image.manifest.xml --name AMI_name --
virtualization-type hvm
```

> **Important**
>
> 如果您先前已指定 ec2-upload-bundle 命令的區域，請在此命令中再次指定該區域。

設置 Amazon EC2 AMI 工具

您可以使用這些工AMI具來建立和管理執行個體商店支援的 Linux。AMIs若要使用工具，您必須在 Linux 執行個體上安裝工具。對於不支RPM援的 Linux 發行版，這些AMI工具可作為RPM和 .zip 檔案使用。

若要使AMI用 RPM

1. 使用 Linux 發行版本的套件管理工具 (例如 yum) 來安裝 Ruby。例如：

```
[ec2-user ~]$ sudo yum install -y ruby
```

2. 使用 RPM wget 或捲曲等工具下載檔案。例如：

```
[ec2-user ~]$ wget https://s3.amazonaws.com/ec2-downloads/ec2-ami-tools.noarch.rpm
```

3. 使用下列指令驗證RPM檔案的簽章：

```
[ec2-user ~]$ rpm -K ec2-ami-tools.noarch.rpm
```

上面的命令應該指示該文件的SHA1和MD5哈希是OK.如果命令指示哈希值NOT OK，請使用以下命令查看文件的 Header SHA1 和MD5哈希值：

```
[ec2-user ~]$ rpm -Kv ec2-ami-tools.noarch.rpm
```

然後，將檔案的標頭SHA1和MD5雜湊與下列驗證AMI工具雜湊進行比較，以確認檔案的真實性：

- Header SHA1: a1f662d6f25f69871104e6a62187fa4df508f880
- MD5: 9faff05258064e2f7909b66142de6782

如果檔案的標頭SHA1和MD5雜湊符合驗證的AMI工具雜湊，請繼續下一步。

4. RPM使用以下命令安裝：

```
[ec2-user ~]$ sudo yum install ec2-ami-tools.noarch.rpm
```

5. 使用ec2-ami-tools-version指令驗證您的AMI工具安裝。

```
[ec2-user ~]$ ec2-ami-tools-version
```

> **Note**
>
> 如果您收到載入錯誤，例如「無法載入此類檔案-ec2/amitools/版本 (LoadError)」，請完成下一個步驟，將工具安裝的AMI位置新增至路徑。RUBYLIB

6. (選擇性) 如果您在上一個步驟中收到錯誤，請將AMI工具安裝位置新增至RUBYLIB路徑。

   a. 執行下列命令來判斷要加入的路徑。

   ```
   [ec2-user ~]$ rpm -qil ec2-ami-tools | grep ec2/amitools/version
   /usr/lib/ruby/site_ruby/ec2/amitools/version.rb
   /usr/lib64/ruby/site_ruby/ec2/amitools/version.rb
   ```

   在上例中，先前載入錯誤所缺少的檔案位在 `/usr/lib/ruby/site_ruby` 和 `/usr/lib64/ruby/site_ruby`。

   b. 在先前的步驟中將位置加入到 `RUBYLIB` 路徑。

   ```
   [ec2-user ~]$ export RUBYLIB=$RUBYLIB:/usr/lib/ruby/site_ruby:/usr/lib64/ruby/site_ruby
   ```

   c. 使用ec2-ami-tools-version指令驗證您的AMI工具安裝。

   ```
   [ec2-user ~]$ ec2-ami-tools-version
   ```

使用 .zip 檔案設定AMI工具

1. 使用 Linux 發行版本的套件管理工具 (例如 apt-get) 來安裝 Ruby 及解壓縮。例如：

   ```
   [ec2-user ~]$ sudo apt-get update -y && sudo apt-get install -y ruby unzip
   ```

2. 使用 wget 或 curl 等工具下載 .zip 檔案。例如：

   ```
   [ec2-user ~]$ wget https://s3.amazonaws.com/ec2-downloads/ec2-ami-tools.zip
   ```

3. 將檔案解壓縮到適當的安裝目錄，例如 `/usr/local/ec2`。

   ```
   [ec2-user ~]$ sudo mkdir -p /usr/local/ec2
   $ sudo unzip ec2-ami-tools.zip -d /usr/local/ec2
   ```

請注意，.zip 文件包含一個文件夾 ec2-阿密工具-*x.x.x*其中，*x.x.x* 是工具的版本號碼 (例如，`ec2-ami-tools-1.5.7`)。

4. 將 `EC2_AMITOOL_HOME` 環境變數設定為工具的安裝目錄。例如：

```
[ec2-user ~]$ export EC2_AMITOOL_HOME=/usr/local/ec2/ec2-ami-tools-x.x.x
```

5. 將工具加入到 PATH 環境變數。例如：

```
[ec2-user ~]$ export PATH=$EC2_AMITOOL_HOME/bin:$PATH
```

6. 您可以使用ec2-ami-tools-version指令驗證您的AMI工具安裝。

```
[ec2-user ~]$ ec2-ami-tools-version
```

管理簽署憑證

AMI工具中的某些指令需要簽署憑證 (也稱為 X.509 憑證)。您必須建立憑證，然後將其上傳至 AWS。例如，您可以使用第三方工具 (例如 Open) SSL 來建立憑證。

建立簽署憑證

1. 安裝並設定「開啟SSL」。
2. 使用 `openssl genrsa` 命令建立私有金鑰，並將輸出儲存至 `.pem` 檔案。我們建議您建立一個 2048 位元或 4096 RSA 位元金鑰。

```
openssl genrsa 2048 > private-key.pem
```

3. 使用 `openssl req` 命令產生憑證。

```
openssl req -new -x509 -nodes -sha256 -days 365 -key private-key.pem -outform PEM -
out certificate.pem
```

若要將憑證上傳至 AWS，請使用upload-signing-certificate指令。

```
aws iam upload-signing-certificate --user-name user-name --certificate-body
 file://path/to/certificate.pem
```

若要列出使用者的憑證，請使用以下list-signing-certificates指令：

```
aws iam list-signing-certificates --user-name user-name
```

若要停用或重新啟用使用者的簽署憑證，請使用指update-signing-certificate令。下列命令可停用憑證：

```
aws iam update-signing-certificate --certificate-id OFHPLP4ZULTHYPMSYEX704BEXAMPLE --
status Inactive --user-name user-name
```

若要刪除憑證，請使用以下delete-signing-certificate指令：

```
aws iam delete-signing-certificate --user-name user-name --certificate-
id OFHPLP4ZULTHYPMSYEX704BEXAMPLE
```

Amazon EC2 AMI 工具參考

您可以使用AMI工具命令來建立和管理執行個體存放區支援的 Linux。AMIs欲設定這些工具，請參閱設置 Amazon EC2 AMI 工具。

如需有關存取金鑰的資訊，請參閱《使IAM用指南》中的〈管理使用IAM者的存取金鑰〉。

命令

- ec2-ami-tools-version
- ec2-bundle-image
- ec2-bundle-vol
- ec2-delete-bundle
- ec2-download-bundle
- ec2-migrate-manifest
- ec2-unbundle
- ec2-upload-bundle
- AMI工具的常用選項

ec2-ami-tools-version

描述

描述AMI工具的版本。

語法

ec2-ami-tools-version

輸出

版本資訊。

範例

此範例指令會顯示您正在使用之AMI工具的版本資訊。

```
[ec2-user ~]$ ec2-ami-tools-version
1.5.2 20071010
```

ec2-bundle-image

描述

AMI從迴路檔案中建立的作業系統映像檔建立執行個體以商店為基礎的 Linux。

語法

ec2-bundle-image -c *path* -k *path* -u *account* -i *path* [-d *path*] [--ec2cert *path*] [-r *architecture*] [--productcodes *code1*,*code2*,...] [-B *mapping*] [-p *prefix*]

選項

`-c, --cert` path

使用者的PEM編碼RSA公開金鑰憑證檔案。

必要：是

`-k, --privatekey` path

PEM編碼RSA金鑰檔案的路徑。您必須指定此金鑰來解開此 bundle，因此請將其置於安全位置。請注意，密鑰不必註冊到您的 AWS 帳戶。

必要：是

`-u, --user` account

使用者的 AWS 帳號 ID，不含破折號。

必要：是

`-i, --image` path

欲綁定的映像路徑。

必要：是

`-d, --destination` path

欲建立 bundle 的目錄。

預設：`/tmp`

必要：否

`--ec2cert` path

用於加密映像資訊清單EC2之 Amazon X.509 公開金鑰憑證的路徑。

`us-gov-west-1` 和 `cn-north-1` 區域使用非預設的公有金鑰憑證，而該憑證的路徑必須以此選項加以指定。憑證的路徑會根據AMI工具的安裝方法而有所不同。以 Amazon Linux 而言，此憑證位於 `/opt/aws/amitools/ec2/etc/ec2/amitools/`。如果您從中的RPM或ZIP檔案安裝AMI工具設置 Amazon EC2 AMI 工具，則憑證位於`$EC2_AMITOOL_HOME/etc/ec2/amitools/`。

必要：僅適用於 `us-gov-west-1` 和 `cn-north-1` 區域。

`-r, --arch` architecture

映像架構。若您未於命令列提供架構，則會在綁定開始時提示您輸入架構。

有效值：`i386` | `x86_64`

必要：否

`--productcodes` code1,code2,...

欲在註冊時連接至映像的產品代碼，以逗號分隔。

必要：否

`-B, --block-device-mapping` mapping

如果區塊裝置的執行個體類型支援指定的裝置，則定義區塊裝置AMI如何公開給此執行個體。

指定以逗號分隔的索引鍵/值組清單，其中索引鍵為虛擬名稱，而值為相對應的設備名稱。虛擬名稱包括下列：

- `ami` - 執行個體所看見的根檔案系統設備
- `root` - 核心所看見的根檔案系統設備
- `swap` - 執行個體所看見的置換裝置
- `ephemeralN` - 第 N 個執行個體存放磁碟區

必要：否

`-p, --prefix` prefix

捆綁AMI文件的文件名前綴。

預設：映像檔案名稱。舉例來說，若映像路徑為 `/var/spool/my-image/version-2/debian.img`，則預設字首為 `debian.img`。

必要：否

`--kernel` kernel_id

已廢除. 使用 register-image 來設定核心。

必要：否

`--ramdisk` ramdisk_id

已廢除。如果需要，使用寄存器映像來設置RAM磁盤。

必要：否

輸出

描述綁定程序階段與狀態的狀態訊息。

範例

此範例會AMI從迴路檔案中建立的作業系統映像建立隨附的。

```
[ec2-user ~]$ ec2-bundle-image -k pk-HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem -c cert-
HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem -u 111122223333 -i image.img -d bundled/ -r x86_64
Please specify a value for arch [i386]:
Bundling image file...
Splitting bundled/image.gz.crypt...
```

```
Created image.part.00
Created image.part.01
Created image.part.02
Created image.part.03
Created image.part.04
Created image.part.05
Created image.part.06
Created image.part.07
Created image.part.08
Created image.part.09
Created image.part.10
Created image.part.11
Created image.part.12
Created image.part.13
Created image.part.14
Generating digests for each part...
Digests generated.
Creating bundle manifest...
ec2-bundle-image complete.
```

ec2-bundle-vol

描述

AMI透過壓縮、加密和簽署執行個體的根裝置磁碟區副本，來建立執行個體存放區支援的 Linux。

Amazon EC2 嘗試從執行個體繼承產品代碼、核心設定、RAM磁碟設定和區塊裝置對應。

根據預設，bundle 程序不包含可能帶有敏感資訊的檔案。這些檔案包含 `*.sw`、`*.swo`、`*.swp`、`*.pem`、`*.priv`、`*id_rsa*`、`*id_dsa*` `*.gpg`、`*.jks`、`*/.ssh/authorized_keys` 和 `*/.bash_history`。若要包含所有這些檔案，請使用 `--no-filter` 選項。若要包含這些檔案中的一部分，請使用 `--include` 選項。

如需詳細資訊，請參閱 建立執行個體存放區支援 AMI。

語法

`ec2-bundle-vol -c path -k path -u account [-d path] [--ec2cert path] [-r architecture] [--productcodes code1,code2,...] [-B mapping] [--all] [-e directory1,directory2,...] [-i file1,file2,...] [--no-filter] [-p prefix] [-s size] [--[no-]inherit] [-v volume] [-P type] [-S script] [--fstab path] [--generate-fstab] [--grub-config path]`

選項

`-c, --cert` path

使用者的PEM編碼RSA公開金鑰憑證檔案。

必要：是

`-k, --privatekey` path

使用者PEM編碼RSA金鑰檔案的路徑。

必要：是

`-u, --user` account

使用者的 AWS 帳號 ID，不含破折號。

必要：是

`-d, --destination` destination

欲建立 bundle 的目錄。

預設：`/tmp`

必要：否

`--ec2cert` path

用於加密映像資訊清單EC2之 Amazon X.509 公開金鑰憑證的路徑。

`us-gov-west-1` 和 `cn-north-1` 區域使用非預設的公有金鑰憑證，而該憑證的路徑必須以此選項加以指定。憑證的路徑會根據AMI工具的安裝方法而有所不同。以 Amazon Linux 而言，此憑證位於 `/opt/aws/amitools/ec2/etc/ec2/amitools/`。如果您從中的RPM或ZIP檔案安裝AMI工具設置 Amazon EC2 AMI 工具，則憑證位於`$EC2_AMITOOL_HOME/etc/ec2/amitools/`。

必要：僅適用於 `us-gov-west-1` 和 `cn-north-1` 區域。

`-r, --arch` architecture

映像架構。若您未於命令列提供架構，則會在綁定開始時提示您提供架構。

有效值：`i386 | x86_64`

必要：否

`--productcodes` code1,code2,...

欲在註冊時連接至映像的產品代碼，以逗號分隔。

必要：否

`-B, --block-device-mapping` mapping

如果區塊裝置的執行個體類型支援指定的裝置，則定義區塊裝置AMI如何公開給此執行個體。

指定以逗號分隔的索引鍵/值組清單，其中索引鍵為虛擬名稱，而值為相對應的設備名稱。虛擬名稱包括下列：

- `ami` - 執行個體所看見的根檔案系統設備
- `root` - 核心所看見的根檔案系統設備
- `swap` - 執行個體所看見的置換裝置
- `ephemeralN` - 第 N 個執行個體存放磁碟區

必要：否

`-a, --all`

綁定所有目錄，包括遠端掛載之檔案系統上的目錄。

必要：否

`-e, --exclude` directory1,directory2,...

欲從 bundle 操作排除的絕對目錄路徑和檔案清單。此參數會覆寫 `--all` 選項。指定排除項目時，隨參數列出的目錄和子目錄將不會與磁碟區一同綁定。

必要：否

`-i, --include` file1,file2,...

欲納入 bundle 操作的檔案清單。否則，指定的檔案會從中排除，AMI因為它們可能包含敏感資訊。

必要：否

`--no-filter`

如果有指定，我們不會從中排除檔案，AMI因為檔案可能包含敏感資訊。

必要：否

`-p, --prefix` prefix

捆綁AMI文件的文件名前綴。

預設：`image`

必要：否

`-s, --size` size

欲建立的映像檔案大小，以 MB (1024 * 1024 位元) 表示。大小上限為 10240 MB。

預設：10240

必要：否

`--[no-]inherit`

指示映像是否繼承執行個體的中繼資料 (預設為繼承)。若啟用 `--inherit` 但無法存取執行個體中繼資料，則綁定會失敗。

必要：否

`-v, --volume` volume

欲從中建立 bundle 的掛載磁碟區絕對路徑。

預設：根目錄 (/)

必要：否

`-P, --partition` type

指示磁碟映像是否使用分割區表格。若未指定分割區表格類型，預設為用於磁碟區父區塊型儲存設備的類型 (若適用)，否則預設為 `gpt`。

有效值：`mbr` | `gpt` | `none`

必要：否

`-S, --script` script

欲在綁定前執行的自訂指令碼。此指令碼必須具備單一引數，亦即磁碟區的掛載點。

必要：否

`--fstab` path

欲綁定至映像的 fstab 路徑。如果沒有指定，Amazon EC2 捆綁包 /etc/fstab。

必要：否

`--generate-fstab`

使用 Amazon EC2 提供的 fstab 捆綁卷。

必要：否

`--grub-config`

欲綁定至映像的替代 grub 組態檔案路徑。根據預設，`ec2-bundle-vol` 預期所複製映像上應存在 `/boot/grub/menu.lst` 或 `/boot/grub/grub.conf`。此選項可讓您指定替代 grub 組態檔案路徑，並將其複製以覆蓋預設值 (如有)。

必要：否

`--kernel` kernel_id

已廢除. 使用 register-image 來設定核心。

必要：否

`--ramdiskramdisk_id`

已廢除。如果需要，使用寄存器映像來設置RAM磁盤。

必要：否

輸出

描述綁定階段與狀態的狀態訊息。

範例

此範例透AMI過壓縮、加密和簽署本機器根檔案系統的快照集來建立隨附的。

```
[ec2-user ~]$ ec2-bundle-vol -d /mnt -k pk-HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem -c
 cert-HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem -u 111122223333 -r x86_64
  Copying / into the image file /mnt/image...
  Excluding:
       sys
       dev/shm
       proc
       dev/pts
       proc/sys/fs/binfmt_misc
```

```
      dev
      media
      mnt
      proc
      sys
      tmp/image
      mnt/img-mnt
1+0 records in
1+0 records out
mke2fs 1.38 (30-Jun-2005)
warning: 256 blocks unused.

Splitting /mnt/image.gz.crypt...
Created image.part.00
Created image.part.01
Created image.part.02
Created image.part.03
...
Created image.part.22
Created image.part.23
Generating digests for each part...
Digests generated.
Creating bundle manifest...
Bundle Volume complete.
```

ec2-delete-bundle

描述

從 Amazon S3 儲存空間刪除指定 bundle。刪除套裝軟體之後，就無法從對應套件啟動執行個體AMI。

語法

ec2-delete-bundle -b *bucket* -a *access_key_id* -s *secret_access_key* [-t *token*] [--url *url*] [--region *region*] [--sigv *version*] [-m *path*] [-p *prefix*] [--clear] [--retry] [-y]

選項

`-b, --bucket` bucket

包含隨附的 Amazon S3 儲存貯體的名稱AMI，後面接著選用的 '/' 分隔路徑前綴

必要：是

`-a, --access-key` access_key_id

AWS 存取金鑰識別碼。

必要：是

`-s, --secret-key` secret_access_key

AWS 秘密存取金鑰。

必要：是

`-t, --delegation-token` token

要傳遞給 AWS 要求的委派權杖。如需詳細資訊，請參閱使用臨時安全登入資料。

必要：僅適用您使用暫時安全登入資料時。

預設：`AWS_DELEGATION_TOKEN` 環境變數的值 (如已設定)。

`--region`region

於請求簽章內使用的區域。

預設：`us-east-1`

必要：若使用簽章第 4 版則為必要

`--sigv`version

簽署請求時使用的簽章版本。

有效值：2 | 4

預設：4

必要：否

`-m, --manifest`path

指向資訊清單檔案的路徑。

必要：您必須指定 `--prefix` 或 `--manifest`。

`-p, --prefix` prefix

綁定的 AMI 檔案名稱字首。請提供完整字首。舉例來說，若字首為 image.img，則使用 `-p image.img` 而非 `-p image`。

必要：您必須指定 `--prefix` 或 `--manifest`。

`--clear`

若 Amazon S3 儲存貯體在刪除指定 bundle 後為空白，則加以刪除。

必要：否

`--retry`

在所有 Amazon S3 錯誤後自動重試，每次操作至多五次。

必要：否

`-y, --yes`

自動假設所有提示的答案為 Yes (是)。

必要：否

輸出

Amazon EC2 會顯示狀態訊息，指出刪除程序的階段和狀態。

範例

此範例會自 Amazon S3 刪除一個 bundle。

```
[ec2-user ~]$ ec2-delete-bundle -b amzn-s3-demo-bucket -a your_access_key_id -
s your_secret_access_key
Deleting files:
amzn-s3-demo-bucket/image.manifest.xml
amzn-s3-demo-bucket/image.part.00
amzn-s3-demo-bucket/image.part.01
amzn-s3-demo-bucket/image.part.02
amzn-s3-demo-bucket/image.part.03
amzn-s3-demo-bucket/image.part.04
amzn-s3-demo-bucket/image.part.05
amzn-s3-demo-bucket/image.part.06
Continue? [y/n]
y
Deleted amzn-s3-demo-bucket/image.manifest.xml
Deleted amzn-s3-demo-bucket/image.part.00
Deleted amzn-s3-demo-bucket/image.part.01
```

```
Deleted amzn-s3-demo-bucket/image.part.02
Deleted amzn-s3-demo-bucket/image.part.03
Deleted amzn-s3-demo-bucket/image.part.04
Deleted amzn-s3-demo-bucket/image.part.05
Deleted amzn-s3-demo-bucket/image.part.06
ec2-delete-bundle complete.
```

ec2-download-bundle

描述

AMIs從 Amazon S3 儲存下載指定的執行個體以商店支援的 Linux。

語法

ec2-download-bundle -b *bucket* -a *access_key_id* -s *secret_access_key* -k *path* [--url *url*] [--region *region*] [--sigv *version*] [-m *file*] [-p *prefix*] [-d *directory*] [--retry]

選項

`-b, --bucket` bucket

bundle 所在的 Amazon S3 儲存貯體名稱，其後為選用「/」以分隔路徑字首。

必要：是

`-a, --access-key` access_key_id

AWS 存取金鑰識別碼。

必要：是

`-s, --secret-key` secret_access_key

AWS 秘密存取金鑰。

必要：是

`-k, --privatekey` path

用於解密資訊清單的私有金鑰。

必要：是

`--url` url

Amazon S3 服務URL。

預設：`https://s3.amazonaws.com/`

必要：否

`--region` region

於請求簽章內使用的區域。

預設：`us-east-1`

必要：若使用簽章第 4 版則為必要

`--sigv` version

簽署請求時使用的簽章版本。

有效值：2 | 4

預設：4

必要：否

`-m, --manifest` file

不含路徑的資訊清單檔案名稱。建議您指定資訊清單 (`-m`) 或字首 (`-p`)。

必要：否

`-p, --prefix` prefix

捆綁AMI文件的文件名前綴。

預設：`image`

必要：否

`-d, --directory` directory

所下載 bundle 的儲存目錄。此目錄必須存在。

預設：目前使用中的目錄。

必要：否

`--retry`

在所有 Amazon S3 錯誤後自動重試，每次操作至多五次。

必要：否

輸出

會顯示指出下載程序各個階段的狀態訊息。

範例

此範例會建立 `bundled` 目錄 (使用 Linux mkdir 命令) 並從 `amzn-s3-demo-bucket` Amazon S3 儲存貯體下載 bundle。

```
[ec2-user ~]$ mkdir bundled
[ec2-user ~]$ ec2-download-bundle -b amzn-s3-demo-bucket/bundles/bundle_name
 -m image.manifest.xml -a your_access_key_id -s your_secret_access_key -k pk-
HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem -d mybundle
Downloading manifest image.manifest.xml from amzn-s3-demo-bucket to mybundle/
image.manifest.xml ...
Downloading part image.part.00 from amzn-s3-demo-bucket/bundles/bundle_name to
 mybundle/image.part.00 ...
Downloaded image.part.00 from amzn-s3-demo-bucket
Downloading part image.part.01 from amzn-s3-demo-bucket/bundles/bundle_name to
 mybundle/image.part.01 ...
Downloaded image.part.01 from amzn-s3-demo-bucket
Downloading part image.part.02 from amzn-s3-demo-bucket/bundles/bundle_name to
 mybundle/image.part.02 ...
Downloaded image.part.02 from amzn-s3-demo-bucket
Downloading part image.part.03 from amzn-s3-demo-bucket/bundles/bundle_name to
 mybundle/image.part.03 ...
Downloaded image.part.03 from amzn-s3-demo-bucket
Downloading part image.part.04 from amzn-s3-demo-bucket/bundles/bundle_name to
 mybundle/image.part.04 ...
Downloaded image.part.04 from amzn-s3-demo-bucket
Downloading part image.part.05 from amzn-s3-demo-bucket/bundles/bundle_name to
 mybundle/image.part.05 ...
Downloaded image.part.05 from amzn-s3-demo-bucket
Downloading part image.part.06 from amzn-s3-demo-bucket/bundles/bundle_name to
 mybundle/image.part.06 ...
Downloaded image.part.06 from amzn-s3-demo-bucket
```

ec2-migrate-manifest

描述

修改執行個體由商店支援的 Linux AMI (例如，其憑證、核心和RAM磁碟)，使其支援不同的區域。

語法

```
ec2-migrate-manifest -c path -k path -m path {(-a access_key_id -
s secret_access_key --region region) | (--no-mapping)} [--ec2cert
ec2_cert_path] [--kernel kernel-id] [--ramdisk ramdisk_id]
```

選項

`-c, --cert` path

使用者的PEM編碼RSA公開金鑰憑證檔案。

必要：是

`-k, --privatekey` path

使用者PEM編碼RSA金鑰檔案的路徑。

必要：是

`--manifest` path

指向資訊清單檔案的路徑。

必要：是

`-a, --access-key` access_key_id

AWS 存取金鑰識別碼。

必要：若使用自動映射則為必要。

`-s, --secret-key` secret_access_key

AWS 秘密存取金鑰。

必要：若使用自動映射則為必要。

`--region` region

欲在映射檔案內查詢的區域。

必要：若使用自動映射則為必要。

`--no-mapping`

停用核心和RAM磁碟的自動對應。

在移轉期間，Amazon EC2 會以專為目的地區域設計的核心和RAM磁碟取代資訊清單檔案中的核心和磁碟。RAM除非提供 `--no-mapping` 參數，否則 `ec2-migrate-bundle` 可能會使用 `DescribeRegions` 和 `DescribeImages` 操作來執行自動映射。

必要：若您未提供用於自動映射的 `-a`、`-s` 和 `--region` 選項則為必要。

`--ec2cert` path

用於加密映像資訊清單EC2之 Amazon X.509 公開金鑰憑證的路徑。

`us-gov-west-1` 和 `cn-north-1` 區域使用非預設的公有金鑰憑證，而該憑證的路徑必須以此選項加以指定。憑證的路徑會根據AMI工具的安裝方法而有所不同。以 Amazon Linux 而言，此憑證位於 `/opt/aws/amitools/ec2/etc/ec2/amitools/`。如果您從中的ZIP檔案安裝AMI工具設置 Amazon EC2 AMI 工具，則憑證位於`$EC2_AMITOOL_HOME/etc/ec2/amitools/`。

必要：僅適用於 `us-gov-west-1` 和 `cn-north-1` 區域。

`--kernel` kernel_id

欲選取的核心 ID。

> **Important**
>
> 我們建議您使用 PV-GRUB 而不是核心和RAM磁碟。如需詳細資訊，請參閱 Amazon Linux 2 使用者指南中的使用者提供的核心。

必要：否

`--ramdisk` ramdisk_id

要選取的RAM磁碟 ID。

> **Important**
>
> 我們建議您使用 PV-GRUB 而不是核心和RAM磁碟。如需詳細資訊，請參閱 Amazon Linux 2 使用者指南中的使用者提供的核心。

必要：否

輸出

描述綁定程序階段與狀態的狀態訊息。

範例

此範例會將`my-ami.manifest.xml`清單中AMI指定的內容從美國複製到歐盟。

```
[ec2-user ~]$ ec2-migrate-manifest --manifest my-ami.manifest.xml
 --cert cert-HKZYKTAIG2ECMXYIBH3HXV4ZBZQ55CLO.pem --privatekey pk-
HKZYKTAIG2ECMXYIBH3HXV4ZBZQ55CLO.pem --region eu-west-1

Backing up manifest...
Successfully migrated my-ami.manifest.xml It is now suitable for use in eu-west-1.
```

ec2-unbundle

描述

從執行個體存放區支援的 Linux 重新建立套裝軟體。AMI

語法

`ec2-unbundle -k path -m path [-s source_directory] [-d destination_directory]`

選項

`-k, --privatekey` path

PEM編碼RSA金鑰檔案的路徑。

必要：是

`-m, --manifest` path

指向資訊清單檔案的路徑。

必要：是

`-s, --source` source_directory

內含 bundle 的目錄。

預設：目前的目錄。

必要：否

`-d, --destination` destination_directory

要在其中取消捆綁的AMI目錄。此目的地目錄必須存在。

預設：目前的目錄。

必要：否

範例

這個 Linux 和UNIX示例解包`image.manifest.xml`文件中AMI指定的。

```
[ec2-user ~]$ mkdir unbundled
$ ec2-unbundle -m mybundle/image.manifest.xml -k pk-
HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem -s mybundle -d unbundled
$ ls -l unbundled
total 1025008
-rw-r--r-- 1 root root 1048578048 Aug 25 23:46 image.img
```

輸出

會顯示指出解開程序各個階段的狀態訊息。

ec2-upload-bundle

描述

將執行個體商店支援之 Linux 的服務包上傳AMI到 Amazon S3，並在上傳的物件上設定適當的存取控制清單 (ACLs)。如需詳細資訊，請參閱建立執行個體存放區支援 AMI。

> **Note**
>
> 若要將物件上傳至執行個體商店支援 Linux 的 S3 儲存貯體AMI，ACLs必須為儲存貯體啟用。否則，Amazon EC2 將無法在要上傳的對象ACLs上進行設置。如果您的目標儲存貯體使用儲存貯體擁有者強制執行 S3 物件擁有權的設定，這將無法運作，因ACLs為已停用。如需詳細資訊，請參閱使用 S3 物件擁有權控制上傳物件的擁有權。

語法

ec2-upload-bundle -b *bucket* -a *access_key_id* -s *secret_access_key* [-t *token*] -m *path* [--url *url*] [--region *region*] [--sigv *version*] [--acl *acl*] [-d *directory*] [--part *part*] [--retry] [--skipmanifest]

選項

`-b, --bucket` bucket

存放 bundle 的 Amazon S3 儲存貯體名稱，其後為選用「/」以分隔路徑字首。若該儲存貯體不存在，則會加以建立 (如該儲存貯體名稱為可用)。此外，如果值區不存在且AMI工具版本為 1.5.18 或更新版本，則此命令會ACLs為值區設定。

必要：是

`-a, --access-key` access_key_id

您的 AWS 存取金鑰 ID。

必要：是

`-s, --secret-key` secret_access_key

您的 AWS 秘密訪問密鑰。

必要：是

`-t, --delegation-token` token

要傳遞給 AWS 要求的委派權杖。如需詳細資訊，請參閱使用臨時安全登入資料。

必要：僅適用您使用暫時安全登入資料時。

預設：`AWS_DELEGATION_TOKEN` 環境變數的值 (如已設定)。

`-m, --manifest` path

指向資訊清單檔案的路徑。資訊清單檔案會於綁定程序期間建立，位於內含 bundle 的目錄。

必要：是

`--url` url

已廢除. 改用 `--region` 選項，除非您的儲存貯體被限制在 EU 位置 (而非 `eu-west-1`)。`--location` 旗標是確定該特定位置限制的唯一方法。

Amazon S3 端點服務URL。

預設：`https://s3.amazonaws.com/`

必要：否

`--region` region

要在請求簽章內針對目的地 S3 儲存貯體使用的區域。

- 若該儲存貯體不存在且您未指定區域，此工具會建立不具位置限制的儲存貯體 (位於`us-east-1`)。
- 若該儲存貯體不存在且您已指定區域，此工具會在該區域建立儲存貯體。
- 若該儲存貯體存在但您未指定區域，此工具會使用該儲存貯體的位置。
- 若該儲存貯體存在且您指定 `us-east-1` 為區域，此工具會使用該儲存貯體的實際位置，不會出現錯誤訊息，而且會覆寫所有現有的相符檔案。
- 若該儲存貯體存在且您指定的區域 (`us-east-1` 之外) 與儲存貯體的實際位置不相符，此工具會退出並顯示錯誤。

若您的儲存貯體被限制在 EU 位置 (而非 `eu-west-1`)，請改用 `--location` 旗標。`--location` 旗標是確定該特定位置限制的唯一方法。

預設：`us-east-1`

必要：若使用簽章第 4 版則為必要

`--sigv` version

簽署請求時使用的簽章版本。

有效值：2 | 4

預設：4

必要：否

`--acl` acl

綁定映像的存取控制清單政策。

有效值：`public-read` | `aws-exec-read`

預設：`aws-exec-read`

必要：否

`-d, --directory` directory

包含隨附AMI零件的目錄。

預設：內含資訊清單檔案的目錄 (請參閱 `-m` 選項)。

必要：否

`--part` part

開始上傳指定組件及所有後續組件。例如：`--part 04`。

必要：否

`--retry`

在所有 Amazon S3 錯誤後自動重試，每次操作至多五次。

必要：否

`--skipmanifest`

不會上傳資訊清單。

必要：否

`--location` location

已廢除. 改用 `--region` 選項，除非您的儲存貯體被限制在 EU 位置 (而非 `eu-west-1`)。`--location` 旗標是確定該特定位置限制的唯一方法。

目的地 Amazon S3儲存貯體的位置限制。若該儲存貯體存在且您指定的位置與儲存貯體的實際位置不相符，此工具會退出並顯示錯誤。若該儲存貯體存在但您未指定位置，此工具會使用該儲存貯體的位置。若該儲存貯體不存在且您已指定位置，此工具會在該位置建立儲存貯體。若該儲存貯體不存在且您未指定位置，此工具會建立不具位置限制的儲存貯體 (位於 `us-east-1`)。

預設：若已指定 `--region`，則位置會設定為該指定區域。若未指定 `--region`，則位置預設為 `us-east-1`。

必要：否

輸出

Amazon EC2 會顯示狀態訊息，指出上傳程序的階段和狀態。

範例

此範例會上傳 `image.manifest.xml` 資訊清單指定的 bundle。

```
[ec2-user ~]$ ec2-upload-bundle -b amzn-s3-demo-bucket/bundles/bundle_name -m
 image.manifest.xml -a your_access_key_id -s your_secret_access_key
Creating bucket...
Uploading bundled image parts to the S3 bucket amzn-s3-demo-bucket ...
Uploaded image.part.00
Uploaded image.part.01
Uploaded image.part.02
Uploaded image.part.03
Uploaded image.part.04
Uploaded image.part.05
Uploaded image.part.06
Uploaded image.part.07
Uploaded image.part.08
Uploaded image.part.09
Uploaded image.part.10
Uploaded image.part.11
Uploaded image.part.12
Uploaded image.part.13
Uploaded image.part.14
Uploading manifest ...
Uploaded manifest.
Bundle upload completed.
```

AMI工具的常用選項

大多數工AMI具都接受以下可選參數。

`--help, -h`

顯示協助訊息。

`--version`

顯示版本及著作權聲明。

`--manual`

顯示手動輸入。

`--batch`

以批次模式執行，不使用互動式提示。

`--debug`

顯示可用於故障排除問題的資訊。

將您的執行個體存放區轉換AMI為支援 EBS AMI

您可以將您擁有的執行個體商店支援AMI的 Linux 轉換為支援 Amazon EBS 的 Linux。AMI

> **Important**
>
> 你不能轉換一AMI個你不擁有的。

若要將執行個體存放區備份轉換為 AMI Amazon 支援 EBS AMI

1. 從 Amazon EBS 支AMI持的 Amazon Linux 實例啟動。如需詳細資訊，請參閱使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體。Amazon Linux 執行個體已預先安裝的 AWS CLI 和AMI工具。
2. 將您用來將執行個體存放區備份的 X.509 私密金鑰上傳AMI到執行個體。我們使用此密鑰來確保只有您和 Amazon 才EC2能訪問您的AMI.

   a. 在執行個體上為 X.509 私有金鑰建立如下的暫時目錄：

   ```
   [ec2-user ~]$ mkdir /tmp/cert
   ```

   b. 使用 scp 等安全複製工具從電腦上將 X.509 私有金鑰複製到執行個體上的 `/tmp/cert` 目錄內。所以此 *`my-private-key`* 下列命令中的參數是您用來連線至執行個體的私密金鑰 SSH。例如：

   ```
   you@your_computer:~ $ scp -i my-private-key.pem /
   path/to/pk-HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem ec2-
   user@ec2-203-0-113-25.compute-1.amazonaws.com:/tmp/cert/
   pk-HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem  100%  717     0.7KB/s   00:00
   ```

3. 設定環境變數以使用 AWS CLI。如需詳細資訊，請參閱環境變數。

   a. (建議使用) 設定 AWS 存取金鑰、私密金鑰和工作階段權杖的環境變數。

   ```
   [ec2-user ~]$ export AWS_ACCESS_KEY_ID=your_access_key_id
   [ec2-user ~]$ export AWS_SECRET_ACCESS_KEY=your_secret_access_key
   ```

```
[ec2-user ~]$ export AWS_SESSION_TOKEN=your_session_token
```

b. 為您的 AWS 訪問密鑰和密鑰設置環境變量。

```
[ec2-user ~]$ export AWS_ACCESS_KEY_ID=your_access_key_id
[ec2-user ~]$ export AWS_SECRET_ACCESS_KEY=your_secret_access_key
```

4. 準備一個 Amazon 彈性塊商店（AmazonEBS）卷為您的新AMI。

a. 使用 create- volume 命令，在與執行個體相同的可用區域中建立空白EBS磁碟區。記下命令輸出中的磁碟區 ID。

> **Important**
>
> 此EBS磁碟區的大小必須相同或大於原始執行個體儲存根磁碟區。

```
[ec2-user ~]$ aws ec2 create-volume --size 10 --region us-west-2 --
availability-zone us-west-2b
```

b. 使用附加磁碟區命令將磁碟區連接到 Amazon EBS 支援的執行個體。

```
[ec2-user ~]$ aws ec2 attach-volume --volume-id volume_id --instance-
id instance_id --device /dev/sdb --region us-west-2
```

5. 建立您的 bundle 資料夾。

```
[ec2-user ~]$ mkdir /tmp/bundle
```

6. `/tmp/bundle`使用指令為您的執行個體存放區AMI下載套裝軟體ec2-download-bundle。

```
[ec2-user ~]$ ec2-download-bundle -b amzn-s3-demo-bucket/bundle_folder/bundle_name
 -m image.manifest.xml -a $AWS_ACCESS_KEY_ID -s $AWS_SECRET_ACCESS_KEY --
privatekey /path/to/pk-HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem -d /tmp/bundle
```

7. 使用 ec2-unbundle 命令從 bundle 重新建構映像檔。

a. 將目錄變更為 bundle 資料夾。

```
[ec2-user ~]$ cd /tmp/bundle/
```

b. 執行 ec2-unbundle 命令。

```
[ec2-user bundle]$ ec2-unbundle -m image.manifest.xml --privatekey /path/to/pk-HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem
```

8. 將檔案從未捆綁的映像複製到新EBS磁碟區。

```
[ec2-user bundle]$ sudo dd if=/tmp/bundle/image of=/dev/sdb bs=1M
```

9. 探測磁碟區中是否有任何解開的新分割區。

```
[ec2-user bundle]$ sudo partprobe /dev/sdb1
```

10. 列出區塊型儲存設備，以找出要掛載的設備名稱。

```
[ec2-user bundle]$ lsblk
NAME         MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
/dev/sda    202:0    0   8G  0 disk
##/dev/sda1 202:1    0   8G  0 part /
/dev/sdb    202:80   0  10G  0 disk
##/dev/sdb1 202:81   0  10G  0 part
```

在此範例中，要掛載的分割區為 /dev/sdb1，但您的設備名稱可能不同。如果磁碟區尚未分割，要掛載的設備會類似 /dev/sdb (沒有設備分割區尾碼)。

11. 建立新EBS磁碟區的掛接點並掛接磁碟區。

```
[ec2-user bundle]$ sudo mkdir /mnt/ebs
[ec2-user bundle]$ sudo mount /dev/sdb1 /mnt/ebs
```

12. 使用您最喜歡的文本編輯器（例如vim或nano）打開EBS卷上的文/etc/fstab件，並刪除例如存儲（臨時）卷的任何條目。由於該加EBS密區已掛載/mnt/ebs，因此fstab檔案位於/mnt/ebs/etc/fstab。

```
[ec2-user bundle]$ sudo nano /mnt/ebs/etc/fstab
#
LABEL=/     /           ext4    defaults,noatime  1   1
tmpfs       /dev/shm    tmpfs   defaults        0   0
devpts      /dev/pts    devpts  gid=5,mode=620  0   0
sysfs       /sys        sysfs   defaults        0   0
proc        /proc       proc    defaults        0   0
```

```
/dev/sdb        /media/ephemeral0       auto    defaults,comment=cloudconfig    0
     2
```

在此範例中，應該移除最後一行。

13. 卸載磁碟區，並從執行個體分離。

```
[ec2-user bundle]$ sudo umount /mnt/ebs
[ec2-user bundle]$ aws ec2 detach-volume --volume-id volume_id --region us-west-2
```

14. AMI從新EBS磁碟區建立，如下所示。

    a. 建立新EBS磁碟區的快照。

    ```
    [ec2-user bundle]$ aws ec2 create-snapshot --region us-west-2 --description
     "your_snapshot_description" --volume-id volume_id
    ```

    b. 檢查快照是否完整。

    ```
    [ec2-user bundle]$ aws ec2 describe-snapshots --region us-west-2 --snapshot-
    id snapshot_id
    ```

    c. 使用describe-images指令識別原始AMI檔案上使用的處理器架構、虛擬化類型和核心映像 (`aki`)。在此步驟中，您需要原始執行個體存放區支援AMI的 AMI ID。

    ```
    [ec2-user bundle]$ aws ec2 describe-images --region us-west-2 --image-id ami-id
     --output text
    IMAGES x86_64 amazon/amzn-ami-pv-2013.09.2.x86_64-s3 ami-8ef297be amazon
     available public machine aki-fc8f11cc instance-store paravirtual xen
    ```

    在此範例中，其架構為 `x86_64`，核心映像 ID 為 `aki-fc8f11cc`。請將這些數值使用在下列步驟。如果上述命令的輸出也列出 `ari` ID，請將其記下。

    d. AMI使用新磁碟區的快照 ID 和上一步的值註冊新EBS磁碟區。如果先前的命令輸出列出 `ari` ID，請用 `--ramdisk-id ari_id` 將其加入到下列命令。

    ```
    [ec2-user bundle]$ aws ec2 register-image --region us-west-2 --
    name your_new_ami_name --block-device-mappings DeviceName=device-
    name,Ebs={SnapshotId=snapshot_id} --virtualization-type paravirtual --
    architecture x86_64 --kernel-id aki-fc8f11cc --root-device-name device-name
    ```

15. (選擇性) 測試完可以從新執行個體啟動執行個體之後AMI，您可以刪除為此程序建立的EBS磁碟區。

```
aws ec2 delete-volume --volume-id volume_id
```

EC2AMI使用視窗系統創建一個 Amazon

Microsoft 系統準備工具 (Windows Sysprep) 工具會建立作業系統的一般化版本，並在擷取新映像之前移除執行個體特定的系統組態。

我們建議您使用 EC2Image Builder 來自動建立、管理和部署自訂、安全和 up-to-date「黃金」伺服器映像，這些影像已預先安裝並預先設定軟體和設定。

您也可以使用 Windows 系統表示建立AMI使用 Windows 啟動代理程式的標準化。如需詳細資訊，請參閱the section called “搭配啟動代理程式使用視窗系統”。

> **Important**
>
> 請勿使用視窗系統建立執行個體備份。Windows Sysprep 會移除系統特定資訊；移除此資訊可能會對執行個體備份造成意外後果。

若要疑難排解視窗系統，請參閱。疑難排解 Amazon EC2 視窗執行個體的系統問題

目錄

- 視窗系統階段
- 開始之前
- 搭配啟動代理程式使用視窗系統

視窗系統階段

視窗系統會執行下列階段：

- 一般化：Sysprep 工具會移除映像特定的資訊和組態。例如，Windows Sysprep 會移除安全性識別碼 (SID)、電腦名稱、事件記錄檔和特定驅動程式，僅舉幾例。完成此階段之後，作業系統 (OS) 就可以建立AMI.

> **Note**
>
> 當您使用 Windows 啟動代理程式執行 Windows Sysprep 時，系統會防止驅動程式遭到移除，因`PersistAllDeviceInstalls`為預設會設定為 true。

- 特製化：隨插即用會掃描電腦，並為任何偵測到的裝置安裝驅動程式。Sysprep 工具會產生作業系統需求，例如電腦名稱和. SID 或者，您可以在此階段執行命令。
- 立即可用的體驗 (OOBE)：系統會執行 Windows 安裝程式的縮寫版本，並要求您輸入系統語言、時區和註冊組織等資訊。當您使用 Windows 啟動代理程式執行 Windows Sysprep 時，回應檔案會自動執行這個階段。

開始之前

- 在執行 Windows Sysprep 之前，我們建議您移除所有本機使用者帳戶和所有帳戶設定檔，而不是執行 Windows Sysprep 的單一系統管理員帳戶。如果您使用其他帳戶和設定檔執行 Windows Sysprep，可能會導致非預期的行為，包括遺失設定檔資料或無法完成 Windows Sysprep。
- 進一步瞭解系統表示概觀。
- 瞭解伺服器角色的哪些 Sysprep Support。

搭配啟動代理程式使用視窗系統

您可以使用 Windows Sysprep 建立標準化的 Amazon 電腦映像 (AMI)，當您開始時，已安裝AMI了其中一個 Windows 啟動代理程式。

搭配 v2 使用視窗系統 EC2Launch

本節包含 EC2Launch v2 服務在準備映像時所執行之工作的詳細資訊。它也包括建立AMI使用 Windows 系統與 EC2Launch v2 服務的標準化的步驟。

具EC2Launch有 v2 主題的視窗系統播放

- 視窗系統表示動作
- Sysprep 後續階段
- 使用 v2 運行視窗系統 EC2Launch

視窗系統表示動作

準備映像時，視窗系統和 EC2Launch v2 會執行下列動作。

1. 當您在EC2Launch設定對話方塊中選擇使用 Sysprep 關機時，系統會執行指`ec2launch sysprep`令。
2. EC2Launchv2 通過讀取註冊表值來編輯`unattend.xml`文件的內容`HKEY_USERS\.DEFAULT\Control Panel\International\LocaleName`。檔案位於以下目錄：`C:\ProgramData\Amazon\EC2Launch\sysprep`。
3. 系統會執行 `BeforeSysprep.cmd`。此命令會建立一個登錄機碼，如下所示：

   reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server" /v fDenyTSConnections /t REG_DWORD /d 1 /f

   登錄機碼會停用RDP連線，直到它們重新啟用為止。停用RDP連線是一項必要的安全性措施，因為在 Windows Sysprep 執行之後的第一個開機工作階段期間，會有短時間RDP允許連線，而系統管理員密碼為空白。
4. EC2Launchv2 服務會執行下列命令來呼叫 Windows 系統：

   sysprep.exe /oobe /generalize /shutdown /unattend: "C:\ProgramData\Amazon\EC2Launch\sysprep\unattend.xml"

一般化階段

- EC2Launchv2 會移除映像特有的資訊和組態，例如電腦名稱和. SID 如果執行個體是網域成員，即會從網域中移除。`unattend.xml` 回應檔案包括下列可影響此階段的設定：
  - PersistAllDeviceInstalls：此設定可防止 Windows 安裝程式移除和重新設定裝置，進而加快映像準備程序的速度，因為 Amazon AMIs 要求特定驅動程式才能執行，並且重新偵測這些驅動程式需要一些時間。
  - DoNotCleanUpNonPresentDevices：此設定會保留目前不存在的裝置的隨插即用資訊。
- 視窗系統會關閉作業系統，因為它準備建立. AMI 系統會啟動新的執行個體，或啟動原始執行個體。

特製化階段

系統會產生作業系統特定的需求，例如電腦名稱和. SID 系統也會根據您在 `unattend.xml` 回應檔案中指定的組態，來執行下列動作。

- CopyProfile：Windows Sysprep 可以設定為刪除所有使用者設定檔，包括內建的系統管理員設定檔。此設定會保留內建管理員帳戶，因此您對該帳戶進行的任何自訂都會帶到新的映像。預設值為 `True`。

  CopyProfile以現有的本機管理員設定檔取代預設設定檔。您在執行 Windows Sysprep 之後登入的所有帳戶都會在第一次登入時收到該設定檔及其內容的複本。

  如果您沒有特定的使用者描述檔自訂要帶到新映像，請將此設定變更為 `False`。Windows Sysprep 將刪除所有用戶配置文件（這樣可以節省時間和磁盤空間）。
- TimeZone：依預設，時區設定為「座標世界時間」(UTC)。
- 順序為 1 的同步命令：系統會執行下列命令，以啟用管理員帳戶並指定密碼要求。

```
net user Administrator /ACTIVE:YES /LOGONPASSWORDCHG:NO /EXPIRES:NEVER /
PASSWORDREQ:YES
```

- 順序為 2 的同步命令：系統會將管理員密碼變碼。此安全性措施的設計目的是防止在 Windows Sysprep 完成後，如果您未設定工作，則無法存取執行個體`setAdminAccount`。

  系統會從您的本機啟動代理程式目錄 (`C:\Program Files\Amazon\EC2Launch\`) 執行下列命令。

```
EC2Launch.exe internal randomize-password --username Administrator
```

- 若要啟用遠端桌面連線，系統會將終端機伺服器`fDenyTSConnections`登錄機碼設為 false。

OOBE階段

1. 系統會使用 EC2Launch v2 回應檔案指定下列組態：
   - `<InputLocale>en-US</InputLocale>`
   - `<SystemLocale>en-US</SystemLocale>`
   - `<UILanguage>en-US</UILanguage>`
   - `<UserLocale>en-US</UserLocale>`
   - `<HideEULAPage>true</HideEULAPage>`
   - `<HideWirelessSetupInOOBE>true</HideWirelessSetupInOOBE>`
   - `<ProtectYourPC>3</ProtectYourPC>`
   - `<BluetoothTaskbarIconEnabled>false</BluetoothTaskbarIconEnabled>`

- `<TimeZone>UTC</TimeZone>`
- `<RegisteredOrganization>Amazon.com</RegisteredOrganization>`
- `<RegisteredOwner>EC2</RegisteredOwner>`

> **Note**
>
> 在概括和專門化階段，EC2Launchv2 監視操作系統的狀態。如果 EC2Launch v2 偵測到作業系統處於 Sysprep 階段，則會將下列訊息發佈至系統記錄檔：
> Windows is being configured. SysprepState= IMAGE _ STATE _ UNDEPLOYABLE

2. 該系統運行 EC2Launch v2。

Sysprep 後續階段

視窗系統表示完成之後，EC2Launchv2 會將下列訊息傳送至主控台輸出：

```
Windows sysprep configuration complete.
```

EC2Launchv2 接著會執行下列動作：

1. 讀取 `agent-config.yml` 檔案的內容並執行已設定的任務。
2. 執行 `preReady` 階段中的所有任務。
3. 完成後，會將 `Windows is ready` 訊息傳送至執行個體系統日誌。
4. 執行 `PostReady` 階段中的所有任務。

如需 EC2Launch v2 的詳細資訊，請參閱使用 EC2Launch v2 代理程式在 EC2 Windows 執行個體啟動期間執行工作。

使用 v2 運行視窗系統 EC2Launch

使用下列程序來建立AMI使用具EC2Launch有 v2 的視窗系統標準化。

1. 在 Amazon EC2 控制台中，找AMI到要複製的。
2. 啟動和連線至 Windows 執行個體。
3. 自訂執行個體。
4. 從 Windows 開始菜單中，搜索並選擇 Amazon EC2Launch 設置。如需 Amazon 設定對話方塊中選項和設EC2Launch定的詳細資訊，請參閱設定視窗執行個體的 EC2Launch V2 設定。

5. 選取 Shutdown with Sysprep (使用 Sysprep 關機) 或 Shutdown without Sysprep (不使用 Sysprep 關機)。

當系統要求您確認是否要執行 Windows Sysprep 並關閉執行個體時，請按一下 [是]。EC2LaunchV2 運行視窗系統。接著，系統會將您登出執行個體，然後執行個體會關機。如果您在 Amazon EC2 主控台中查看「執行個體」頁面，執行個體狀態會從變更`Running``Stopping`為`Stopped`。在這一點上，AMI從這個實例創建一個是安全的。

您可以使用下列命令，從命令列手動呼叫 Windows 系統表示工具：

```
"%programfiles%\amazon\ec2launch\ec2launch.exe" sysprep --shutdown=true
```

使用視窗系統 EC2Launch

EC2Launch提供 Windows Sysprep 的預設回應檔案和批次檔案，可自動化並保護您的. AMI 您可選擇性修改這些檔案。這些檔案預設位於下列目錄：`C:\ProgramData\Amazon\EC2-Windows\Launch\Sysprep`。

> **Important**
>
> 請勿使用視窗系統建立執行個體備份。視窗系統會移除系統特定的資訊。如果您移除此資訊，執行個體備份可能產生意外結果。

視窗系統與主題 EC2Launch

- EC2Launch視窗系統的答案和批處理文件
- 運行視窗系統 EC2Launch
- 在啟動自訂時，更新伺服器 2016 及更新版本的中繼資料/KMS路由 AMI

EC2Launch視窗系統的答案和批處理文件

適用於 Windows 系統的EC2Launch回應檔案和批次檔案包括下列項目：

`Unattend.xml`

這是預設回應檔案。如果您在使用者介面ShutdownWithSysprep中執行`SysprepInstance.ps1`或選擇，系統會從此檔案讀取設定。

`BeforeSysprep.cmd`

自定義此批處理文件以在運行 Windows 系統之前EC2Launch運行命令。

`SysprepSpecialize.cmd`

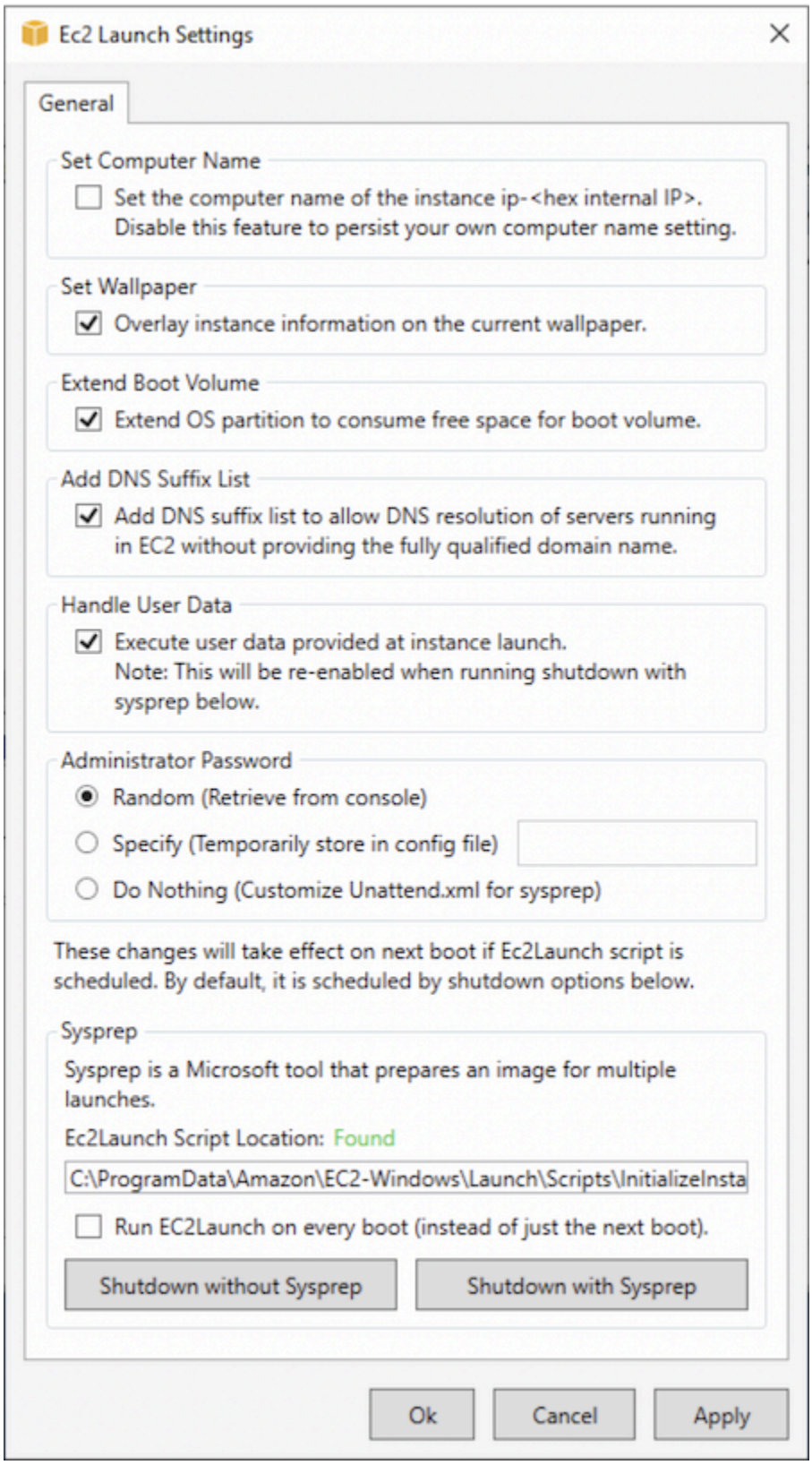
自訂此批次檔案，以便在 Windows Sysprep 專門化階段執行命令。

運行視窗系統 EC2Launch

在 Windows 伺服器 2016 年及更新版本的完整安裝 (具有桌面體驗) 上，您可以使用手動方式或使用EC2啟EC2Launch動設定應用程式來執行 Windows 系統表示。

使用「EC2Launch設定」應用程式執行視窗系統

1. 在 Amazon EC2 控制台中，找到或創建一個視窗服務器 2016 或更高版本AMI。
2. 從啟動視窗執行個體AMI。
3. 連線到您的 Windows 執行個體並加以自訂。
4. 搜索並運行EC2LaunchSettings應用程序。根據預設，其位於下列目錄：`C:\ProgramData\Amazon\EC2-Windows\Launch\Settings`。



5. 視需要選取或清除選項。這些設定存放在 `LaunchConfig.json` 檔案中。
6. 針對 Administrator Password (管理員密碼)，請執行下列其中一項作業：

- 選擇 Random (隨機)。EC2Launch生成密碼並使用用戶的密鑰對其進行加密。系統會在執行個體啟動後停用此設定，以便執行個體重新開機或停止再啟動時，能持續使用這個密碼。
- 選擇 Specify (指定) 並輸入符合系統要求的密碼。密碼會`LaunchConfig.json`以純文字格式儲存，並在 Windows Sysprep 設定系統管理員密碼之後刪除。如果您立即關機，即會立即設定密碼。EC2Launch使用用戶的密鑰加密密碼。
- 在檔案中選擇DoNothing並指定密`unattend.xml`碼。如果您未在 `unattend.xml` 中指定密碼，會停用管理員帳戶。

7. 選擇 Shutdown with Sysprep (使用 Sysprep 關機)。

若要使用手動執行視窗系統 EC2Launch

1. 在 Amazon EC2 控制台中找到或創建您要複製的 Windows 服務器 2016 或更高版本AMI的數據中心版本。
2. 啟動和連線至 Windows 執行個體。
3. 自訂執行個體。
4. 在 `LaunchConfig.json` 檔案中指定設定。根據預設，此檔案位於 `C:\ProgramData\Amazon\EC2-Windows\Launch\Config` 目錄。

   針對 `adminPasswordType`，請指定下列其中一個值：

   `Random`

   EC2Launch生成密碼並使用用戶的密鑰對其進行加密。系統會在執行個體啟動後停用此設定，以便執行個體重新開機或停止再啟動時，能持續使用這個密碼。

   `Specify`

   EC2Launch使用您在中指定的密碼`adminPassword`。如果密碼不符合系統需求，請改為EC2Lauch產生隨機密碼。密碼會`LaunchConfig.json`以純文字格式儲存，並在 Windows Sysprep 設定系統管理員密碼之後刪除。EC2Launch使用用戶的密鑰加密密碼。

   `DoNothing`

   EC2Launch使用您在`unattend.xml`檔案中指定的密碼。如果您未在 `unattend.xml` 中指定密碼，會停用管理員帳戶。

5. (選用) 請在 `unattend.xml` 和其他組態檔案中指定設定。如果您計畫安裝，就不需要變更這些檔案。根據預設，檔案位於下列目錄：`C:\ProgramData\Amazon\EC2-Windows\Launch\Sysprep`。

6. 在視窗中 PowerShell，執行`./InitializeInstance.ps1 -Schedule`。根據預設，指令碼位於下列目錄：`C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts`。此指令碼會排程執行個體在下一次開機期間初始化。您必須在執行下一步的 `SysprepInstance.ps1` 指令碼前，執行此指令碼。
7. 在視窗中 PowerShell，執行`./SysprepInstance.ps1`。根據預設，指令碼位於下列目錄：`C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts`。

系統會將您登出執行個體，然後執行個體會關機。如果您在 Amazon EC2 主控台中查看「執行個體」頁面，則執行個體狀態會從變`Running`為`Stopping`，然後變為`Stopped`。在這一點上，AMI從這個實例創建一個是安全的。

在啟動自訂時，更新伺服器 2016 及更新版本的中繼資料/KMS路由 AMI

若要在啟動自訂時更新 Server 2016 及更新版本的中繼資料/KMS路由AMI，請執行下列其中一項動作：

- 運行 EC2LaunchSettingsGUI（C：\\ AmazonProgramData\ EC2-視窗\ 啟動\ 設置\ EC2 LaunchSettings .exe），然後選擇關閉與視窗系統的選項。
- 在建立之前，在沒有視窗系統的情況下執行EC2LaunchSettings並關閉。AMI這會將EC2啟動初始化工作設定為在下次開機時執行，這會根據執行個體的子網路設定路由。
- 在建立AMI來PowerShell源之前，手動重新排程 EC2 Launch 初始化工作。

> **Important**
>
> 在重新排定任務之前，請記下預設密碼重設行為。

- 若要在發生 Windows 啟用失敗或與執行個體中繼資料通訊失敗的執行中的執行個體上更新路由，請參閱「無法啟用 Windows」。

使用視窗系統 EC2Config

本節包含EC2Config服務在準備映像時所執行之工作的詳細資訊。它還包括創建AMI使用 Windows Sysprep 與服務的標準化的EC2Config步驟。

視窗系統與主題 EC2Config

- 視窗系統表示動作
- Sysprep 後續階段

- 使用該服務運行視窗系統 EC2Config

視窗系統表示動作

準備映像時，視窗系統和EC2Config服務會執行下列動作。

1. 當您在 [EC2服務內容] 對話方塊中選擇使用 Sysprep 關機時，系統會執行 ec2config.exe- sysprep 命令。
2. 服EC2Config務會讀取`BundleConfig.xml`檔案的內容。此檔案預設位在下列目錄：`C:\Program Files\Amazon\Ec2ConfigService\Settings`。

   `BundleConfig.xml` 檔案包括下列設定。您可以變更這些設定：

   - AutoSysprep：指出是否要自動使用視窗系統。如果您是從 [EC2服務屬性] 對話方塊執行 Windows Sysprep，則不需要變更此值。預設值為 No。
   - S etRDPCertificate：設定遠端桌面伺服器的自我簽署憑證。這可讓您安全地使用遠端桌面通訊協定 (RDP) 連線至執行個體。如果新的執行個體應該使用憑證，請將值變更為 Yes。這個設定不適用於 Windows 伺服器 2012 執行個體，因為這些作業系統可以產生自己的憑證。預設值為 No。
   - SetPasswordAfterSysprep：在新啟動的執行個體上設定隨機密碼，使用使用者啟動金鑰加密密碼，然後將加密的密碼輸出至主控台。如果新執行個體不應該設定為隨機加密密碼，請將值變更為 No。預設值為 Yes。
   - PreSysprepRunCmd：要執行的命令的位置。根據預設，命令位於下列目錄：`C:\Program Files\Amazon\Ec2ConfigService\Scripts\BeforeSysprep.cmd`。
3. 系統會執行 `BeforeSysprep.cmd`。此命令會建立一個登錄機碼，如下所示：

   ```
   reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server" /v
    fDenyTSConnections /t REG_DWORD /d 1 /f
   ```

   登錄機碼會停用RDP連線，直到它們重新啟用為止。停用RDP連線是一項必要的安全性措施，因為在 Windows Sysprep 執行之後的第一個開機工作階段期間，會有短時間RDP允許連線，而系統管理員密碼為空白。
4. 此EC2Config服務會執行下列命令來呼叫 Windows 系統表示：

   ```
   sysprep.exe /unattend: "C:\Program Files\Amazon\Ec2ConfigService\sysprep2008.xml" /
   oobe /generalize /shutdown
   ```

一般化階段

- 此工具會移除影像特有的資訊和組態，例如電腦名稱和. SID 如果執行個體是網域成員，即會從網域中移除。`sysprep2008.xml` 回應檔案包括下列可影響此階段的設定：
  - PersistAllDeviceInstalls：此設定可防止 Windows 安裝程式移除和重新設定裝置，進而加快映像準備程序的速度，因為 Amazon AMIs 要求特定驅動程式才能執行，並且重新偵測這些驅動程式需要一些時間。
  - DoNotCleanUpNonPresentDevices：此設定會保留目前不存在的裝置的隨插即用資訊。
- 視窗系統會關閉作業系統，因為它準備建立. AMI 系統會啟動新的執行個體，或啟動原始執行個體。

特製化階段

系統會產生作業系統特定需求，例如電腦名稱和SID. 系統也會根據您在 sysprep2008.xml 回應檔案中指定的組態，來執行下列動作。

- CopyProfile：Windows Sysprep 可以設定為刪除所有使用者設定檔，包括內建的系統管理員設定檔。此設定會保留內建管理員帳戶，因此您對該帳戶進行的任何自訂都會帶到新的映像。預設值為 True。

  CopyProfile以現有的本機管理員設定檔取代預設設定檔。執行 Windows Sysprep 之後登入的所有帳戶都會在第一次登入時收到該設定檔及其內容的複本。

  如果您沒有特定的使用者描述檔自訂要帶到新映像，請將此設定變更為 False。Windows Sysprep 會移除所有使用者設定檔；這樣可以節省時間和磁碟空間。
- TimeZone：依預設，時區設定為「座標世界時間」(UTC)。
- 順序為 1 的同步命令：系統會執行下列命令，以啟用管理員帳戶並指定密碼要求。

  net user Administrator /ACTIVE:YES /LOGONPASSWORDCHG:NO /EXPIRES:NEVER /PASSWORDREQ:YES
- 順序為 2 的同步命令：系統會將管理員密碼變碼。如果您未啟用 ec2setpassword 設定，此安全性措施是為了防止在 Windows Sysprep 完成之後存取執行個體。

  C:\Program 文件\ Amazon\ EC2ConfigService\ ScramblePassword .exe」-u 管理員
- 順序為 3 的同步命令：系統會執行下列命令：

  C:\Program 文件\ Amazon\ EC2ConfigService\ 腳本\ SysprepSpecializePhase .cmd

  此命令會新增下列登錄機碼，以重新啟用RDP：

註冊添加「HKEY_ LOCAL _MACHINE\\SYSTEM\ 控制CurrentControlSet\ 終端服務器」/V/ fDenyTSConnectionsT REG _ /d 0 DWORD /F

OOBE階段

1. 使用EC2Config服務回應檔案，系統會指定下列組態：
   - < InputLocale > 一般美國 InputLocale </ >
   - < SystemLocale > 一般美國 SystemLocale </ >
   - < UILanguage > 一般美國 UILanguage </ >
   - < UserLocale > 一般美國 UserLocale </ >
   - <H ideEULAPage > 真</H > ideEULAPage
   - < HideWirelessSetupIn OOBE > 真實 </ > HideWirelessSetupIn OOBE
   - < NetworkLocation > 其他 </ > NetworkLocation
   - < ProtectYour 個人電腦 >3</ 個人電腦 > ProtectYour
   - BluetoothTaskbarIconEnabled假假 </ > BluetoothTaskbarIconEnabled
   - <TimeZone>UTC</TimeZone>
   - < RegisteredOrganization > 亞馬遜網站 </ > RegisteredOrganization
   - < RegisteredOwner > Amazon </ > RegisteredOwner

   > **Note**
   >
   > 在一般化和專門化階段期間，EC2Config服務會監視作業系統的狀態。如果EC2Config偵測到作業系統處於 Sysprep 階段，則會將下列訊息發佈至系統記錄檔：EC2ConfigMonitorState: 0 視窗正在設定中。SysprepState= IMAGE _ STATE _ UNDEPLOYABLE

2. OOBE階段完成後，系統會`SetupComplete.cmd`從下列位置執`C:\Windows\Setup\Scripts\SetupComplete.cmd`行：在 2015 年 4 月AMIs之前的 Amazon 公眾，這個文件是空的，圖像上什麼也沒有運行。在 2015 年 4 AMIs 月之後的公開版本中，檔案包含下列值：call "C:\Program Files\Amazon\Ec2ConfigService\Scripts\PostSysprep.cmd"。
3. 系統會執行 `PostSysprep.cmd`，執行下列操作：
   - 將本機管理員密碼設定為不過期。如果密碼過期，管理員可能會無法登入。
   - 設定MSSQLServer電腦名稱 (如果已安裝)，以便名稱與AMI.

Sysprep 後續階段

Windows Sysprep 完成之後，EC2Config服務會將下列訊息傳送至主控台輸出：

```
Windows sysprep configuration complete.
   Message: Sysprep Start
   Message: Sysprep End
```

EC2Config 接著會執行下列動作：

1. 讀取 config.xml 檔案的內容，並列出所有已啟用的外掛程式。
2. 同時執行所有「Windows 就緒之前」(“Before Windows is ready”) 外掛程式。
   - EC2 SetPassword
   - EC2 SetComputerName
   - EC2 InitializeDrives
   - EC2 EventLog
   - EC2 配置 RDP
   - Ec2O utputRDPCert
   - EC2 SetDriveLetter
   - EC2 WindowsActivate
   - EC2 DynamicBootVolumeSize
3. 完成之後，會將「Windows 就緒」(“Windows is ready”) 訊息傳送至執行個體系統日誌。
4. 同時執行所有「Windows 就緒之後」(“After Windows is ready”) 外掛程式。
   - Amazon CloudWatch 日誌
   - UserData
   - AWS Systems Manager (Systems Manager)

如需 Windows 外掛程式的詳細資訊，請參閱在EC2舊版 Windows 作業系統執行個體啟動期間，使用此EC2Config服務執行工作。

使用該服務運行視窗系統 EC2Config

使用下列程序來建立AMI使用 Windows Sysprep 和服務的標準化。EC2Config

1. 在 Amazon 主EC2控台中，找到或建立您AMI要複製的檔案。
2. 啟動和連線至 Windows 執行個體。

3. 自訂執行個體。
4. 在EC2Config服務回應檔案中指定組態設定：

   `C:\Program Files\Amazon\Ec2ConfigService\sysprep2008.xml`
5. 從 Windows「開始」功能表中，選擇「所有程式」，然後選擇「EC2ConfigService設定」。
6. 選擇 Ec2 Service Properties (Ec2 服務屬性) 對話方塊中的 Image (映像) 標籤。如需 Ec2 Service Properties (Ec2 服務屬性) 對話方塊中之選項和設定的詳細資訊，請參閱 Ec2 服務屬性。
7. 選取管理員密碼的選項，然後選取 Shutdown with Sysprep (使用 Sysprep 關機) 或 Shutdown without Sysprep (不使用 Sysprep 關機)。EC2Config根據您選取的密碼選項編輯設定檔案。
   - 隨機：EC2Config生成密碼，使用用戶的密鑰對其進行加密，並向控制台顯示加密的密碼。我們會在第一次啟動後停用此設定，以便執行個體重新開機或是停止後再啟動時，能持續使用這個密碼。
   - 指定：密碼會以未加密的格式 (純文字) 儲存在 Windows Sysprep 回應檔案中。當 Windows 系統表示下次執行時，它會設定系統管理員密碼。如果您立即關機，即會立即設定密碼。在服務重新啟動時，會移除管理員密碼。請務必記住此密碼，之後即無法再擷取此密碼。
   - 保持現有：當 Windows Sysprep 執行或EC2Config重新啟動時，系統管理員帳戶的現有密碼不會變更。請務必記住此密碼，之後即無法再擷取此密碼。
8. 選擇 OK (確定)。

當系統要求您確認是否要執行 Windows Sysprep 並關閉執行個體時，請按一下 [是]。你會注意到，EC2Config運行視窗系統。接著，系統會將您登出執行個體，然後執行個體會關機。如果您在 Amazon EC2 主控台中查看 [執行個體] 頁面，執行個體狀態會從變更`Running`為`Stopping`，最後變更為`Stopped`。在這一點上，AMI從這個實例創建一個是安全的。

您可以使用下列命令，從命令列手動呼叫 Windows 系統表示工具：

```
"%programfiles%\amazon\ec2configservice\"ec2config.exe -sysprep""
```

> **Note**
>
> 如果您的命令CMD介面已經在 C:\Program 檔案\ AmazonEC2ConfigService\ 目錄中，則不需要使用命令中的雙引號。

但是，您必須非常小心，`Ec2ConfigService\Settings`資料夾中指定的XML檔案選項正確無誤；否則，您可能無法連線至執行個體。如需設定檔案的詳細資訊，請參閱 EC2Config設定檔案。

如需從命令列設定並執行 Windows Sysprep 的範例，請參閱。`Ec2ConfigService\Scripts\InstallUpdates.ps1`

複製 Amazon EC2 AMI

您可以在相同區域內或相同分割區中的區域建立 Amazon 機器映像 (AMI) 副本。若要複製AMI到另一個分割區，請參閱存儲和恢復 AMI。

目錄

- 考量事項
- 成本
- 授予複製 Amazon 的許可 EC2 AMIs
- 複製 AMI
- 停用待定的 AMI 複製作業
- Amazon EC2 AMI 副本如何工作

考量事項

- 複製權限 AMIs — 您可以使用IAM策略授與或拒絕使用者複製的權限AMIs。為動作指定的資源層級權限僅適`CopyImage`用於新的權限。AMI您無法指定來源的資源層級權限。AMI
- 啟動許可和 Amazon S3 儲存貯體許可 — AWS 不會將啟動許可或 Amazon S3 儲存貯體許可從來源複製AMI到新儲存貯體許可AMI。複製操作完成後，您可以將啟動許可和 Amazon S3 儲存貯體許可套用至新的儲存貯體許可AMI。
- 標籤 — 您只能複製附加至來源的使用者定義AMI標籤AMI。將不會複製連接至其他 AWS 帳戶 的系統標籤 (字首為 `aws:`) 和使用者定義的標籤。複製時AMI，您可以將新標記附加至目標AMI及其後備快照。

成本

複製的AMI. 然而，將收取標準儲存和資料傳輸費用。如果您複製EBS備份AMI，則會產生任何額外EBS快照的儲存費用。

授予複製 Amazon 的許可 EC2 AMIs

若要複製EBS支援的執行個體或執行個體存放區支援AMI，您需要下列權限：IAM

- `ec2:CopyImage`— 若要複製AMI. 對於 EBS-backedAMIs，它還授予複製支持快照AMI的權限。

- `ec2:CreateTags`— 標記目標AMI。對於 EBS-backedAMIs，它還授予標記目標後備快照AMI的權限。

如果您要複製儲存支援的執行個體AMI，則需要下列額外IAM權限：

- `s3:CreateBucket`— 若要在目標區域中建立新的 S3 儲存貯體 AMI
- `s3:GetBucketAcl`— 讀取來源值區的ACL權限
- `s3:ListAllMyBuckets`— 在目標區域中尋找現有AMIs的 S3 儲存貯體
- `s3:GetObject`— 讀取來源值區中的物件
- `s3:PutObject`— 在目標值區中寫入物件
- `s3:PutObjectAcl`— 寫入目標值區中新物件的權限

複製EBS支援AMI並標記目標AMI和快照的範例IAM原則

下列範例原則授予您複製任何EBS支援AMI並標記目標AMI及其支援快照的權限。

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Sid": "PermissionToCopyAllImages",
        "Effect": "Allow",
        "Action": [
            "ec2:CopyImage",
            "ec2:CreateTags"
        ],
        "Resource": "arn:aws:ec2:*::image/*"
    }]
}
```

複製EBS已備份AMI但拒絕標記新快照的範例IAM原則

當您取得`ec2:CopySnapshot`權限時，系統會自動授與`ec2:CopyImage`權限。這包括標記目標新支援快照的權限AMI。可明確拒絕標記新支援快照的權限。

下列範例原則授予您複製任何EBS支援的權限AMI，但拒絕您標記目標的新支援快照。AMI

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

```
            "Effect": "Allow",
            "Action": [
                "ec2:CopyImage",
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:*::image/*"
        },
        {
            "Effect": "Deny",
            "Action": "ec2:CreateTags",
            "Resource": "arn:aws:ec2:::snapshot/*"
        }
    ]
}
```

複製執行個體以商店支援AMI並標記目標的範例IAM政策 AMI

下列範例政策授予您將指定來源儲存貯體AMI中任何執行個體商店支援的權限複製到指定的 Region，並標記目標。AMI

```
{
    "Version": "2012-10-17",
    "Statement": [{
            "Sid": "PermissionToCopyAllImages",
            "Effect": "Allow",
            "Action": [
                "ec2:CopyImage",
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:*::image/*"
        },
        {
            "Effect": "Allow",
            "Action": "s3:ListAllMyBuckets",
            "Resource": [
                "arn:aws:s3:::*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": "s3:GetObject",
            "Resource": [
                "arn:aws:s3:::ami-source-bucket/*"
```

```
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:CreateBucket",
                "s3:GetBucketAcl",
                "s3:PutObjectAcl",
                "s3:PutObject"
            ],
            "Resource": [
                "arn:aws:s3:::amis-for-account-in-region-hash"
            ]
        }
    ]
}
```

若要尋找AMI來源儲存貯體的 Amazon 資源名稱 (ARN)，請開啟 Amazon EC2 主控台 https://console.aws.amazon.com/ec2/，在導覽窗格中選擇 AMIs，然後在「來源」欄中找到值區名稱。

> **Note**
>
> 只有在您第一次將執行個體存放區備份複製到個別區域時，才需AMI要此`s3:CreateBucket`權限。之後，已在區域中建立的 Amazon S3 儲存貯體將用於存放您複製到AMIs該區域的所有 future 資料。

複製 AMI

您可以AMI使用下列程序複製。

Console

若要複製 AMI

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在主控台導覽列中，選取包含AMI.
3. 在導覽窗格中，選擇AMIs顯示 [地區] 中AMIs可供您使用的清單。
4. 如果沒有看到AMI要拷貝的項目，請選擇其他篩選器。您可以依 [我的AMIs擁有者]、[私人影像]、[公開影像] 和 [已停用的影像] 篩選。

5. 選取AMI要複製的，然後選擇「動作」>「複製」AMI。
6. 在「複製 AMI」頁面上，指定下列資訊：

   a. AMI複製名稱：新的名稱AMI。您可以在名稱中包含作業系統資訊，因為 Amazon EC2 在顯示AMI.

   b. AMI複製描述：依預設，描述會包含有關來源的資訊，以AMI便您可以區分複本與原始副本。您可視需要變更此描述。

   c. 目的地區域：要在其中複製的區域AMI。如需詳細資訊，請參閱跨區域複製。

   d. 複製標籤：選取此核取方塊可在複製時包含使用者定義的AMI標記AMI。將不會複製連接至其他 AWS 帳戶 的系統標籤 (字首為 `aws:`) 和使用者定義的標籤。

   e. (AMIs僅限 EBS-backed) 加密AMI複製的EBS快照：選取此核取方塊可加密目標快照，或使用不同的金鑰重新加密目標快照。如果預設為啟用加密，則會選取 [加密AMI副本EBS快照] 核取方塊，且無法清除。如需詳細資訊，請參閱加密和複製。

   f. (AMIs僅限 EBS-backed) KMS金鑰：用來加密目標快照的KMS金鑰。

   g. 標籤：您可以使用相同的標籤來標記新快照AMI和新快照，也可以使用不同的標籤來標記它們。

      - 若要使用相同標籤標記新快照AMI和新快照，請選擇「將影像和快照標記在一起」。相同的標籤會套用至新的快照AMI和所建立的每個快照。
      - 若要使用不同的標籤來標記新快照AMI和新快照，請分別選擇「標記影像和快照」。不同的標籤會套用至新的AMI和建立的快照。不過請注意，建立的所有新快照都會取得相同的標籤；您無法使用不同的標籤來標記每個新快照。

      若要新增標籤，請選擇 Add tag (新增標籤)，然後輸入標籤的鍵和值。針對每個標籤，重複執行上述動作。

   h. 當您準備好複製時AMI，請選擇「複製」AMI。

      新的初始狀態AMI為`Pending`。當狀態為時，AMI複製作業即完成`Available`。

AWS CLI

若要AMI使用複製 AWS CLI

您可以AMI使用「複製影像」指令來複製。您必須同時指定來源和目的地區域。請使用 `--source-region` 參數指定來源區域。請使用 `--region` 參數或環境變數指定目的地區域。若要取得更多資訊，請參閱〈規劃 AWS 指令行介面〉。

(AMIs僅限 EBS-backed) 在複製期間加密目標快照時，必須指定下列其他參數：`--encrypted`和`--kms-key-id`。

如需範例命令，請參閱 AWS CLI 命令參考 中的 copy-image 下的 Examples (範例)。

PowerShell

若要AMI使用視窗的工具複製 PowerShell

您可以AMI使用Copy-EC2Image指令複製。您必須同時指定來源和目的地區域。請使用 `-SourceRegion` 參數指定來源區域。請使用 `-Region` 參數或 `Set-AWSDefaultRegion` 命令指定目的地區域。如需詳細資訊，請參閱指定 AWS 區域。

(AMIs僅限 EBS-backed) 在複製期間加密目標快照時，必須指定下列其他參數：`-Encrypted`和`-KmsKeyId`。

停用待定的 AMI 複製作業

您可以使用下列程序停止擱置的AMI複本。

Console

使用控制台停止AMI複製作業

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 從導覽列中的區域選取器，選取目的地區域。
3. 在導覽窗格中，選擇AMIs。
4. 選取停AMI止複製，然後選擇動作，取消註冊AMI。
5. 當系統要求確認時，請選擇「取消註冊 AMI」。

Command line

使用指令行停止AMI複製作業

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱訪問 Amazon EC2。

- deregister-image (AWS CLI)
- Unregister-EC2Image (AWS Tools for Windows PowerShell)

Amazon EC2 AMI 副本如何工作

複製源代碼AMI會產生相同但不同AMI的新內容，我們也稱之為目標AMI。目標AMI有自己的唯一 AMI ID。您可以變更或取消註冊來源，而不會AMI對目標AMI造成任何影響。反之亦然。

使用 EBS-backed 時AMI，其每個支援快照都會複製到相同但不同的目標快照。如果您將一個複製AMI到新區域，則快照是完整的（非增量）副本。如果您加密未加密的備份快照或將其加密為新KMS金鑰，則快照會是完整 (非增量) 複本。後續的複製作業會AMI產生支援快照的增量複本。

目錄

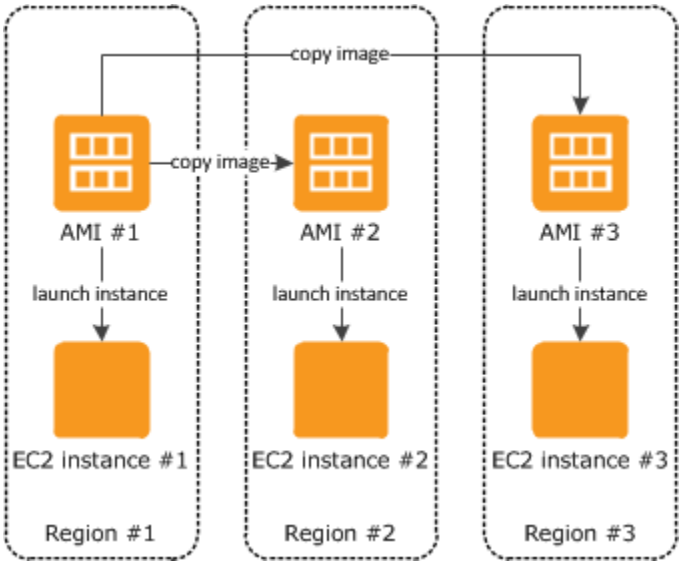
- 跨區域複製
- 跨帳戶複製
- 加密和複製

跨區域複製

AMI跨地理位置不同的區域複製具有以下好處：

- 一致的全球部署：將一個AMI區域複製到另一個區域可讓您根據相同區域在不同區域中啟動一致的執行個體AMI。
- 擴展能力：無論使用者的位置，設計並建置滿足全球使用者需求的應用程式將變得更輕鬆。
- 效能：您可分散應用程式，並將應用程式的重要元件放置於靠近使用者的位置，藉此增加效能。您也可以利用特定區域的功能，例如執行個體類型或其他 AWS 服務。
- 高可用性：您可跨 AWS 區域設計並部署應用程式，增加可用性。

下圖顯示來源AMI與AMIs在不同「區域」中複製的兩個來源之間的關係，以及從每個區域啟動的EC2執行環境。當您從執行個體啟動時AMI，它會位於該執行個體所在的相同區域中AMI。如果您對來源進行變更，AMI並希望這些變更反映AMIs在目標區域中，則必須將來源重新複製AMI到目標區域。



當您第一次將執行個體存放區備份AMI到某個區域時，我們會為AMIs複製到該區域的 Amazon S3 儲存貯體建立一個 Amazon S3 儲存貯體。您複製到AMIs該區域的所有執行個體商店支援，都會儲存在此值區中。值區名稱的格式如下：amis-for-*account*-在-*region-hash*。 例如：`amis-for-123456789012-in-us-east-2-yhjmxvp6`.

先決條件

在複製之前AMI，您必須確保來源AMI的內容已更新，以支援在不同區域中執行。例如，您應更新資料庫連線字串或類似的應用程式組態資料，以指向適當的資源。否則，從目的地區域AMI中的新執行個體啟動的執行個體仍可能會使用來源區域的資源，這可能會影響效能和成本。

限制

- 目的地區域限制為 100 個並行AMI複本。
- 您無法將半虛擬 (PV) 複製AMI到不支援 PV AMIs 的區域。如需詳細資訊，請參閱虛擬化類型。

跨帳戶複製

如果AMI來自另一 AWS 帳戶 個與您的共用 AWS 帳戶，您可以複製共用AMI。這稱為跨帳戶複製。與AMI您共享的是來源AMI。當您複製來源時AMI，您會建立新的來源AMI。新的AMI通常稱為目標AMI。

AMI成本

- 如果是共用帳戶AMI，AMI則會向該共用帳戶收取該區域中的儲存空間費用。
- 如果您複製與帳AMI戶共用的帳戶，您就是帳戶AMI中目標的擁有者。
  - 來源AMI所有者需支付標準的 Amazon EBS 或 Amazon S3 傳輸費。

- 您需支付目標AMI在目的地區域中儲存的費用。

資源許可

若要從其他帳戶複製與您共用的帳戶，來源的擁有者AMI必須授與您備份的儲存區的讀取權限AMI。AMI儲存可以是關聯的EBS快照 (針對 Amazon EBS 支援AMI) 或關聯的 S3 儲存貯體 (針對執行個體存放AMI區支援)。如果共用AMI具有加密快照，則擁有者也必須與您共用金鑰。如需授予資源許可的詳細資訊，對於EBS快照，請參閱 Amazon EBS使用者指南中的共用 Amazon EBS 快照，對於 S3 儲存貯體，請參閱 Amazon 簡單儲存服務使用者指南中的 Amazon S3 中的身分和存取管理。

> **Note**
> 附加至來源的標記不AMI會跨帳戶複製到目標AMI。

加密和複製

下表顯示各種AMI複製案例的加密支援。雖然您可將未加密快照複製為加密快照，但是無法將加密快照複製為未加密快照。

案例	描述	支援
1	U nencrypted-to-unencrypted	是
2	E ncrypted-to-encrypted	是
3	U nencrypted-to-encrypted	是
4	E ncrypted-to-unencrypted	否

> **Note**
> `CopyImage`動作期間的加密僅適用於支援 Amazon EBS 的功能。AMIs由於存放區備份的執行個體AMI不依賴快照，因此您無法使用複製來變更其加密狀態。

根據預設 (亦即，未指定加密參數)，系統會以其原始加密狀態複製的後備快照。AMI複製未加密快照AMI支援的結果會產生相同的目標快照，也未加密。如果來源AMI是由加密的快照所支援，則複製後會

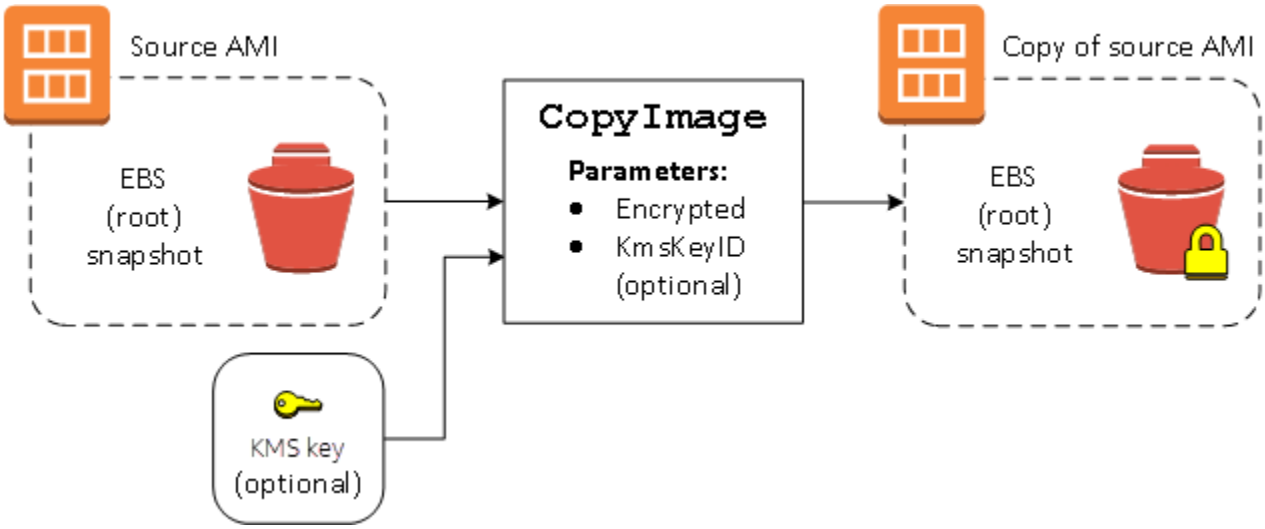
產生相同的目標快照，並以相同的 AWS KMS 金鑰加密。依預設，複製多個快照AMI支援的來源加密狀態會保留每個目標快照中的來源加密狀態。

如果您在複製時指定加密參數AMI，則可以加密或重新加密其後備快照。下列範例顯示非預設案例，該案例會為`CopyImage`動作提供加密參數，以變更目標AMI的加密狀態。

將未加密的來源複製AMI到加密的目標 AMI

在這個案例中，會將未加密的根快照所AMI支援的根快照複製到AMI具有加密根快照的。透過兩個加密參數叫用 `CopyImage` 動作，包括客戶受管金鑰。因此，根快照的加密狀態會變更，因此目標AMI由包含與來源快照相同資料的根快照進行支援，但使用指定的金鑰加密。您會產生兩種快照的儲存成本AMIs，以及從其中一個啟動的任何執行個體的費用AMI。

> **Note**
>
> 依預設啟用加密與將`true`中所有快照的`Encrypted`參數設定為的效果相同AMI。



設定 `Encrypted` 參數會加密此執行個體的單一快照。若您未指定 `KmsKeyId` 參數，則會使用預設的客戶受管金鑰來加密快照複本。

如需AMIs使用加密快照進行複製的詳細資訊，請參閱對 EBS 後端 AMI 使用加密。

AMI使用 S3 存放和恢復

您可以將 Amazon 機器映像 (AMI) 存放在 Amazon S3 儲存貯體中，將其複製AMI到另一個 S3 儲存貯體，然後從 S3 儲存貯體還原。透過存放和還原AMI使用 S3 儲存貯體，您可以AMIs從一個 AWS 分割

區複製到另一個分割區，例如，從主要商業分割區複製到 AWS GovCloud (US) 分割區。您也可以將它們存放在 S3 儲存貯體中AMIs來製作封存副本。

支APIs援儲存和還原AMI使用 S3 的`CreateStoreImageTaskDescribeStoreImageTasks`、和`CreateRestoreImageTask`。

`CopyImage`建議用API於在 AWS 分區AMIs內進行複製。但是，`CopyImage`無法將其複製AMI到另一個分區。

如需 AWS 分割區的相關資訊，請參閱 *partition* 在IAM使用者指南中的 Amazon 資源名稱 (ARNs) 頁面上。

> **Warning**
>
> 在分 AWS 區或 AWS 區域之間移動資料時，請確保您遵守所有適用的法律和業務要求，包括但不限於任何適用的政府法規和資料落地要求。

目錄

- 使用案例
- 限制
- 成本
- AMI儲存與還原的運作方式
- 建立商店映像檔工作

使用案例

使用儲存區和還原APIs來執行下列動作：

- AMI在 AWS 分割區之間複製
- 製作的歸檔副本 AMIs

AMI在 AWS 分割區之間複製

透過AMI使用 S3 儲存貯體存放和還原，您可以AMI從一個 AWS 分割區複製到另一個分割區，或從一個 AWS 區域複製到另一個區域。在下列範例中，您將AMI從主要商業分割區複製到分割區，特別是從「`us-east-2`區域」複製到「`us-gov-east-1`區域」。 AWS GovCloud (US)

要AMI從一個分割區複製到另一個分割區，請依照下列步驟執行：

- 使用將其存放AMI在目前區域的 S3 儲存貯體中`CreateStoreImageTask`。在此範例中，S3 儲存貯體位於 `us-east-2`。
- 使用 `DescribeStoreImageTasks` 監控存放任務的進度。當任務完成時，該物件在 S3 儲存貯體中變得可見。
- 使用您選擇的程序，將存放的AMI物件複製到目標磁碟分割中的 S3 儲存貯體。在此範例中，S3 儲存貯體位於 `us-gov-east-1`。

> **Note**
>
> 因為每個分割區需要不同的 AWS 登入資料，因此無法將 S3 物件直接從一個分割區複製到另一個分割區。跨分割區複製 S3 物件的程序超出本文件的範圍。我們提供下列複製程序作為範例，但您必須使用符合安全性需求的複製程序。
>
> - 若要AMI跨分割區進行複製，複製程序可能如下所示：將物件從來源值區下載至中繼主機 (例如EC2執行個體或膝上型電腦)，然後將物件從中繼主機上傳至目標值區。針對程序的每個階段，請使用磁碟分割的 AWS 認證。
> - 為了更能夠持續使用，請考慮開發一個能管理複製的應用程式，最好可使用 S3 分段下載和上傳。

- 使用AMI從目標磁碟分割中的 S3 儲存貯體還原`CreateRestoreImageTask`。在此範例中，S3 儲存貯體位於 `us-gov-east-1`。
- 透過描述AMI要檢查還原工作狀態何時可用，來監視還原工作的進度。您也可以AMI透過描述快照來監視組成還原之快照的進度百分比。

製作的歸檔副本 AMIs

您可以將它們存放在 S3 儲存貯體中AMIs來製作封存副本。會AMI封裝到 S3 中的單一物件中，而且所有中AMI繼資料 (不包括共用資訊) 都會保留為儲存的一部分AMI。資AMI料會壓縮為儲存程序的一部分。AMIs包含可以輕鬆壓縮的資料，將導致 S3 中的物件變小。為了降低成本，您可以使用較低成本的 S3 儲存層。如需詳細資訊，請參閱 Amazon S3 儲存類別和 Amazon S3 定價

限制

- 若要儲存AMI，您 AWS 帳戶 必須擁有AMI及其快照，或是AMI與其快照直接與您的帳戶共用。如果僅公開共享，AMI則無法存儲它。
- 只有 EBS-backed AMIs 可以使用這些APIs存儲。

- 不支援半虛擬 (AMIsPV)。
- 可以存儲的AMI（壓縮之前）的大小限制為 5,000 GB。
- 存放映像請求的配額：600 GB 的進行中儲存工作 (快照資料)。
- 還原映像請求的配額：300 GB 的進行中還原工作 (快照資料)。
- 在儲存區工作期間，不得刪除快照，且執行儲存區的IAM主體必須能夠存取快照，否則儲存程序將會失敗。
- 您無法在同一個 S3 儲存貯體AMI中建立多個副本。
- 存儲AMI在 S3 存儲桶中的存儲不能使用其原始 AMI ID 恢復。您可以通過使用AMI別名來緩解此問題。
- 目前只APIs有使用 AWS Command Line Interface、 AWS SDKs和 Amazon 才能支援存放區和還原EC2API。您無法AMI使用 Amazon EC2 控制台存儲和恢復。

成本

AMIs使用 S3 存放和還原時，需支付存放區和還原APIs所使用的服務以及資料傳輸的費用。APIs使用 S3 和 EBS DirectAPI（由這些內部用APIs於訪問快照數據）。如需詳細資訊，請參閱 Amazon S3 定價和 Amazon EBS 定價。

AMI儲存與還原的運作方式

若要使用 S3 存放和還原，請AMI使用下列指令APIs：

- `CreateStoreImageTask`— 將其存儲AMI在 S3 存儲桶中
- `DescribeStoreImageTasks`— 提供存AMI儲任務的進度
- `CreateRestoreImageTask`— AMI 從 S3 儲存貯體還原

如何工APIs作

- CreateStoreImageTask
- DescribeStoreImageTasks
- CreateRestoreImageTask
- 檔案路徑

CreateStoreImageTask

將AMI作為單個對象`CreateStoreImageTask`API存儲在 S3 存儲桶中。

這會API建立一個從AMI及其快照讀取所有資料的工作，然後使用 S3 多部分上傳將資料存放在 S3 物件中。會擷API取的所有元件AMI，包括大多數非區域特定的中AMI繼資料，以及中包含的所有EBS快照AMI，並將它們封裝到 S3 中的單一物件中。資料會作為上傳程序的一部分進行壓縮，以減少 S3 中使用的空間量，因此 S3 中的物件可能小於中快照大小的總和AMI。

如果調用此標籤的帳戶可見AMI和快照標籤API，則會保留它們。

S3 中的物件具有與相同的 IDAMI，但具有`.bin`副檔名。下列資料也會以 S3 中繼資料標籤的形式存放在 S3 物件上：AMI名稱、AMI說明、AMI註冊日期、AMI擁有者帳戶和存放區作業的時間戳記。

完成任務所花費的時間取決於AMI. 這也取決於有多少其他任務正在進行中，因為任務已排入佇列。您可以呼叫來追蹤工作的進度`DescribeStoreImageTasks`API。

所有進行AMIs中的大小總和限制為每個帳戶 600 GB 的EBS快照資料。系統會拒絕更多的任務建立，除非進行中的任務數量小於限制數量。例如，如果目前儲存的快照資料AMI有 100 GB，另一個快照資料為 200 GB，則會接受另一個要求，因為進行中的總計為 300 GB，小於限制。AMI但是，如果目前正在儲存AMI具有 800 GB 快照資料的單一資料，則會拒絕進一步的工作，直到工作完成為止。

DescribeStoreImageTasks

`DescribeStoreImageTasks`API說明AMI儲存區工作的進度。您可以描述指定的工作AMIs。如果未指定AMIs，則會取得過去 31 天內處理過的所有商店映像檔工作的分頁清單。

對於每個AMI任務，響應指示任務是`InProgress``Completed`、或`Failed`。對於任務`InProgress`，回應會以百分比顯示預估進度。

任務會以相反的時間順序列出。

目前，只能檢視上個月的任務。

CreateRestoreImageTask

會`CreateRestoreImageTask`API啟動AMI從先前使用`CreateStoreImageTask`要求建立的 S3 物件還原的工作。

還原任務可以在執行存放任務的相同或不同區域中執行。

要從中還原AMI物件的 S3 儲存貯體必須位於要求還原任務的相同區域。AMI將在此區域中恢復。

系統AMI會使用其中繼資料來還原，例如名稱、描述和與儲存值對應的區塊裝置對應AMI。此帳戶的名稱AMIs在「區域」中必須是唯一的。如果您未提供名稱，新名稱AMI會取得與原始名稱相同的名稱AMI。會AMI取得在還原程序時產生的新 AMI ID。

完成AMI還原工作所需的時間取決於AMI. 這也取決於有多少其他任務正在進行中，因為任務已排入佇列。您可以透過描述 (描述影像) 或其EBS快照 AMI (描述- 快照) 來檢視工作的進度。如果工作失敗，AMI和快照會移至失敗狀態。

所有進行AMIs中的大小總和限制為每個帳戶 300 GB (根據還原後的EBS大小)。系統會拒絕更多的任務建立，除非進行中的任務數量小於限制數量。

檔案路徑

您可以透過下列方式在儲存和還原AMIs時使用檔案路徑：

- AMI在 S3 中存儲時，可以將文件路徑添加到存儲桶名稱中。在內部，系統會將路徑與值區名稱分隔開來，然後將路徑新增至所產生的物件索引鍵以儲存。AMI完整的物件路徑會顯示在API呼叫的回應中。
- 還原時AMI，由於物件索引鍵參數可用，因此可以將路徑加入至物件索引鍵值的開頭。

範例：儲存和還原 AMI (AWS CLI) 時使用檔案路徑

下列範例會先將 a 儲存AMI在 S3 中，檔案路徑會附加至儲存貯體名稱。然後，範例會AMI從 S3 還原，並在物件索引鍵參數前面加上檔案路徑。

儲存時AMI，請在值區名稱後指定檔案路徑，如下所示：

```
aws ec2 create-store-image-task \
    --image-id ami-1234567890abcdef0 \
    --bucket amzn-s3-demo-bucket/path1/path2
```

下列為範例輸出。

```
{
  "ObjectKey": "path1/path2/ami-1234567890abcdef0.bin"
}
```

當您還原時AMI，請在上一個步驟中指定輸出的值，其中包括檔案路徑。

```
aws ec2 create-restore-image-task \
    --object-key path1/path2/ami-1234567890abcdef0.bin \
    --bucket amzn-s3-demo-bucket \
    --name "New AMI Name"
```

建立商店映像檔工作

當您將儲存AMI在 S3 儲存貯體中時，會建立存放區映像任務。您可以使用存放區映像工作來監視程序的進度和結果。

目錄

- 保護您的 AMIs
- AMIs使用 S3 存放和還原的許可
- 建立儲存區和還原映像工作

保護您的 AMIs

請務必確保 S3 儲存貯體設定有足夠的安全性來保護儲存貯體的內容，AMI並且只要AMI物件保留在儲存貯體中，就能維持安全性。如果不能這樣做，不建議使APIs用這些。確保不允許公開存取 S3 儲存貯體。我們建議您為存放所在的 S3 儲存貯體啟用伺服器端加密AMIs，但這不是必要的。

如需如何為 S3 儲存貯體設置適當安全性設定的相關資訊，請檢閱下列安全性主題：

- 封鎖對 Amazon S3 儲存體的公開存取權
- 對 Amazon S3 儲存貯體設定預設伺服器端加密行為
- 我可以使用哪些 S3 儲存貯體政策來遵守 s3- AWS Config 規則bucket-ssl-requests-only？
- 啟用 Amazon S3 伺服器存取記錄日誌

將AMI快照複製到 S3 物件時，資料便會透過TLS連線複製。您可以AMIs使用加密的快照進行儲存，但快照會在儲存過程中解密。

AMIs使用 S3 存放和還原的許可

如果您的IAM主體將AMIs使用 Amazon S3 存放或還原，則需要授予他們所需的許可。

下列範例原則包含允許IAM主參與者執行儲存和還原工作所需的所有動作。

您也可以建立IAM原則，僅授與主參與者存取特定資源。如需更多原則範例，請參閱《IAM使用指南》中的 AWS 資源存取管理。

> **Note**
>
> 如果構成的快照已加密，或者您的帳戶預設為啟用加密功能，則您的IAM主體必須具有使用KMS金鑰的權限。AMI

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:DeleteObject",
                "s3:GetObject",
                "s3:ListBucket",
                "s3:PutObject",
                "s3:PutObjectTagging",
                "s3:AbortMultipartUpload",
                "ebs:CompleteSnapshot",
                "ebs:GetSnapshotBlock",
                "ebs:ListChangedBlocks",
                "ebs:ListSnapshotBlocks",
                "ebs:PutSnapshotBlock",
                "ebs:StartSnapshot",
                "ec2:CreateStoreImageTask",
                "ec2:DescribeStoreImageTasks",
                "ec2:CreateRestoreImageTask",
                "ec2:GetEbsEncryptionByDefault",
                "ec2:DescribeTags",
                "ec2:CreateTags"
            ],
            "Resource": "*"
        }
    ]
}
```

建立儲存區和還原映像工作

若要將儲存AMI在 S3 儲存貯體中，請先建立存放映像任務。完成任務所花費的時間取決於AMI. 您可以追蹤任務的進度，直到工作成功或失敗為止。

若要建立存放區映像工作

使用指create-store-image-task令。指定要在其中存放的 S3 儲存貯體的識別碼AMI和名稱AMI。

```
aws ec2 create-store-image-task \
    --image-id ami-1234567890abcdef0 \
    --bucket amzn-s3-demo-bucket
```

下列為範例輸出。

```
{
  "ObjectKey": "ami-1234567890abcdef0.bin"
}
```

描述存放區映像工作的進度

使用指describe-store-image-tasks令。

```
aws ec2 describe-store-image-tasks
```

下列為範例輸出。

```
{
    "StoreImageTaskResults": [
        {
            "AmiId": "ami-1234567890abcdef0",
            "Bucket": "amzn-s3-demo-bucket",
            "ProgressPercentage": 17,
            "S3objectKey": "ami-1234567890abcdef0.bin",
            "StoreTaskState": "InProgress",
            "StoreTaskFailureReason": null,
            "TaskStartTime": "2022-01-01T01:01:01.001Z"
        }
    ]
}
```

建立還原影像工作

使用指create-restore-image-task令。使用輸出`S3ObjectKey`和`describe-store-image-tasks`輸`Bucket`出的值，指定要複製到AMI的 S3 儲存貯體的物件金鑰和名稱。AMI還指定恢復的名稱AMI。此帳戶的名稱AMIs在「區域」中必須是唯一的。

> **Note**
>
> 恢復將AMI獲得一個新的 AMI ID。

```
aws ec2 create-restore-image-task \
    --object-key ami-1234567890abcdef0.bin \
    --bucket amzn-s3-demo-bucket \
    --name "New AMI Name"
```

下列為範例輸出。

```
{
   "ImageId": "ami-0eab20fe36f83e1a8"
}
```

檢查 Amazon 上EC2AMI次使用的時間

Amazon 會追EC2蹤您AMI上次用來啟動執行個體的日期和時間。如果您有長時間未用於啟動執行個體的執行個體，請考慮AMI該執行個體AMI是否適合取消註冊或淘汰。

考量事項

- 使用啟動執行個體時，會有 24 小時的延遲時間，才會報告該使用量。AMI
- 您必須是的擁有者才能取AMI得上次啟動的時間。
- 有關AMI使用情況的資料將於 2017 年 4 月開始提供。

Console

若要檢視上次啟動的時間 AMI

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側導覽窗格中，選擇AMIs。
3. 從篩選條件列中，選擇 Owned by me (由我擁有)。
4. 選取AMI，然後核取 [上次啟動時間] 欄位 (如果您已選取 [詳細資料] 索引標籤旁的核取方塊)。AMI此欄位會顯示上次用來啟動執行個體的日期和時間。AMI

AWS CLI

您可以使用描述影像或describe-image-attribute指令來檢視您擁AMI有的上次啟動時間。

使用描述影像AMI檢視上次啟動時間的步驟

使用描述影像指令並指定的 ID。AMI

```
aws ec2 describe-images --image-id ami-0123456789example --query
 "Images[*].LastLaunchedTime[].Value"
```

下列為範例輸出。

```
[
    "2024-04-02T02:03:18Z"
]
```

如果`LastLaunchedTime`不存在，請確認您擁有AMI.

若要檢視上次啟動的時間 AMI

使用指describe-image-attribute令並指定`--attribute lastLaunchedTime`。您必須是的擁有者AMI才能執行此命令。

```
aws ec2 describe-image-attribute \
    --image-id ami-0123456789example \
    --attribute lastLaunchedTime
```

下列為範例輸出。

```
{
    "ImageId": "ami-1234567890example",
    "LastLaunchedTime": {
        "Value": "2022-02-10T02:03:18Z"
    }
}
```

棄用 Amazon EC2 AMI

您可以棄用，AMI以表示它已過期且不應該使用。您也可以指定 future 的取代日期AMI，指出何時AMI會過期。例如，您可能會棄用不再主動維護AMI的版本，或者您可能會棄用已被較新版本取代AMI的版

本。根據預設，已取代AMIs不會出現在AMI清單中，因此新使用者無法使用 out-of-date AMIs。不過，現有使用者和啟動服務 (例如啟動範本和 Auto Scaling 群組) 可以AMI透過指定其 ID 來繼續使用已取代的。若要刪除AMI讓使用者和服務無法使用它，您必須取消註冊。

在被棄AMI用之後：

- 對於AMI使用者，除非您指定其 ID 或指定AMIs必須顯示已取代的項目，否則已取代的項目不AMI會出現在DescribeImagesAPI呼叫中。AMI擁有者AMIs在DescribeImagesAPI呼叫中繼續看到已淘汰。
- 對於AMI使用者，已取代AMI的項目無法透過EC2主控台選取。例如，已取代的項目AMI不會出現在啟動實例精靈的AMI目錄中。AMI所有者繼續在EC2控制台AMIs中看到已棄用。
- 對於使用AMI者，如果您知道已取代的 IDAMI，您可以使用、或繼續使用已取代AMI的執行個體API來啟動執行個體SDKs。CLI
- 啟動服務 (例如啟動範本和 Auto Scaling 群組) 可以繼續參考已取代AMIs。
- EC2使用隨後被取代的啟動AMI的實例不會受到影響，而且可以停止、啟動和重新啟動。

您可以棄用私人和公共。AMIs

您也可以建立 Amazon Data Lifecycle Manager EBS 支援的AMI政策，以自動化已備份的EBS棄用。AMIs如需詳細資訊，請參閱自動化AMI生命週期。

> **Note**
>
> 默認情況下，所有公共的棄用日期AMIs都設置為從AMI創建日期起兩年。您可將棄用日期設定為早於兩年。若要取消淘汰日期，或將淘汰日期移至較晚的日期，您必須僅與特定 AWS 帳戶共用，將其設為AMI私有。

目錄

- 成本
- 限制
- 棄用 AMI
- 描述已棄用 AMIs
- 取消AMI棄用

成本

當您取代時AMI，不會刪除。AMIAMI擁有者會繼續支付快AMI照費用。若要停止支付快照費用，AMI擁有者必須AMI透過取消註冊來刪除快照。

限制

- 若要取代AMI，您必須是. 的擁有者。AMI

棄用 AMI

您可以在特定日期和時間取代。AMI您必須是AMI擁有者才能執行此程序。

Console

若要在特定AMI日期淘汰

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側導覽器中，選擇AMIs。
3. 從篩選條件列中，選擇 Owned by me (由我擁有)。
4. 選取AMI，然後選擇 [動作]、[管理AMI淘汰]。您可以選擇多個AMIs以一次設置多AMIs個相同的棄用日期。
5. 選取 Enable (啟用) 核取方塊，然後輸入取代日期和時間。

   棄用日期的上限為 10 年後，公眾除外AMIs，其上限是自創建日期起 2 年。您無法指定過去的日期。
6. 選擇 Save (儲存)。

AWS CLI

若要在特定AMI日期淘汰

使用指enable-image-deprecation令。指定要取代的 AMI ID 以及取代的日期和時間。AMI如果您指定以秒為單位的值，Amazon 會將秒數EC2四捨五入到最接近的分鐘。

上限`deprecate-at`為 10 年後，公眾除外AMIs，其上限為創建日期起 2 年。您無法指定過去的日期。

```
aws ec2 enable-image-deprecation \
```

```
    --image-id ami-1234567890abcdef0 \
    --deprecate-at "2021-10-15T13:17:12.000Z"
```

預期的輸出結果

```
{
  "Return": "true"
}
```

檢查上AMI次使用的時間

`LastLaunchedTime`是一個時間戳記，指出您上AMI次用於啟動實例的時間。AMIs最近沒有用於啟動實例的可能是棄用或取消註冊的好候選人。

> **Note**
>
> - 使用 a 啟動執行個體時，會有 24 小時的延遲時間，才會報告該使用量。AMI
> - `lastLaunchedTime` 資料從 2017 年 4 月開始提供。

Console

若要檢視上次啟動的時間 AMI

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側導覽器中，選擇AMIs。
3. 從篩選條件列中，選擇 Owned by me (由我擁有)。
4. 選取AMI，然後核取 [上次啟動時間] 欄位 (如果您已選取 [詳細資料] 索引標籤旁的核取方塊)。AMI此欄位會顯示上次用來啟動執行個體的日期和時間。AMI

AWS CLI

若要檢視上次啟動的時間 AMI

執行命describe-image-attribute令並指定`--attribute lastLaunchedTime`。您必須是執行此命令的AMI擁有者。

```
aws ec2 describe-image-attribute \
```

```
    --image-id ami-1234567890example \
    --attribute lastLaunchedTime
```

範例輸出

```
{
    "LastLaunchedTime": {
        "Value": "2022-02-10T02:03:18Z"
    },
    "ImageId": "ami-1234567890example",
}
```

描述已棄用 AMIs

您可以查看的棄用日期和時間AMI，並AMIs按棄用日期過濾所有。您也可以使用 AWS CLI 來描述所有AMIs已被棄用的，其中的棄用日期是過去的。

Console

若要檢視的取代日期 AMI

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側瀏覽器中 AMIs，選擇，然後選取AMI。
3. 核取 [棄用時間] 欄位 (如果您選取了 [詳細資料] 索引標籤旁的核取方塊)。AMI此欄位會顯示. AMI 如果欄位為空白，則AMI不會取代。

若要AMIs依棄用日期篩選

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側導覽器中，選擇AMIs。
3. 在篩選列中，選擇「我擁有」或「私人影像」(包AMIs含與您共用以及您擁有的私人影像)。
4. 在搜尋列中，輸入 **`Deprecation time`** (當您輸入字母時，Deprecation time (取代時間) 篩選條件隨即顯示)，隨後選擇一個運算子及一個日期和時間。

AWS CLI

當您AMIs使用描述-images 指令描述所有項目時，結果會根據您是使用者還是擁有AMI者而有所不同。AMI

- 如果您是AMI使用者：

  根據預設，當您AMIs使用 describe-images 指令描述所有內容時，不AMIs是您擁有但與您共用的已取代，則不會出現在結果中。這是因為預設值為 `--no-include-deprecated`。若要在結果AMIs中包含已取代，您必須指定`--include-deprecated`參數。

- 如果您是擁AMI有者：

  當您AMIs使用 describe-images 指令描述所有項目時，您擁有的所有內AMIs容 (包括已取代AMIs) 都會出現在結果中。您不需要指定 `--include-deprecated` 參數。此外，您不能使用從結果中排除您擁有的已棄AMIs用`--no-include-deprecated`。

如果AMI已取代，則`DeprecationTime`欄位會顯示在結果中。

> **Note**
>
> 棄用的AMIAMI是棄用日期是過去的。如果您已將棄用日期設定為 future 的日期，AMI則尚未取代。

若要AMIs在描述全部時包含所有已停用 AMIs

使用 describe-images 指令並指定`--include-deprecated`參數，以便在結果中包含您未擁有的所有AMIs已取代的已取代項目。

```
aws ec2 describe-images \
    --region us-east-1 \
    --owners 123456example
    --include-deprecated
```

描述的棄用日期 AMI

使用描述影像指令並指定的 ID。AMI

請注意，如果您與 AMI ID `--no-include-deprecated` 一起指定，則不推薦使用的AMI將在結果中返回。

```
aws ec2 describe-images \
    --region us-east-1 \
    --image-ids ami-1234567890EXAMPLE
```

預期的輸出結果

此DeprecationTime欄位會顯示設定AMI為取代的日期。如果未設定為已取代，則DeprecationTime欄位不會顯示在輸出中。AMI

```
{
    "Images": [
        {
            "VirtualizationType": "hvm",
            "Description": "Provided by Red Hat, Inc.",
            "PlatformDetails": "Red Hat Enterprise Linux",
            "EnaSupport": true,
            "Hypervisor": "xen",
            "State": "available",
            "SriovNetSupport": "simple",
            "ImageId": "ami-1234567890EXAMPLE",
            "DeprecationTime": "2021-05-10T13:17:12.000Z"
            "UsageOperation": "RunInstances:0010",
            "BlockDeviceMappings": [
                {
                    "DeviceName": "/dev/sda1",
                    "Ebs": {
                        "SnapshotId": "snap-111222333444aaabb",
                        "DeleteOnTermination": true,
                        "VolumeType": "gp2",
                        "VolumeSize": 10,
                        "Encrypted": false
                    }
                }
            ],
            "Architecture": "x86_64",
            "ImageLocation": "123456789012/RHEL-8.0.0_HVM-20190618-x86_64-1-Hourly2-
GP2",
            "RootDeviceType": "ebs",
            "OwnerId": "123456789012",
            "RootDeviceName": "/dev/sda1",
            "CreationDate": "2019-05-10T13:17:12.000Z",
            "Public": true,
            "ImageType": "machine",
            "Name": "RHEL-8.0.0_HVM-20190618-x86_64-1-Hourly2-GP2"
        }
    ]
}
```

取消AMI棄用

您可以取消淘汰AMI，從「棄用時間」字段（控制台）中刪除日期和時間，或者從 describe- images 輸出（）中刪除`DeprecationTime`字段。AWS CLI您必須是AMI擁有者才能執行此程序。

Console

若要取消淘汰 AMI

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側導覽器中，選擇AMIs。
3. 從篩選條件列中，選擇 Owned by me (由我擁有)。
4. 選取AMI，然後選擇 [動作]、[管理AMI淘汰]。您可以選擇多個AMIs以一次取消多個棄用。AMIs
5. 清除 Enable (啟用) 核取方塊，然後選擇 Save (儲存)。

AWS CLI

若要取消淘汰 AMI

使用指disable-image-deprecation令並指定的 ID AMI。

```
aws ec2 disable-image-deprecation \
    --image-id ami-1234567890abcdef0
```

預期的輸出結果

```
{
  "Return": "true"
}
```

禁用 Amazon EC2 AMI

您可以禁用AMI以防止將其用於實例啟動。您無法從已停用的執行個體啟動新的執行個體AMI。您可以重新啟用禁用的功能，以AMI便在實例啟動時再次使用它。

> **Warning**
> 停用會AMI移除其所有啟動權限。

當一個AMI被禁用時：

- AMI的狀態會變更為`disabled`。
- 禁用的AMI無法共享。如果AMI是公開的或先前共用的，則會將其設為私有。如果與 AWS 帳戶、組織或組織單位共用，則他們將失去對禁用者的存取權AMI。AMI
- 根據預設，通DescribeImagesAPI話中AMI不會顯示停用。
- 已停用AMI不會顯示在 [由我擁有] 主控台篩選器下方。若要尋找停用AMIs，請使用 [已停用的映像] 主控台篩選
- 停用無AMI法在EC2主控台中選取執行個體啟動。例如，已停用的項目AMI不會出現在啟動執行個體精靈的AMI目錄中，或建立啟動範本時。
- 啟動服務 (例如啟動範本和 Auto Scaling 群組) 可繼續停用參照AMIs。從停用的執行個體後續啟動AMI將會失敗，因此我們建議將啟動範本和 Auto Scaling 群組更新為AMIs僅可參考。
- EC2先前使用隨後停用的執行處理啟動AMI的執行處理不會受到影響，而且可以停止、啟動和重新啟動。
- 您無法刪除與停用相關聯的快照AMIs。嘗試刪除關聯的快照會導致`snapshot is currently in use`錯誤。

當重新啟用時：AMI

- AMI的狀態會變更為`available`，而且可用來啟動執行個體。
- 可AMI以共用。
- AWS 帳戶、組織和組織單位在停用AMI時失去存取權限的組織和組織單位不會自動重新獲得存取權，但AMI可以再次與他們共用。

您可以禁用私人和公共AMIs。

您可以封存與已停用的EBS備份AMIs相關聯的快照。這可以幫助您降低需要長期保留的罕見使用AMIs相關的存儲成本。如需詳細資訊，請參閱 Amazon EBS使用者指南中的封存 Amazon EBS 快照。

目錄

- 成本
- 必要條件
- 必要的IAM權限
- 禁用 AMI

- 描述已禁用 AMIs
- 重新啟用已禁用 AMI

成本

當您停用時AMI，不AMI會刪除。如果AMI是 EBS-backedAMI，您將繼續支付EBS快照費用。AMI如果您想要保留AMI，可以透過封存快照來降低儲存成本。如需詳細資訊，請參閱 Amazon EBS使用者指南中的封存 Amazon EBS 快照。如果您不想保留AMI及其快照，則必須取消註冊AMI並刪除快照。如需詳細資訊，請參閱EBS-支持 AMIs。

必要條件

若要停用或重新啟用AMI，您必須是的擁有者。AMI

必要的IAM權限

若要停用並重新啟用AMI，您必須具備下列IAM權限：

- `ec2:DisableImage`
- `ec2:EnableImage`

禁用 AMI

您可以使AMI用EC2控制台或 AWS Command Line Interface （AWS CLI）來禁用。您必須是AMI擁有者才能執行此程序。

Console

若要停用 AMI

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側導覽窗格中，選擇AMIs。
3. 從篩選條件列中，選擇 Owned by me (由我擁有)。
4. 選取AMI，然後選擇 [動作]、[停用] AMI。您可以選擇多個AMIs以一次禁用。
5. 在「停用AMI」視窗中，選擇「停用」AMI。

AWS CLI

若要停用 AMI

使用指disable-image令並指定的 ID AMI。

```
aws ec2 disable-image --image-id ami-1234567890abcdef0
```

預期的輸出結果

```
{
  "Return": "true"
}
```

描述已禁用 AMIs

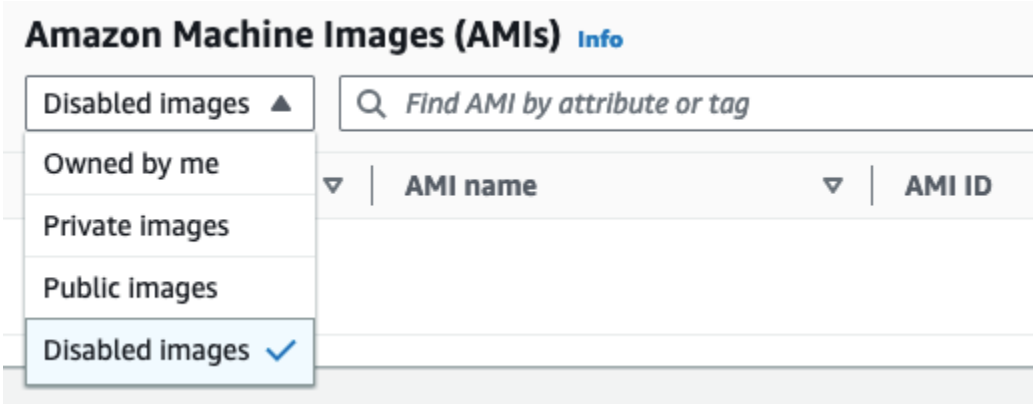
您可以在EC2主控台AMIs中檢視已停用，並使用 AWS CLI.

您必須是AMI擁有者才能檢視停用AMIs。由於禁用AMIs被設為私人，因此AMIs如果您不是所有者，則無法查看禁用。

Console

若要檢視已停用 AMIs

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側導覽窗格中，選擇AMIs。
3. 在篩選列中，選擇 [停用的影像]。



AWS CLI

默認情況下，當您使用命describe-images令來描述所有內容時AMIs，禁用AMIs不會出現在結果中。這是因為預設值為 `--no-include-disabled`。若要在結果AMIs中包含禁用，您必須指定`--include-disabled`參數。

描述全部AMIs時包含所有已停用的步驟 AMIs

除了所有其他`--include-disabled`參數之外，使用指describe-images令並指定要擷取停AMIs用的參數AMIs。或者，指`--owners self`定僅擷取AMIs您擁有的。

```
aws ec2 describe-images \
    --region us-east-1 \
    --owners self
    --include-disabled
```

如果您指定停用的 IDAMI，但未指定`--include-disabled`，則會在結果中傳回停用AMI的 ID。

```
aws ec2 describe-images \
    --region us-east-1 \
    --image-ids ami-1234567890EXAMPLE
```

僅擷取已停用 AMIs

指定 `--filters Name=state,Values=disabled`。還必須指定 `--include-disabled`，否則會收到錯誤。

```
aws ec2 describe-images \
    --include-disabled \
    --filters Name=state,Values=disabled
```

範例輸出

此`State`欄位會顯示的狀態AMI。 `disabled`表示已AMI停用。

```
{
    "Images": [
        {
            "VirtualizationType": "hvm",
            "Description": "Provided by Red Hat, Inc.",
            "PlatformDetails": "Red Hat Enterprise Linux",
            "EnaSupport": true,
```

```
            "Hypervisor": "xen",
            "State": "disabled",
            "SriovNetSupport": "simple",
            "ImageId": "ami-1234567890EXAMPLE",
            "DeprecationTime": "2023-05-10T13:17:12.000Z"
            "UsageOperation": "RunInstances:0010",
            "BlockDeviceMappings": [
                {
                    "DeviceName": "/dev/sda1",
                    "Ebs": {
                        "SnapshotId": "snap-111222333444aaabb",
                        "DeleteOnTermination": true,
                        "VolumeType": "gp2",
                        "VolumeSize": 10,
                        "Encrypted": false
                    }
                }
            ],
            "Architecture": "x86_64",
            "ImageLocation": "123456789012/RHEL-8.0.0_HVM-20190618-x86_64-1-Hourly2-
GP2",
            "RootDeviceType": "ebs",
            "OwnerId": "123456789012",
            "RootDeviceName": "/dev/sda1",
            "CreationDate": "2019-05-10T13:17:12.000Z",
            "Public": false,
            "ImageType": "machine",
            "Name": "RHEL-8.0.0_HVM-20190618-x86_64-1-Hourly2-GP2"
        }
    ]
}
```

重新啟用已禁用 AMI

您可以重新啟用已停AMI用的功能。您必須是AMI擁有者才能執行此程序。

Console

若要重新啟用已停用 AMI

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側導覽窗格中，選擇AMIs。

3. 在篩選列中，選擇 [停用的影像]。
4. 選取AMI，然後選擇 [動作] > [啟用] AMI。您可以選取多個AMIs以一次重新啟用多AMIs個功能。
5. 在「啟用AMI」視窗中，選擇「啟用」。

AWS CLI

若要重新啟用已停用 AMI

使用指enable-image令並指定的 ID AMI。

```
aws ec2 enable-image --image-id ami-1234567890abcdef0
```

預期的輸出結果

```
{
  "Return": "true"
}
```

註銷 Amazon EC2 AMI

當您取消註冊時AMI，Amazon 會將其EC2永久刪除。取消註冊之後AMI，您無法使用它來啟動新的執行個體。您可以考慮在完成使用後取消註冊。AMI

若要防止意外或惡意取消註冊AMI，您可以開啟取消註冊保護。如果您不小心取消註冊 EBS-backedAMI，則只有在永久刪除之前在允許的時間內還原它時，才可以使用資源回收筒還原它。

取AMI消註冊不會影響從啟動的任何執行個AMI體。您可以繼續使用這些執行個體。取AMI消註冊AMI也不會影響建立程序期間建立的任何快照。您將繼續支付這些執行個體的使用費用，以及快照的儲存成本。因此，為避免產生不必要的費用，我們建議您終止任何執行個體，並刪除不需要的快照。如需詳細資訊，請參閱避免未使用資源的成本。

目錄

- 考量事項
- 取消註冊 AMI
- 避免未使用資源的成本
- 保護 Amazon 免受EC2AMI註銷

考量事項

- 您無法取消註冊不屬AMI於您帳戶的帳戶。
- 您不能使用 Amazon EC2 取消註冊由AMI該 AWS Backup 服務管理的。請改用 AWS Backup 來刪除備份儲存庫中對應的復原點。如需詳細資訊，請參閱《AWS Backup 開發人員指南》中的刪除備份。

取消註冊 AMI

使用下列任一方法取消註冊 EBS-backed AMI 或執行個體存放區支援。AMI

> **Tip**
>
> 為避免產生不必要的成本，您應該刪除不需要的任何資源。例如，對於 EBS-backedAMIs，如果您不需要與已取消註冊相關聯的快照AMI，則應將其刪除。如需詳細資訊，請參閱避免未使用資源的成本。

Console

將 AMI取消註冊

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇AMIs。
3. 在篩選列中，選擇 [我擁有] 以列出可用的影像AMIs，或選擇 [停用的影像] 以列出已停用的影像AMIs。
4. 選取AMI要取消註冊的。
5. 選擇 Actions (動作)、DeregisterAMI (取消註冊)。
6. 當系統提示您進行確認時，請選擇「取消註冊 AMI」。

   控制台可能需要幾分鐘的時間才能AMI從清單中移除。選擇 Refresh (重新整理) 以重新整理狀態。

AWS CLI

將 AMI取消註冊

使用取消註冊影像指令，並指定要取消註冊的 AMI ID。

```
aws ec2 deregister-image --image-id ami-0123456789example
```

PowerShell

將 AMI取消註冊

使用指Unregister-EC2Image令程式並指定AMI要取消註冊的識別碼。

```
Unregister-EC2Image -ImageId ami-0123456789example
```

避免未使用資源的成本

取消註冊時AMI，不會刪除與. AMI 這些資源包括支援的快照，以AMIs及 EBS Amazon S3 中針對執行個體存放AMIs區支援的檔案。當您取消註冊時AMI，也不會終止或停止從AMI.

您將繼續支付儲存快照和檔案的成本，而且任何執行中的執行個體都會產生費用。

為避免產生這些類型的不必要成本，建議您刪除不需要的任何資源。

EBS-支持 AMIs

使用以下任何一種方法刪除與 EBS-backed AMI 相關聯的資源。

Console

若要刪除與您的支援相關聯的EBS資源 AMI

1. 取AMI消註冊.

   記下 AMI ID-這可協助您在下一個步驟中尋找要刪除的快照。
2. 刪除不需要的快照。

   相關聯的識別碼會顯示AMI在「快照」畫面的「說明」欄中。
3. 終止您不需要的執行個體。

AWS CLI

若要刪除與您的支援相關聯的EBS資源 AMI

1. 使用取消註冊影AMI像指令來取消註冊。

```
aws ec2 deregister-image --image-id ami-0123456789example
```

2. 使用刪除快照指令刪除不需要的快照。

```
aws ec2 delete-snapshot --snapshot-id snap-0123456789example
```

3. 通過使用終端實例命令終止不需要的實例。

```
aws ec2 terminate-instances --instance-ids i-0123456789example
```

PowerShell

若要刪除與您的支援相關聯的EBS資源 AMI

1. 使用Unregister-EC2Image指令程式取AMI消註冊。

```
Unregister-EC2Image -ImageId ami-0123456789example
```

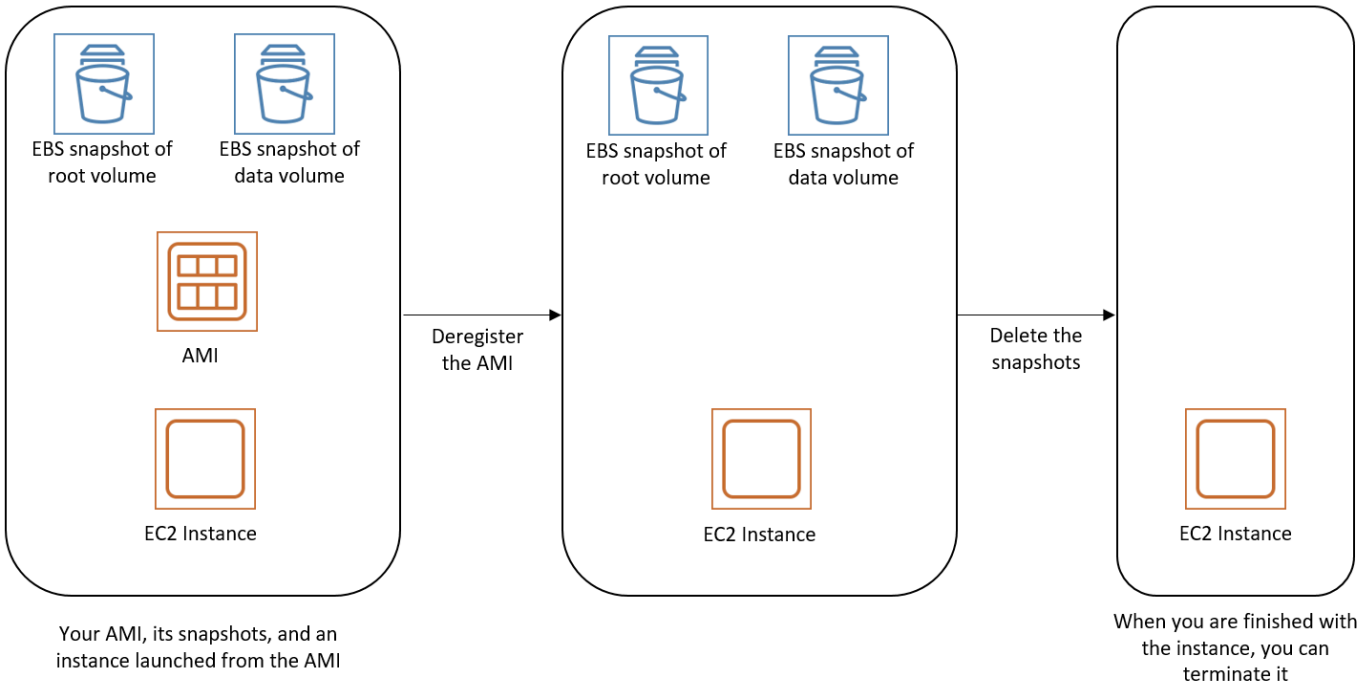
2. 使用Remove-EC2Snapshot指令程式刪除不需要的快照。

```
Remove-EC2Snapshot -SnapshotId snap-0123456789example
```

3. 使用Remove-EC2Instance指令程式終止不需要的執行個體。

```
Remove-EC2Instance -InstanceId i-0123456789example
```

下圖說明刪除與EBS已備份AMI相關資源的流程。



執行個體存放區支援 AMI

使用下列方法刪除與執行個體存放AMI區支援相關聯的資源。

若要刪除與執行個體商店支援相關聯的資源 AMI

1. 使用取消註冊影AMI像指令來取消註冊。

```
aws ec2 deregister-image --image-id ami-0123456789example
```

2. 使用 ec2-delete-bundle (AMI工具) 命令刪除 Amazon S3 中的套裝軟體。

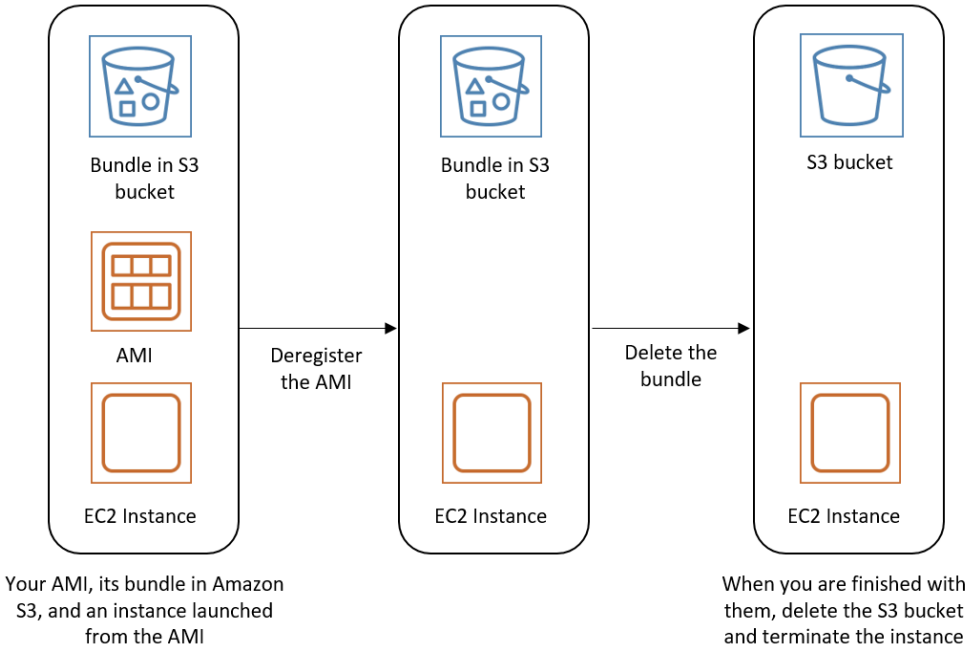
```
ec2-delete-bundle -b amzn-s3-demo-bucket/myami -a your_access_key_id -
s your_secret_access_key -p image
```

3. 通過使用終端實例命令終止不需要的實例。

```
aws ec2 terminate-instances --instance-ids i-0123456789example
```

4. 如果您已完成將服務包上傳到的 Amazon S3 儲存貯體，則可以刪除該儲存貯體。若要刪除 Amazon S3 儲存貯體，請開啟 Amazon S3 主控台，選取儲存貯體，然後依序選取 Actions (動作) 和 Delete (刪除)。

下圖說明刪除與執行個體商店AMI支援相關資源的流程。



保護 Amazon 免受EC2AMI註銷

您可以開啟取消註冊保護，以防AMI止意外或惡意刪除。當您開啟取消註冊保護時，AMI無論使用者的權限為何，都無法取消註冊。IAM如果要取消註冊AMI，必須先關閉其上的取消註冊保護。

當您在上開啟註銷保護時AMI，您可以選擇包含 24 小時的冷卻時間。這個冷卻時間是指停止註冊保護在您關閉後保持有效的時間。在此冷卻時間內，AMI無法取消註冊。當冷卻時間結束時，AMI可以撤銷註冊。

所有現有和新AMIs的預設情況下都會關閉取消註冊保護。

開啟取消註冊保護

使用下列程序開啟取消註冊保護。

Console

若要開啟取消註冊保護 AMI

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇AMIs。

3. 在篩選列中，選擇 [我擁有] 以列出可用的影像AMIs，或選擇 [停用的影像] 以列出已停用的影像AMIs。
4. 選取要開啟取消註冊保護的項目，然後選擇 [動作] > [管理取AMI消註冊保護]。AMI
5. 在「管理取AMI消註冊保護」對話方塊中，您可以開啟取消註冊保護 (無論是否有冷卻時間)。請選擇下列其中一個選項：
   - 啟用 24 小時的冷卻時間 — 使用冷卻時間後AMI，如果關閉註冊保護，則無法在 24 小時內取消註冊。
   - 無冷卻時間啟用 — 在沒有冷卻時間的情況下，AMI可以在停用註冊保護關閉時立即取消註冊。
6. 選擇 Save (儲存)。

AWS CLI

若要開啟取消註冊保護 AMI

使用指enable-image-deregistration-protection令並指定 AMI ID。若要包含可選的 24 小時冷卻時間，請將`--with-cooldown`設定為`true`。若要排除冷卻時間，請省略`--with-cooldown`參數。

```
aws ec2 enable-image-deregistration-protection \
    --image-id ami-0123456789example \
    --with-cooldown true
```

關閉取消註冊保護

請使用下列程序來關閉取消註冊保護。

如果您選擇在啟用的取消註冊保護時包含 24 小時的冷卻時間AMI，則當您關閉註冊保護時，您將無法立即取消註冊。AMI冷卻時間是 24 小時的時間段，即使在您關閉註冊保護之後，註銷保護仍然有效。在此冷卻時間內，AMI無法取消註冊。冷卻時間結束後，AMI可以撤銷註冊。

Console

若要關閉 AMI

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇AMIs。

3. 在篩選列中，選擇 [我擁有] 以列出可用的影像AMIs，或選擇 [停用的影像] 以列出已停用的影像AMIs。
4. 選取AMI以關閉取消註冊保護，然後選擇 [動作]、[管理取AMI消註冊保護]。
5. 在 [管理取AMI消註冊保護] 對話方塊中，選擇 [停用]。
6. 選擇 Save (儲存)。

AWS CLI

若要關閉 AMI

使用指disable-image-deregistration-protection令並指定 AMI ID。

```
aws ec2 disable-image-deregistration-protection --image-id ami-0123456789example
```

使用 Amazon EC2 開機模式執行個體啟動行為

電腦開機時，第一個執行的軟體會負責初始化平台，並提供作業系統執行平台特定作業的介面。

在 Amazon 中EC2，支援兩種開機模式軟體版本：統一可延伸韌體界面 (UEFI) 和舊版BIOS。

可能的啟動模式參數 AMI

可AMI以具有下列其中一個啟動模式參數值：`uefilegacy-bios`、或`uefi-preferred`。AMI啟動模式參數是可選的。對於沒AMIs有開機模式參數的情況下，從這些模式啟動的執行個體會AMIs使用執行個體類型的預設開機模式值。

AMI開機模式參數的用途

AMI啟動模式參數會向 Amazon 發出訊號EC2，啟動執行個體時要使用的啟動模式。當啟動模式參數設定為時`uefi`，會EC2嘗試在上啟動執行個體UEFI。如果作業系統未設定為支援UEFI，則執行個體啟動將不成功。

UEFI偏好的開機模式參數

您可以使AMIs用`uefi-preferred`引導模式參數創建支持UEFI和傳BIOS統。當啟動模式參數設定為`uefi-preferred` (如果執行個體類型支援)UEFI，則會在上啟動執行個體UEFI。如果執行個體類型不支援UEFI，則會在 Legacy 上啟動執行個體BIOS。

> **Warning**
>
> 某些功能 (例如「UEFI安全開機」) 僅適用於開機的執行個體UEFI。當您將`uefi-preferred`AMI啟動模式參數與不支援的執行個體類型搭配使用時UEFI，執行個體會以 Legacy 的身分啟動，而BIOS且會停用UEFI相依功能。如果您依賴UEFI相依功能的可用性，請將您的AMI開機模式參數設定為`uefi`。

執行個體類型的預設開機模式

- 重力彈例證類型：UEFI
- Intel 和AMD執行個體類型：舊版 BIOS

區域支援

UEFI局部區域、Wavelength 區域或 AWS Outposts不支援開機。

開機模式主題

- 在UEFI開機模式下啟動EC2執行個體的需求
- 確定 Amazon 的啟動模式參數 EC2 AMI
- 判斷EC2執行個體類型支援的開機模式
- 判斷EC2執行個體的開機模式
- 判斷EC2執行個體作業系統的開機模式
- 設置 Amazon 的啟動模式 EC2 AMI
- UEFIAmazon EC2 實例的變量
- UEFIAmazon EC2 實例的安全啟動

在UEFI開機模式下啟動EC2執行個體的需求

執行個體的啟動模式取決於執行個體的組態AMI、其中包含的作業系統以及執行個體類型。若要以UEFI開機模式啟動執行個體，您必須符合下列需求。

AMI

必AMI須配置UEFI如下：

- 作業系統 — AMI 必須將包含在中的作業系統設定為使用UEFI；否則，執行個體啟動將會失敗。如需詳細資訊，請參閱判斷EC2執行個體作業系統的開機模式。
- AMI開機模式參數 — AMI 必須將的開機模式參數設定為`uefi`或`uefi-preferred`。如需詳細資訊，請參閱確定 Amazon 的啟動模式參數 EC2 AMI。

下列 Linux AMIs 支援UEFI：

- Amazon Linux 2023
- Amazon Linux 2 (僅限重力子執行個體類型)

對於其他 LinuxAMIs，您必須設定AMI、AMI透過虛擬機器匯入/匯入，或匯入AMI到CloudEndure。

視窗 — 下列視窗AMIs支援UEFI：

- TPM-視窗 _ 伺服器 2022-英文全基
- TPM-視窗 _ 伺服器 2022-英文核心基礎
- TPM-視窗服務器 2019-英語全基
- TPM-視窗伺服器 2019-英文核心基礎
- TPM-視窗伺服器 2016-英文全基
- TPM-視窗伺服器 2016-英文核心基礎

執行個體類型

在 AWS 硝基系統上建置的所有執行個體都支援UEFI和傳統BIOS，但下列項目除外：裸機執行個體、G4ad DL1、P4、u-3tb1、u-6tb1、u-9tb1、u-12tb1、u-18tb1、u-24tb1 和。VT1如需詳細資訊，請參閱the section called “執行個體類型開機模式”。

下表顯示執行處理的啟動模式 (由「產生的執行處理開機模式」資料欄指示) 是由 AMI (欄 1) 的開機模式參數、(欄 2) 所包含之作業系統的開機模式組態，以及執行處理類型 AMI (第 3 欄) 的開機模式支援所決定。

AMI開機模式參數	作業系統開機模式組態	執行個體類型開機模式支援	產生的執行個體開機模式
UEFI	UEFI	UEFI	UEFI
舊版 BIOS	舊版 BIOS	舊版 BIOS	舊版 BIOS

AMI開機模式參數	作業系統開機模式組態	執行個體類型開機模式支援	產生的執行個體開機模式
UEFI偏好	UEFI	UEFI	UEFI
UEFI偏好	UEFI	UEFI和舊版 BIOS	UEFI
UEFI偏好	舊版 BIOS	舊版 BIOS	舊版 BIOS
UEFI偏好	舊版 BIOS	UEFI和舊版 BIOS	舊版 BIOS
未指定開機模式-ARM	UEFI	UEFI	UEFI
未指定開機模式 - x86	舊版 BIOS	UEFI和舊版 BIOS	舊版 BIOS

確定 Amazon 的啟動模式參數 EC2 AMI

AMI啟動模式參數是可選的。可AMI以具有下列其中一個啟動模式參數值：`uefilegacy-bios`、或`uefi-preferred`。

有些AMIs沒有啟動模式參數。如果沒AMI有開機模式參數，從啟動的執行個體會AMI使用執行個體類型的預設值 (位`uefi`於 Graviton 上) 以及在 Intel 和AMD執行個體類型`legacy-bios`上。

Console

確定AMI（控制台）的啟動模式參數

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 AMIs，然後選取AMI。
3. 檢查開機模式欄位。
   - uefi 的值表示AMI支援UEFI。
   - UEFI 優先選項的值表示AMI支持和UEFI傳統。BIOS
   - 如果沒有值，從啟動的執行個體會AMI使用例證類型的預設值。

若要判斷啟動執行個體AMI時的開機模式參數 (主控台)

使用啟動執行個體精靈啟動執行個AMI體時，請在選取步驟中檢查 [開機模式] 欄位。如需詳細資訊，請參閱應用程式和作業系統映像 (Amazon Machine Image)。

AWS CLI

若要判斷 AMI (AWS CLI) 的開機模式參數

使用此describe-images作業決定AMI.

```
aws ec2 describe-images --region us-east-1 --image-id ami-0abcdef1234567890

{
    "Images": [
        {
         ...
            ],
            "EnaSupport": true,
            "Hypervisor": "xen",
            "ImageOwnerAlias": "amazon",
            "Name": "UEFI_Boot_Mode_Enabled-Windows_Server-2016-English-Full-
Base-2020.09.30",
            "RootDeviceName": "/dev/sda1",
            "RootDeviceType": "ebs",
            "SriovNetSupport": "simple",
            "VirtualizationType": "hvm",
            "BootMode":
 "uefi"
        }
    ]
}
```

在輸出中，`BootMode`欄位指示的啟動模式AMI。的值`uefi`表示AMI支撐UEFI。的值`uefi-preferred`表示AMI支援UEFI和舊式兩者BIOS。如果沒有值，從啟動的執行個體會AMI使用例證類型的預設值。

PowerShell

要確定的啟動模式參數AMI（工具 PowerShell）

使用Get-EC2Image指令程式來判斷. AMI

```
PS C:\> Get-EC2Image -Region us-east-1 -ImageId ami-0abcdef1234567890 | Format-List
 Name, BootMode, TpmSupport

Name        : TPM-Windows_Server-2016-English-Full-Base-2023.05.10
BootMode    : uefi
```

```
TpmSupport : v2.0
```

在輸出中，`BootMode`欄位指示的啟動模式AMI。的值`uefi`表示AMI支撐UEFI。的值`uefi-preferred`表示AMI支援UEFI和舊式兩者BIOS。如果沒有值，從啟動的執行個體會AMI使用例證類型的預設值。

判斷EC2執行個體類型支援的開機模式

您可以使用 AWS CLI 或 [工具] PowerShell 來判斷執行個體類型支援的開機模式。

確定執行個體類型的支援開機模式

您可以使用下列方法來確定執行個體類型的支援開機模式。

AWS CLI

使用 describe-instance-types 命令確定執行個體類型的支援開機模式。`--query`參數會篩選輸出，以僅傳回支援的開機模式。

下列範例顯示同時`m5.2xlarge`支援UEFI和舊版BIOS啟動模式。

```
aws ec2 describe-instance-types --region us-east-1 --instance-types m5.2xlarge --
query "InstanceTypes[*].SupportedBootModes"
```

下列為範例輸出。

```
[
    [
        "legacy-bios",
        "uefi"
    ]
]
```

下列範例顯示僅`t2.xlarge`支援 Legacy BIOS。

```
aws ec2 describe-instance-types --region us-east-1 --instance-types t2.xlarge --
query "InstanceTypes[*].SupportedBootModes"
```

下列為範例輸出。

```
[
    [
        "legacy-bios"
    ]
]
```

PowerShell

使用 Get-EC2InstanceType(工具 PowerShell) 指令程式來判斷執行個體類型支援的開機模式。

下列範例顯示同時`m5.2xlarge`支援UEFI和舊版BIOS啟動模式。

```
Get-EC2InstanceType -Region us-east-1 -InstanceType m5.2xlarge | Format-List
 InstanceType, SupportedBootModes
```

下列為範例輸出。

```
InstanceType       : m5.2xlarge
SupportedBootModes : {legacy-bios, uefi}
```

下列範例顯示僅`t2.xlarge`支援 Legacy BIOS。

```
Get-EC2InstanceType -Region us-east-1 -InstanceType t2.xlarge | Format-List
 InstanceType, SupportedBootModes
```

下列為範例輸出。

```
InstanceType       : t2.xlarge
SupportedBootModes : {legacy-bios}
```

判斷支援的執行個體類型 UEFI

您可以使用以下方法來確定支持的實例類型UEFI;

AWS CLI

可用的執行個體類型因 AWS 區域而異。若要查看區域UEFI中支援的可用例證類型，請將describe-instance-types指令與`--region`參數搭配使用。如果您省略`--region`參數，則會在請求中使用您設定的預設「地區」(Region)。包括`--filters`參數，將結果範圍定為支援的執行個體類型，以UEFI及將輸出範圍限定為的`--query`參數`InstanceType`。

```
aws ec2 describe-instance-types --filters Name=supported-boot-mode,Values=uefi --
query "InstanceTypes[*].[InstanceType]" --output text | sort
```

下列為範例輸出。

```
a1.2xlarge
a1.4xlarge
a1.large
a1.medium
a1.metal
a1.xlarge
c5.12xlarge
...
```

PowerShell

```
PS C:\> Get-EC2InstanceType | `
 Where-Object {$_.SupportedBootModes -Contains "uefi"} | `
 Sort-Object InstanceType | `
 Format-Table InstanceType -GroupBy CurrentGeneration
```

下列為範例輸出。

```
CurrentGeneration: False

InstanceType
------------
a1.2xlarge
a1.4xlarge
a1.large
a1.medium
a1.metal
a1.xlarge

CurrentGeneration: True

InstanceType
------------
c5.12xlarge
c5.18xlarge
c5.24xlarge
```

```
c5.2xlarge
c5.4xlarge
c5.9xlarge
...
```

判斷支援UEFI安全開機並保留非揮發性變數的執行個體類型

裸機執行個體不支援UEFI安全開機和非揮發性變數，因此這些範例會將它們從輸出中排除。如需有關UEFI安全開機的資訊，請參閱UEFIAmazon EC2 實例的安全啟動。

AWS CLI

使用指describe-instance-types令，並透過包括`Name=bare-metal,Values=false`篩選器，將裸機例證從輸出中排除。

```
aws ec2 describe-instance-types --filters Name=supported-boot-mode,Values=uefi
 Name=bare-metal,Values=false --query "InstanceTypes[*].[InstanceType]" --output
 text | sort
```

下列為範例輸出。

```
a1.2xlarge
a1.4xlarge
a1.large
a1.medium
...
```

PowerShell

```
PS C:\> Get-EC2InstanceType | `
    Where-Object { `
        $_.SupportedBootModes -Contains "uefi" -and `
        $_.BareMetal -eq $False
        } | `
    Sort-Object InstanceType  | `
    Format-Table InstanceType, SupportedBootModes, BareMetal,
 @{Name="SupportedArchitectures";
 Expression={$_.ProcessorInfo.SupportedArchitectures}}
```

```
InstanceType     SupportedBootModes  BareMetal SupportedArchitectures
```

```
------------     ------------------  --------- ----------------------
a1.2xlarge       {uefi}                  False arm64
a1.4xlarge       {uefi}                  False arm64
a1.large         {uefi}                  False arm64
a1.medium        {uefi}                  False arm64
a1.xlarge        {uefi}                  False arm64
c5.12xlarge      {legacy-bios, uefi}     False x86_64
c5.18xlarge      {legacy-bios, uefi}     False x86_64
```

判斷EC2執行個體的開機模式

執行個體的開機模式會顯示在 Amazon EC2 主控台的「開機模式」欄位中，並顯示在中的`currentInstanceBootMode`參數 AWS CLI。

啟動執行個體時，其啟動模式參數的值由AMI用來啟動它的 boot mode 參數值決定，如下所示：

- AMI具有的啟動模式參數`uefi`會建立參`currentInstanceBootMode`數為的執行個體`uefi`。
- AMI具有的啟動模式參數`legacy-bios`會建立參`currentInstanceBootMode`數為的執行個體`legacy-bios`。
- `uefi`如果執行個AMI體類型支援，則使用的啟動模式`currentInstanceBootMode`參數為`uefi-preferred`建立執行個體UEFI；否則，會建立`currentInstanceBootMode`參數為的執行個體`legacy-bios`。
- 如果沒AMI有啟動模式參數值，則會建立具有`currentInstanceBootMode`參數值的執行個體，該參數值取決於AMI架構是 ARM x86 以及執行個體類型支援的啟動模式。預設的開機模式為`uefi`引力執行個體類型，以及 Intel 和`legacy-bios`AMD執行個體類型。

Console

確定執行個體的開機模式 (主控台)

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選取 Instances (執行個體)，然後選取您的執行個體。
3. 在 Details (詳細資訊) 標籤上，檢查 Boot mode (開機模式) 欄位。

AWS CLI

確定執行個體的開機模式 (AWS CLI)

使用 describe-instances 命令確定執行個體的開機模式。您也可以決定用來建立執行個體的開機模式。AMI

```
aws ec2 describe-instances --region us-east-1 --instance-ids i-1234567890abcdef0

{
    "Reservations": [
        {
            "Groups": [],
            "Instances": [
                {
                    "AmiLaunchIndex": 0,
                    "ImageId": "ami-0e2063e7f6dc3bee8",
                    "InstanceId": "i-1234567890abcdef0",
                    "InstanceType": "m5.2xlarge",
                    ...
                    },
                   "BootMode": "uefi",
                   "CurrentInstanceBootMode": "uefi"
                }
            ],
            "OwnerId": "1234567890",
            "ReservationId": "r-1234567890abcdef0"
        }
    ]
}
```

PowerShell

判斷執行個體的開機模式 (適用於的工具 PowerShell)

請使用 Get-EC2Image Cmdlet 來確定執行個體的開機模式。您也可以決定用來建立執行個體的開機模式。AMI

Get-EC2Image (AWS Tools for Windows PowerShell)

```
(Get-EC2Instance -InstanceId i-1234567890abcdef0).Instances | Format-List BootMode,
 CurrentInstanceBootMode, InstanceType, ImageId

BootMode                : uefi
CurrentInstanceBootMode : uefi
InstanceType            : c5a.large
ImageId                 : ami-0265446f88eb4021b
```

在輸出中，以下參數描述開機模式：

- `BootMode`— 用來建立執行個體的啟動模式。AMI
- `CurrentInstanceBootMode` - 用於在啟動或開始時讓執行個體開機的開機模式。

判斷EC2執行個體作業系統的開機模式

的啟動模式指AMI導 Amazon EC2 使用哪種啟動模式來啟動實例。若要檢視執行個體的作業系統是否已設定UEFI，您必須使用 SSH (Linux 執行個體) 或 RDP (Windows 執行個體) 連線至執行個體。

請遵循於執行個體作業系統的指示。

Linux

確定執行個體作業系統的開機模式

1. 使用 Connect 到您的 Linux 執行個體SSH。
2. 若要檢視作業系統的開機模式，請嘗試下列其中一個動作：

   - 執行下列命令。

     ```
     [ec2-user ~]$ sudo /usr/sbin/efibootmgr
     ```

     在UEFI開機模式下啟動的執行個體的預期輸出

     ```
     BootCurrent: 0001
     Timeout: 0 seconds
     BootOrder: 0000,0001
     Boot0000* UiApp
     Boot0001* UEFI Amazon Elastic Block Store vol-xyz
     ```

   - 執行下列命令，以驗證 `/sys/firmware/efi` 目錄是否存在。只有在執行處理使用啟動時，此目錄才會存在UEFI。如果此目錄不存在，命令會傳回 `Legacy BIOS Boot Detected`。

     ```
     [ec2-user ~]$ [ -d /sys/firmware/efi ] && echo "UEFI Boot Detected" || echo
      "Legacy BIOS Boot Detected"
     ```

     在UEFI開機模式下啟動的執行個體的預期輸出

```
UEFI Boot Detected
```

在舊版BIOS開機模式下啟動的執行個體的預期輸出

```
Legacy BIOS Boot Detected
```

- 執行下列命令以確認是否EFI出現在dmesg輸出中。

```
[ec2-user ~]$ dmesg | grep -i "EFI"
```

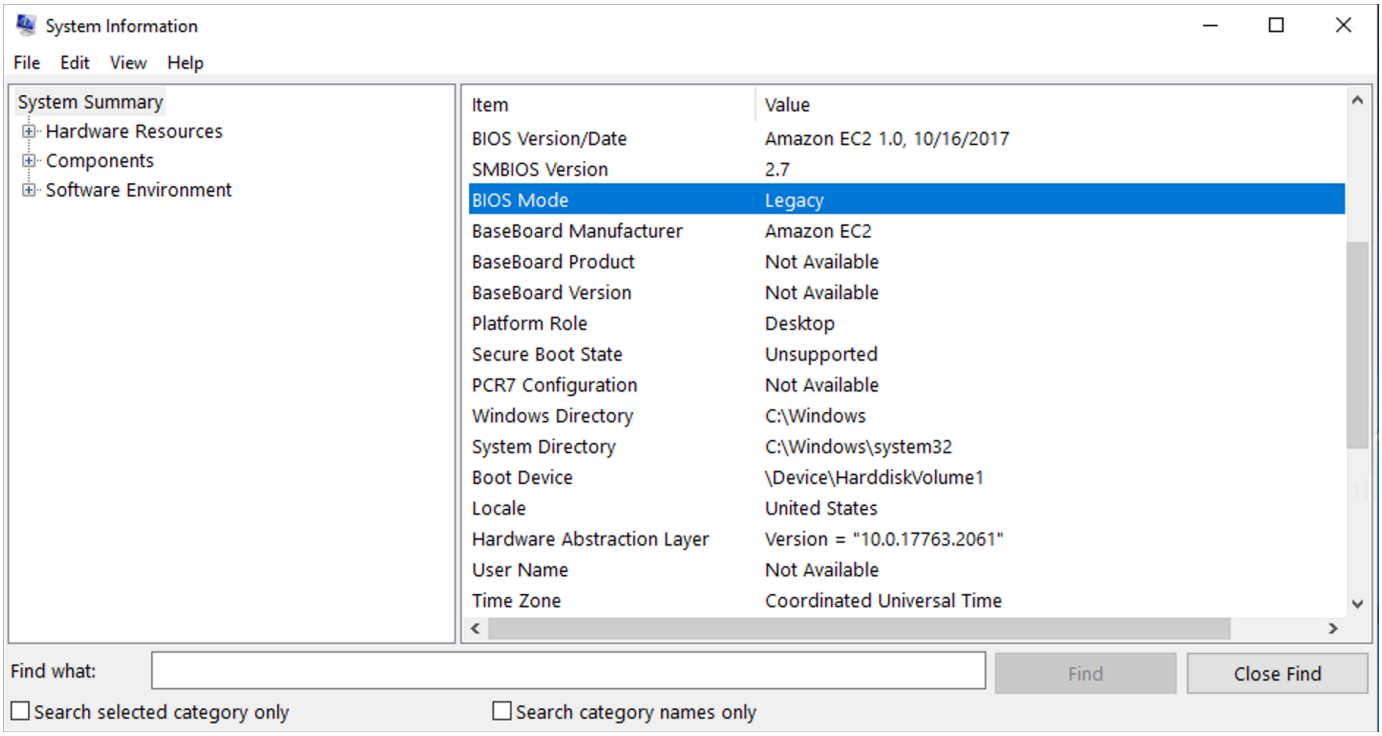
在UEFI開機模式下啟動的執行個體的預期輸出

```
[    0.000000] efi: Getting EFI parameters from FDT:
[    0.000000] efi: EFI v2.70 by EDK II
```

Windows

確定執行個體作業系統的開機模式

1. 使用 Connect 至您的 Windows 執行個體RDP。
2. 轉到「系統信息」，然後檢查「BIOS模式」行。



設置 Amazon 的啟動模式 EC2 AMI

當您AMI使用register-image指令建立時，您可以AMI將的啟動模式設定為`uefi`、`legacy-bios`、或`uefi-preferred`。

當AMI啟動模式設定為時`uefi-preferred`，執行個體會以下列方式啟動：

- 對於同時支援UEFI和 Legacy BIOS (例如`m5.large`) 的執行個體類型，執行個體會使用UEFI.
- 對於僅支援 Legacy BIOS (例如`m4.large`) 的執行個體類型，執行個體會使用 Legacy 啟動BIOS。

> **Note**
>
> 如果將AMI開機模式設定為`uefi-preferred`，作業系統必須支援啟動UEFI和 Legacy 的功能BIOS。
>
> 目前，您無法使用該register-image命令來創建同時支持 Nitro TPM 和UEFI首選的命令。AMI

> **Warning**
>
> 某些功能 (例如「UEFI安全開機」) 僅適用於開機的執行個體UEFI。當您將`uefi-preferred`AMI啟動模式參數與不支援的執行個體類型搭配使用時UEFI，執行個體會以 Legacy 的身分啟動，而BIOS且會停用UEFI相依功能。如果您依賴UEFI相依功能的可用性，請將您的AMI開機模式參數設定為`uefi`。

若要將現有 Legacy BIOS 型執行個體轉換為 LegacyUEFI，或將現有的執行個體轉換UEFI為 LegacyBIOS，您需要執行以下幾個步驟：首先，修改執行個體的磁碟區和作業系統，以支援選取的開機模式。然後，建立磁碟區的快照。最後，使register-image用建立AMI使用快照。

您無法AMI使用create-image指令設定的開機模式。使用 create-image，AMI會繼承用於EC2建立. AMI 例如，如果您AMI從在 Legacy 上執行的EC2執行個體建立BIOS，則AMI開機模式會設定為`legacy-bios`。如果您AMI從使用啟動模式設為啟動的EC2執行個AMI體建立執行個體`uefi-preferred`，則建立的AMI執行個體也會將其開機模式設定為`uefi-preferred`。

> **Warning**
>
> 設定AMI開機模式參數並不會自動為指定的開機模式設定作業系統。在繼續執行這些步驟之前，您必須先對執行個體的磁碟區和作業系統進行適當的修改，以支援使用選取的開機模式進行開機；否則，產生的結果AMI將無法使用。例如，如果您要將舊版 BIOS Windows 執行個體轉換為UEFI，您可以使用 Microsoft 提供的MBR2GPT工具將系統磁碟從轉換MBR為GPT。所需的修改視作業系統而異。如需詳細資訊，請參閱作業系統的手冊。

若要設定 AMI (AWS CLI) 的開機模式

1. 對執行個體的磁碟區和作業系統進行適當的修改，以支援透過選取的開機模式開機。所需的修改視作業系統而異。如需詳細資訊，請參閱作業系統的手冊。

   > **Note**
   >
   > 如果您不執行此步驟，AMI將無法使用。

2. 若要尋找執行個體的磁碟區 ID，請使用 describe-instances 命令。您將在下一個步驟中建立此磁碟區的快照。

```
aws ec2 describe-instances --region us-east-1 --instance-ids i-1234567890abcdef0
```

預期的輸出結果

```
...
            "BlockDeviceMappings": [
                        {
                            "DeviceName": "/dev/sda1",
                            "Ebs": {
                                "AttachTime": "",
                                "DeleteOnTermination": true,
                                "Status": "attached",
                                "VolumeId": "vol-1234567890abcdef0"
                            }
                        }
                        ...
```

3. 若要建立磁碟區的快照，請使用 create-snapshot 命令。使用上一個步驟的磁碟區 ID。

```
aws ec2 create-snapshot --region us-east-1 --volume-id vol-1234567890abcdef0 --
description "add text"
```

預期的輸出結果

```
{
 "Description": "add text",
 "Encrypted": false,
 "OwnerId": "123",
 "Progress": "",
 "SnapshotId": "snap-01234567890abcdef",
 "StartTime": "",
 "State": "pending",
 "VolumeId": "vol-1234567890abcdef0",
 "VolumeSize": 30,
 "Tags": []
}
```

4. 請注意上一個步驟輸出中的快照 ID。
5. 等到快照建立completed後，再進行下一個步驟。若要查詢快照的狀態，請使用 describe-snapshots 命令。

```
aws ec2 describe-snapshots --region us-east-1 --snapshot-ids snap-01234567890abcdef
```

範例輸出

```
{
    "Snapshots": [
        {
            "Description": "This is my snapshot",
            "Encrypted": false,
            "VolumeId": "vol-049df61146c4d7901",
            "State": "completed",
            "VolumeSize": 8,
            "StartTime": "2019-02-28T21:28:32.000Z",
            "Progress": "100%",
            "OwnerId": "012345678910",
            "SnapshotId": "snap-01234567890abcdef",
...
```

6. 若要建立新的AMI，請使用register-image指令。使用您在之前步驟中記下的快照 ID。

   - 若要將啟動模式設定為UEFI，請將`--boot-mode`參數增加至指令並指定`uefi`為值。

```
aws ec2 register-image \
   --region us-east-1 \
   --description "add description" \
   --name "add name" \
   --block-device-mappings "DeviceName=/dev/
sda1,Ebs={SnapshotId=snap-01234567890abcdef,DeleteOnTermination=true}" \
   --architecture x86_64 \
   --root-device-name /dev/sda1 \
   --virtualization-type hvm \
   --ena-support \
   --boot-mode uefi
```

   - 若要將開機模式設定為 `uefi-preferred`，請將 `--boot-mode` 參數新增至命令並將 `uefi-preferred` 指定為值。

```
aws ec2 register-image \
   --region us-east-1 \
   --description "add description" \
   --name "add name" \
```

```
   --block-device-mappings "DeviceName=/dev/
sda1,Ebs={SnapshotId=snap-01234567890abcdef,DeleteOnTermination=true}" \
   --architecture x86_64 \
   --root-device-name /dev/sda1 \
   --virtualization-type hvm \
   --ena-support \
   --boot-mode uefi-preferred
```

預期的輸出結果

```
{
"ImageId": "ami-new_ami_123"
}
```

7. 若要確認新建立的啟動模式是否AMI具有您在上一個步驟中指定的開機模式，請使用指describe-images令。

```
aws ec2 describe-images --region us-east-1 --image-id ami-new_ami_123
```

預期的輸出結果

```
{
  "Images": [
   {
   "Architecture": "x86_64",
   "CreationDate": "2021-01-06T14:31:04.000Z",
   "ImageId": "ami-new_ami_123",
   "ImageLocation": "",
   ...
   "BootMode": "uefi"
   }
   ]
}
```

8. 使用新建立的執行個體啟動新AMI執行個體。

   如果AMI啟動模式為`uefi`或`legacy-bios`，則從此模式建立的執行個體AMI將具有與AMI. 如果AMI啟動模式為`uefi-preferred`，則執行個體將使用UEFI執行個體類型支援的情況下啟動UEFI；否則，執行個體將會使用 Legacy 啟動BIOS。

9. 若要驗證新執行個體是否具有預期的開機模式，請使用 describe-instances 命令。

UEFIAmazon EC2 實例的變量

當您啟動啟動模式設定為的執行個體時UEFI，會建立變數的索引鍵值存放區。該存儲可以通過UEFI和實例操作系統用於存儲UEFI變量。

UEFI開機載入程式和作業系統會使用變數來設定早期系統啟動。它們可讓作業系統管理開機程序的特定設定，例如開機順序，或管理UEFI安全開機的金鑰。

> **Warning**
>
> 任何可以連線至執行個體的使用者 (以及可能在執行個體上執行的任何軟體)，或任何有權在執行個體GetInstanceUefiDataAPI上使用的人都可以讀取變數。您絕對不應在UEFI變數儲存區中儲存敏感資料，例如密碼或個人識別資訊。

UEFI變量持久性

- 對於在 2022 年 5 月 10 日或之前啟動的執行個體，在重新啟動或停止時會清除UEFI變數。
- 對於在 2022 年 5 月 11 日或之後啟動的執行個體，標記為非揮發性的UEFI變數會在重新啟動和停止/啟動時保留。
- 裸機實例不會在實例停止/啟動操作中保留UEFI非易失性變量。

UEFIAmazon EC2 實例的安全啟動

UEFI安全開機 defense-in-depth 是建立在 Amazon 長期安全開機程序的基礎上EC2，並提供其他功能，協助客戶保護軟體免於在重新開機期間持續存在的威脅。它確保執行個體僅對使用密碼編譯金鑰簽署的軟體進行開機。這些密鑰存儲在UEFI非易失性變量存儲的密鑰數據庫中。UEFI安全開機可防止未經授權修改執行個體開機流程。

目錄

- UEFI安全開機如何搭配 Amazon EC2 執行個體運作
- 啟動具有UEFI安全開機支援的 Amazon EC2 執行個體
- 確認 Amazon EC2 執行個體是否已啟用UEFI安全開機
- AMI使用自訂UEFI安全開機金鑰建立 Linux
- 為UEFI安全啟動創建 AWS 二進制 blob

UEFI安全開機如何搭配 Amazon EC2 執行個體運作

UEFI安全開機是中指定的一項功能UEFI，可提供有關開機鏈狀態的驗證。它的設計是為了確保在韌體的自我初始化之後，只會執行經過密碼編譯驗證的UEFI二進位檔案。這些二進製文件包括UEFI驅動程序和主引導加載程序，以及鏈接加載的組件。

UEFI安全啟動指定四個關鍵數據庫，這些數據庫在信任鏈中使用。數據庫存儲在UEFI變量存儲。

信任鏈如下：

平台金鑰 (PK) 資料庫

PK 資料庫是信任根。它包含用於更新金鑰交換金鑰 (KEK) 資料庫的信任鏈中的單一公開 PK 金鑰。

若要變更 PK 資料庫，您必須具有私有 PK 金鑰才能簽署更新要求。這包括透過寫入空 PK 金鑰來刪除 PK 資料庫。

金鑰交換金鑰 (KEK) 資料庫

該KEK數據庫是在信任鏈中用於更新簽名（db）和 denylist（dbx）數據庫的公KEK鑰列表。

若要變更公用KEK資料庫，您必須擁有私密 PK 金鑰才能簽署更新要求。

簽章 (db) 資料庫

db 資料庫是在信任鏈中用來驗證所有UEFI開機二進位檔案的公開金鑰和雜湊清單。

若要變更資料庫，您必須擁有私密 PK 金鑰或任何私密KEK金鑰才能簽署更新要求。

簽章拒絕清單 (dbx) 資料庫

dbx 資料庫是一份不受信任的公有金鑰和二進位檔案雜湊清單，在信任鏈中作為撤銷檔案使用。

dbx 資料庫始終優先於所有其他金鑰資料庫。

若要變更 dbx 資料庫，您必須擁有私密 PK 金鑰或任何私密KEK金鑰，才能簽署更新要求。

UEFI論壇在 https://uefi.org/revocationlistfile 上為許多已知的不良二進製文件和證書維護了一個公開可用的 dbx。

> **Important**
>
> UEFI安全開機會對任何UEFI二進位檔案強制執行簽章驗證。要允許在UEFI安全啟動中執行UEFI二進製文件，請使用上述任何私有數據庫密鑰對其進行簽名。

預設情況下，「UEFI安全開機」為停用狀態，且系統處於中狀態`SetupMode`。當系統處於`SetupMode` 模式時，所有金鑰變數都可以在沒有密碼編譯簽章的情況下更新。設定 PK 時，會啟用「UEFI安全開機」SetupMode 並結束。

啟動具有UEFI安全開機支援的 Amazon EC2 執行個體

當您使用受支援AMI和受支援的EC2執行個體類型啟動 Amazon 執行個體時，該執行個體將會根據其UEFI安全開機資料庫自動驗證UEFI啟動二進位檔案。不需任何其他設定。您也可以在啟動後在執行個體上設定UEFI安全開機。

> **Note**
>
> UEFI安全開機可保護您的執行個體及其作業系統，防止開機流程修改。如果您AMI從已啟用「UEFI安全開機」AMI 的來源建立新的，並在複製程序期間修改某些參數，例如在`UefiData`中變更AMI，您可以停用「UEFI安全開機」。

目錄

- 支援 AMIs
- 支援的執行個體類型

支援 AMIs

Linux AMIs

若要啟動 Linux 執行個體，Linux AMI 必須啟用UEFI安全開機。

從 AL2 023 版開始，Amazon Linux 支援UEFI安全開機。但是，默認情況下未啟用UEFI安全啟動 AMIs。如需詳細資訊，請參閱《AL2023 使用者指南》中的 < UEFI安全開機 > 。舊版 Amazon Linux AMIs 未啟用UEFI安全開機功能。若要使用支援的AMI，您必須在自己的 Linux 上執行許多設定步驟 AMI。如需詳細資訊，請參閱AMI使用自訂UEFI安全開機金鑰建立 Linux。

視窗 AMIs

若要啟動 Windows 執行個體，視窗AMI必須啟用UEFI安全開機。下列視窗AMIs已預先設定為使用 Microsoft 金鑰啟用UEFI安全開機：

- TPM-視窗 _ 伺服器 2022-英文核心基礎
- TPM-視窗 _ 伺服器 2022-英文全基
- TPM-視窗 _ 服務器 2022-英語-全版-企業版 SQL
- TPM-視窗 _ 伺服器 2022-英文全版-標準版 SQL
- TPM-視窗伺服器 2019-英文核心基礎
- TPM-視窗服務器 2019-英語全基
- TPM-視窗 _ 伺服器 -2019-英文版-完整版 _ 企業版 SQL
- TPM-視窗伺服器 2019-英文版-完整版 SQL
- TPM-視窗伺服器 2016-英文核心基礎
- TPM-視窗伺服器 2016-英文全基

目前，我們不支援使用import-image指令匯入具有UEFI安全開機功能的 Windows。

支援的執行個體類型

所有支援UEFIUEFI安全開機的虛擬化執行個體類型。如需支援UEFI安全開機的執行個體類型，請參閱UEFI開機模式的需求。

> **Note**
>
> 裸機執行個體類型不支援UEFI安全開機。

確認 Amazon EC2 執行個體是否已啟用UEFI安全開機

您可以使用下列程序來判斷 Amazon EC2 是否已啟用UEFI安全開機。

Linux 執行個體

您可以使用此公用`mokutil`程式來驗證 Linux 執行個體是否已啟用UEFI安全開機。如果執行個體上沒有安裝 `mokutil`，則必須安裝。如需 Amazon Linux 2 的安裝說明，請參閱在 Amazon Linux 2 執行個體上尋找和安裝軟體套件。如需其他 Linux 發行版，請參閱其特定文件。

確認 Linux 執行個體是否已啟用UEFI安全開機

Connect 至您的執行個體，並如同`root`在終端機視窗中執行下列指令。

```
mokutil --sb-state
```

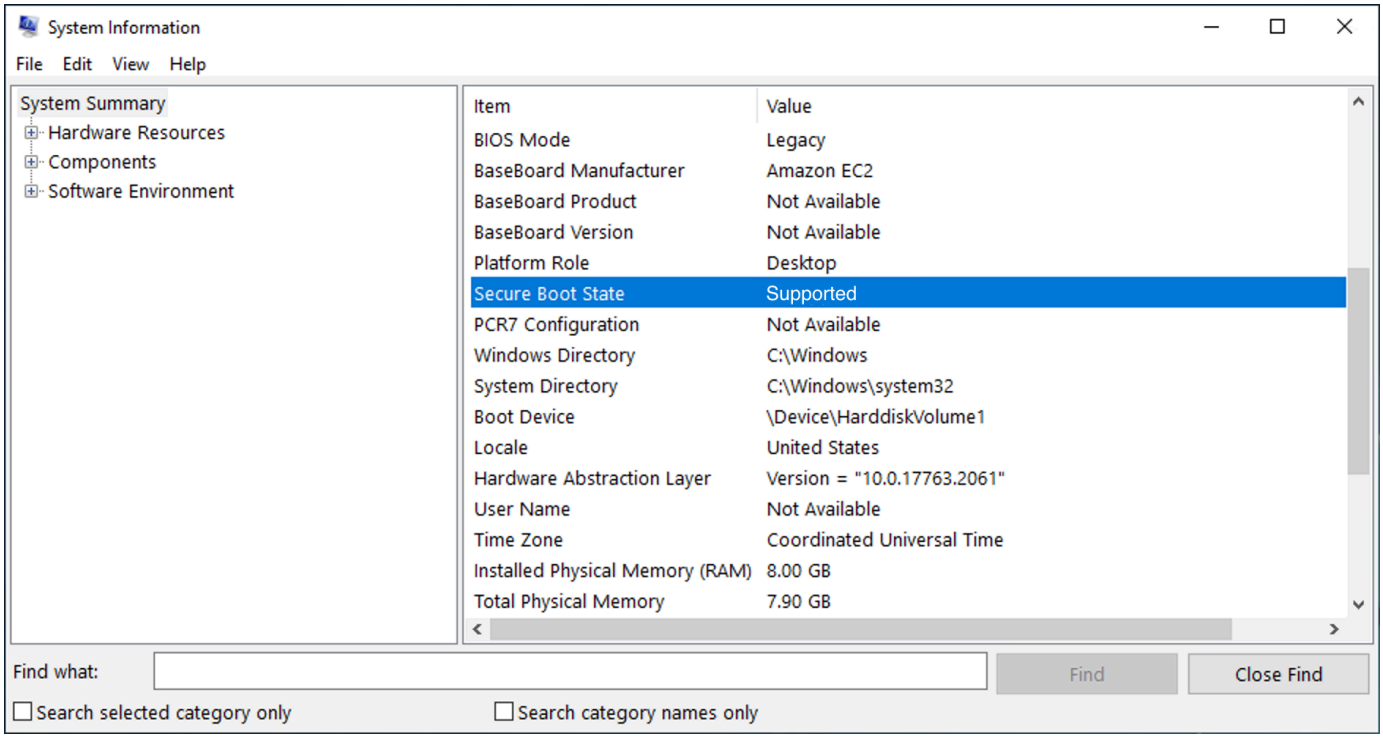
下列為範例輸出。

- 如果啟用「UEFI安全開機」，則輸出會包含`SecureBoot enabled`。
- 如果未啟用UEFI安全開機，則輸出會包含`SecureBoot disabled`或`Failed to read SecureBoot`。

Windows 執行個體

確認 Windows 執行個體是否已啟用UEFI安全開機

1. 連線到您的執行個體。
2. 開啟 msinfo32 工具。
3. 檢查 Secure Boot State (安全開機狀態) 欄位。如果啟用「UEFI安全開機」，則會支援此值，如下圖所示。



您也可以使用 Windows 指 PowerShell 令程式`Confirm-SecureBootUEFI`來檢查安全開機狀態。如需有關指令程式的詳細資訊，請參閱 Microsoft 文件SecureBootUEFI中的確認。

AMI使用自訂UEFI安全開機金鑰建立 Linux

此程序說明如何建立AMI具有UEFI安全開機和自訂私密金鑰的 Linux。從 AL2 023 版開始，Amazon Linux 支援UEFI安全開機。如需詳細資訊，請參閱《AL2023 使用者指南》中的 < UEFI安全開機 >。

> **Important**
>
> 下列程序僅適用於進階使用者。您必須具備足夠的 SSL Linux 發行版開機流程知識，才能使用這些程序。

必要條件

- 將會使用下列工具：
  - 開放 SSL — https://www.openssl.org/
  - 埃菲瓦爾 — 埃菲瓦爾 https://github.com/rhboot/
  - efitools - https://git.kernel.org/pub/scm/linux/kernel/git/jejb/efitools.git/
  - get-instance-uefi-data AWS CLI 命令
- 您的 Linux 執行個體必須使用支援UEFI開機模式AMI的 Linux 啟動，並且存在非揮發性資料。

不含UEFI安全開機金鑰的新建立執行個體會在中建立`SetupMode`，讓您註冊自己的金鑰。有些預先AMIs配置了UEFI安全啟動，您無法更改現有密鑰。如果要更改密鑰，則必須AMI根據原始密鑰創建一個新密鑰AMI。

您有兩種在變數存放區傳播金鑰的方法，這些方法在接下來的選項 A 和選項 B 中說明。選項 A 說明如何在執行個體中執行此操作，並模仿真實硬體的流程。選項 B 說明如何建立二進位 blob，然後在您建立. AMI 對於這兩個選項，您必須先建立用於信任鏈的三對金鑰對。

步驟 1：建立三個金鑰配對

UEFISecure Boot 是以下三個用於信任鏈的關鍵資料庫為基礎：平台金鑰 (PK)、金鑰交換金鑰 (KEK) 和簽章資料庫 (db) [1]

您需在執行個體上建立每個金鑰。若要準備適用於 UEFI Secure Boot 標準的格式的公開金鑰，您必須為每個金鑰建立憑證。 DER定義格SSL式（格式的二進制編碼）。然後，您可以將每個憑證轉換為UEFI簽名清單，這是 UEFI Secure Boot 所瞭解的二進位格式。最後，您需使用相關金鑰簽署每個憑證。

主題

- 準備建立金鑰對
- 金鑰對 1：建立平台金鑰 (PK)
- 金鑰配對 2：建立金鑰交換金鑰 (KEK)
- 金鑰對 3：建立簽章資料庫 (db)
- 使用私有金鑰簽署開機映像 (核心)

準備建立金鑰對

在建立金鑰配對之前，請先建立要用於金鑰產生的全域唯一識別碼 (GUID)。

1. 連線到執行個體。
2. 在 shell 提示中執行下列命令。

```
uuidgen --random > GUID.txt
```

金鑰對 1：建立平台金鑰 (PK)

PK 是UEFI安全開機執行個體的信任根。私有 PK 用於更新KEK，這反過來可用於將授權密鑰添加到簽名數據庫（db）。

X.509 標準用於建立金鑰對。如需有關標準的資訊，請參閱維基百科上的 X.509。

建立 PK

1. 建立金鑰。您必須將變數命名為 PK。

```
openssl req -newkey rsa:4096 -nodes -keyout PK.key -new -x509 -sha256 -days 3650 -
subj "/CN=Platform key/" -out PK.crt
```

   指定了下列參數：

   - `-keyout PK.key` - 私有金鑰檔案。
   - `-days 3650` - 憑證有效的天數。
   - `-out PK.crt`— 用來建立UEFI變數的憑證。
   - CN=*Platform key* - 金鑰的通用名稱 (CN)。您可以輸入自己組織的名稱，而不是 *Platform key*.

2. 建立憑證。

```
openssl x509 -outform DER -in PK.crt -out PK.cer
```

3. 將憑證轉換為UEFI簽名清單。

```
cert-to-efi-sig-list -g "$(< GUID.txt)" PK.crt PK.esl
```

4. 使用私有 PK（自簽）簽署簽UEFI名列表。

```
sign-efi-sig-list -g "$(< GUID.txt)" -k PK.key -c PK.crt PK PK.esl PK.auth
```

金鑰配對 2：建立金鑰交換金鑰 (KEK)

private 用KEK於將密鑰添加到數據庫中，該數據庫是要在系統上啟動的授權簽名列表。

若要建立 KEK

1. 建立金鑰。

```
openssl req -newkey rsa:4096 -nodes -keyout KEK.key -new -x509 -sha256 -days 3650 -
subj "/CN=Key Exchange Key/" -out KEK.crt
```

2. 建立憑證。

```
openssl x509 -outform DER -in KEK.crt -out KEK.cer
```

3. 將憑證轉換為UEFI簽名清單。

```
cert-to-efi-sig-list -g "$(< GUID.txt)" KEK.crt KEK.esl
```

4. 使用私有 PK 簽署簽章清單。

```
sign-efi-sig-list -g "$(< GUID.txt)" -k PK.key -c PK.crt KEK KEK.esl KEK.auth
```

金鑰對 3：建立簽章資料庫 (db)

db 清單包含授權在系統上開機的授權金鑰。要修改列表，私有KEK是必要的。將使用在此步驟中建立的私有金鑰簽署開機映像。

建立 db

1. 建立金鑰。

```
openssl req -newkey rsa:4096 -nodes -keyout db.key -new -x509 -sha256 -days 3650 -
subj "/CN=Signature Database key/" -out db.crt
```

2. 建立憑證。

```
openssl x509 -outform DER -in db.crt -out db.cer
```

3. 將憑證轉換為UEFI簽名清單。

```
cert-to-efi-sig-list -g "$(< GUID.txt)" db.crt db.esl
```

4. 使用私人簽署簽名清單KEK。

```
sign-efi-sig-list -g "$(< GUID.txt)" -k KEK.key -c KEK.crt db db.esl db.auth
```

使用私有金鑰簽署開機映像 (核心)

若為 Ubuntu 22.04，下列映像需要簽章。

```
/boot/efi/EFI/ubuntu/shimx64.efi
/boot/efi/EFI/ubuntu/mmx64.efi
/boot/efi/EFI/ubuntu/grubx64.efi
/boot/vmlinuz
```

簽署映像

使用下列語法簽署映像。

```
sbsign --key db.key --cert db.crt --output /boot/vmlinuz /boot/vmlinuz
```

> **Note**
>
> 您必須簽署所有新的核心。*/##/vmlinu#* 通常會符號鏈接到上次安裝的內核。

請參閱您的發行版本說明文件，了解您的開機鏈和所需映像。

[1] 感謝 ArchWiki 社區所做的所有工作。用於建立 PK、建立、建立資料庫和簽署映像檔的指令來自建立金鑰，由 ArchWiki 維護團隊和/或 ArchWiki 貢獻者撰寫。KEK

步驟 2（選項 A）：從實例內將密鑰添加到變量存儲

建立三個金鑰對後，您即可以連接到執行個體，並透過完成以下步驟從執行個體中將金鑰新增至變數存放區。

選項 A 步驟：

- 步驟 1：啟動支援UEFI安全開機的執行個體
- 步驟 2：設定執行個體以支援UEFI安全開機
- 步驟 3：AMI從執行個體建立

步驟 1：啟動支援UEFI安全開機的執行個體

當您啟動具有下列先決條件的執行個體時，執行個體就可以設定為支援 UEFI Secure Boot。您只能在啟動時在執行個體上啟用「UEFI安全開機」支援；稍後無法啟用。

必要條件

- AMI— Linux AMI 必須支援UEFI開機模式。若要確認是否AMI支援UEFI開機模式，開AMI機模式參數必須是 uefi。如需詳細資訊，請參閱確定 Amazon 的啟動模式參數 EC2 AMI。

  請注意， AWS 僅提供AMIs設定UEFI為支援以重力為基礎的執行個體類型的 Linux。 AWS 目前不提供支援UEFI開機AMIs模式的 x86_64。您可以AMI將自己的設定為支援所有架構的UEFI開機模式。若要將您自己AMI的設定為支援UEFI開機模式，您必須自行執行許多設定步驟AMI。如需詳細資訊，請參閱設置 Amazon 的啟動模式 EC2 AMI。
- 執行個體類型 — UEFI 也支援UEFI安全開機的所有虛擬化執行個體類型。裸機執行個體類型不支援UEFI安全開機。如需支援UEFI安全開機的執行個體類型，請參閱UEFI開機模式的需求。
- UEFI安全啟動發布後啟動您的實例。只有在 2022 年 5 月 10 日之後啟動（UEFI安全開機發行時）啟動的執行個體才能支援UEFI安全開機。

啟動執行個體之後，您可以檢查UEFI資料是否存在，以確認執行個體是否已準備好支援 UEFI Secure Boot (換句話說，您可以繼續執行步驟 2)。UEFI數據的存在表明非易失性數據是持久的。

驗證執行個體是否準備好執行步驟 2

使用 get-instance-uefi-data 命令並指定執行個體 ID。

```
aws ec2 get-instance-uefi-data --instance-id i-0123456789example
```

如果輸出中有UEFI資料，則執行個體已準備好進行步驟 2。如果輸出為空白，則無法將執行個體設定為支援UEFI安全開機。如果您的執行個體在UEFI安全開機支援可用之前啟動，就會發生這種情況。啟動新的執行個體，然後再試一次。

步驟 2：設定執行個體以支援UEFI安全開機

在執行個體的UEFI變數存放區中註冊金鑰配對

> **Warning**
> 您必須在註冊金鑰之後簽署開機映像，否則您將無法啟動執行個體。

建立已簽署的UEFI簽名清單 (PK、和db) 之後KEK，必須在UEFI韌體中註冊這些簽名清單。

只有在以下情況下才能寫入 PK 變數：

- 如果 SetupMode 變數為 1，表示尚未註冊 PK。透過以下命令來檢查這一點：輸出為 1 或 0。

  ```
  efivar -d -n 8be4df61-93ca-11d2-aa0d-00e098032b8c-SetupMode
  ```

- 新 PK 由現有 PK 的私有金鑰簽署。

在UEFI變數存放區中註冊金鑰

必須在執行個體上執行以下命令。

如果 SetupMode 啟用 (值為1)，則可以在執行個體上執行下列命令來註冊金鑰：

```
[ec2-user ~]$ efi-updatevar -f db.auth db
```

```
[ec2-user ~]$ efi-updatevar -f KEK.auth KEK
```

```
[ec2-user ~]$ efi-updatevar -f PK.auth PK
```

確認已啟用UEFI安全開機

若要驗證是否已啟用UEFI安全開機，請遵循中的步驟確認 Amazon EC2 執行個體是否已啟用UEFI安全開機。

您現在可以使用get-instance-uefi-dataCLI指令匯出UEFI變數存放區，或者繼續執行下一個步驟並簽署您的開機映像，以重新開機進入啟用UEFI安全開機的執行個體。

步驟 3：AMI從執行個體建立

若要AMI從執行個體建立，您可以使用主控台或`CreateImage`APICLI、或SDKs。如需主控台的說明，請參閱 創建一個 Amazon 支EBS持 AMI。如需API指示，請參閱CreateImage。

> **Note**
>
> 會`CreateImage`API自動將執行個體的UEFI變數存放區複製到AMI。主控台會使用`CreateImage`API. 使用此功能啟動執行個體之後AMI，執行個體將擁有相同的UEFI變數存放區。

步驟 2（選項 B）：創建一個包含預填充變量存儲的二進制 blob

建立三個金鑰配對之後，您可以建立二進位 blob，其中包含預先填入的變數存放區，其中包含UEFI安全開機金鑰。

> **Warning**
>
> 您必須在註冊金鑰之前簽署開機映像，否則您將無法啟動執行個體。

選項 B 步驟：

- 步驟 1：建立新的變數存放區或更新現有變數存放區
- 第 2 步：在AMI創建時上傳二進制 blob

步驟 1：建立新的變數存放區或更新現有變數存放區

您可以使用 python-uefivars 工具，在沒有執行個體執行的情況下離線建立變數存放區。該工具可以從您的金鑰建立一個新的變數存放區。指令碼目前支援使用較高層級的工具更容易編輯的 AWS 格式、格式和JSON表示。EDK2

在沒有執行個體執行的情況下離線建立變數存放區

1. 在下列連結下載該工具。

```
https://github.com/awslabs/python-uefivars
```

2. 通過執行下列命令從您的金鑰建立新的變數存放區。這將在中創建一個 base64 編碼的二進制塊 *your_binary_blob*. 斌。該工具還支援透過 `-I` 參數更新二進位 blob。

```
./uefivars.py -i none -o aws -O your_binary_blob.bin -P PK.esl -K KEK.esl --db
 db.esl --dbx dbx.esl
```

第 2 步：在AMI創建時上傳二進制 blob

用register-image於傳遞UEFI變量存儲數據。對於 `--uefi-data` 參數，請指定您的二進位 blob，而對於 `--boot-mode` 參數，則指定 `uefi`。

```
aws ec2 register-image \
    --name uefi_sb_tpm_register_image_test \
    --uefi-data $(cat your_binary_blob.bin) \
    --block-device-mappings "DeviceName=/dev/sda1,Ebs=
 {SnapshotId=snap-0123456789example,DeleteOnTermination=true}" \
    --architecture x86_64 \
    --root-device-name /dev/sda1 \
    --virtualization-type hvm \
    --ena-support \
    --boot-mode uefi
```

為UEFI安全啟動創建 AWS 二進制 blob

您可以使用下列步驟在AMI建立期間自訂UEFI安全開機變數。這些步驟中使用的是截至 2021 年 9 月的最新版本。KEK如果 Microsoft 更新KEK，您必須使用最新的KEK.

若要建立二 AWS 進位 blob

1. 建立一個空白的 PK 簽章清單。

```
touch empty_key.crt
cert-to-efi-sig-list empty_key.crt PK.esl
```

2. 下載KEK憑證。

```
https://go.microsoft.com/fwlink/?LinkId=321185
```

3. 將KEK憑證包裝在UEFI簽名清單 (`siglist`) 中。

```
sbsiglist --owner 77fa9abd-0359-4d32-bd60-28f4e78f784b --type x509 --output
 MS_Win_KEK.esl MicCorKEKCA2011_2011-06-24.crt
```

4. 下載 Microsoft 的 db 憑證。

```
https://www.microsoft.com/pkiops/certs/MicWinProPCA2011_2011-10-19.crt
https://www.microsoft.com/pkiops/certs/MicCorUEFCA2011_2011-06-27.crt
```

5. 產生 db 簽章清單。

```
sbsiglist --owner 77fa9abd-0359-4d32-bd60-28f4e78f784b --type x509 --output
 MS_Win_db.esl MicWinProPCA2011_2011-10-19.crt
sbsiglist --owner 77fa9abd-0359-4d32-bd60-28f4e78f784b --type x509 --output
 MS_UEFI_db.esl MicCorUEFCA2011_2011-06-27.crt
cat MS_Win_db.esl MS_UEFI_db.esl > MS_db.esl
```

6. 從下列連結下載更新後的 dbx 變更要求。

```
https://uefi.org/revocationlistfile
```

7. 您在上一個步驟中下載的 dbx 變更要求已經與 Microsoft 簽署KEK，因此您必須將其移除或解壓縮。您可以使用下列連結。

```
https://gist.github.com/out0xb2/f8e0bae94214889a89ac67fceb37f8c0
```

```
https://support.microsoft.com/en-us/topic/microsoft-guidance-for-applying-secure-
boot-dbx-update-e3b9e4cb-a330-b3ba-a602-15083965d9ca
```

8. 使用`uefivars.py`指令碼建立UEFI變數存放區。

```
./uefivars.py -i none -o aws -O uefiblob-microsoft-keys-empty-pk.bin -P ~/PK.esl -K
 ~/MS_Win_KEK.esl --db ~/MS_db.esl  --dbx ~/dbx-2021-April.bin
```

9. 檢查二進制 blob 和UEFI變量存儲。

```
./uefivars.py -i aws -I uefiblob-microsoft-keys-empty-pk.bin -o json | less
```

10. 您可以透過再次將 blob 傳遞給同一工具來更新 Blob。

```
./uefivars.py -i aws -I uefiblob-microsoft-keys-empty-pk.bin -o aws -O uefiblob-
microsoft-keys-empty-pk.bin -P ~/PK.esl -K ~/MS_Win_KEK.esl --db ~/MS_db.esl  --dbx
 ~/dbx-2021-April.bin
```

預期的輸出結果

```
Replacing PK
Replacing KEK
Replacing db
Replacing dbx
```

對 EBS 後端 AMI 使用加密

以 Amazon EBS 快照為後端的 AMI 可以利用 Amazon EBS 加密。資料和根磁碟區的快照可以經過加密再連接至 AMI。您可以啟動執行個體並複製映像，而且包括完整 EBS 加密支援在內。所有可用的區域都支援這些作業的 AWS KMS 加密參數。

具有 EBS 磁碟區的 EC2 執行個體從 AMIs 啟動的方式，如同其他執行個體一樣。此外，從以未加密 EBS 快照為後端的 AMI 啟動執行個體時，您可以在啟動期間加密部分或所有磁碟區。

與 EBS 磁碟區一樣，AMI 中的快照可以依預設加密 AWS KMS key，也可以加密至您指定的客戶管理金鑰。無論是哪種情況，您都必須擁有許可才能使用所選取的 KMS 金鑰。

具有加密快照的 AMI 可以跨 AWS 帳戶共用。如需詳細資訊，請參閱 了解 Amazon 中的共享AMI用量 EC2。

對 EBS 後端 AMI 加密主題

- 執行個體啟動案例
- 映像複製案例

執行個體啟動案例

Amazon EC2 執行個體是透過區塊裝置對應提供的參數的`RunInstances`動作，透過 AWS Management Console 或直接使用 Amazon EC2 API 或 CLI 來啟動 Amazon EC2 執行個體。如需詳細

資訊，請參閱 Amazon EC2 執行個體上磁碟區的區塊裝置對映。如需控制區塊裝置對應的範例 AWS CLI，請參閱啟動、列出和終止 EC2 執行個體。

根據預設，如果沒有明確的加密參數，從 AMI 的來源快照還原 EBS 磁碟區時，`RunInstances` 動作會維護快照的現有加密狀態。如果預設啟用加密，則從 AMI 建立的所有磁碟區 (無論是從加密或未加密的快照) 都會加密。如果預設未啟用加密，則執行個體會維持 AMI 的加密狀態。

您也可以啟動執行個體並提供加密參數，以同時將新的加密狀態套用到產生的磁碟區。因此，可看到有下列行為：

在無加密參數的情況下啟動

- 未加密快照會還原到未加密的磁碟區，除非預設加密已啟用，在此情況下，所有新建立的磁碟區都加密。
- 您擁有的加密快照會還原到以相同 KMS 金鑰 所加密的磁碟區。
- 您不擁有的加密快照 (例如，AMI 會與您共用) 會還原至由您 AWS 帳戶預設 KMS 金鑰加密的磁碟區。

您可以提供加密參數來覆寫預設行為。可用的參數是 `Encrypted` 和 `KmsKeyId`。只設定 `Encrypted` 參數會產生下列行為：

已設定 **`Encrypted`** 但未指定 **`KmsKeyId`** 的執行個體啟動行為

- 未加密快照會還原到以您的 AWS 帳戶的預設 KMS 金鑰所加密的 EBS 磁碟區。
- 您擁有的加密快照會還原到以相同 KMS 金鑰 所加密的 EBS 磁碟區。(換言之，`Encrypted` 參數沒有效果。)
- 您不擁有的加密快照 (也就是說，AMI 會與您共用) 還原至由您 AWS 帳戶預設 KMS 金鑰加密的磁碟區。(換言之，`Encrypted` 參數沒有效果。)

同時設定 `Encrypted` 和 `KmsKeyId` 參數可讓您為加密操作指定非預設 KMS 金鑰。產生下列行為：

`Encrypted` 和 **`KmsKeyId`** 都設定的執行個體

- 未加密快照會還原到以指定的 KMS 金鑰 所加密的 EBS 磁碟區。
- 加密快照會還原到不是以原始 KMS 金鑰 所加密的 EBS 磁碟區，而是以指定的 KMS 金鑰 所加密的 EBS 磁碟區。

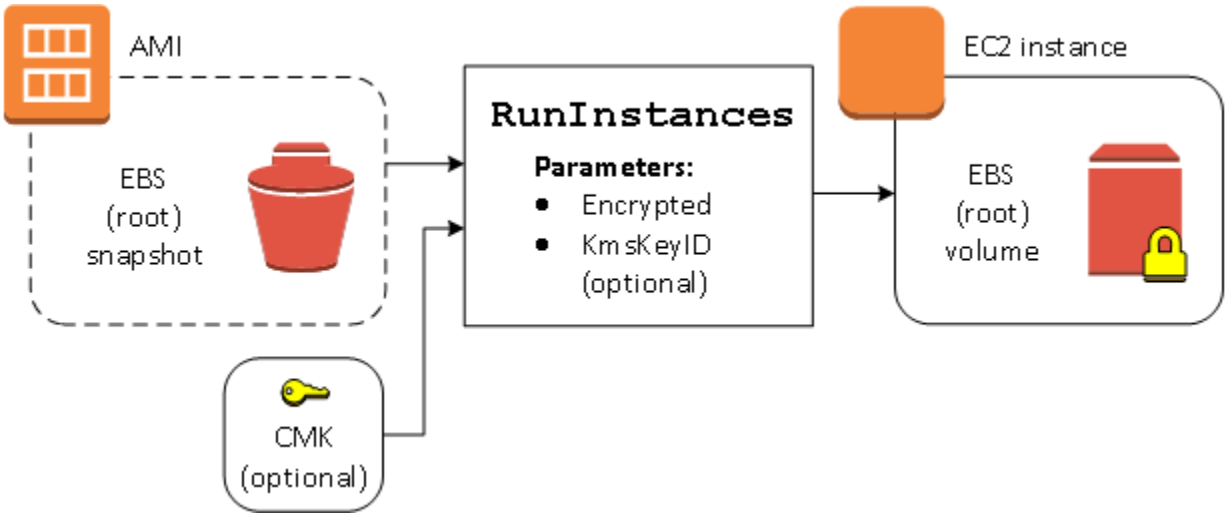
提交 `KmsKeyId` 但未同時設定 `Encrypted` 參數會導致錯誤。

下列各節提供從 AMI 使用非預設加密參數來啟動執行個體的範例。在每個案例中，提供給 `RunInstances` 動作的參數會導致從快照還原磁碟區期間變更加密狀態。

如需使用主控台從 AMI 啟動執行個體的資訊，請參閱 啟動 Amazon EC2 實例。

啟動期間加密磁碟區

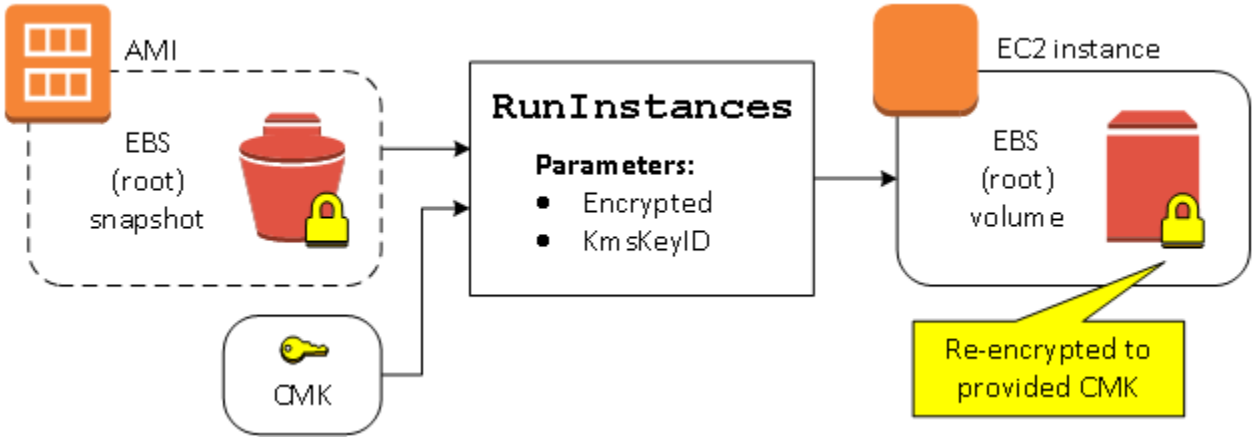
在此範例中，以未加密快照為後端的 AMI 用於啟動具有加密 EBS 磁碟區的 EC2 執行個體。



單獨 `Encrypted` 參數會導致加密此執行個體的磁碟區。提供 `KmsKeyId` 參數為選擇性。如果未指定 KMS 金鑰識別碼，則會使用 AWS 帳戶的預設 KMS 金鑰來加密磁碟區。若要以您擁有的不同 KMS 金鑰 來加密磁碟區，請提供 `KmsKeyId` 參數。

啟動期間重新加密磁碟區

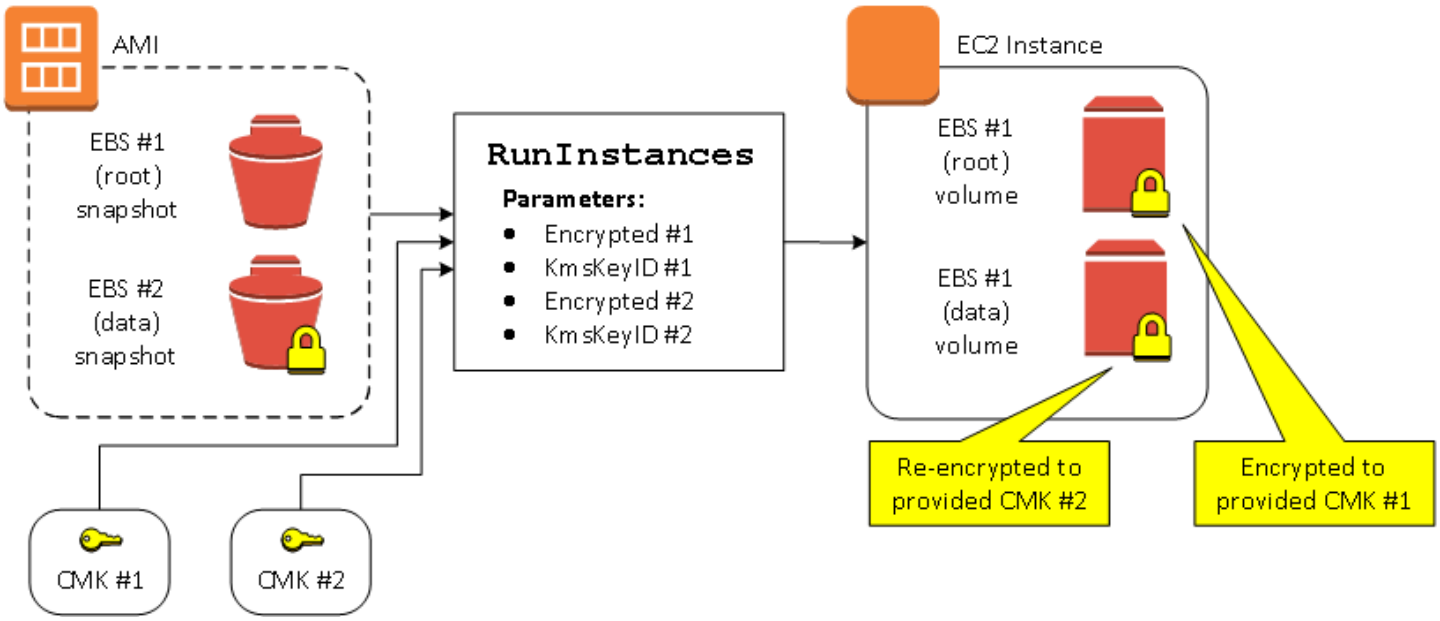
在此範例中，受加密快照支援的 AMI 用於啟動 EC2 執行個體，而此執行個體具有以新的 KMS 金鑰 所加密的 EBS 磁碟區。



如果您擁有 AMI 但未提供加密參數，則產生的執行個體具有使用與快照相同的 KMS 金鑰 所加密的磁碟區。如果 AMI 是共享，而非您所擁有，而且您未提供加密參數，則會以您的預設 KMS 金鑰 來加密磁碟區。按上圖所示提供加密參數，磁碟區會以指定的 KMS 金鑰 加密。

啟動期間變更多個磁碟區的加密狀態

在這個較複雜的範例中，以多個快照 (各有其自己的加密狀態) 為後端的 AMI 用於啟動 EC2 執行個體，而此執行個體具新加密的磁碟區和重新加密的磁碟區。



在此案例中，提供每個來源快照的加密參數給 `RunInstances` 動作。指定所有可能的加密參數時，無論您是否擁有 AMI，產生的執行個體都相同。

映像複製案例

透過 AWS Management Console 或直接使用 Amazon EC2 API 或 CLI，使用 `CopyImage` 動作來複製 Amazon EC2 AMI。

根據預設，如果沒有明確的加密參數，在複製期間，`CopyImage` 動作會維護 AMI 的來源快照的現有加密狀態。您也可以啟動 AMI 並提供加密參數，以同時將新的加密狀態套用到其相關聯的 EBS 快照。因此，可看到有下列行為：

在無加密參數的情況下複製

- 未加密快照會複製到另一個未加密的快照，除非預設加密已啟用，在此情況下，所有新建立的快照都加密。
- 您擁有的加密快照會複製到以相同 KMS 金鑰 加密的快照。

- 您不擁有的加密快照 (也就是說，AMI 會與您共用) 複製到由您 AWS 帳戶的預設 KMS 金鑰加密的快照集。

您可以提供加密參數來覆寫所有這些預設行為。可用的參數是 `Encrypted` 和 `KmsKeyId`。只設定 `Encrypted` 參數會產生下列行為：

已設定 **`Encrypted`** 但未指定 **`KmsKeyId`** 的複製映像行為

- 未加密快照會複製到以 AWS 帳戶的預設 KMS 金鑰所加密的快照。
- 加密快照會複製到以相同 KMS 金鑰 所加密的快照。(換言之，`Encrypted` 參數沒有效果。)
- 您不擁有的加密快照 (也就是說，AMI 會與您共用) 複製到由 AWS 帳戶預設 KMS 金鑰加密的磁碟區。(換言之，`Encrypted` 參數沒有效果。)

同時設定 `Encrypted` 和 `KmsKeyId` 參數可讓您為加密操作指定客戶受管 KMS 金鑰。產生下列行為：

`Encrypted` 和 **`KmsKeyId`** 都設定的複製映像行為

- 未加密快照會複製到以指定的 KMS 金鑰 所加密的快照。
- 加密快照會複製到不是以原始 KMS 金鑰 所加密的快照，而是以指定的 KMS 金鑰 加密的快照。

提交 `KmsKeyId` 但未同時設定 `Encrypted` 參數會導致錯誤。

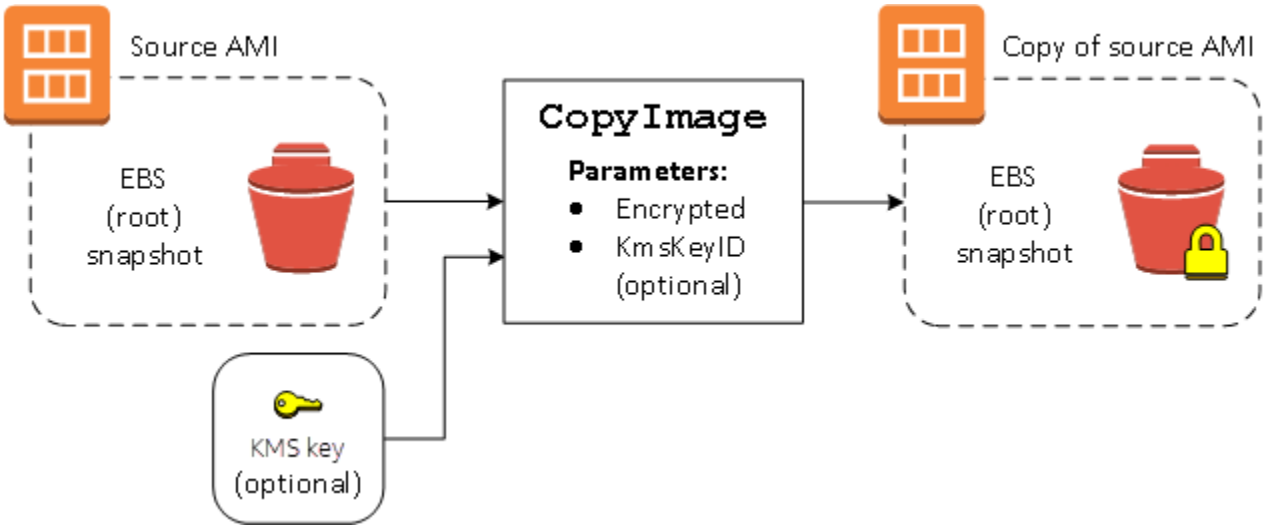
下一節提供使用非預設加密參數來複製 AMI 的範例，結果導致變更加密狀態。

如需使用主控台的詳細說明，請參閱 複製 Amazon EC2 AMI。

複製期間加密未加密的映像

在此案例中，未加密根快照所支援的 AMI 將複製到具有加密根快照的 AMI。透過兩個加密參數叫用 `CopyImage` 動作，包括客戶受管金鑰。因此，根快照的加密狀態會變更，目標 AMI 會受到包含與來源快照相同資料的根快照支援，但使用指定的金鑰進行加密。您需要為兩個 AMI 中的快照支付儲存成本，以及從任一 AMI 啟動的任何執行個體的費用。

> **Note**
>
> 依預設啟用加密，與將`Encrypted``true`參數設定為 AMI 中所有快照的效果相同。



設定 `Encrypted` 參數會加密此執行個體的單一快照。若您未指定 `KmsKeyId` 參數，則會使用預設的客戶受管金鑰來加密快照複本。

> **Note**
>
> 您也可以複製具有多個快照的映像，並個別設定各快照的加密狀態。

了解 Amazon 中的共享AMI用量 EC2

共享AMI是開發AMI人員創建並提供給其他人使用的。開始使用 Amazon 的最簡單方法之一EC2是使用具有所需元件的共AMI用，然後新增自訂內容。您也可以創建自己的AMIs並與他人共享。

您使用共享AMI的風險由您自擔。Amazon 不能保證其他 Amazon EC2 用戶AMIs共享的完整性或安全性。因此，您應該將 shared 視AMIs為可能會考慮在自己的資料中心部署的任何外部程式碼，並執行適當的盡職調查。我們建議您AMI從受信任的來源取得，例如經過驗證的提供者。

已驗證的供應商

在 Amazon EC2 控制台中，由 Amazon 或經過驗證的 Amazon 合作夥伴擁有的公共AMIs標記為「已驗證的提供商」。

您也可以使用描述影像 AWS CLI 指令來識別來自已驗證提供者AMIs的公用。Amazon 或已驗證的合作夥伴擁有的公有映像有一個採用別名的擁有者，即 `amazon` 或 `aws-marketplace`。在CLI輸出中，這些值會顯示為`ImageOwnerAlias`。其他使用者無法為他們的別名AMIs。這使您可以輕鬆地AMIs從 Amazon 或經過驗證的合作夥伴找到。

若要成為已驗證的供應商，您必須在 AWS Marketplace上註冊為賣方。註冊後，您可以AMI在 AWS Marketplace. 如需更多資訊，請參閱「賣家指南」中的「以賣家身分開始使用」和「以AWS Marketplace 賣家為AMI基礎

共享AMI主題

- 尋找共用AMIs以用於 Amazon EC2 執行個體
- 準備使用 Linux AMIs 的共用功能
- 讓您AMI公開可在 Amazon 使用 EC2
- 了解阻止公共訪問 AMIs
- 與組織和組織單位共AMI用
- AMI與特定 AWS 帳戶共用
- 取消與您的AMI共享 AWS 帳戶
- 建立共用 Linux 的建議 AMIs

如果您正在尋找有關其他主題的資訊

- 若要取得有關建立的資訊AMI，請參閱the section called “建立執行個體存放區支援 AMI”或the section called “創建一個 AMI”。
- 如需有關在 AWS Marketplace上建置、交付和維護應用程式的資訊，請參閱 AWS Marketplace 文件。

尋找共用AMIs以用於 Amazon EC2 執行個體

您可以使用 Amazon EC2 主控台或命令列尋找與 Amazon EC2 執行個體搭配使AMIs用的公開或私有共用。

AMIs是一個區域資源。當您搜尋共用 AMI (公開或私人) 時，您必須從共用來源的相同區域進行搜尋。若要在不同的「區域」中使AMI用，請將「區域」複製AMI到「區域」，然後共用它。如需詳細資訊，請參閱複製 Amazon EC2 AMI。

尋找共用 AMI (主控台)

使用主控台尋找共AMI用的私人

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇AMIs。

3. 在第一個篩選條件中，選擇 Private images (私有映像)。所有AMIs已與您共享的內容都會列出。若要縮小搜尋範圍，請選擇 Search (搜尋) 列並使用選單中提供的篩選條件選項。

使用主控台尋找共AMI用的公用

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇AMIs。
3. 在第一個篩選條件中，選擇 Public images (公有映像)。若要縮小搜尋範圍，請選擇 Search (搜尋) 欄位並使用功能表中提供的篩選條件選項。

使用控制台查找亞馬遜AMIs的共享公共

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇AMIs。
3. 在第一個篩選條件中，選擇 Public images (公有映像)。
4. 選擇 Search (搜尋) 欄位，然後從顯示的功能表選項中依序選擇 Owner alias (擁有者別名)、= 和 amazon，以僅顯示 Amazon 的公有映像。

使用主控台AMI從經過驗證的提供者尋找共用的公用

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在功能窗格中，選擇 [AMI目錄]。
3. 選擇「社群」AMIs。
4. 已驗證的供應商標籤表AMIs示來自 Amazon 或經過驗證的合作夥伴。

尋找共用的 AMI (AWS CLI)

使用描述影像指令 (AWS CLI) 進行列示。AMIs您可以將清單範圍設定為您感興趣AMIs的類型，如下列範例所示。

範例：列出所有公開 AMIs

以下命令列出了所有公開的AMIs，包括您擁有AMIs的任何公共。

```
aws ec2 describe-images --executable-users all
```

範例：AMIs具有明確啟動權限的清單

下列命令會AMIs列出您具有明確啟動權限的項目。此列表不包括您擁有AMIs的任何內容。

```
aws ec2 describe-images --executable-users self
```

範例：已驗證提供者AMIs擁有的清單

下面的命令列出了驗證提供者AMIs所擁有的。AMIs由經過驗證的供應商 (Amazon 或經過驗證的合作夥伴) 所擁有的公開擁有者擁有，其顯示為`amazon`或`aws-marketplace`在帳戶欄位中。這可幫助您輕鬆地AMIs從經過驗證的提供商處找到。其他使用者無法為他們的別名AMIs。

```
aws ec2 describe-images \
    --owners amazon aws-marketplace \
    --query 'Images[*].[ImageId]' \
    --output text
```

範例：帳戶AMIs擁有的清單

下面的命令列出由指定的AMIs擁有 AWS 帳戶。

```
aws ec2 describe-images --owners 123456789012
```

範例：AMIs使用篩選器進行範圍

要減少顯示的數量AMIs，請使用過濾器僅列出您感興趣AMIs的類型。例如，使用下列篩選器僅顯示 EBS-backed AMIs。

```
--filters "Name=root-device-type,Values=ebs"
```

尋找共用 AMI (適用於視窗的工具 PowerShell)

使用指Get-EC2Image令 (適用於 Windows 的工具 PowerShell) 進行列示AMIs。您可以將清單範圍設定為您感興趣AMIs的類型，如下列範例所示。

範例：列出所有公開 AMIs

以下命令列出了所有公開的AMIs，包括您擁有AMIs的任何公共。

```
PS C:\> Get-EC2Image -ExecutableUser all
```

範例：AMIs具有明確啟動權限的清單

下列命令會AMIs列出您具有明確啟動權限的項目。此列表不包括您擁有AMIs的任何內容。

```
PS C:\> Get-EC2Image -ExecutableUser self
```

範例：已驗證提供者AMIs擁有的清單

下面的命令列出了驗證提供者AMIs所擁有的。AMIs由經過驗證的供應商 (Amazon 或經過驗證的合作夥伴) 所擁有的公開擁有者擁有，其顯示為`amazon`或`aws-marketplace`在帳戶欄位中。這可幫助您輕鬆地AMIs從經過驗證的提供商處找到。其他使用者無法為他們的別名AMIs。

```
PS C:\> Get-EC2Image -Owner amazon aws-marketplace
```

範例：帳戶AMIs擁有的清單

下面的命令列出由指定的AMIs擁有 AWS 帳戶。

```
PS C:\> Get-EC2Image -Owner 123456789012
```

範例：AMIs使用篩選器進行範圍

要減少顯示的數量AMIs，請使用過濾器僅列出您感興趣AMIs的類型。例如，使用下列篩選器僅顯示 EBS-backed AMIs。

```
-Filter @{ Name="root-device-type"; Values="ebs" }
```

準備使用 Linux AMIs 的共用功能

在您使用 Linux 的共AMI用資料之前，請執行下列步驟，確認沒有預先安裝的認證可允許第三方不必要的存取您的執行個體，也沒有預先設定的遠端記錄可以傳輸敏感資料給第三方。如需有關改善系統安全性的資訊，請查閱所使用之 Linux 發行版本的文件。AMI

為了確保您不會意外失去對執行個體的存取權，建議您啟動兩個工SSH作階段，並保持第二個工作階段開啟，直到您移除了無法識別的認證，並確認您仍然可以使用登入執行個體為止SSH。

1. 識別並停用任何未經授權的公開SSH金鑰。檔案中唯一的金鑰應該是您用來啟動AMI. 以下命令可找出 `authorized_keys` 檔案：

```
[ec2-user ~]$ sudo find / -name "authorized_keys" -print -exec cat {} \;
```

2. 停用根使用者的密碼型態身分驗證。開啟 `sshd_config` 檔案並編輯 `PermitRootLogin` 行，如下所示：

   ```
   PermitRootLogin without-password
   ```

   或者，您可停用公用程式，以根使用者身分登入執行個體：

   ```
   PermitRootLogin No
   ```

   重新啟動 sshd 服務。
3. 檢查是否有任何其他使用者能夠登入您的執行個體。具有超級使用者權限的使用者特別危險。移除或鎖定任何未知帳戶的密碼。
4. 檢查未使用且正在執行網路服務偵聽是否有傳入連線的開放中連接埠。
5. 若要避免預先設定的遠端登入，您應刪除現有的組態檔案，然後重新啟動 `rsyslog` 服務。例如：

   ```
   [ec2-user ~]$ sudo rm /etc/rsyslog.conf
   [ec2-user ~]$ sudo service rsyslog restart
   ```
6. 確認所有的 cron 任務均為合法。

如果您發現自己覺得AMI存在安全性風險的公眾，請聯絡 AWS 安全性團隊。如需詳細資訊，請參閱 AWS 安全中心。

讓您AMI公開可在 Amazon 使用 EC2

您可以通過與所有人共享使用您的AMI公開 AWS 帳戶。

如果您想要防止公開共用您的AMIs，您可以啟用的封鎖公用存取AMIs。這可以阻止任何AMI公開的嘗試，有助於防止未經授權的訪問和潛在的AMI數據濫用。請注意，啟用封鎖公開存取並不會影響您AMIs已經公開使用的存取；它們仍然可以公開使用。如需詳細資訊，請參閱了解阻止公共訪問 AMIs。

若要僅允許特定帳戶使用您AMI啟動執行個體，請參閱AMI與特定 AWS 帳戶共用。

目錄

- 考量事項

- AMI與所有 AWS 帳戶共用 (公開分享)

考量事項

AMI公開之前，請考慮以下事項。

- 所有權 — 若要AMI公開，您 AWS 帳戶 必須擁有AMI.
- 地區-AMIs 是一個區域資源。當您共享時AMI，僅在您分享它的地區中可用。若要在不同的「區域」中使AMI用，請將「區域」複製AMI到「區域」，然後共用它。如需詳細資訊，請參閱複製 Amazon EC2 AMI。
- 封鎖公開存取 — 若要公開共用AMI，AMIs必須在每個AMI將公開共用的區域中停用的封鎖公開存取。公開分享之後AMI，您可以重新啟用封鎖公用存取，以防止AMIs進一步公開分享您AMIs的.
- 有些不AMIs能公開-如果您AMI包含以下組件之一，則無法將其公開（但您可以AMI與特定組件共享 AWS 帳戶）：
  - 加密磁碟區
  - 加密磁碟區的快照
  - 產品代碼
- 避免暴露敏感資料 — 若要避免在共用時暴露敏感資料AMI，請閱讀中的安全性考量，建立共用 Linux 的建議 AMIs並遵循建議的動作。
- 使用情況 — 當您共用時AMI，使用者只能從中啟動執行個體AMI。他們無法刪除、共用或修改它。不過，在使用您啟動執行個體之後AMI，他們就可以AMI從他們啟動的執行個體建立執行個體。
- 自動棄用 — 默認情況下，所有公共的棄用日期設置AMIs為從AMI創建日期起兩年。您可將棄用日期設定為早於兩年。要取消棄用日期或將棄用日期移至較晚的日期，您必須僅與特定日期共享來將其設為AMI私有。 AWS 帳戶
- 移除過時 AMIs — 在公用AMI達到淘汰日期之後，如果六個月或更長時間未啟動任何新執行個體，AWS 最終會移除公用共用屬性，以便過時AMIs不會出現在公開AMI清單中。AMI
- 帳單 — 當其 AWS 帳戶 他執行個體用於啟動執行個體時，我們不會向您AMI收取費用。使用執行個體啟動執行個體的AMI帳戶會按其啟動的執行個體計費。

AMI與所有 AWS 帳戶共用 (公開分享)

AMI公開之後，您可以在主控台的社群AMIs中使用此功能，您可以從主控台左側導覽器中的AMICatalog 存取，或使用EC2主控台啟動執行個體時。請注意，在您公開社群AMIs之後，可能需AMI要一段時間才會出現在社群中。

Console

AMI公開

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇AMIs。
3. AMI從清單中選取您的，然後選擇 [動作] > [編輯AMI權限]。
4. 在AMI可用性下，選擇 [公開]。
5. 選擇 Save changes (儲存變更)。

AWS CLI

每個都AMI有一個`launchPermission`屬性 AWS 帳戶，它控制除了所有者之外，允許哪些屬性使用它AMI來啟動實例。透過修改的`launchPermission`屬性AMI，您可以將其設為AMI公用 (這將啟動權限授與所有人 AWS 帳戶)，或僅與您指定的 AWS 帳戶 屬性共用。

您可以IDs從具有. 啟動權限的帳戶清單中新增或移除帳戶AMI。若要AMI公開，請指定`all`群組。您可同時指定公有和明確啟動許可。

AMI公開

1. 使用如下指modify-image-attribute令，將`all`群組加入至指定的`launchPermission`清單中AMI。

```
aws ec2 modify-image-attribute \
    --image-id ami-0abcdef1234567890 \
    --launch-permission "Add=[{Group=all}]"
```

2. 若要驗證的啟動權限AMI，請使用describe-image-attribute指令。

```
aws ec2 describe-image-attribute \
    --image-id ami-0abcdef1234567890 \
    --attribute launchPermission
```

3. (選擇性) 若要再次AMI設為私人，請將`all`群組從其啟動權限中移除。請注意，的擁有者AMI始終具有啟動權限，因此不受此命令的影響。

```
aws ec2 modify-image-attribute \
    --image-id ami-0abcdef1234567890 \
```

```
    --launch-permission "Remove=[{Group=all}]"
```

PowerShell

每個都AMI有一個`launchPermission`屬性 AWS 帳戶，它控制除了所有者之外，允許哪些屬性使用它AMI來啟動實例。透過修改的`launchPermission`屬性AMI，您可以將其設為AMI公用 (這將啟動權限授與所有人 AWS 帳戶)，或僅與您指定的 AWS 帳戶 屬性共用。

您可以IDs從具有. 啟動權限的帳戶清單中新增或移除帳戶AMI。若要AMI公開，請指定`all`群組。您可同時指定公有和明確啟動許可。

AMI公開

1. 使用如下指Edit-EC2ImageAttribute令，將`all`群組加入至指定的`launchPermission`清單中AMI。

```
PS C:\> Edit-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -Attribute
 launchPermission -OperationType add -UserGroup all
```

2. 若要驗證的啟動權限AMI，請使用下列Get-EC2ImageAttribute命令。

```
PS C:\> Get-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -Attribute
 launchPermission
```

3. (選擇性) 若要再次AMI設為私人，請將`all`群組從其啟動權限中移除。請注意，的擁有者AMI始終具有啟動權限，因此不受此命令的影響。

```
PS C:\> Edit-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -Attribute
 launchPermission -OperationType remove -UserGroup all
```

了解阻止公共訪問 AMIs

若要防止公開共用您的AMIs，您可以啟用的封鎖公用存取AMIs。此設定已在帳戶層級啟用，但您必須在每個 AWS 區域 要防止公開共用您的帳戶層級啟用此設定AMIs。

啟用封鎖公用存取時，任何嘗試AMI公開的嘗試都會自動封鎖。但是，如果您已經公開AMIs，則它們仍然可以公開使用。

若要公開分享AMIs，您必須停用封鎖公開存取。完成共用後，最佳做法是重新啟用封鎖公用存取，以防止任何意外的公開共用您的. AMIs

您可以將IAM權限限制為管理員使用者，以便只有他們可以啟用或停用的封鎖公用存取AMIs。

主題

- 預設設定
- 管理下列項目的封鎖公用存取設定 AMIs

預設設定

根據您的帳戶是新帳戶還是現有帳戶，以及您是否擁有公開，預AMIs設情況下會啟用或停用 [封鎖公用存取] 設定AMIs。下表列出了預設的設定：

AWS 帳戶	封鎖預設AMIs設定的公開存取
新帳戶	已啟用
沒有公開 AMIs [1] 的現有帳戶	已啟用
擁有一個或多個公開的現有帳戶 AMIs	已停用

[1] 如果您的帳戶在 2023 年 7 月 15 AMIs 日或之後有一個或多個公開帳戶，即使您之後將所有設AMIs為AMIs私人帳戶，依預設會停用您帳戶的封鎖公開存取。

您可以管理阻止公共訪問設置，AMIs以控制是否可以公開共享它們。您可以使用 Amazon EC2 主控台或檢視目前的區塊公AMIs用存取狀態 AWS CLI。

管理下列項目的封鎖公用存取設定 AMIs

檢視下列項目的封鎖公用存取狀態 AMIs

若要查看您帳戶中的公開共用AMIs是否遭到封鎖，您可以檢視封鎖公開存取的狀態AMIs。您必須查看每 AWS 區域 個狀態，以查看您的公共共享AMIs是否被阻止。

所需的許可

若要取得的目前封鎖公用存取設定AMIs，您必須擁有`GetImageBlockPublicAccessState`IAM權限。

Console

若要檢視指定區域中的封鎖AMIs公用存取狀態

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 從導覽列 (位於畫面頂端)，選取要檢視其中封鎖公開存取狀態的區域AMIs。
3. 如果儀表板未顯示，請在導覽窗格中選擇「EC2儀表板」。
4. 在帳戶屬性下，選擇資料保護和安全性。
5. 在 [封鎖公用存取權限] 下方AMIs，勾選 [公用存取] 欄位。值可能是已封鎖新的公開共用，也可能是已允許新的公開共用。

AWS CLI

若要取得下列項目的封鎖公用存取狀態 AMIs

使用 get-image-block-public-訪問狀態命令。

- 對於特定區域

```
aws ec2 get-image-block-public-access-state --region us-east-1
```

預期的輸出結果 – 值為 `block-new-sharing` 或 `unblocked`。

```
{
    "ImageBlockPublicAccessState": "block-new-sharing"
}
```

- 對於您帳戶中的所有區域

```
echo -e "Region   \t Public Access State" ; \
echo -e "-------------- \t ----------------------" ; \
for region in $(
    aws ec2 describe-regions \
        --region us-east-1 \
        --query "Regions[*].[RegionName]" \
        --output text
```

```
    );
    do (output=$(
        aws ec2 get-image-block-public-access-state \
            --region $region \
            --output text)
        echo -e "$region \t $output"
    );
done
```

預期的輸出結果 – 值為 `block-new-sharing` 或 `unblocked`。

```
Region           Public Access State
--------------   ----------------------
ap-south-1       block-new-sharing
eu-north-1       unblocked
eu-west-3        block-new-sharing
...
```

PowerShell

若要取得下列項目的封鎖公用存取狀態 AMIs

使用指Get-EC2ImageBlockPublicAccessState令程式。

- 對於特定區域

```
Get-EC2ImageBlockPublicAccessState -Region us-east-1
```

預期的輸出結果

```
block-new-sharing
```

- 對於您帳戶中的所有區域

```
(Get-EC2Region).RegionName | `
    ForEach-Object {
        [PSCustomObject]@{
            Region   = $_
            PublicAccessState = (Get-EC2ImageBlockPublicAccessState -Region $_)
        }
} | `
```

```
Format-Table -AutoSize
```

預期的輸出結果

```
Region         PublicAccessState
------         -----------------
ap-south-1     block-new-sharing
eu-north-1     block-new-sharing
eu-west-3      block-new-sharing
…
```

啟用封鎖的公用存取 AMIs

為了防止公開共享您的AMIs，請AMIs在帳戶級別啟用阻止公共訪問。您必須AMIs AWS 區域 在每個要阻止公共共享您的AMIs. 如果您已經有公開AMIs，它們將保持公開可用。

所需的許可

若要啟用的封鎖公用存取設定AMIs，您必須擁有`EnableImageBlockPublicAccess`IAM權限。

Console

啟用指定區域AMIs中的封鎖公用存取

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 從導覽列 (位於畫面頂端)，選取要在其中啟用封鎖公開存取的區域AMIs。
3. 如果儀表板未顯示，請在導覽窗格中選擇「EC2儀表板」。
4. 在帳戶屬性下，選擇資料保護和安全性。
5. 在 [封鎖的公用存取] 下方AMIs，選擇 [管理]
6. 選取封鎖新的公開共用核取方塊，然後選擇更新。

> **Note**
> 設定此設定最多API可能需要 10 分鐘的時間。在此期間，該值將是已允許新的公開共用。完API成設定後，該值會自動變更為 [新增公用共用已封鎖]。

AWS CLI

若要啟用封鎖公用存取 AMIs

使用 enable-image-block-public-access 命令。

- 對於特定區域

```
aws ec2 enable-image-block-public-access \
--region us-east-1 \
--image-block-public-access-state block-new-sharing
```

預期的輸出結果

```
{
    "ImageBlockPublicAccessState": "block-new-sharing"
}
```

- 對於您帳戶中的所有區域

```
echo -e "Region   \t Public Access State" ; \
echo -e "-------------- \t ----------------------" ; \
for region in $(
    aws ec2 describe-regions \
        --region us-east-1 \
        --query "Regions[*].[RegionName]" \
        --output text
    );
    do (output=$(
        aws ec2 enable-image-block-public-access \
            --region $region \
            --image-block-public-access-state block-new-sharing \
            --output text)
        echo -e "$region \t $output"
    );
done
```

預期的輸出結果

```
Region           Public Access State
--------------   ----------------------
ap-south-1       block-new-sharing
```

```
eu-north-1       block-new-sharing
eu-west-3        block-new-sharing
...
```

> **Note**
>
> 設定此設定最多API可能需要 10 分鐘的時間。在此期間，如果你運行 get-image-block-public-access-state 命令，響應將是。`unblocked`完API成配置後，響應將是`block-new-sharing`。

PowerShell

若要啟用封鎖公用存取 AMIs

使用指Enable-EC2ImageBlockPublicAccess令。

- 對於特定區域

```
Enable-EC2ImageBlockPublicAccess `
    -Region us-east-1 `
    -ImageBlockPublicAccessState block-new-sharing
```

預期的輸出結果

```
Value
-----
block-new-sharing
```

- 對於您帳戶中的所有區域

```
(Get-EC2Region).RegionName | `
    ForEach-Object {
    [PSCustomObject]@{
        Region            = $_
        PublicAccessState = (
        Enable-EC2ImageBlockPublicAccess `
         -Region $_ `
         -ImageBlockPublicAccessState block-new-sharing)
    }
```

```
} | `
Format-Table -AutoSize
```

預期的輸出結果

```
Region         PublicAccessState
------         -----------------
ap-south-1     block-new-sharing
eu-north-1     block-new-sharing
eu-west-3      block-new-sharing
…
```

禁用的阻止公共訪問 AMIs

若要允許您帳戶中的使用者公開共用您的帳戶AMIs，請在帳戶層級停用封鎖公開存取。您必須禁用阻止公共訪問AMIs AWS 區域 在每個您想要允許公共共享您的AMIs.

所需的許可

若要停用的封鎖公用存取設定AMIs，您必須擁有`DisableImageBlockPublicAccess`IAM權限。

Console

若要在指定的區域AMIs中停用封鎖公用存取

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 從導覽列 (位於畫面頂端)，選取要停用其中封鎖公開存取的區域AMIs。
3. 如果儀表板未顯示，請在導覽窗格中選擇「EC2儀表板」。
4. 在帳戶屬性下，選擇資料保護和安全性。
5. 在 [封鎖的公用存取] 下方AMIs，選擇 [管理]
6. 清除封鎖新的公開共用核取方塊，然後選擇更新。
7. 出現確認提示時，請輸入 **`confirm`**，然後選擇允許公開共用。

> **Note**
> 設定此設定最多API可能需要 10 分鐘的時間。在此期間，該值將是已封鎖新的公開共用。完成API設定後，值會自動變更為 [允許新增公用共用]。

AWS CLI

若要停用的封鎖公用存取 AMIs

使用 disable-image-block-public-access 命令。

- 對於特定區域

```
aws ec2 disable-image-block-public-access --region us-east-1
```

預期的輸出結果

```
{
   "ImageBlockPublicAccessState": "unblocked"
}
```

- 對於您帳戶中的所有區域

```
echo -e "Region   \t Public Access State" ; \
echo -e "-------------- \t ----------------------" ; \
for region in $(
    aws ec2 describe-regions \
        --region us-east-1 \
        --query "Regions[*].[RegionName]" \
        --output text
    );
    do (output=$(
        aws ec2 disable-image-block-public-access \
            --region $region \
            --output text)
        echo -e "$region \t $output"
    );
done
```

預期的輸出結果

```
Region           Public Access State
--------------   ----------------------
ap-south-1       unblocked
eu-north-1       unblocked
eu-west-3        unblocked
```

```
...
```

> **Note**
>
> 設定此設定最多API可能需要 10 分鐘的時間。在此期間，如果你運行 get-image-block-public-access-state 命令，響應將是。`block-new-sharing`完API成配置後，響應將是`unblocked`。

PowerShell

若要停用的封鎖公用存取 AMIs

使用指Disable-EC2ImageBlockPublicAccess令程式。

- 對於特定區域

```
Disable-EC2ImageBlockPublicAccess -Region us-east-1
```

預期的輸出結果

```
Value
-----
unblocked
```

- 對於您帳戶中的所有區域

```
(Get-EC2Region).RegionName | `
    ForEach-Object {
    [PSCustomObject]@{
        Region            = $_
        PublicAccessState = (Disable-EC2ImageBlockPublicAccess -Region $_)
    }
} | `
Format-Table -AutoSize
```

預期的輸出結果

```
Region         PublicAccessState
```

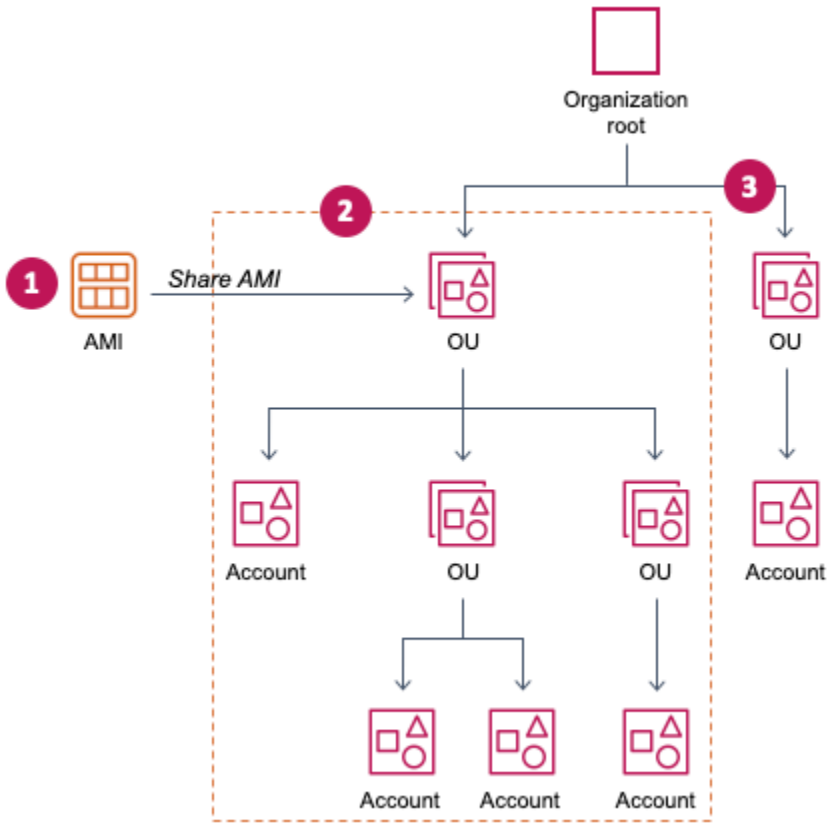
```
------          -----------------
ap-south-1      unblocked
eu-north-1      unblocked
eu-west-3       unblocked
…
```

與組織和組織單位共AMI用

AWS Organizations是一種帳戶管理服務，可讓您 AWS 帳戶 將多個組織合併到您建立並集中管理的組織中。除了AMI與特定帳戶共用外，您還可以與您建立的組織或組織單位 (OU) 共用。

組織是您建立來合併並集中管理 AWS 帳戶的實體。您可以將帳戶組織為階層式、類似樹狀目錄的結構，其中的根位於頂層，而組織單位在組織根下形成巢狀。每個帳號都可以直接新增至根帳號，或放置在階層中的其OUs中一個帳號。如需帳戶類型的詳細資訊，請參閱 AWS Organizations 使用者指南中的 AWS Organizations 術語與概念。

當您AMI與組織或 OU 共用時，所有子系帳戶都可以存取AMI. 例如，在下圖中，與頂層 OU 共用 (由數字 1 處的箭頭表示)。AMI巢狀於該頂層 OU (以數字 2 的虛線表示) 之下的所有OUs和帳戶也可以存取AMI. 組織中的帳戶和虛線外的 OU (以數字 3 表示) 無法存取，AMI因為它們不AMI是與共用之 OU 的子系。



主題

- 取得ARN組織或組織單位
- 考量事項
- 允許組織和OUs使用KMS金鑰
- 管理AMI與組織或 OU 的共用

取得ARN組織或組織單位

組織和組織單位ARNs包含 12 位數的管理帳號。如果您不知道管理帳戶號碼，您可以描述要取得每個帳戶的組織和組織單位。ARN在下列範例中，`123456789012` 是管理帳號。

您必須先擁有描述組織和組織單位的權限ARNs，才能取得。下列政策提供必要的許可。

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

```
            "Action": [
                "organizations:Describe*"
            ],
            "Resource": "*"
        }
    ]
}
```

為了得到ARN一個組織

使用describe-organization指令和`--query`參數設定`'Organization.Arn'`為可僅傳回組織ARN。

```
aws organizations describe-organization --query 'Organization.Arn'
```

回應範例

```
"arn:aws:organizations::123456789012:organization/o-123example"
```

若要取得ARN組織單位

使用命describe-organizational-unit令，指定 OU ID，然後將`--query`參數設定`'OrganizationalUnit.Arn'`為僅傳回組織單位ARN。

```
aws organizations describe-organizational-unit --organizational-unit-
id ou-1234-5example --query 'OrganizationalUnit.Arn'
```

回應範例

```
"arn:aws:organizations::123456789012:ou/o-123example/ou-1234-5example"
```

考量事項

與特定組織或組織單位共AMIs用時，請考慮下列事項。

- 擁有權 — 若要共用AMI，您 AWS 帳戶 必須擁有AMI.
- 共用限制 — AMI 擁有者可以AMI與任何組織或 OU 共用，包括組織OUs，且他們不是的成員。

  如需AMI可在區域內共用的實體數目上限，請參閱 Amazon EC2 服務配額。
- 標籤 — 您無法共用使用者定義的標籤 (附加至標籤AMI)。當您共用時AMI，您使用者定義的標籤不適用 AWS 帳戶 於與之共用的組織或 OU 中的AMI任何一個。

- ARNformat — 當您在命令中指定組織或 OU 時，請確定使用正確的ARN格式。如果只指定 ID，則您會收到錯誤，例如，如果只指定 `o-123example` 或 `ou-1234-5example`。

  正確的ARN格式：

  - 組織ARN：`arn:aws:organizations::account-id:organization/organization-id`
  - OUARN: `arn:aws:organizations::account-id:ou/organization-id/ou-id`

  其中：

  - *`account-id`* 是 12 位數字的管理帳戶號碼，`123456789012`例如。如果您不知道管理帳戶編號，可以描述要取得的組織或組織單位ARN，其中包括管理帳戶編號。如需詳細資訊，請參閱取得ARN組織或組織單位。
  - *`#####`* 是組織 ID，例如，`o-123example`。
  - *`#####`* 是組織單位 ID，例如，`ou-1234-5example`。

  如需有關的格式的詳細資訊ARNs，請參閱IAM使用者指南中的 Amazon 資源名稱 (ARNs)。
- 加密和金鑰 — 您可以共AMIs用未加密和加密快照支援的快照。
  - 加密快照必須透過客戶受管金鑰進行加密。您無法共用使AMIs用預設 AWS 受管理金鑰加密的快照所支援的快照集。
  - 如果您共AMI用由加密快照支援的快照，您必須允許組織或使用用OUs來加密快照的客戶管理金鑰。 如需詳細資訊，請參閱允許組織和OUs使用KMS金鑰。
- 地區-AMIs 是一個區域資源。當您共享時AMI，僅在您分享它的地區中可用。若要在不同的「區域」中使AMI用，請將「區域」複製AMI到「區域」，然後共用它。如需詳細資訊，請參閱複製 Amazon EC2 AMI。
- 使用情況 — 當您共用時AMI，使用者只能從中啟動執行個體AMI。他們無法刪除、共用或修改它。不過，在使用您啟動執行個體之後AMI，他們就可以AMI從他們啟動的執行個體建立執行個體。
- 帳單 — 當其 AWS 帳戶 他執行個體用於啟動執行個體時，我們不會向您AMI收取費用。使用執行個體啟動執行個體的AMI帳戶會按其啟動的執行個體計費。

允許組織和OUs使用KMS金鑰

如果您共AMI用由加密快照支援的快照，您也必須允許組織或OUs使用 AWS KMS keys 用來加密快照的組織。

使用`aws:PrincipalOrgID`和索`aws:PrincipalOrgPaths`引鍵來比較要求原則中 AWS Organizations 路徑的主參與者路徑的路徑。該主體可以是使用者、IAM角色、同盟使用者或 AWS 帳戶 根使用者。在策略中，此條件索引鍵可確保要求者是指定組織根目錄或OUs中 AWS Organizations

的帳戶成員。如需條件陳述式的更多範例，請參閱《IAM使用指南》aws:PrincipalOrgPaths中的aws:PrincipalOrgID和。

如需有關編輯金鑰政策的詳細資訊，請參閱AWS Key Management Service 開發人員指南中的允許其他帳戶中的使用者使用KMS金鑰。

若要授與組織或 OU 使用KMS金鑰的權限，請將下列陳述式新增至金鑰原則。

```
{
    "Sid": "Allow access for organization root",
    "Effect": "Allow",
    "Principal": "*",
    "Action": [
        "kms:Describe*",
        "kms:List*",
        "kms:Get*",
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "aws:PrincipalOrgID": "o-123example"
        }
    }
}
```

若要與多個KMS金鑰共用OUs，您可以使用類似下列範例的原則。

```
{
        "Sid": "Allow access for specific OUs and their descendants",
        "Effect": "Allow",
        "Principal": "*",
        "Action": [
            "kms:Describe*",
            "kms:List*",
            "kms:Get*",
            "kms:Encrypt",
            "kms:Decrypt",
            "kms:ReEncrypt*",
            "kms:GenerateDataKey*"
```

```
        ],
        "Resource": "*",
        "Condition": {
            "StringEquals": {
                "aws:PrincipalOrgID": "o-123example"
            },
            "ForAnyValue:StringLike": {
                "aws:PrincipalOrgPaths": [
                    "o-123example/r-ab12/ou-ab12-33333333/*",
                    "o-123example/r-ab12/ou-ab12-22222222/*"
                ]
            }
        }
}
```

管理AMI與組織或 OU 的共用

檢視OUs與共用的組織 AMI

您可以使用 Amazon EC2 主控台或檢 AWS CLI 查與哪些組織和OUs您共用了AMI.

檢視共用的組織 (主控台) OUs AMI

使用主控台檢查與哪些組織和OUs您AMI共享了您的

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇AMIs。
3. 在列表AMI中選擇您的，選擇權限選項卡，然後向下滾動到共享組織/OUs。

   若要尋找AMIs與您共用的項目，請參閱尋找共用AMIs以用於 Amazon EC2 執行個體。

檢視OUs與共用的組織 (AWS CLI) AMI

您可以使用describe-image-attribute命令 (AWS CLI) 和OUs`launchPermission`屬性來檢查哪些組織和您已AMI與之共用。

若要檢查哪些組織和OUs您已AMI使用 AWS CLI

此命describe-image-attribute令描述指定的`launchPermission`屬性AMI，並傳回OUs與您共用的組織AMI。

```
aws ec2 describe-image-attribute \
```

```
    --image-id ami-0abcdef1234567890 \
    --attribute launchPermission
```

回應範例

```
{
    "ImageId": "ami-0abcdef1234567890",
    "LaunchPermissions": [
        {
            "OrganizationalUnitArn": "arn:aws:organizations::111122223333:ou/
o-123example/ou-1234-5example"
        }
    ]
}
```

AMI與組織或 OU 共用

您可以使用 Amazon EC2 主控台或AMI與組織或 OU 共用。 AWS CLI

共用 AMI (主控台)

使用主控台AMI與組織或 OU 共用

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇AMIs。
3. 在清單AMI中選取您的，然後選擇 [動作] > [編輯AMI權限]。
4. 在AMI可用性下，選擇「私人」。
5. 在 [共用的組織/] 旁邊OUs，選擇 [新增組織/OU]。ARN
6. 針對組織 /OU ARN，輸入您要ARN與其共用的組織ARN或 OUAMI，然後選擇 [共用]。AMI請注意，您必須指定完整的ARN，而不僅僅是 ID。

   若要AMI與多個組織共用OUs，或重複此步驟，直到您新增了所有必要的組織或OUs。

   > **Note**
   > 您不需要共用AMI參考的 Amazon EBS 快照，即可共用AMI. 只需共用AMI本身，系統就會自動提供執行個體存取啟動時所參考的 Amazon EBS 快照。不過，您確實需要共用用來加密參照之快AMI照的KMS金鑰。 如需詳細資訊，請參閱允許組織和OUs使用KMS金鑰。

7. 完成後，請選擇 Save changes (儲存變更)。

8. (選擇性) 若要檢視或OUs與您共用的組織AMI，請在清單AMI中選取，選擇權限索引標籤，然後向下捲動至共用組織/OUs。若要尋找AMIs與您共用的項目，請參閱尋找共用AMIs以用於 Amazon EC2 執行個體。

分享一個AMI（AWS CLI）

使用modify-image-attribute指令 (AWS CLI) 來共用AMI.

若要使用AMI與組織共用 AWS CLI

此命modify-image-attribute令會授與指定組織的指AMI定的啟動權限。請注意，您必須指定完整的 ARN，而不僅僅是 ID。

```
aws ec2 modify-image-attribute \
    --image-id ami-0abcdef1234567890 \
    --launch-permission
 "Add=[{OrganizationArn=arn:aws:organizations::123456789012:organization/
o-123example}]"
```

若要使用AMI與 OU 共用 AWS CLI

此命modify-image-attribute令會授與指定 OU 的指AMI定的啟動權限。請注意，您必須指定完整的 ARN，而不僅僅是 ID。

```
aws ec2 modify-image-attribute \
    --image-id ami-0abcdef1234567890 \
    --launch-permission
 "Add=[{OrganizationalUnitArn=arn:aws:organizations::123456789012:ou/o-123example/
ou-1234-5example}]"
```

> **Note**
>
> 您不需要共用AMI參考的 Amazon EBS 快照，即可共用AMI. 只需共用AMI本身，系統就會自動提供執行個體存取啟動時所參考的 Amazon EBS 快照。不過，您確實需要共用用來加密參照之快AMI照的KMS金鑰。如需詳細資訊，請參閱允許組織和OUs使用KMS金鑰。

共用 AMI (視窗工具 PowerShell)

使用Edit-EC2ImageAttribute命令 (Windows 工具 PowerShell) 來共用，AMI如下列範例所示。

若要AMI與組織或 OU 共用

下列命令會授與指定組織的指AMI定的啟動權限。

```
PS C:\> Edit-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -
Attribute launchPermission -OperationType add -OrganizationArn
 "arn:aws:organizations::123456789012:organization/o-123example"
```

> **Note**
>
> 您不需要共用AMI參考的 Amazon EBS 快照，即可共用AMI. 只需共用AMI本身，系統就會自動提供執行個體存取啟動時所參考的 Amazon EBS 快照。不過，您確實需要共用用來加密參照之快AMI照的KMS金鑰。如需詳細資訊，請參閱允許組織和OUs使用KMS金鑰。

若要停止AMI與組織或 OU 共用

下列命令會從指定的組織移AMI除指定的啟動權限：

```
PS C:\> Edit-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -
Attribute launchPermission -OperationType remove -OrganizationArn
 "arn:aws:organizations::123456789012:organization/o-123example"
```

若要停止AMI與所有組織共用OUs，以及 AWS 帳戶

下列命令會從指定的移除所有公用和明確的啟動權限AMI。請注意，的擁有者AMI始終具有啟動權限，因此不受此命令的影響。

```
PS C:\> Reset-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -Attribute
 launchPermission
```

停止AMI與組織或 OU 共用

您可以使用 Amazon EC2 主控台或停止AMI與組織或 OU 共用。 AWS CLI

停止共用 AMI (主控台)

使用主控台停止AMI與組織或 OU 共用

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇AMIs。

3. 在清單AMI中選取您的，然後選擇 [動作] > [編輯AMI權限]。
4. 在 [共用組織/] 底下OUs，選取您要停止共OUs用的組織或組織AMI，然後選擇 [移除選取項目]。
5. 完成後，請選擇 Save changes (儲存變更)。
6. (選擇性) 若要確認您已停止AMI與組織共用OUs，或選取清單AMI中的，選擇權限索引標籤，然後向下捲動至共用組織/OUs。

停止共用 AMI (AWS CLI)

使用modify-image-attribute或reset-image-attribute指令 (AWS CLI) 停止共用AMI.

若要使用停止AMI與組織或 OU 共用 AWS CLI

命modify-image-attribute令會從指定的組織移AMI除指定的啟動權限。請注意，您必須指定ARN.

```
aws ec2 modify-image-attribute \
    --image-id ami-0abcdef1234567890 \
    --launch-permission
 "Remove=[{OrganizationArn=arn:aws:organizations::123456789012:organization/
o-123example}]"
```

若要停止AMI與所有組織共 AWS 帳戶 用OUs，請使用 AWS CLI

命reset-image-attribute令會從指定的移除所有公用和明確的啟動權限AMI。請注意，的擁有者AMI始終具有啟動權限，因此不受此命令的影響。

```
aws ec2 reset-image-attribute \
    --image-id ami-0abcdef1234567890 \
    --attribute launchPermission
```

> **Note**
> 如果特定帳戶位於AMI與其共用的組織或 OU 中，您就無法停止與該帳戶共用AMI該帳戶。如果您嘗試AMI透過移除帳戶的啟動許可來停止共用，Amazon 會EC2傳回成功訊息。但是，會AMI繼續與帳戶共用。

AMI與特定 AWS 帳戶共用

您可以在不公開的 AWS 帳戶 情況下AMI與特定AMI共享。所有你需要的是 AWS 帳戶 IDs.

AWS 帳戶 ID 是一個 12 位數的數字，例如012345678901，可唯一 AWS 帳戶識別. 如需詳細資訊，請參閱《AWS Account Management 參考指南》中的檢視 AWS 帳戶 識別符。

考量事項

AMIs與特定的共享時，請考慮以下事項 AWS 帳戶。

- 擁有權 — 若要共用AMI，您 AWS 帳戶 必須擁有AMI.
- 共用限制 — 有關AMI可在區域內共用的實體數量上限，請參閱 Amazon EC2 服務配額。
- 標籤 — 您無法共用使用者定義的標籤 (附加至標籤AMI)。當您共用時AMI，您使用者定義的標籤不適用於與 AWS 帳戶 之共用AMI的任何標籤。
- 加密和金鑰 — 您可以共AMIs用未加密和加密快照支援的快照。
  - 加密的快照必須使用KMS金鑰加密。您無法共用使AMIs用預設 AWS 受管理金鑰加密的快照所支援的快照集。
  - 如果您共AMI用由加密快照支援的快照，您必須允許使用用來加密快照的KMS金鑰。 AWS 帳戶 如需詳細資訊，請參閱允許組織和OUs使用KMS金鑰。若要設定在使用客戶受管金鑰進行加密時啟動 Auto Scaling 執行個體所需的金鑰 AWS KMS key 政策，請參閱 Amazon EC2 Auto Scaling 使用者指南中的加密磁碟區使用的必要政策。
- 地區-AMIs 是一個區域資源。當您共享時AMI，它僅適用於該地區。若要在不同的「區域」中使AMI用，請將「區域」複製AMI到「區域」，然後共用它。如需詳細資訊，請參閱複製 Amazon EC2 AMI。
- 使用情況 — 當您共用時AMI，使用者只能從中啟動執行個體AMI。他們無法刪除、共用或修改它。不過，在他們使用您的執行個體啟動執行個體之後AMI，就可以AMI從其執行個體建立執行個體。
- 複製共用 AMIs — 如果其他帳戶中的使用者想要複製共用AMI，您必須授與他們備份的儲存區的讀取權限AMI。如需詳細資訊，請參閱跨帳戶複製。
- 帳單 — 當其 AWS 帳戶 他執行個體用於啟動執行個體時，我們不會向您AMI收取費用。使用執行個體啟動執行個體的AMI帳戶會按其啟動的執行個體計費。

共用 AMI (主控台)

使用主控台授予明確啟動許可

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇AMIs。

3. 在清單AMI中選取您的，然後選擇 [動作] > [編輯AMI權限]。
4. 選擇 Private (私有)。
5. 在 Shared accounts (共享帳戶) 中，選擇 Add account ID (新增帳戶 ID)。
6. 針對 AWS 帳戶 ID，輸入您要共用的 AWS 帳戶 IDAMI，然後選擇 [共用] AMI。

   若要AMI與多個帳戶共用，請重複步驟 5 和 6，直到您新增所有必要的帳戶為止IDs。

   > **Note**
   >
   > 您不需要共用AMI參考的 Amazon EBS 快照，即可共用AMI. 只需共用AMI本身；系統會自動為啟動提供執行個體存取參考的 Amazon EBS 快照。不過，您確實需要共用任何用來加密AMI參照之快照的KMS金鑰。如需詳細資訊，請參閱 Amazon EBS使用者指南中的共用 Amazon EBS 快照。

7. 完成後，請選擇 Save changes (儲存變更)。
8. (選擇性) 若要檢視 AWS 帳戶 IDs與之共用的AMI，請在清單AMI中選取，然後選擇「權限」索引標籤。若要尋找AMIs與您共用的項目，請參閱尋找共用AMIs以用於 Amazon EC2 執行個體。

分享一個AMI（AWS CLI）

使用modify-image-attribute指令 (AWS CLI) 來共用，AMI如下列範例所示。

授予明確啟動許可

下列命令會授與指定的指定AMI的啟動權限 AWS 帳戶。在下列範例中，將範例 AMI ID 取代為有效 AMI ID，並以 12 位數 AWS 帳戶 ID 取代*account-id*。

```
aws ec2 modify-image-attribute \
    --image-id ami-0abcdef1234567890 \
    --launch-permission "Add=[{UserId=account-id}]"
```

> **Note**
>
> 您不需要共用AMI參考的 Amazon EBS 快照，即可共用AMI. 只需共用AMI本身；系統會自動為啟動提供執行個體存取參考的 Amazon EBS 快照。不過，您確實需要共用任何用來加密AMI參照之快照的KMS金鑰。如需詳細資訊，請參閱 Amazon EBS使用者指南中的共用 Amazon EBS 快照。

移除帳戶的啟動許可

下列命令會從指定的移除指定AMI的啟動權限 AWS 帳戶。在下列範例中，將範例 AMI ID 取代為有效 AMI ID，並以 12 位數 AWS 帳戶 ID 取代*account-id*。

```
aws ec2 modify-image-attribute \
    --image-id ami-0abcdef1234567890 \
    --launch-permission "Remove=[{UserId=account-id}]"
```

移除所有的啟動許可

下列命令會從指定的移除所有公用和明確的啟動權限AMI。請注意，的擁有者AMI始終具有啟動權限，因此不受此命令的影響。在下列範例中，將範例 AMI ID 取代為有效 AMI ID。

```
aws ec2 reset-image-attribute \
    --image-id ami-0abcdef1234567890 \
    --attribute launchPermission
```

共用 AMI (視窗工具 PowerShell)

使用Edit-EC2ImageAttribute命令 (Windows 工具 PowerShell) 來共用，AMI如下列範例所示。

授予明確啟動許可

下列命令會授與指定的指定AMI的啟動權限 AWS 帳戶。在下列範例中，將範例 AMI ID 取代為有效 AMI ID，並以 12 位數 AWS 帳戶 ID 取代*account-id*。

```
PS C:\> Edit-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -Attribute
 launchPermission -OperationType add -UserId "account-id"
```

> **Note**
>
> 您不需要共用AMI參考的 Amazon EBS 快照，即可共用AMI. 只需共用AMI本身；系統會自動為啟動提供執行個體存取參考的 Amazon EBS 快照。不過，您確實需要共用任何用來加密AMI參照之快照的KMS金鑰。如需詳細資訊，請參閱 Amazon EBS使用者指南中的共用 Amazon EBS 快照。

移除帳戶的啟動許可

下列命令會從指定的移除指定AMI的啟動權限 AWS 帳戶。在下列範例中，將範例 AMI ID 取代為有效 AMI ID，並以 12 位數 AWS 帳戶 ID 取代*account-id*。

```
PS C:\> Edit-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -Attribute
 launchPermission -OperationType remove -UserId "account-id"
```

移除所有的啟動許可

下列命令會從指定的移除所有公用和明確的啟動權限AMI。請注意，的擁有者AMI始終具有啟動權限，因此不受此命令的影響。在下列範例中，將範例 AMI ID 取代為有效 AMI ID。

```
PS C:\> Reset-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -Attribute
 launchPermission
```

取消與您的AMI共享 AWS 帳戶

Amazon 機器映像 (AMI) 可以 AWS 帳戶透過將帳戶新增至AMI的啟動許可，與特定人員共用。如果AMI已與您共享，但您不想再 AWS 帳戶 與您的帳戶共享該帳戶，則可以從啟動權限中刪除您AMI的帳戶。您可以透過執行`cancel-image-launch-permission` AWS CLI 命令來執行此操作。執行此命令時，您 AWS 帳戶 會從指定的啟動權限中移除AMI。若要尋找與AMIs您共用的 AWS 帳戶，請參閱尋找共用AMIs以用於 Amazon EC2 執行個體。

您可以取消與您的帳戶AMI共享的內容，例如，為了減少啟動與您共享的未使用或已棄AMI用的執行個體的可能性。當您取消與您的帳戶AMI共享時，它不會再出現在EC2控制台的任何AMI列表或描述圖像的輸出中。

主題

- 限制
- 取消與您的帳戶AMI共享

限制

- 您可以從僅與您共享的帳戶的啟動權限中刪除您的帳 AWS 帳戶 戶。AMI 您無法使用`cancel-image-launch-permission`從與組織或組織單位 (OU) AMI 共用的啟動權限中移除帳戶，也無法移除對公用的存取權AMIs。
- 您無法從的啟動權限中永久移除您的帳戶AMI。擁AMI有者可以再次AMI與您的帳戶共享。
- AMIs是一個區域資源。執行時`cancel-image-launch-permission`，您必須指定所在的 [AMI區域]。在指令中指定「區域」，或使用 AWS _ DEFAULT _ REGION 環境變數。

- 只有 AWS CLI 和SDKs支援將您的帳戶從AMI. 主EC2控台目前不支援此動作。

取消與您的帳戶AMI共享

> **Note**
> 取消與您的帳戶AMI共享後，您將無法復原。若要重新取得對的存取權AMI，AMI擁有者必須與您的帳戶共用它。

AWS CLI

若要取消與您的AMI分享內容 AWS 帳戶

使用指cancel-image-launch-permission令並指定 AMI ID。

```
aws ec2 cancel-image-launch-permission \
    --image-id ami-0123456789example \
    --region us-east-1
```

預期的輸出結果

```
{
    "Return": true
}
```

PowerShell

若要取消與您的AMI共用，請 AWS 帳戶 使用 AWS Tools for PowerShell

使用指Stop-EC2ImageLaunchPermission令並指定 AMI ID。

```
Stop-EC2ImageLaunchPermission `
    -ImageId ami-0123456789example `
    -Region us-east-1
```

預期的輸出結果

```
True
```

建立共用 Linux 的建議 AMIs

請使用下列準則來減少攻擊面，並改善AMIs您建立之攻擊面的可靠性。

> **Important**
>
> 沒有任何一份清單可詳盡列出所有的安全性準則。AMIs仔細建立您的共用內容，並花點時間考慮您可能會在哪裡公開敏感資料。

目錄

- 停用根使用者的密碼型態遠端登入
- 停用本機根存取
- 移除SSH主機金鑰配對
- 安裝公有金鑰登入資料
- 停用 sshd DNS 檢查 (選擇性)
- 移除敏感資料

如果您正在建造 AMIs AWS Marketplace，請參閱《AWS Marketplace 賣家指南》AMIs中建立的最佳做法，了解指南、政策和最佳做法。

如需有關AMIs安全共用的其他資訊，請參閱下列文章：

- 如何以安全的方式AMIs共享和使用公共
- 公共AMI出版：強化和清理要求

停用根使用者的密碼型態遠端登入

為公用使用固定的 root 密碼AMI是一種安全風險，可能會迅速成為人所知。即便仰賴使用者於第一次登入時變更密碼，都會留下一些可能遭到濫用的空窗。

若要解決此問題，請停用根使用者的密碼型態遠端登入。

停用根使用者的密碼型態遠端登入

1. 使用文字編輯器開啟 `/etc/ssh/sshd_config` 檔案，並找出下行：

```
#PermitRootLogin yes
```

2. 將行變為：

```
PermitRootLogin without-password
```

此組態檔案的位置可能會因您的發行版本而有所不同，或者如果您未執行「開啟」SSH。若是這種情況，請參閱相關文件。

停用本機根存取

當您使用共用時AMIs，最佳做法是停用直接根登入。若要停用，請登入運作中的執行個體，然後發出下列命令：

```
[ec2-user ~]$ sudo passwd -l root
```

> **Note**
>
> 此命令不會影響 sudo 的使用。

移除SSH主機金鑰配對

如果您打算共用AMI衍生自公用的AMI，請移除位於中的現有SSH主機金鑰配對`/etc/ssh`。當有人使SSH用您的執行個體啟動執行個體時，這會強制產生新的唯一SSH金鑰配對AMI，進而改善安全性並降低「man-in-the-middle」攻擊的可能性。

移除位在您系統中所有下列的金鑰檔案。

- ssh_host_dsa_key
- ssh_host_dsa_key.pub
- ssh_host_key
- ssh_host_key.pub
- ssh_host_rsa_key
- ssh_host_rsa_key.pub
- ssh_host_ecdsa_key

- ssh_host_ecdsa_key.pub
- ssh_host_ed25519_key
- ssh_host_ed25519_key.pub

您可以使用下列命令安全地移除所有這些檔案：

```
[ec2-user ~]$ sudo shred -u /etc/ssh/*_key /etc/ssh/*_key.pub
```

> **Warning**
>
> 像是 **shred** 等安全移除公用程式可能不會從儲存媒體中移除檔案的所有複本。檔案的隱藏副本可以透過記錄檔案系統 (包括 Amazon Linux 預設 ext4)、快照、RAID備份和暫時快取來建立。如需詳細資訊，請參閱 **shred** 文件。

> **Important**
>
> 如果您忘記從公開移除現有的SSH主機金鑰組AMI，我們的例行稽核程序會通知您和所有執行執行個體的客戶，瞭解您AMI的潛在安全風險。在短暫的寬限期後，我們紀念AMI私人。

安裝公有金鑰登入資料

設定AMI為防止使用密碼登入之後，您必須確定使用者可以使用其他機制登入。

Amazon EC2 允許使用者在啟動執行個體時指定公開-私 key pair 名稱。當有效的 key pair 名稱提供給`RunInstances`API呼叫時 (或透過命令列API工具)，公開金鑰 (Amazon 在呼叫`CreateKeyPair`或之後在伺服器上EC2保留的金鑰組部分`ImportKeyPair`) 可透過對執行個體中繼資料的HTTP查詢提供給執行個體。

若要透過登入SSH，您AMI必須在開機時擷取金鑰值，並將其附加至 `/root/.ssh/authorized_keys` (或對於上的任何其他使用者帳戶的對等值AMI)。使用者可以使用 key pair 啟動您AMI的執行個體並登入，而不需要 root 密碼。

包括 Amazon Linux 和 Ubuntu 等許多發行版本皆使用 `cloud-init` 套件來注入所設定使用者的公有金鑰登入資料。如果您的發行版本不支援 `cloud-init`，您可新增下列程式碼至系統啟動指令碼 (例如 `/etc/rc.local`)，以納入您在啟動時為根使用者指定的公有金鑰。

> **Note**
>
> 在下列範例中，IP 地址 http://169.254.169.254/ 是 link-local 地址且僅在執行個體中有效。

IMDSv2

```
if [ ! -d /root/.ssh ] ; then
        mkdir -p /root/.ssh
        chmod 700 /root/.ssh
fi
# Fetch public key using HTTP
TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-
token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-
data/public-keys/0/openssh-key > /tmp/my-key
if [ $? -eq 0 ] ; then
        cat /tmp/my-key >> /root/.ssh/authorized_keys
        chmod 700 /root/.ssh/authorized_keys
        rm /tmp/my-key
fi
```

IMDSv1

```
if [ ! -d /root/.ssh ] ; then
        mkdir -p /root/.ssh
        chmod 700 /root/.ssh
fi
# Fetch public key using HTTP
curl http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key > /tmp/my-key
if [ $? -eq 0 ] ; then
        cat /tmp/my-key >> /root/.ssh/authorized_keys
        chmod 700 /root/.ssh/authorized_keys
        rm /tmp/my-key
fi
```

這會套用到任何使用者；無需將其限制到 `root` 使用者。

> **Note**
> 根據此項重新組合執行個體，AMI包括啟動執行個體時所使用的金鑰。若要避免加入金鑰，您必須清除 (或刪除) `authorized_keys` 檔案或將此檔案排除於重新綁定之外。

停用 sshd DNS 檢查 (選擇性)

停用 sshd DNS 檢查會稍微削弱您的 sshd 安全性。但是，如果DNS解決失敗，SSH登錄仍然可以工作。如果您未停用 sshd 檢查，DNS解析失敗會阻止所有登入。

若要停用 sshd 檢查 DNS

1. 使用文字編輯器開啟 `/etc/ssh/sshd_config` 檔案，並找出下行：

```
#UseDNS yes
```

2. 將行變為：

```
UseDNS no
```

> **Note**
> 此組態檔案的位置可能會因您的發行版本或您未執行「開啟」而有所不同SSH。若是這種情況，請參閱相關文件。

移除敏感資料

我們建議您不要將敏感數據或軟件存儲在您共享AMI的任何內容上。啟動共用的使用者AMI可以將其重新捆綁，並將其註冊為自己的共享。請按照以下準則幫助您避免一些容易遭到忽視的安全風險：

- 建議使用 `--exclude` *`directory`* 的 `ec2-bundle-vol` 選項跳過含有您不想加入到套件內之機密資訊的任何目錄和子目錄。特別是，在捆綁映像時，排除所有用戶擁有的SSH公鑰/私鑰對和SSHauthorized_keys文件。Amazon 公共AMIs存儲這些`/root/.ssh`對於根用戶和普通`/home/`*`user_name`*`/.ssh/`用戶。如需詳細資訊，請參閱ec2-bundle-vol。
- 綁定前務必刪除 shell 歷程記錄。如果您嘗試上傳一個以上的捆綁包AMI，shell 歷史記錄包含您的訪問密鑰。以下範例應該是您在從執行個體內綁定之前執行的最後一項命令。

```
[ec2-user ~]$ shred -u ~/.*history
```

> **Warning**
>
> 上面警告中提及的 **shred** 限制同樣適用於此處。
> 請注意，bash 會在退出時將目前工作階段的歷程記錄寫入到磁碟。如果您在刪除 `~/.bash_history` 後登出執行個體，然後再重新登入，您會發現 `~/.bash_history` 已重新建立，並包含在先前工作階段中執行的所有命令。
> bash 以外的其他程式也會將歷程記錄寫入到磁碟。請小心使用，並移除或排除不必要以點開頭的檔案或以點開頭的目錄。

- 綁定執行中的執行個體需要您的私有金鑰和 X.509 憑證。因此請將這些憑證和其他登入資料放在不會加入綁定的位置 (例如執行個體存放區)。

使用 Amazon 監控AMI事件 EventBridge

當 Amazon 機器映像（AMI）的狀態發生變化時，Amazon EC2 生成一個發送到 Amazon 的事件 EventBridge （以前稱為 Amazon CloudWatch 活動）。事件會以JSON格式傳送至預設 EventBridge 事件匯流排。您可以使用 Amazon EventBridge 偵測這些事件並做出回應。您可以透過在其中建立規則 EventBridge 來觸發動作以回應事件來執行此操作。例如，您可以建立 EventBridge 規則來偵測建AMI立程序何時完成，然後叫用 Amazon SNS 主題以傳送電子郵件通知給您。

Amazon 在AMI進入以下任何狀態時EC2生成一個`EC2 AMI State Change`事件：

- `available`
- `failed`
- `deregistered`
- `disabled`

盡可能產生事件。

下表列出AMI可以輸入的AMI作業和狀態。在表格中，「是」 指出在對應作業執行時AMI可以輸入的狀態。

AMI操作	available	failed	deregistered	disabled
CopyImage	是	是		
CreateImage	是	是		
CreateRestoreImageTask	是	是		
DeregisterImage			是	
DisableImage				是
EnableImage	是			
RegisterImage	是	是		

EC2 AMI State Change 事件

- 事件詳細資訊
- available 事件
- failed 事件
- deregistered 事件
- disabled 事件

事件詳細資訊

您可以在事件中使用下列欄位來建立觸發動作的規則：

`"source": "aws.ec2"`

識別該事件來自 Amazon EC2。

`"detail-type": "EC2 AMI State Change"`

識別事件名稱。

`"detail": { "ImageId": "ami-0123456789example", "State": "available", }`

提供 AMI (`available`、、`failed`或`disabled`) 的 AMI ID 和狀態。`deregistered`

如需詳細資訊，請參閱 Amazon EventBridge 使用者指南中的以下內容：

- Amazon EventBridge 活動
- Amazon EventBridge 事件模式
- Amazon EventBridge 規則

如需如何建立 Lambda 函數的教學課程以及執行 Lambda 函數的 EventBridge 規則，請參閱AWS Lambda 開發人員指南 EventBridge中的教學課程：使用記錄 Amazon EC2 執行個體的狀態。

available 事件

以下是 Amazon 在成功`CreateImage`、、`CopyImage`或`EnableImage`操作後AMI進入`available`狀態時EC2產生的事件範例。`RegisterImage` `CreateRestoreImageTask`

`"State": "available"` 表示操作成功。

```
{
    "version": "0",
    "id": "example-9f07-51db-246b-d8b8441bcdf0",
    "detail-type": "EC2 AMI State Change",
    "source": "aws.ec2",
    "account": "012345678901",
    "time": "yyyy-mm-ddThh:mm:ssZ",
    "region": "us-east-1",
    "resources": ["arn:aws:ec2:us-east-1::image/ami-0123456789example"],
    "detail": {
        "RequestId": "example-9dcc-40a6-aa77-7ce457d5442b",
        "ImageId": "ami-0123456789example",
        "State": "available",
        "ErrorMessage": ""
    }
}
```

failed 事件

以下是 Amazon 在失敗`CreateImage`、、`CopyImage`或`CreateRestoreImageTask`操作後AMI進入`failed`狀態時EC2產生的事件範例。`RegisterImage`

下列欄位會提供相關資訊：

- `"State": "failed"` - 表示操作失敗。

- `"ErrorMessage": ""` - 提供操作失敗的原因。

```
{
    "version": "0",
    "id": "example-9f07-51db-246b-d8b8441bcdf0",
    "detail-type": "EC2 AMI State Change",
    "source": "aws.ec2",
    "account": "012345678901",
    "time": "yyyy-mm-ddThh:mm:ssZ",
    "region": "us-east-1",
    "resources": ["arn:aws:ec2:us-east-1::image/ami-0123456789example"],
    "detail": {
        "RequestId": "example-9dcc-40a6-aa77-7ce457d5442b",
        "ImageId": "ami-0123456789example",
        "State": "failed",
        "ErrorMessage": "Description of failure"
    }
}
```

deregistered 事件

以下是 Amazon 在成功`DeregisterImage`操作後AMI進入`deregistered`狀態時EC2產生的事件範例。如果操作失敗，則不會產生任何事件。因為 `DeregisterImage` 是一個同步操作，所以會立刻發現所有失敗。

`"State": "deregistered"` 表示 `DeregisterImage` 操作成功。

```
{
    "version": "0",
    "id": "example-9f07-51db-246b-d8b8441bcdf0",
    "detail-type": "EC2 AMI State Change",
    "source": "aws.ec2",
    "account": "012345678901",
    "time": "yyyy-mm-ddThh:mm:ssZ",
    "region": "us-east-1",
    "resources": ["arn:aws:ec2:us-east-1::image/ami-0123456789example"],
    "detail": {
        "RequestId": "example-9dcc-40a6-aa77-7ce457d5442b",
        "ImageId": "ami-0123456789example",
        "State": "deregistered",
        "ErrorMessage": ""
    }
```

```
}
```

disabled 事件

以下是 Amazon 在成功DisableImage操作後AMI進入disabled狀態時EC2產生的事件範例。如果操作失敗，則不會產生任何事件。因為 DisableImage 是一個同步操作，所以會立刻發現所有失敗。

"State": "disabled" 表示 DisableImage 操作成功。

```
{
    "version": "0",
    "id": "example-9f07-51db-246b-d8b8441bcdf0",
    "detail-type": "EC2 AMI State Change",
    "source": "aws.ec2",
    "account": "012345678901",
    "time": "yyyy-mm-ddThh:mm:ssZ",
    "region": "us-east-1",
    "resources": ["arn:aws:ec2:us-east-1::image/ami-0123456789example"],
    "detail": {
        "RequestId": "example-9dcc-40a6-aa77-7ce457d5442b",
        "ImageId": "ami-0123456789example",
        "State": "disabled",
        "ErrorMessage": ""
    }
}
```

了解 AMI 帳單資訊

啟動執行個體時，有多種 Amazon Machine Image (AMI) 可供選擇，而且它們支援各種作業系統平台和功能。若要瞭解您在啟動執行個體時選擇的 AMI 如何影響帳 AWS 單的底線，您可以研究相關的作業系統平台和帳單資訊。在啟動任何隨需或 競價型執行個體，或購買 預留執行個體 之前執行此動作。

以下兩個例子說明事先研究 AMI 如何幫助您選擇最符合需求的 AMI：

- 針對 競價型執行個體，您可以使用 AMI 平台詳細資訊來確認 競價型執行個體 是否支援 AMI。
- 購買 預留執行個體 時，您可確保對應 AMI 平台詳細資訊選取操作系統平台 (平台)。

如需執行個體定價的詳細資訊，請參閱 Amazon EC2 定價。

內容

- AMI 帳單資訊欄位
- 尋找 AMI 帳單和用量詳細資訊
- 驗證帳單上的 AMI 費用

AMI 帳單資訊欄位

下列欄位提供與 AMI 相關的帳單資訊：

平台詳細資料

與 AMI 帳單代碼相關的平台詳細資訊。例如：`Red Hat Enterprise Linux`。

使用操作

Amazon EC2 執行個體的操作以及與 AMI 相關聯的帳單代碼。例如：`RunInstances:0010`。使用量作業與「成本與用途報表」(CUR) 與「 AWS 價目表 API」中的「生產線料號/作業」欄位相對 AWS 應。

您可以在 Amazon EC2 主控台的執行個體或 AMI 頁面上檢視這些欄位，或在描述影像或命令傳回的回應中檢視這些欄位。Get-EC2Image

範例資料：使用操作 (依平台)

下表列出可在 Amazon EC2 主控台的執行個體或 AMI 頁面上顯示的一些平台詳細資訊和使用情況操作值，或在描述影像或命令傳回的回應中顯示。Get-EC2Image

平台詳細資料	使用操作 [2]
Linux/UNIX	RunInstances
Red Hat BYOL Linux	RunInstances:00g0[3]
Red Hat Enterprise Linux	RunInstances:0010
Red Hat Enterprise Linux with HA	RunInstances:1010

平台詳細資料	使用操作 [2]
Red Hat Enterprise Linux with SQL Server Standard and HA	RunInstances:1014
Red Hat Enterprise Linux with SQL Server Enterprise and HA	RunInstances:1110
Red Hat Enterprise Linux with SQL Server Standard	RunInstances:0014
Red Hat Enterprise Linux with SQL Server Web	RunInstances:0210
Red Hat Enterprise Linux with SQL Server Enterprise	RunInstances:0110
SQL Server Enterprise	RunInstances:0100
SQL Server Standard	RunInstances:0004
SQL Server Web	RunInstances:0200
SUSE Linux	RunInstances:000g
Ubuntu Pro	RunInstances:0g00
Windows	RunInstances:0002
Windows BYOL	RunInstances:0800
Windows with SQL Server Enterprise[1]	RunInstances:0102
Windows with SQL Server Standard[1]	RunInstances:0006
Windows with SQL Server Web[1]	RunInstances:0202

[1] 如果兩個軟體授權與 AMI 相關聯，平台詳細資料欄位會同時顯示兩者。

[2] 如果您正在執行 Spot 執行個體，「AWS 成本和使用量報告」lineitem/Operation上的可能與此處列出的使用量作業值不同。例如，如果`lineitem/Operation`顯示`RunInstances:0010:SV006`，表示 Amazon EC2 在區域 6 的美國東部 (維吉尼亞北部) 執行 RHEL Spot 執行個體小時數。

[3] 這會顯示RunInstances (Linux/UNIX)在您的使用情況報告中。

尋找 AMI 帳單和用量詳細資訊

在 Amazon EC2 主控台中，您可以從 AMI 頁面或執行個體頁面檢視 AMI 帳單資訊。您也可以使用 AWS CLI 或執行個體中繼資料服務尋找帳單資訊。

下列欄位可協助您驗證帳單上的 AMI 費用：

- 平台詳細資料
- 使用操作
- AMI ID

尋找 AMI 帳單資訊 (主控台)

請按照下列步驟在 Amazon EC2 主控台中檢視 AMI 帳單資訊：

從 AMI 頁面查詢 AMI 帳單資訊

1. 在 https://console.aws.amazon.com/ec2/ 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選取 AMI，然後選取一個 AMI。
3. 在 Details (詳細資料) 索引標籤上，檢查 Platform details (平台詳細資料) 和 Usage operation (使用操作) 的值。

從 Instances (執行個體) 頁面查詢 AMI 帳單資訊

1. 在 https://console.aws.amazon.com/ec2/ 開啟 Amazon EC2 主控台。
2. 在導覽窗格中，選取 Instances (執行個體)，然後選取一個執行個體。
3. 在 Details (詳細資訊) 索引標籤 (如果您使用的是舊版主控台，則是 Description (說明) 索引標籤) 上，查看 Platform details (平台詳細資訊) 和 Usage operation (使用操作) 的值。

尋找 AMI 帳單資訊 (AWS CLI)

若要使用尋找 AMI 帳單資訊 AWS CLI，您必須知道 AMI 識別碼。若不知道 AMI ID，可以使用 describe-instances 命令來從執行個體取得。

尋找 AMI ID

如果知道執行個體 ID，可以使用 describe-instances 命令來為執行個體取得 AMI ID。

```
aws ec2 describe-instances --instance-ids i-123456789abcde123
```

在輸出中，AMI ID 會在 `ImageId` 欄位中指定。

```
..."Instances": [
{
    "AmiLaunchIndex": 0,
    "ImageId": "ami-0123456789EXAMPLE",
    "InstanceId": "i-123456789abcde123",
    ...
}]
```

尋找 AMI 帳單資訊

如果知道 AMI ID，可以使用 describe-images 命令取得 AMI 平台和使用操作詳細資訊：

```
$ aws ec2 describe-images --image-ids ami-0123456789EXAMPLE
```

下列範例輸出顯示 `PlatformDetails` 和 `UsageOperation` 欄位。在此範例中，ami-0123456789EXAMPLE 平台是 `Red Hat Enterprise Linux`，而使用操作和計費代碼為 `RunInstances:0010`。

```
{
    "Images": [
        {
            "VirtualizationType": "hvm",
            "Description": "Provided by Red Hat, Inc.",
            "Hypervisor": "xen",
            "EnaSupport": true,
            "SriovNetSupport": "simple",
            "ImageId": "ami-0123456789EXAMPLE",
            "State": "available",
            "BlockDeviceMappings": [
```

```
                {
                    "DeviceName": "/dev/sda1",
                    "Ebs": {
                        "SnapshotId": "snap-111222333444aaabb",
                        "DeleteOnTermination": true,
                        "VolumeType": "gp2",
                        "VolumeSize": 10,
                        "Encrypted": false
                    }
                }
            ],
            "Architecture": "x86_64",
            "ImageLocation": "123456789012/RHEL-8.0.0_HVM-20190618-x86_64-1-Hourly2-
GP2",
            "RootDeviceType": "ebs",
            "OwnerId": "123456789012",
            "PlatformDetails": "Red Hat Enterprise Linux",
            "UsageOperation": "RunInstances:0010",
            "RootDeviceName": "/dev/sda1",
            "CreationDate": "2019-05-10T13:17:12.000Z",
            "Public": true,
            "ImageType": "machine",
            "Name": "RHEL-8.0.0_HVM-20190618-x86_64-1-Hourly2-GP2"
        }
    ]
}
```

驗證帳單上的 AMI 費用

為了確保您不會產生意外費用，您可以驗證 AWS 成本和用量報告 (CUR) 中執行個體的帳單資訊是否與您用來啟動執行個體的 AMI 相關聯的帳單資訊相符。

若要確認帳單資訊，請在 CUR 中尋找執行個體 ID，然後檢查 `lineitem/Operation` 欄中對應的值。該值應該符合 AMI 相關聯使用操作 的值。

例如，AMI `ami-0123456789EXAMPLE` 具有下列帳單資訊：

- 平台詳細資訊 = `Red Hat Enterprise Linux`
- 使用操作 = `RunInstances:0010`

如果使用此 AMI 啟動執行個體，可在 CUR 中找到該執行個體 ID，然後檢查 `lineitem/Operation` 欄中的相應值。在此範例中，此值應為 `RunInstances:0010`。

AMIAmazon 的配額 EC2

下列配額適用於建立和共用AMIs。這些配額適用於每個 AWS 區域。

配額名稱	描述	每個區域的預設配額
AMIs	每個區域AMIs允許的最大公共和私人數量。這些包括可用、擱置中和已停用AMIs，以及AMIs在資源回收筒中。	50,000
公眾 AMIs	每個區域允許的公開人數上限AMIs，包括資源回收筒AMIs中的公開人數。	5
AMI分享	區域中AMI可與之共用的實體 (組織、組織單位 (OUs) 和帳戶) 的最大數目。請注意，如果您AMI與組織或 OU 共用，組織或 OU 中的帳號數目不會計入配額中。	1,000

如果您超過配額且想要建立或共用更多配額AMIs，可以執行下列動作：

- 如果您超出總額AMIs或公開AMIs配額，請考慮取消註冊未使用的映像檔。
- 如果您超出公開AMIs配額，請考慮將一個或多個公開AMIs設為私有。
- 如果您超出AMI共用配額，請考慮AMIs與組織或 OU 共用，而不是與個別帳戶共用。
- 要求增加的配額AMIs。

要求增加配額 AMIs

如果您需要的預設配額超過AMIs，您可以要求提高配額。

若要要求提高配額 AMIs

1. 在開啟「Service Quotas」主控台https://console.aws.amazon.com/servicequotas/。

2. 在導覽窗格中，選擇 AWS services (AWS 服務)。
3. 從清單中選擇 Amazon 彈性運算雲端 (AmazonEC2)，或在搜尋方塊中輸入服務的名稱。
4. 選擇要求增加的AMI配額。您可以選取的AMI配額為：
   - AMIs
   - 公眾 AMIs
   - AMI分享
5. 選擇 Request quota increase (請求增加配額)。
6. 對於 Change quota value (變更配額值)，輸入新的配額值，然後選擇 Request (請求)。

若要檢視任何擱置或最近解決的請求，請從瀏覽窗格中選擇 Dashboard (儀表板)。對於擱置的請求，請選擇請求狀態以開啟請求回條。請求的初始狀態為 Pending (待定)。狀態變更為 Quota requested (已請求配額) 後，您會在 Support Center case number (支援中心案例編號) 中看到案例編號。選擇案例編號，為請求開啟票證。

解決請求之後，配額的 Applied quota value (套用的配額值) 會設定為新值。

如需詳細資訊，請參閱 Service Quotas 使用者指南。

Amazon EC2 實例

Amazon EC2 執行個體是 AWS 雲端環境中的虛擬伺服器。從您第一次啟動執行個體 (稱為啟動執行個體) 到刪除執行個體 (稱為終止執行個體)，您就可以完全掌控執行個體。啟動執行個體時，您可以從各種作業系統中進行選擇。您可以連線至執行個體，並加以自訂以符合您的需求。例如，您可以設定作業系統、安裝作業系統更新，以及在執行個體上安裝應用程式。

Amazon EC2 提供了廣泛的實例類型。您可以選擇執行應用程式所需的運算資源、記憶體、儲存和網路效能的執行個體類型。

使用 AmazonEC2，您只需為使用量付費。當您啟動執行個體並轉換為執行中狀態時，就會開始計費執行個體。停止執行個體時會停止計費，並在您啟動執行個體時繼續執行。當您終止執行個體時，會在執行個體轉換至關閉狀態時停止計費。

Amazon EC2 提供的功能可讓您用來優化執行個體的效能和成本。例如，您可以使用 Amazon EC2 Fleet 或 Amazon EC2 Auto Scaling，隨著執行個體使用率的變化而擴展或縮減容量。您可以使用 Spot 執行個體或 Savings Plans 降低執行個體的成本。

功能和任務

- Amazon EC2 實例類型
- Amazon EC2 帳單和購買選項
- 在 Amazon 啟動範本中存放執行個體EC2啟動參數
- 啟動 Amazon EC2 實例
- Connect 至您的EC2執行個體
- Amazon EC2 實例狀態更改
- 使用執行個體中繼資料管理EC2執行個
- 偵測主機是否為EC2執行個體
- Amazon EC2 執行個體的執行個體身分證
- EC2執行個體上的精確時脈和時間同步
- 管理EC2執行個體的裝置驅動程式
- 設定您的EC2視 Amazon 執行個體
- 將EC2視窗執行個體升級至較新版本的視窗伺服器
- 教學：將 Amazon EC2 執行個體 Connect 到 Amazon RDS 資料庫

Amazon EC2 實例類型

啟動執行個體時，您指定的執行個體類型將決定執行個體所用的主機電腦硬體。每個執行個體類型均提供不同的運算、記憶體和儲存功能，且會依照這些功能分組為不同的執行個體系列。請根據您計劃在執行個體上執行之應用程式或軟體的需求選取執行個體類型。

Amazon EC2 將主機電腦的一些資源專用於特定執行個體CPU，例如記憶體和執行個體儲存。Amazon 會在執行個體之間EC2共用主機電腦的其他資源，例如網路和磁碟子系統。如果主機電腦上的每個執行個體嘗試對其中一項共用資源作最大的利用，該資源會平均分配給每個執行個體。但是，當某一資源使用率較低時，執行個體便能使用該資源較高的比例 (需有資源可用)。

每個執行個體類型從共同資源獲得的最低效能可能更高或更低。例如，具有高 I/O 效能的執行個體類型會分配到較多的共用資源。分配較多的共用資源，也可減少 I/O 效能的變動。對多數的應用程式來說，中等的 I/O 效能即堪使用。但是，對於需要更高或更一致 I/O 效能的應用程式，請考慮使用具有較高 I/O 效能的執行個體類型。

內容

- 可用的執行個體類型
- 硬體規格
- AMI虛擬化類型
- 尋找 Amazon 執行個EC2體類型
- 從EC2執行個體類型尋找工具取得
- 從 Compute Optimizer 取得EC2執行個體
- Amazon EC2 實例類型更改
- 爆量效能執行個體
- GPU執行個體的效能加速
- Amazon EC2 Mac 實例
- Amazon EBS 優化執行個體類型
- CPUAmazon EC2 實例的選項
- AMDSEV-SNP 適用於 Amazon EC2 實例
- Amazon EC2 Linux 實例的處理器狀態控制

可用的執行個體類型

Amazon EC2 提供多種針對不同使用案例進行優化的執行個體類型選擇。執行個體類型包含不同的記憶體CPU、儲存和網路容量組合，讓您彈性選擇適合應用程式的資源組合。每個執行個體類型都包含一或多個執行個體大小，可讓您根據目標工作負載的需求擴展資源。 如需功能和使用案例的詳細資訊，請參閱 Amazon EC2 執行個體類型詳細資訊。

實例類型命名慣例

名稱取決於執行個體系列、層代、處理器系列、功能和大小。如需詳細資訊，請參閱 Amazon EC2 執行個體類型指南中的命名慣例。

尋找執行個體類型

若要判斷哪些執行個體類型符合您的需求，例如支援的區域、運算資源或儲存資源，請參閱 Amazon EC2 執行個體類型指南中的尋找 Amazon 執行個EC2體類型和 Amazon EC2 執行個體類型規格。

最新一代的執行個體

- 一般用途：M5 | M5Ad | 兆廣播公司 | M5dN | 兆豐 | M6g | M6gD | M6i | M6i | M6-柔性 | M6-柔性 | M6 英寸 | M7a | 麥克澳門平方米 | 澳門 2-平方米 | T2 | T3 | T4G
- 計算最佳化:C5 | C5Ad | C5 毫升 | C5 毫升 | C5 安 | C6 克重 | 氯化碳 | 氯化碳 | C6 英寸 | C6 英寸 | C7 克 | 氯化碳 | 氯化碳 | 氯化碳
- 記憶體最佳化:R5 | R5 安培 | R5 倍 | R5dN | R5dn | R5n | R6 克 | R6gd | R6 億 | R6 英寸 | U 型 U-9TB1 | U-12 兆比 1 | U-18 兆比 1 | U-24 兆比 1 | 烏 7i-十二 TB | 烏 7 英寸 -24 TB | 烏 7 英寸 32 TB | X2 國際化數據分析 | X2 國際化 DN | X2 國際化 DN | X2iedn | X1e | Z1D
- 儲存最佳化：D2 | D3 | 第 3 天 | H1 | I3 | I3en | I4G | I4i | IM4GN | i4GN
- 加速運算:DL1| | F1 DL2q | 四倍 | G4dN | G5 | G5G | G6 | G6E | GR6 | 資料一次 | 二級二 | P3 | P4dN | P4dN | P4de | P4de | P5 | 中一號 | TRN1N | VT1
- 高效能運算：高效能運算 | HPC6a | HPC7g

上一代執行個體

- 一般用途：A1 | M1 | M2 | 立方米 | M4
- 運算最佳化：C1 | C3
- 記憶體最佳化：
- 儲存最佳化：I2

- 加速運算：G3

硬體規格

如需詳細的執行個體類型規格，請參閱 Amazon EC2 執行個體類型指南中的規格。如需定價資訊，請參閱 Amazon EC2 隨需定價。

為了決定最符合您需求的執行個體類型，我們建議您啟動執行個體並使用您自己的基準參考應用程式。由於您必須為執行個體按秒數付費，因此在作決定前先測試多個執行個體類型是比較方便且省錢的方式。假如您在作決定後需要變更，也可稍後再變更執行個體類型。如需詳細資訊，請參閱Amazon EC2 實例類型更改。

Intel 處理器功能

在 Intel 處理器上執行的 Amazon 執行EC2個體可能包含下列功能。並非所有下列處理器功能都受到所有執行個體類型的支援。如需每種執行個體類型可用功能的詳細資訊，請參閱 Amazon EC2 執行個體類型。

- Intel AES 新指令 (AES-NI) — Intel AES-NI 加密指令集改良了原始的進階加密標準 (AES) 演算法，可提供更快速的資料保護與更高的安全性。目前所有的EC2執行個體都支援此處理器功能。
- 英特爾高級矢量擴展（英特爾 AVXAVX2，英特爾和英特爾 AVX -512）— 英特爾AVX和英特爾 AVX2是 256 位，英特爾 AVX -512 是 512 位指令集擴展，專為浮點（FP）密集型應用而設計。Intel AVX 指令可改善影像與音訊/視訊處理、科學模擬、財務分析，以及 3D 建模與分析等應用程式的效能。這些功能僅適用於使用啟動的執行個體HVMAMIs。
- Intel Turbo Boost Technology (Intel 渦輪加速技術) – Intel Turbo Boost Technology (Intel 渦輪加速技術) 處理器會自動執行核心，比基礎操作頻率更快。
- Intel Deep Learning Boost (Intel DL Boost) – 可加速 AI 深度學習使用案例。第 2 代 Intel Xeon 可擴充處理器以全新的向量神經網路指令 (VNNI/INT8) 擴充 Intel AVX -512，相較於前一代 Intel Xeon 可擴充處理器 (含FP32)，可大幅提升深度學習推論效能，適用於影像辨識/分段、物件偵測、語音辨識、語言翻譯、推薦系統、強化學習等。VNNI可能與所有 Linux 發行版都不相容。

  下列執行個體支援VNNI：`M5nR5nM5dnM5zn`、`R5b`、`R5dn`、、`D3`、`D3en`、和`C6i`。 `C5`和`C5d`執行個體僅VNNI支援`12xlarge``24xlarge`、和`metal`執行個體。

64 位元的產業命名慣例可能會造成混淆CPUs。晶片製造商進階微型裝置 (AMD) 推出了第一個以 Intel x86 指令集為基礎的商業上成功的 64 位元架構。因此，無論芯片製造商如何，該架構AMD64都被廣泛

稱為。Windows 和多個 Linux 發行版本也遵照此作法。這說明為何執行 Ubuntu 或 Windows 執行個體上的內部系統資訊會顯示CPU架構，AMD64就像執行個體在 Intel 硬體上執行一樣。

AWS 重力子處理器

AWS Graviton 是一系列處理器，旨在為您在 Amazon EC2 執行個體上執行的工作負載提供最優惠的價格效能。

如需詳細資訊，請參閱重力子入門。

AWS 草屬

由 AWS Trainium 提供支援的執行個體專為高效能、具成本效益的深度學習訓練而打造。您可以使用這些執行個體來訓練自然語言處理、電腦視覺，以及廣泛應用程式所使用的推薦模型，例如語音辨識、建議、詐騙偵測，以及影像和影片分類。在常見的 ML 架構中使用您現有的工作流程，例如 PyTorch 和 TensorFlow。

AWS 推論

由 AWS Inferentia 提供支援的執行個體專為加速機器學習而設計。它們提供高效能和低延遲的機器學習推論。這些執行個體已進行最佳化，適用於部署不同應用的深度學習 (DL) 模型，例如自然語言處理、物件偵測與分類、內容個人化與篩選，以及語音辨識等應用。

您可以透過多種方式開始使用：

- 使用完全受管理的服務 SageMaker，是開始使用機器學習模型的最簡單方法。如需詳細資訊，請參閱 Amazon 開 SageMaker 發人員指南 SageMaker中的入門使用。
- 使用深度學習啟動 Inf1 或 Inf2 執行個體。AMI如需詳細資訊，請參閱AWS Deep Learning AMI 開發人員指AWS 南DLAMI中的推論。
- 使用您自己的 Inf1 或 Inf2 執行個體AMI並安裝 AWS Neuron SDK，以便編譯、執行和分析推論的深度學習模型。 AWS
- 使用 Inf1 或 Inf2 執行個體以及 Amazon ECS 最佳化的執行個體啟動容器執行個體。AMI如需詳細資訊，請參閱 Amazon 彈性容器服務開發人員指南AMIs中的 Amazon Linux 2 (推論)。
- 使用執行 Inf1 執行個體的節點建立 Amazon EKS 叢集。如需詳細資訊，請參閱 Amazon EKS 使用者指南中的推論支援。

AMI虛擬化類型

執行個體的虛擬化類型取決於您用來啟動執行個體的虛擬化類型。AMI目前一代的執行個體類型僅支援硬體虛擬機器 (HVM)。上一代的某些執行個體類型支援半虛擬 (PV)，有些 AWS 區域支援 PV 執行個體。如需詳細資訊，請參閱虛擬化類型。

為了獲得最佳效能，建議您使用 HVMAMI. 此外，HVMAMIs還需要利用增強型網路功能。HVM虛擬化使用 AWS 平台提供的硬體輔助技術。使用HVM虛擬化時，客體虛擬機器會像在原生硬體平台上一樣執行，但它仍然使用 PV 網路和儲存驅動程式來提升效能。

尋找 Amazon 執行個EC2體類型

您必須先選取要使用的執行個體類型，才能啟動執行個體。您選擇的執行個體類型可能取決於您的工作負載所需的資源，如運算、記憶體或儲存資源。識別數種可能適合您的工作負載的執行個體類型，並評估其在測試環境中的效能，可能是有利的做法。沒有任何替代方法可以測量您的應用程式在負載條件下的效能。

您可以使用執行個體類型尋找工具取得EC2執行個EC2體類型的建議和指引。如需詳細資訊，請參閱the section called “EC2實體類型尋找工”。

如果您已經有執行中的EC2執行個體，您可以用 AWS Compute Optimizer 來取得執行個體類型的建議，以提升效能、節省成本，或兩者兼而有之。如需詳細資訊，請參閱 the section called “Compute Optimizer 建議”。

任務

- 使用主控台尋找執行個體類型
- 使用描述執行個體類型 AWS CLI
- 使用尋找執行個體類型 AWS CLI

使用主控台尋找執行個體類型

您可以使用 Amazon EC2 主控台找到符合您需求的執行個體類型。

使用主控台尋找執行個體類型

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 從導覽列選取要在其中啟動執行個體的區域。無論您的位置為何，皆可選取任何可用的區域。
3. 在導覽窗格中，選擇 Instance Types (執行個體類型)。

4. (選用) 選取偏好設定 (齒輪) 圖示來選取要顯示的執行個體類型屬性，例如隨需 Linux 定價，然後選取 Confirm (確認)。或者，選取執行個體類型的名稱，以開啟其詳細資訊頁面，並透過主控台檢視可用的所有屬性。主控台不會顯示所有可透過API或指令列使用的屬性。
5. 使用執行個體類型屬性，將顯示的執行個體類型清單篩選為僅符合您需求的執行個體類型。例如，您可以篩選下列屬性：
   - Availability zones (可用區域) – 可用區域、本機區域或 Wavelength 區域的名稱。如需詳細資訊，請參閱the section called “區域 (Region) 和區域 (Zone)”。
   - vCPUs或 [核心] — vCPUs 或核心的數目。
   - Memory (GiB) (記憶體 (GiB)) – 記憶體大小 (以 GiB 為單位)。
   - Network performance (網路效能) – 網路效能 (以 GB 為單位)。
   - Local instance storage (本機執行個體儲存體) – 表示執行個體類型是否具有本機執行個體儲存體 (`true` | `false`)。
6. (選擇性) 若要查看 side-by-side 比較，請選取多個執行個體類型的核取方塊。比較結果顯示在螢幕畫面的底部。
7. (選擇性) 若要將執行個體類型清單儲存為逗號分隔值 (.csv) 檔案以供進一步檢閱，請選擇動作，下載清單。CSV檔案包括符合您所設定篩選的所有執行個體類型。
8. (選用) 若要使用滿足您需求的執行個體類型來啟動執行個體，請選取執行個體類型的核取方塊，然後選取 Actions (動作)、Launch instance (啟動執行個體)。如需詳細資訊，請參閱使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體。

使用描述執行個體類型 AWS CLI

您可以使用指describe-instance-types令來描述特定執行個體類型。

若要完整描述執行個體類型

下列命令會顯示指定執行個體類型的所有可用詳細資料。輸出很長，所以在這裡省略。

```
aws ec2 describe-instance-types \
    --instance-types t2.micro \
    --region us-east-2
```

描述執行個體類型並篩選輸出

下列命令會顯示指定執行個體類型的網路詳細資料。

```
aws ec2 describe-instance-types \
    --instance-types t2.micro \
    --region us-east-2 \
    --query "InstanceTypes[].NetworkInfo"
```

下列為範例輸出。

```
[
    {
        "NetworkPerformance": "Low to Moderate",
        "MaximumNetworkInterfaces": 2,
        "MaximumNetworkCards": 1,
        "DefaultNetworkCardIndex": 0,
        "NetworkCards": [
            {
                "NetworkCardIndex": 0,
                "NetworkPerformance": "Low to Moderate",
                "MaximumNetworkInterfaces": 2,
                "BaselineBandwidthInGbps": 0.064,
                "PeakBandwidthInGbps": 1.024
            }
        ],
        "Ipv4AddressesPerInterface": 2,
        "Ipv6AddressesPerInterface": 2,
        "Ipv6Supported": true,
        "EnaSupport": "unsupported",
        "EfaSupported": false,
        "EncryptionInTransitSupported": false,
        "EnaSrdSupported": false
    }
]
```

下列命令會顯示指定執行個體類型的可用記憶體。

```
aws ec2 describe-instance-types \
    --instance-types t2.micro \
    --region us-east-2 \
    --query "InstanceTypes[].MemoryInfo"
```

下列為範例輸出。

```
[
```

```
    {
        "SizeInMiB": 1024
    }
]
```

使用尋找執行個體類型 AWS CLI

您可以使用describe-instance-types和describe-instance-type-offerings指令尋找符合您需求的執行個體類型。

範例

- 範例 1：依可用區域尋找執行個體類型
- 範例 2：依可用記憶體大小尋找執行個體類型
- 範例 3：依可用執行個體儲存體尋找執行個體類型
- 範例 4：尋找支援休眠的執行個體類型

範例 1：依可用區域尋找執行個體類型

下列範例只會顯示指定可用區域中提供的執行個體類型。

```
aws ec2 describe-instance-type-offerings --location-type "availability-zone" \
    --filters "Name=location,Values=us-east-2a" \
    --region us-east-2 \
    --query "InstanceTypeOfferings[*].[InstanceType]" --output text | sort
```

輸出為執行個體類型清單，依字母順序排序。以下僅是輸出的開始。

```
a1.2xlarge
a1.4xlarge
a1.large
a1.medium
a1.metal
a1.xlarge
c4.2xlarge
   ...
```

範例 2：依可用記憶體大小尋找執行個體類型

下列範例只會顯示具有 64 GiB (65536 MiB) 記憶體的目前一代執行個體類型。

```
aws ec2 describe-instance-types \
    --filters "Name=current-generation,Values=true" "Name=memory-info.size-in-
mib,Values=65536" \
    --region us-east-2 \
    --query "InstanceTypes[*].[InstanceType]" --output text | sort
```

輸出為執行個體類型清單，依字母順序排序。以下僅是輸出的開始。

```
c5a.8xlarge
c5ad.8xlarge
c6a.8xlarge
c6g.8xlarge
c6gd.8xlarge
c6gn.8xlarge
c6i.8xlarge
c6id.8xlarge
c6in.8xlarge
   ...
```

範例 3：依可用執行個體儲存體尋找執行個體類型

下列範例顯示具有執行個體存放磁碟區之所有 R7 執行個體的執行個體儲存體總大小。

```
aws ec2 describe-instance-types \
    --filters "Name=instance-type,Values=r7*" "Name=instance-storage-
supported,Values=true" \
    --region us-east-2 \
    --query "InstanceTypes[].[InstanceType, InstanceStorageInfo.TotalSizeInGB]" \
    --output table
```

下列為範例輸出。

```
----------------------------
|  DescribeInstanceTypes  |
+----------------+--------+
|  r7gd.xlarge   |  237   |
|  r7gd.8xlarge  |  1900  |
|  r7gd.16xlarge |  3800  |
|  r7gd.medium   |  59    |
|  r7gd.4xlarge  |  950   |
|  r7gd.2xlarge  |  474   |
```

```
|  r7gd.metal    |  3800  |
|  r7gd.large    |  118   |
|  r7gd.12xlarge |  2850  |
+----------------+--------+
```

範例 4：尋找支援休眠的執行個體類型

下列範例會顯示支援休眠的執行個體類型。

```
aws ec2 describe-instance-types \
    --filters "Name=hibernation-supported,Values=true" \
    --region us-east-2 \
    --query "InstanceTypes[*].[InstanceType]" \
    --output text | sort
```

輸出為執行個體類型清單，依字母順序排序。以下僅是輸出的開始。

```
c4.2xlarge
c4.4xlarge
c4.8xlarge
c4.large
c4.xlarge
c5.12xlarge
c5.18xlarge
c5.2xlarge
c5.4xlarge
c5.9xlarge
...
```

從EC2執行個體類型尋找工具取得

EC2執行個體類型尋找工具會考量您的使用案例、工作負載類型、CPU製造商偏好設定，以及如何設定價格和效能的優先順序，以及您可以指定的其他參數。然後，它會使用這些資料，針對最適合您的新工作負載的 Amazon EC2 執行個體類型提供建議和指導。

如此眾多的執行個體類型可供使用，尋找適合您工作負載的執行個體類型可能非常耗時且複雜。透過使用EC2執行個體類型尋找工具，您可以隨時掌握最新的執行個體類型，並為您的工作負載實現最佳價格效能。

您可以使用 Amazon EC2 主控台取得EC2執行個體類型的建議和指導。您也可以直接前往 Amazon Q 尋求執行個體類型建議。如需詳細資訊，請參閱 Amazon Q 開發人員使用者指南。

如果您正在尋找適用於現有工作負載的執行個體類型建議，請使用 AWS Compute Optimizer. 如需詳細資訊，請參閱從 Compute Optimizer 取得EC2執行個體。

使用EC2實體類型尋找工具

在 Amazon EC2 主控台中，您可以從啟動執行個體精靈中的EC2執行個體類型尋找工具、建立啟動範本時，或在 [執行個體類型] 頁面上取得執行個體類型建議。

使用下列指示，透過 Amazon EC2 主控台中的EC2執行個體類型尋找工具取得EC2執行個體類型的建議和指導。若要檢視步驟的動畫，請參閱檢視動畫：使用執行個體類型尋找工具取得EC2實體類型建議。

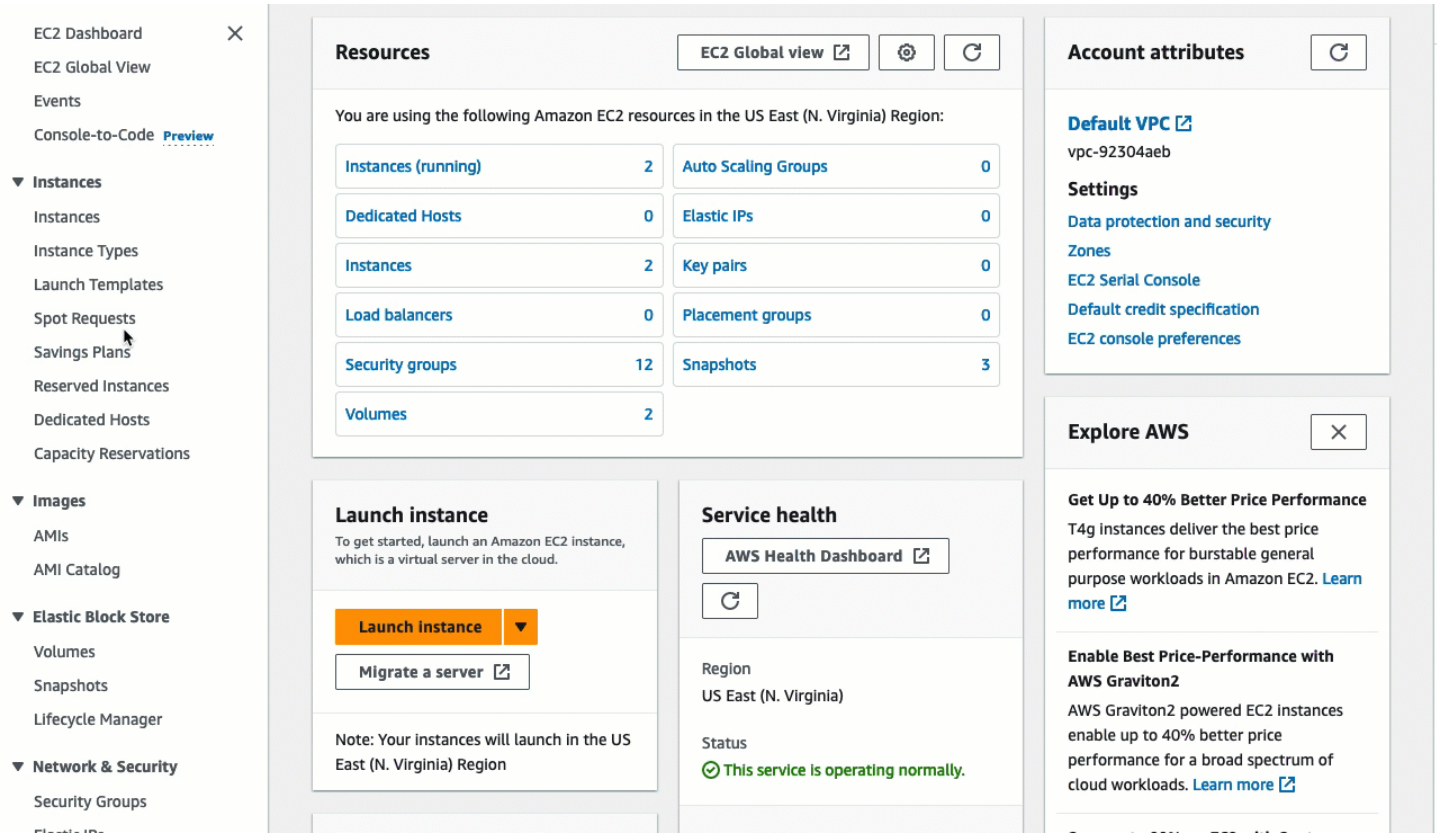
使用執行個體類型尋找器取得EC2執行個體類型建議

1. 使用以下任何一種方式開始您的流程：
   - 請遵循該程序來啟動執行個體。在執行個體類型旁，選擇取得建議連結。
   - 依照程序建立啟動範本。在執行個體類型旁，選擇取得建議連結。
   - 在瀏覽窗格中，選擇「實體類型」，然後選擇「實體類型尋找工具」按鈕。
2. 在「取得執行個體類型選取建議」畫面中，執行下列動作：
   a. 選取「工作負載類型」、「使用案例」、「優先順序」和「CPU製造商」等選項，以指定執行個體
   b. (選擇性) 若要為工作負載指定更詳細的需求，請執行下列動作：
      i. 展開 [進階參數]。
      ii. 若要加入參數，請選取參數，選擇「新增」，然後指定參數值。對您要加入的每個其他參數重複上述步驟。若要指示沒有最小值或最大值，請將欄位保留空白。
      iii. 若要在新增參數後移除參數，請選擇參數旁邊的 X。
   c. 選擇取得執行個體類型建議。

      Amazon 為您EC2提供符合指定需求的執行個體系列建議。
3. 若要檢視建議的執行個體系列中每個執行個體類型的詳細資料，請選擇 [檢視建議的執行個體系列
4. 選取符合您需求的執行個體類型，然後選擇「動作」、「啟動執行個體」或「動作」、「建立啟動範本」。

或者，如果您是在啟動執行個體精靈或啟動範本頁面中啟動程序，而且想要回到原始流程，請記下您要使用的執行個體類型。接著，在啟動執行個體精靈或啟動範本中，針對「執行個體類型」選擇執行個體類型，並完成啟動執行個體或建立啟動範本的程序。

檢視動畫：使用執行個體類型尋找工具取得EC2實體類型建議



從 Compute Optimizer 取得EC2執行個體

AWS Compute Optimizer 提供 Amazon EC2 建議以協助您改善效能、節省金錢或兩者兼而有之。您可以使用這些建議來決定是否要變更為新執行個體類型。

若要提出建議，Compute Optimizer 會分析您現有的執行個體規格和使用率指標。然後，編譯後的資料會用來建議哪些 Amazon EC2 執行個體類型最能處理現有的工作負載。建議會連同每小時執行個體定價一起傳回。如需詳細資訊，請參閱AWS Compute Optimizer 使用者指南中的 Amazon EC2 執行個體指標。

目錄

- 要求
- 尋找分類

- 檢視建議
- 評估建議的考量事項

要求

若要從 Compute Optimizer 取得建議，您必須先選擇加入 Compute Optimizer。如需詳細資訊，請參閱《 AWS Compute Optimizer使用者指南》中的 AWS Compute Optimizer 入門。

Compute Optimizer 會針對某些執行個體類型產生建議，但並非所有執行個 如果您使用不支援的執行個體類型，Compute Optimizer 將不會產生建議。如需支援執行個體類型的清單，請參閱AWS Compute Optimizer 使用者指南中的 Amazon EC2 執行個體要求。

尋找分類

Compute Optimizer 會將EC2執行個體的發現項目分類如下：

- 佈建不足 — 當EC2執行個體的至少一個規格 (例如記憶體或網路) 不符合工作CPU負載的效能需求時，就會將執行個體視為佈建不足。佈建不足的EC2執行個體可能導致應用程式效能不佳
- 過度佈建 — 如果您的EC2執行個體至少有一項規格 (例如記憶體或網路) 可以縮減大小CPU，同時仍符合工作負載的效能需求，且沒有規格佈建不足時，則會將執行個體視為過度佈建。過度佈建的EC2執行個體可能導致不必要的基礎結構
- 最佳化 — 如果EC2執行個體的所有規格 (例如記憶體和網路) 都符合工作負載的效能需求，且執行個體未過度佈建，則會將執行個體視為CPU最佳化。最佳化EC2執行個體以最佳效能和基礎架構成本執行您的工作負載 對於最佳化的執行個體，Compute Optimizer 有時可能會建議新一代執行個體類型。
- None (無)–沒有此執行個體的建議。如果您已在 12 小時內選擇加入 Compute Optimizer，或是執行個體執行時間不到 30 小時，或 Compute Optimizer 不支援執行個體類型，就可能會發生這種情況。

檢視建議

選擇使用 Compute Optimizer 後，您可以在 Amazon EC2 主控台中檢視 Compute Optimizer 為EC2執行個體產生的發現結果。然後，您可以存取 Compute Optimizer 主控台來檢視建議。如果您最近選擇加入，最多 12 小時內發現項目可能不會反映在EC2主控台中。

使用 Amazon EC2 主控台檢視執行個體的建議

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。

3. 選擇執行個體 ID ，以開啟執行個體詳細資訊頁面。
4. 在執行個體詳細資料頁面的頂端摘要區段中，AWS Compute Optimizer 找出發現項目。如果有發現項目，我們會顯示發現項目分類和檢視詳細資料的連結。否則，我們會顯示此執行個體沒有可用的建議。
5. 如果有發現項目，請選擇「檢視詳細資料」。這會開啟 [運算最佳化處理程式] 主控台中的EC2執行處理建議頁 目前的例證類型會標示為「目前」。還有最多三種執行個體類型建議，標示為 「選項 1」、「選項 2」 和 「選項 3」。此頁面也會顯示執行處理最近的 CloudWatch 測量結果資料。

檢視所有區域中所有執行處理的建議

您可以使用 Compute Optimizer 主控台檢視所有區域中所有 Amazon EC2 執行個體的建議。如需詳細資訊，請參閱AWS Compute Optimizer 使用指南中的檢視EC2EC2執行個體建議和檢視執行個體詳細資訊

評估建議的考量事項

當您收到建議時，您必須決定是否採取行動。變更執行個體類型之前，請考慮下列事項：

- 建議不會預測您的使用量。建議是根據您最近 14 天期間的歷史使用情況而定。請務必選擇符合未來資源需求的執行個體類型。
- 著重於圖形指標，以判斷實際使用量是否低於執行個體容量。您也可以在 CloudWatch 中檢視指標資料 (平均值、峰值、百分位數)，以進一步評估EC2執行個體建議。例如，請注意CPU百分比量度在一天中的變化情況，以及是否有需要容納的尖峰。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的檢視可用指標。
- Compute Optimizer 可能會提供高載效能執行個體的建議，亦即 T3、T3a 和 T2 執行個體。如果您定期在基準線之上成組分解，請確定您可以根據新執行個體類型繼續執行此操作。 vCPUs 如需詳細資訊，請參閱高載效能執行個體的關鍵概念。
- 如果您購買的是預留執行個體，隨需執行個體可能會以預留執行個體方式計費。變更目前的執行個體類型之前，請先評估對預留執行個體使用率和涵蓋範圍的影響。
- 盡可能考慮轉換為較新一代的執行個體。
- 移轉至不同的執行個體系列時，請確定目前的執行個體類型和新的執行個體類型是相容的，例如在虛擬化、架構或網路類型方面。如需詳細資訊，請參閱變更執行個體類型的相容性。
- 最後，請考慮針對每個建議所提供的效能風險評分。效能風險指出您可能需要花費的精力，以驗證建議的執行個體類型是否符合工作負載的效能需求。我們也建議在進行任何變更之前和之後進行嚴格的負載和效能測試。

Amazon EC2 實例類型更改

隨著您的需求變更，您可能會發現您的執行個體過度利用 (執行個體類型太小)，或利用不足 (執行個體類型太大)。若出現此情況，您可以透過變更執行個體類型來調整執行個體的大小。例如，如果您的 `t2.micro` 執行個體對於其工作負載來說太小，則您可以透過將其變更為較大的 T2 執行個體類型 (例如 `t2.large`) 來增加其大小。或者，您可以將其變更為另一個執行個體類型，例如 `m5.large`。您也可能想要從上一代變更為目前一代的執行個體類型，以利用某些功能，例如支援IPv6。

如果想要獲得有關最能處理現有工作負載的執行個體類型建議，可以使用 AWS Compute Optimizer。如需詳細資訊，請參閱從 Compute Optimizer 取得EC2執行個體。

若變更執行個體類型，您將開始支付新執行個體類型的費率。如需所有執行個體類型的隨需費率，請參閱 Amazon EC2 隨需定價。

若要在不變更執行個體類型的情況下新增額外的儲存空間，請將EBS磁碟區新增至執行個體。如需詳細資訊，請參閱 Amazon EBS使用者指南中的將 Amazon EBS 磁碟區連接到執行個體。

要遵循哪些指示？

變更執行個體類型有不同的指示。要使用的指示取決於執行個體的根磁碟區，以及執行個體類型是否與執行個體的目前組態相容。如需如何判斷相容性的相關資訊，請參閱 變更執行個體類型的相容性。

使用下表判斷要遵循哪些指示。

根磁碟區	相容性	使用這些指示
EBS	相容	變更執行個體類型
EBS	不相容	移轉至新的執行個體類型
執行個體存放區	不適用	移轉至新的執行個體類型

變更執行個體類型的相容性

只在執行個體的目前組態與您想要的執行個體類型相容時，您才能變更執行個體類型。如果您想要的執行個體類型與執行個體的目前狀態不相容，您必須搭配與執行個體類型相容的組態，啟動新的執行個體，然後將您的應用程式遷移到新的執行個體。

相容性是透過下列方式決定的：

虛擬化類型

Linux AMIs 使用兩種虛擬化類型之一：半虛擬 (PV) 或硬體虛擬機器 (HVM)。如果執行個體是從 PV 啟動的AMI，則無法變更為HVM只有執行個體類型。如需詳細資訊，請參閱虛擬化類型。若要檢查執行個體的虛擬化類型，請在 Amazon EC2 主控台的「執行個體」畫面的詳細資料窗格中查看虛擬化值。

架構

AMIs特定於處理器的架構，因此您必須選取與目前執行個體類型具有相同處理器架構的執行個體類型。例如：

- 如果目前執行個體類型具有 Arm 架構處理器，您會受限於支援 Arm 架構處理器的執行個體類型，例如 C6g 和 M6g。
- 下列執行個體類型是唯一支援 32 位元的執行個體類型 AMIs：`t2.nanot2.microt2.smallt2.mediumc3.large`、`t1.micro`、、`m1.small`、`m1.mediu` 和`c1.medium`。如果您要變更 32 位元執行個體的執行個體類型，則會受限於這些執行個體類型。

網路介面卡

如果您從某個網路介面卡的驅動程式切換到另一個網路介面卡，該網路介面卡設定會在作業系統建立新的介面卡時重設。若要重新進行設定，您可能需要存取具有系統管理員許可的本機帳戶。以下是從某個網路介面卡移至另一個網路介面卡的範例：

- AWS 光伏（T2 實例）到英特爾 82599 VF（M4 實例）
- 英特爾 82599 VF（大多數 M4 實例）到ENA（M5 實例）
- ENA(M5 執行個體) 至高頻寬 ENA (M5n 執行個體)

網路卡

某些執行個體類型支援多個網路卡。您必須選取一個執行個體類型，它支援與目前執行個體類型相同數量的網路卡。

增強型網路

支援增強型聯網的執行個體類型需要安裝必要的驅動程式。例如，建立在 AWS Nitro System 上的執行個體需要EBS安裝彈性網路介面卡 (ENA) 驅動程式。AMIs若要從不支援增強型聯網的執行個體類型變更為支援增強型聯網的執行個體類型，您必須視情況在執行個體上安裝ENA驅動程式或ixgbevf 驅動程式。

> **Note**
>
> 在啟用 ENA Express 的情況下調整執行個體大小時，新執行個體類型也必須支援 ENA Express。如需支援 ENA Express 的執行個體類型清單，請參閱ENA快速支援的執行個體類型。
> 若要從支援 Express 的執行個體類型變更為不支援 ENA Express 的執行個體類型，請在調整執行個體大小之前，請確定目前未啟用 ENA Express。

NVMe

EBS在 AWS Nitro 系統上建置的執行個體上，磁碟區會以區NVMe塊裝置的形式公開。如果您從不支援的執行個體類型變更NVMe為支援的執行個體類型NVMe，您必須先在執行個體上安裝NVMe驅動程式。此外，您在區塊裝置對應中指定之裝置的裝置名稱也會使用裝NVMe置名稱 (`/dev/nvme[0-26]n1`) 來重新命名。

[Linux 執行個體] 因此，若要在開機時使用掛載檔案系統`/etc/fstab`，您必須使用 UUID /Label 而非裝置名稱。

磁碟區限制

您可以連接到執行個體的最大 Amazon EBS 磁碟區數量取決於執行個體類型和執行個體大小。如需詳細資訊，請參閱Amazon EC2 實例的EBS數量限制。

只有執行個體類型或執行個體大小支援的磁碟區數量，等於或大於目前連接至執行個體的磁碟區數量，您才能變更為這樣的執行個體類型或執行個體大小。如果您變更為不支援目前連接磁碟區數量的執行個體類型或執行個體大小，則該請求會失敗。例如，如果您從具有 32 個連接磁碟區的 `m7i.4xlarge` 執行個體變更為最多支援 27 個磁碟區的 `m6i.4xlarge`，則該請求會失敗。

硝基 TPM

如果您使用已啟用 Nitro 的執行個AMI體和支援 Nitro 的執行個體類型來TPM啟動執行個體TPM，則執行個體會在啟用 Nitro 的情況下啟動。TPM您只能變更為也支援 Nitro TPM 的執行個體類型。

變更 Amazon 執行個體的執EC2行個體類型

如果您需要的執行個體類型與執行個體目前的組態相容，請使用下列指示變更 Amazon EBS 支援執行個體的執行個體類型。如需詳細資訊，請參閱the section called “相容性”。

考量事項

- 您必須先停止執行個體，才能變更執行個體類型。請確認您已為執行個體停止時的停機時間做好規劃。停止執行個體並變更其執行個體類型可能會花費數分鐘，重新啟動您的執行個體所需要花費的時間則會根據您應用程式的啟動指令碼而有所不同。如需詳細資訊，請參閱 停止並啟動 Amazon EC2 執行個體。
- 當您停止並啟動執行個體時，我們會將執行個體移至新硬體。如果您的執行個體有公開IPv4位址，我們會釋出該地址，並為您的執行個體提供新的公開IPv4地址。
- 您無法變更 Spot 執行個體的執行個體類型。
- [Windows 執行個體] 建議您在變更執行個體類型之前更新 AWS PV 驅動程式套件。如需詳細資訊，請參閱the section called “升級 PV 驅動程式”。
- 如果您的執行個體位於 Auto Scaling 群組中，Amazon EC2 Auto Scaling 服務會將停止的執行個體標記為運作狀態不良，並可能會終止執行個體並啟動替換執行個體。若要避免這種情況，您可以在變更執行個體類型時，暫停群組的擴展程序。如需詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南中的暫停和繼續 Auto Scaling 群組的程序。
- 當您使用執行個體存放磁碟區變更執行個體的NVMe執行個體類型時，更新後的執行個體可能會有額外的執行個體儲存磁碟區，因為即使未在AMI或NVMe執行個體區塊裝置對映中指定，所有執行個體儲存磁碟區也可以使用。否則，更新後的執行個體通常會有與您在啟動原始執行個體時指定之執行個體儲存體磁碟區相同數目的執行個體儲存體磁碟區。
- 您可以連接到執行個體的最大 Amazon EBS 磁碟區數量取決於執行個體類型和執行個體大小。如果執行個體類型或執行個體大小不支援已連接至執行個體的磁碟區數量，您就無法變更為這樣的執行個體類型或執行個體大小。如需詳細資訊，請參閱Amazon EC2 實例的EBS數量限制。
- [Linux 執行個體] 您可以使用`AWSSupport-MigrateXenToNitroLinux`執行本，將相容的 Linux 執行個體從 Xen 執行個體類型移轉為 Nitro 執行個體類型。如需詳細資訊，請參閱 AWS Systems Manager Automation Runbook 參考 中的 AWSSupport-MigrateXenToNitroLinux runbook。
- [Windows 執行個體] 如需將相容 Windows 執行個體從 Xen 執行個體類型移轉至 Nitro 執行個體類型的其他指引，請參閱移轉至最新一代執行個體類型。

變更 Amazon EBS 支援執行個體的執行個體類型

1. (選用) 若新執行個體類型需要的驅動程式並未安裝於現有的執行個體，您必須先連接您的執行個體並安裝驅動程式。如需詳細資訊，請參閱變更執行個體類型的相容性。
2. [Windows 執行個體] 如果您將 Windows 執行個體設定為使用靜態 IP 位址，且從不支援增強型聯網的執行個體類型變更為支援增強型聯網的執行個體類型，則當您重新設定靜態 IP 位址位址時，可能會收到有關潛在 IP 位址衝突的警告。若要避免這種情況發生，請DHCP在變更執行個體類型

之前，在執行個體的網路介面上啟用。從您的執行個體開啟 [網路和共用中心]，為網路介面開啟 [網際網路通訊協定第 4 版 (TCP/IPv4) 內容]，然後選擇 [自動取得 IP 位址]。在網路介面上變更執行個體類型，然後重新設定靜態 IP 地址。

3. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
4. 在導覽窗格中，選擇 Instances (執行個體)。
5. 選取執行個體，並選取 Instance state (執行個體狀態)、Stop instance (停止執行個體)。出現確認提示時，請選擇 Stop (停止)。停止執行個體可能需要幾分鐘。
6. 在仍然選取執行個體的情況下，選取 Actions (動作)、Instance settings (執行個體設定)、Change instance type (變更執行個體類型)。如果執行個體狀態不是 `stopped`，此選項會呈現灰色。
7. 在 Change instance type (變更執行個體類型) 頁面上，執行下列動作：

   a. 針對 Instance type (執行個體類型)，選取您想要的執行個體類型。

      如果執行個體類型不在清單中，表示其與您執行個體的組態不相容。請改用下列指示：透過啟動新執行個體移轉至新的執行個EC2體類型。

   b. (選擇性) 如果您選取的執行個體類型支援EBS最佳EBS化，請選取 [- optimization] 以啟用最佳EBS化，或取消選取 [EBS-optimization] 以停用EBS最佳化。如果您選取的執行個體類型預設為EBS最佳化，就會選取 EBS-optimized，而您無法取消選取該執行個體類型。

   c. 選擇 Apply (套用) 以接受新的設定。
8. 若要啟動執行個體，請選取執行個體，然後選取 Instance state (執行個體狀態)、Start instance (啟動執行個體)。執行個體進入 `running` 狀態可能需要數分鐘。若您的執行個體未啟動，請參閱對變更執行個體類型進行疑難排解。
9. [Windows 執行個體] 如果您的執行個體以 EC2Launch v1 執行 Windows 伺服器 2016 或 Windows 伺服器 2019，請連線到您的 Windows 執行個體，並在執行個體類型變更後執行下列EC2Launch PowerShell 指令碼以設定執行個體。

> **Important**
>
> 當您啟用初始化執行個體 EC2 Launch 指令碼時，管理員密碼將會重設。您可以修改組態檔來停用管理員密碼重設，方法為在初始化任務的設定中指定它。如需如何停用密碼重設的步驟，請參閱設定初始化工作 (EC2Launch) 或變更設定 (EC2Launchv2)。

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 -
Schedule
```

透過啟動新執行個體移轉至新的執行個EC2體類型

只有當執行個體是具有與您想要的新EC2執行個EBS體類型不相容的組態時，您才能變更執行個體的執行個體類型。否則，如果設定或執行個體與新執行個體類型不相容，或是執行個體存放區型執行個體，您必須啟動與您想要的執行個體類型相容的替代執行個體。若要取得有關如何確定相容性的更多資訊，請參閱變更執行個體類型的相容性。

遷移程序概觀

- 備份原始執行個體上的資料。
- 使用與您想要的新執行個體類型相容的組態啟動新執行個體，並附加任何附加至原始執行個體的EBS磁碟區。
- 在新執行個體上安裝應用程式。
- 還原任何資料。
- 如果原始執行個體具有彈性 IP 位址，您必須將其與新執行個體建立關聯，以確保使用者可以繼續使用您的應用程式而不會中斷。

將執行個體移轉至新執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 備份您仍然需要的任何數據，如下所示：
   - Connect 至您的執行個體，並將執行個體儲存磁碟區上的資料複製到永久儲存。
   - 建立EBS磁碟區的快照，以便使用相同資料建立新磁碟區，或將磁碟區與原始執行個體分離，以便將磁碟區附加至新執行個體。
3. 在導覽窗格中，選擇 Instances (執行個體)。
4. 選擇啟動執行個體。設定執行個體時，請執行下列動作：
   a. 選取支援AMI所需執行個體類型的執行個體類型。例如，您必須選取AMI支援新執行個體類型的處理器類型。此外，目前一代的執行個體類型需要 HVMAMI.
   b. 選取您想要的新執行個體類型。如果您想要的執行個體類型無法使用，表示該執行個體類型與您所選取的執行個體設定不相容。AMI
   c. 如果您想要允許相同的流量傳送到新執行個體，請選取VPC與原始執行個體搭配使用的相同安全群組。
   d. 完成設定新的執行個體後，請完成步驟來選取金鑰對並啟動您的執行個體。執行個體進入 `running` 狀態可能需要數分鐘。

5. 如果您將資料備份到EBS快照，請從快照建立磁碟區，然後將磁碟區連接至新執行個體。

   若要將EBS磁碟區從原始執行個體移至新執行個體，請將磁碟區與原始執行個體分離，然後將該磁碟區附加至新執行個體。
6. 在新的執行個體上安裝您的應用程式及任何必要的軟體。
7. 還原您從原始執行個體的執行個體儲存體磁碟區備份的任何資料。
8. 如果原始執行個體具有彈性 IP 位址，請將其指派給新執行個體，如下所示：

   a. 在導覽窗格中，選擇 [彈性] IPs。
   b. 選取與原始執行個體相關聯的彈性 IP 地址，選取 Actions (動作)，然後選取 Disassociate Elastic IP address (與彈性 IP 地址取消關聯)。出現確認提示時，請選擇 Disassociate (取消關聯)。
   c. 在彈性 IP 地址仍在選取狀態的情況下，選取 Actions (動作)，然後選取 Associate Elastic IP address (與彈性 IP 地址建立關聯)。
   d. 對於 Resource type (資源類型)，選擇 Instance (執行個體)。
   e. 對於「例如」，選擇新的執行個體。
   f. (選用) 對於 Private IP address (私人 IP 地址)，指定要與彈性 IP 地址建立關聯的私人 IP 地址。
   g. 選擇 Associate (關聯)。
9. (選用) 若您不再需要原始執行個體，則可將其終止。選取執行個體，確認您要終止的是原始執行個體，而非新的執行個體 (例如，檢查名稱或啟動時間)，然後選取 Instance state (執行個體狀態)、Terminate instance (終止執行個體)。

對變更執行個體類型進行疑難排解

請使用以下資訊來協助您診斷並修正在變更執行個體類型時可能遇到的問題。

在變更執行個體類型之後執行個體不會啟動

可能原因：未符合新執行個體類型的需求

如果您的執行個體未啟動，則原因可能是未符合新執行個體類型的其中一個需求。如需詳細資訊，請參閱為何我的 Linux 執行個體在我變更其類型之後未啟動？

可能的原因：AMI不支援執行個體類型

如果您使用EC2主控台變更執行個體類型，則只有所選執行個體支援的執行個體類型可AMI供使用。不過，如果您使用啟動執行個體，您可以指定不相容的執AMI行個體類型。 AWS CLI 如果AMI

和執行個體類型不相容，執行個體就無法啟動。如需詳細資訊，請參閱變更執行個體類型的相容性。

可能原因：執行個體位於集群放置群組中

如果您的執行個體位於集群放置群組中，並在變更執行個體類型之後，執行個體無法啟動，請嘗試下列動作：

1. 停止集群放置群組中的所有執行個體。
2. 變更受影響執行個體的執行個體類型。
3. 啟動集群放置群組中的所有執行個體。

在變更執行個體類型之後，無法從網際網路連線到應用程式或網站

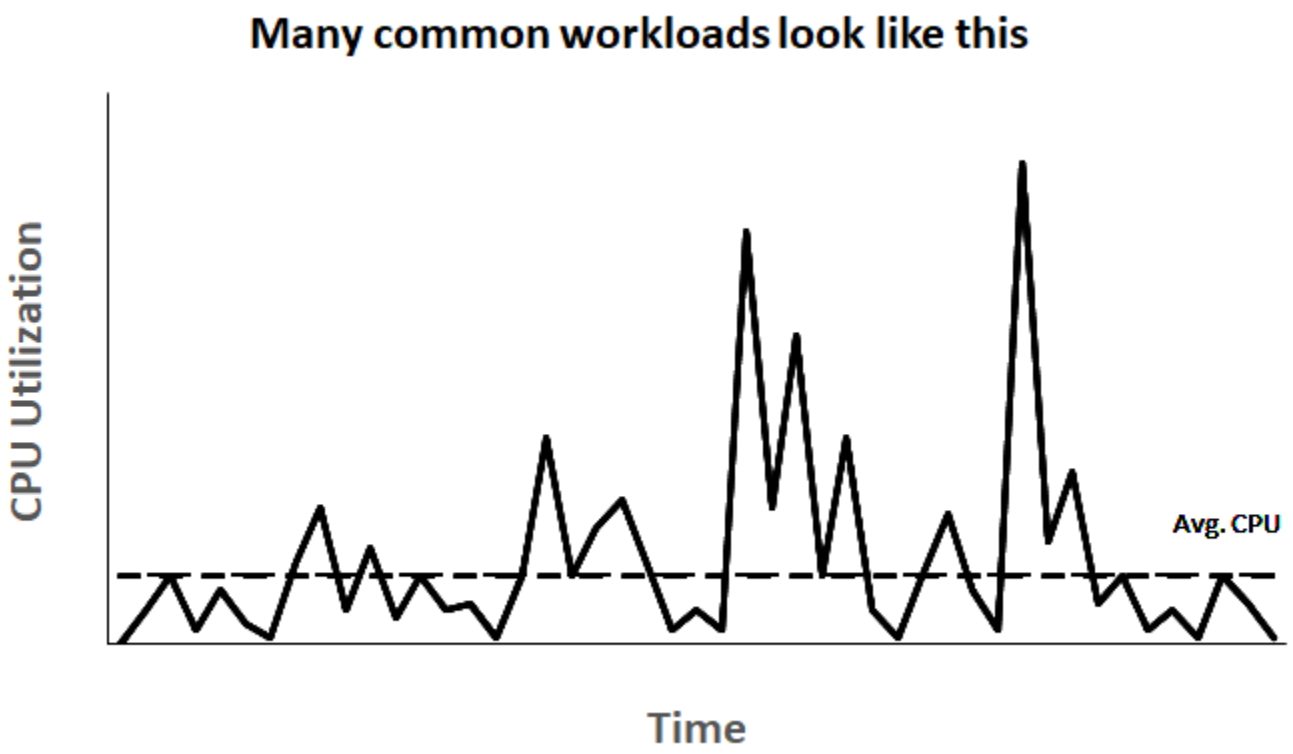
可能的原因：公共IPv4地址被釋放

變更執行個體類型時，您必須先停止執行個體。當您停止執行個體時，我們會釋出公開IPv4地址，並為您的執行個體提供新的公開IPv4地址。

若要在執行個體停止和啟動之間保留公用位IPv4址，建議您使用 Elastic IP 位址，但如果您的執行個體正在執行，則無須支付額外費用。如需詳細資訊，請參閱彈性 IP 地址。

爆量效能執行個體

許多一般用途的工作負載平均而言並不繁忙，而且不需要高層級的持續CPU效能。下圖說明當今客戶在 AWS 雲端執行的許多常見工作負載的CPU使用率。



這些使用 low-to-moderate CPU率工作負載會導致CPU週期浪費，因此，您需要支付超過使用量的費用。為了克服此問題，您可以利用低成本爆量一般用途執行個體，也就是 T 執行個體。

T 執行個體系列提供基準CPU效能，並能在需要時隨時超出基準線成組分解。此基準的定義CPU是為了滿足大多數一般用途工作負載的需求，包括大型微服務、Web 伺服器、中小型資料庫、資料記錄、程式碼存放庫、虛擬桌面、開發和測試環境，以及業務關鍵應用程式。T 執行個體可在運算、記憶體和網路資源之間取得平衡，並提供最具成本效益的方式，讓您以最具成本效益的方式來執行各種具有 low-to-moderate CPU用途的一般用途應用程式。與 M 執行個體相比，它們可以為您節省高達 15% 的成本，而且透過更小、更經濟實惠的執行個體大小，提供低至 2 vCPUs 和 0.5 GiB 的記憶體，可節省更多成本。較小的 T 執行個體大小 (例如 nano、微型、小型和中型) 非常適合需要少量記憶體且不需要高CPU使用率的工作負載。

> **Note**
>
> 本主題說明可爆發。CPU如需有關爆量網路效能的詳細資訊，請參閱 Amazon EC2 實例網絡帶寬。

EC2高載執行個體類型

高載執行EC2個體包含 T4G、T3A 和 T3 執行個體類型，以及上一代 T2 執行個體類型。

T4g 執行個體類型是最新一代的爆量執行個體。它們提供最優惠的性能價格，並為您提供所有EC2實例類型中最低的成本。T4G 執行個體類型由以 ARM 為基礎的 AWS Graviton2 處理器提供支援，並提供來自作業系統廠商、獨立軟體廠商以及熱門服務和應用程式的廣泛生態系統支援。 AWS

下表摘要列出爆量執行個體類型之間的主要差異。

類型	描述	處理器系列
最新一代		
T4g	與 T3 相比，成本最低的EC2執行個體類型，價格/效能提高 40%，成本降低 20%	AWS 重力 2 處理器配備臂 NEOVERSE N1 核心
T3a	成本最低的 x86 型執行個體，成本比 T3 執行個體低 10%	AMD第 1 代EPYC處理器
T3	適用於 x86 工作負載的最佳峰值價格/效能，比上一代 T2 執行個體的價格/效能低 30%	Intel Xeon 可擴充處理器（Skylake、Cascade Lake 處理器）
上一代		
T2	上一代爆量執行個體	Intel Xeon 處理器

如需執行個體定價和其他規格的相關資訊，請參閱 Amazon EC2 定價和 Amazon EC2 執行個體類型。如需有關爆量網路效能的詳細資訊，請參閱 Amazon EC2 實例網絡帶寬。

如果您的帳戶剛建立不到 12 個月，可以在一定的使用限額內免費使用 `t2.micro` 執行個體 (或 `t3.micro` 無法使用的區域中的 `t2.micro` 執行個體)。如需詳細資訊，請參閱AWS 免費方案。

T 執行個體支援的購買選項

- On-Demand Instances
- Reserved Instances
- 專用執行個體 (僅限 T3)

- 專用執行個體 (僅限 T3，僅限 `standard` 模式)
- Spot 執行個體

如需詳細資訊，請參閱Amazon EC2 帳單和購買選項。

內容

- 最佳實務
- 高載效能執行個體的關鍵概念
- 爆量效能執行個體的無限制模式
- 爆量效能執行個體的標準模式
- 使用爆量效能執行個體
- 監控高CPU載執行個體的積分

最佳實務

請遵循這些最佳實務，讓高載效能執行個體發揮最大的效益。

- 請確定您所選擇的執行個體容量，高於作業系統和應用程式的最低記憶體容量需求。具有圖形化使用者介面且耗用大量記憶體和CPU資源 (例如 Windows) 的作業系統，對於許多使用案例而言，可能需要較大的執行個`t3.micro`體大小。隨著工作負載的記憶體和CPU需求隨著時間的推移而增加，您可以靈活地將 T 執行個體擴展到相同執行個體類型的較大執行個體大小，或選擇其他執行個體類型。
- 為您的帳戶啟用 AWS Compute Optimizer，並檢閱工作負載的 Compute Optimizer 建議。Compute Optimizer 可協助評估執行個體是否應該擴大規模以提高效能，還是應縮小規模以節省成本。Compute Optimizer 也可能會根據您的情況建議不同的執行個體類型。如需詳細資訊，請參閱AWS Compute Optimizer 使用指南中的檢視EC2執行個體建議。

高載效能執行個體的關鍵概念

傳統的 Amazon EC2 執行個體類型提供固定CPU資源，而高載效能執行個體則提供基準CPU使用率，並且能夠將CPU使用率提升到基準層級以上。這可確保您只需為基準CPU加上任何額外的突發CPU用量付費，從而降低運算成本。基準使用率和成組分解的能力由CPU學分控制。高載效能執行個體是唯一使用積分進CPU行使用的執行個體類型。

每個高載效能執行個體在低於基準時都會持續獲得積分，並在超出CPU基準時持續花費積分。賺取或花費的積分取決於執行個體的CPU使用率：

- 如果使CPU用率低於基準，則獲得的積分會大於花費的積分。
- 如果使CPU用率等於基準，則獲得的積分等於花費的積分。
- 如果使CPU用率高於基準，則花費的積分會高於獲得的積分。

當獲得的積分大於花費的信用額度時，差異稱為應計積分，稍後可用於突破基準使用CPU率。同樣地，如果消耗的額度大於獲得的額度，則執行個體行為取決於額度組態模式 – 標準模式或無限制模式。

在標準模式中，當花費的點數超過賺取的積分時，執行個體會使用累積的積分來超出基準使用CPU率。如果沒有剩餘的累積積分，則執行個體逐漸歸結為基準使CPU用率，並且在累積更多積分之前無法超出基準線。

在無限制模式中，如果執行個體突發超過基準使用CPU率，則執行個體會先使用累積的積分來爆發。如果沒有剩餘的累積額度，則執行個體會消耗剩餘額度進行超出。當它的使用CPU率低於基準時，它會使用它賺CPU取的信用來支付早些時候花費的剩餘信用。Amazon 能夠賺取CPU積分以支付剩餘積分，讓 Amazon 能EC2夠在 24 小時內平均執行個體的CPU使用率。如果 24 小時期間的平均CPU使用量超過基準，則會按固定額外費率按每 v CPU 小時固定附加費率固計費執行個體的。

目錄

- 重要概念和定義
- 賺取CPU積分
- CPU信貸賺取率
- CPU信貸累積限額
- 累CPU積學分壽命
- 基準使用率

重要概念和定義

下列重要概念和定義適用於爆量效能執行個體。

CPU利用率

CPU使用率是指目前在執行個體上使用的已配置EC2運算單元百分比。此指標會測量執行個體使用的已配置CPU週期百分比。CPU使用CPU率 CloudWatch 測量結果會顯示每個執行個體的CPU使用量，而非每個核心 執行個體的基準線CPU規格也是以每個執行個體的CPU使用量為基礎。若要使用 AWS Management Console 或來測量使用CPU率 AWS CLI，請參閱取得特定執行個體的統計資料。

CPU信貸

v CPU 時間的單位。

範例：

1 個點CPU數 = 1 v CPU * 100% 使用率 * 1 分鐘。

1 個點CPU數 = 1 v CPU * 50% 使用率 * 2 分鐘

1 個點CPU數 = 2 v CPU * 25% 的使用率 * 2 分鐘

基準使用率

當認列的貸方數目與使用的貸方數目相符時，審核使用率是CPU可用於淨貸方餘額為零的CPU層次。CPU基準使用率也稱為基準。基準使用率以 v 使用CPU率的百分比表示，計算方式如下：基準使用率百分比 = (獲得的學分數/數目vCPUs) /60 分鐘。

如需每個爆量效能執行個體類型的基準使用率，請參閱點數表。

獲得的點數

執行個體在執行時持續獲得的額度。

每小時獲得的積分數 = 基準使用率百分比 vCPUs * 60 分鐘數

範例：

具有 2 的 t3.nano vCPUs 和 5% 的基線使用率可獲得每小時 6 個學分，計算方式如下：

2 vCPUs *5% 基準線 * 60 分鐘 = 每小時 6 個學分

消耗或使用的額度

執行個體在執行時持續使用的額度。

CPU每分鐘花費的點數 vCPUs = CPU 使用量 * 1 分鐘

累積額度

如果執行個體使用的CPU積分少於基準使用量所需的點數，則為未使用的積分。換句話說，累積額度 = (獲得的額度 - 使用的額度) 低於基準。

範例：

如果 t3.nano 以 2% 的CPU使用率執行 (低於其一小時 5% 的基準線)，則累積的學分計算方式如下：

累CPU積積分 = (每小時獲得的積分 — 每小時使用的積分) = 6 — 2 vCPUs * 2% 使用CPU率 * 60 分鐘 = 6 — 2.4 = 每小時 3.6 個累積積分

額度累積限額

它取決於執行個體大小，但通常等於 24 小時內獲得的最大額度。

範例：

對於 t3.nano，額度累積限額 = 24 * 6 = 144 個額度

啟動額度

僅適用於為標準模式設定的 T2 執行個體。啟動積分是指配置給新 T2 執行個體的有限數量CPU積分，以便在標準模式下啟動時，它可以超出基準。

剩餘額度

執行個體耗盡其累積額度餘額之後所消耗的額度。剩餘額度是專為爆量執行個體所設計，以便長時間維持高效能，並且僅用於無限制模式。剩餘額度餘額用於決定執行個體在無限制模式下爆量使用了多少額度。

標準模式

額度組態模式，透過消耗掉額度餘額中所累積的額度，此模式可讓執行個體超出基準。

無限制模式

信用組態模式，可讓執行處理在需要的時間內維持高CPU使用率，從而超出基準線。如果執行個體的平均CPU使用CPU率在累計 24 小時期間或執行個體生命週期 (以較短者為準) 達到或低於基準，則每小時執行個體價格會自動涵蓋所有使用尖峰值。

下表摘要列出爆量執行個體類型之間的主要額度差異。

Type	支援的CPU學分類型	額度組態模式	執行個體啟動和停止之間的累CPU積積分壽命
最新一代			
T4g	獲得的額度、累積的額度、消耗的額度、剩餘額度 (僅限無限制模式)	標準、無限制 (預設)	7 天 (額度會在執行個體停止後保留 7 天)

Type	支援的CPU學分類型	額度組態模式	執行個體啟動和停止之間的累CPU積積分壽命
T3a	獲得的額度、累積的額度、消耗的額度、剩餘額度 (僅限無限制模式)	標準、無限制 (預設)	7 天 (額度會在執行個體停止後保留 7 天)
T3	獲得的額度、累積的額度、消耗的額度、剩餘額度 (僅限無限制模式)	標準、無限制 (預設)	7 天 (額度會在執行個體停止後保留 7 天)
上一代			
T2	獲得的額度、累積的額度、消耗的額度、啟動額度 (僅限標準模式)、剩餘額度 (僅限無限制模式)	標準 (預設)、無限制	0 天 (當執行個體停止時，額度會消失)

> **Note**
>
> 專用執行個體上啟動的 T3 執行個體不支援無限制模式。

賺取CPU積分

視執行個體大小而定，每個高載效能執行個體每小時都會持續獲得 (以毫秒級解析度) 設定的CPU點數。無論是積分還是花費的會計過程也會以毫秒級的分辨率進行，因此您不必擔心超支CPU信用額度；短暫的一小部分CPU使用信用額度的一小部分。CPU

如果高載績效執行環境使用的CPU資源少於基準使用率所需的資源 (例如閒置時)，則未使用的貸方會在CPU貸方餘CPU額中累積。如果爆量效能執行個體的效能需要突增到超過基準使用率，將會用掉累積的額度。高載效能執行個體所累積的信用越多，當需要更多的CPU使用率時，就能超越其基準線的時間就越長。

下表列出高載效能執行個體類型、每小時獲得積CPU分的比率、執行個體可累積的最大獲得CPU積分數、 vCPUs 每個執行個體的數量，以及以完整核心 (使用單一 vCPU) 百分比表示的基準使用率。

執行個體類型	CPU每小時獲得的積分	可累積的額度上限*	vCPUs***	每 v 的基準使用率 CPU
T2				
t2.nano	3	72	1	5%
t2.micro	6	144	1	10%
t2.small	12	288	1	20%
t2.medium	24	576	2	20%**
t2.large	36	864	2	30%**
t2.xlarge	54	1296	4	22.5%**
t2.2xlarge	81.6	1958.4	8	17%**
T3				
t3.nano	6	144	2	5%**
t3.micro	12	288	2	10%**
t3.small	24	576	2	20%**
t3.medium	24	576	2	20%**
t3.large	36	864	2	30%**
t3.xlarge	96	2304	4	40%**
t3.2xlarge	192	4608	8	40%**
T3a				
t3a.nano	6	144	2	5%**

執行個體類型	CPU每小時獲得的積分	可累積的額度上限*	vCPUs***	每 v 的基準使用率 CPU
t3a.micro	12	288	2	10%**
t3a.small	24	576	2	20%**
t3a.medium	24	576	2	20%**
t3a.large	36	864	2	30%**
t3a.xlarge	96	2304	4	40%**
t3a.2xlarge	192	4608	8	40%**
T4g				
t4g.nano	6	144	2	5%**
t4g.micro	12	288	2	10%**
t4g.small	24	576	2	20%**
t4g.medium	24	576	2	20%**
t4g.large	36	864	2	30%**
t4g.xlarge	96	2304	4	40%**
t4g.2xlarge	192	4608	8	40%**

* 可累計的額度等於在 24 小時期間可獲得的額度。

** 表格中的百分比基準使用率為每 v CPU。在中 CloudWatch，CPU使用率顯示為每 v CPU。例如，在基準層級運作的`t3.large`執行個體CPU使用率顯示為 30% 的 CloudWatch CPU指標。如需如何計算基準使用率的資訊，請參閱基準使用率。

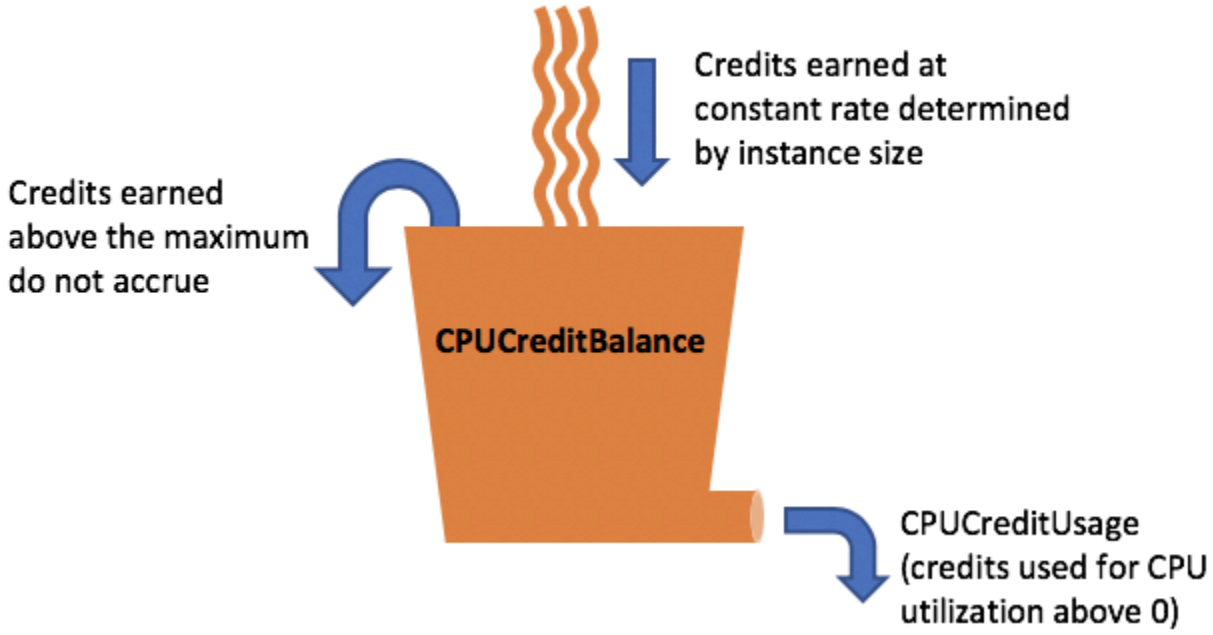
*** 每個 v 都CPU是 Intel Xeon 核心或核心的執行AMDEPYC緒，T2 和 T4G 執行個體除外。

CPU信貸賺取率

每小時獲得的CPU積分數量取決於執行個體大小。例如，`t3.nano` 每小時可獲得 6 點，而 `t3.small` 每小時可獲得 24 點。上表列出所有執行個體的額度獲得率。

CPU信貸累積限額

執行中的執行個體所獲得的額度，雖然永遠不會過期，但是執行個體可累積獲得的額度有其上限。限額由CPU信貸結餘限額決定。在達到上限之後，任何新獲得的額度都會遭到捨棄，如下圖所示。完整時段表示CPU信用餘額限制，溢出指示超出限制的新賺取的信用額度。



每個執行個體大小的CPU信用餘額上限不同。例如，`t3.micro`執行個體在積CPU分餘額中最多可累積 288 個已獲得的CPU積分。上面的表格列出了每個 執行個體的可累積獲得的額度上限。

T2 標準執行個體也會獲得啟動額度。啟動點數不會計入CPU信用餘額限制。如果 T2 執行個體尚未使用其啟動積分，且在累積已獲得積分的同時在 24 小時內仍處於閒置狀態，則其積分餘CPU額會顯示為超過限制。如需詳細資訊，請參閱啟動額度。

T4g、T3a 和 T3 執行個體不會獲得啟動額度。這些執行個體會根據預設啟動為 `unlimited`，因此啟動時可立即大幅提升效能，而不需任何啟動額度。依預設，在專用執行個體啟動上啟動的 T3 執行個體為 `standard`；專用執行個體上的 T3 執行個體不支援 `unlimited` 模式。

累CPU積學分壽命

CPU執行中執行個體的積分不會過期。

對於 T2，在執行個體停止和啟動之間不會持續存在CPU積分餘額。如果您停止 T2 執行個體，則執行個體會失去其所有累績的額度。

對於 T4G、T3A 和 T3，在執行個體停止後，CPU積分餘額會維持七天，之後積分會遺失。如果您在七天內啟動執行個體，則不會失去任何額度。

如需詳細資訊，請參閱CloudWatch 測量結果表格`CPUCreditBalance`中的。

基準使用率

當認列的貸方數目與使用的貸方數目相符時，審核使用率是CPU可用於淨貸方餘額為零的CPU層次。CPU基準使用率也稱為基準。

基準使用率以 v 使CPU用率的百分比表示，計算方式如下：

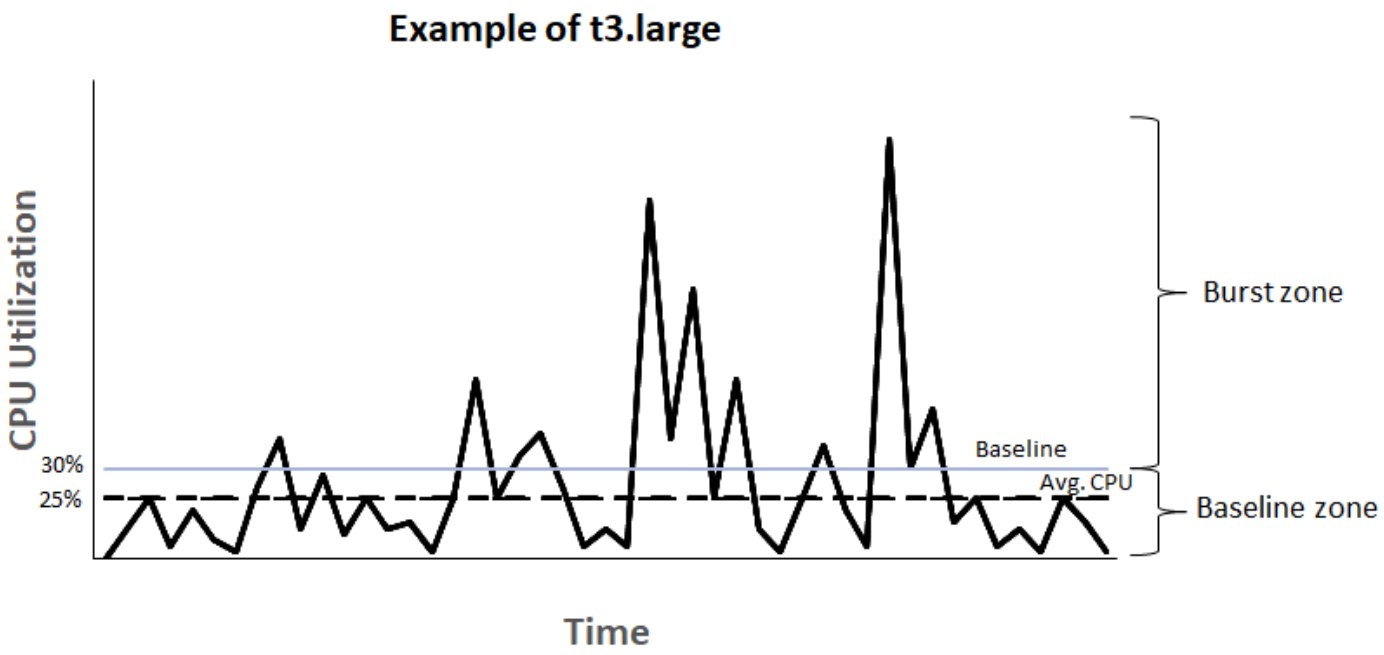
```
(number of credits earned/number of vCPUs)/60 minutes = % baseline
utilization
```

例如，具有 2 vCPUs 的`t3.nano`執行個體每小時可獲得 6 個學分，因此基準使用率為 5%，計算方式如下：

```
(6 credits earned/2 vCPUs)/60 minutes = 5% baseline utilization
```

使用 2 vCPUs 的`t3.large`執行個體每小時可獲得 36 個點數，因此基準使用率為 30% (`(36/2)/60`)。

下圖提供平均CPU使用率低於基準線的範例。`t3.large`



爆量效能執行個體的無限制模式

高載效能執行個體設定為unlimited可在需要的時間內維持高CPU使用率。如果執行個體的平均CPU使用CPU率在累計 24 小時期間或執行個體生命週期 (以較短者為準) 達到或低於基準，則每小時執行個體價格會自動涵蓋所有使用尖峰值。

針對絕大多數的一般用途工作負載，設定為 unlimited 的執行個體可提供足夠效能，而不需支付任何額外費用。如果執行個體長時間以較高的CPU使用率執行，則可以每 v CPU 小時以固定的額外費率執行。

如果您使用選件下的t2.micro或t3.micro執行個體並以unlimited模式使用，如果您在累計 24 小時期間內的平均使用率超過執行個體的基準使用率，則可能需要支付費用。AWS 免費方案

T4G、T3a 和 T3 執行個體unlimited依預設啟動 (除非您變更預設值)。如果 24 小時期間內的平均CPU使用量超過基準，您就會產生剩餘額度的費用。如果您將競價型執行個體啟動為，unlimited並計劃立即使用這些執行個體並且在沒有閒置時間累積積CPU分的情況下使用，則會產生剩餘額度的費用。建議您在標準模式下啟動 Spot 執行個體，以免需要支付更高的費用。如需詳細資訊，請參閱剩餘額度可能會產生費用及啟動高載效能執行個體。

> **Note**
>
> 依預設，在專用執行個體啟動上啟動的 T3 執行個體為 `standard`；專用執行個體上的 T3 執行個體不支援 `unlimited` 模式。

目錄

- 高載執行個體的無限模式概念
  - 無限制爆量效能執行個體如何運作
  - 何時使用無限模式與固定模式 CPU
  - 剩餘額度可能會產生費用
  - T2 無限制執行個體沒有啟動額度
  - 啟用無限制模式
  - 在無限制與標準之間切換時，額度會有什麼變化
  - 監控額度用量
- 高載執行個體的無限模式範例
  - 範例 1：說明 T3 Unlimited 的額度使用
  - 範例 2：說明 T2 Unlimited 的額度使用

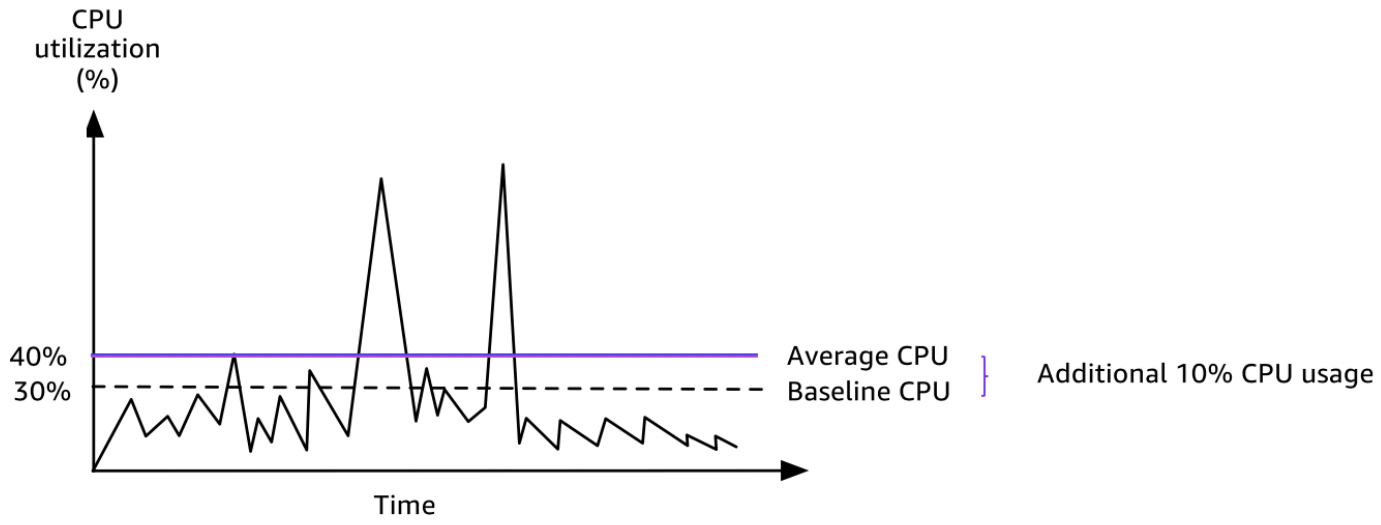
高載執行個體的無限模式概念

`unlimited` 模式是爆量效能執行個體的額度組態選項。對於執行中或已停止的執行個體，隨時都可以啟用或停用它。您可以在每個 AWS 區域、每個高載績效執行環境系列的帳戶層次設`unlimited`定為預設信用選項，以便使用預設信用選項來啟動帳戶中的所有新高載績效執行環境。

無限制爆量效能執行個體如何運作

如果設定為`unlimited`耗盡其信用餘額的高載效能執行個體，它可以花費剩餘的CPU信用額度超出基準線。當它的使用CPU率低於基準時，它會使用它賺CPU取的信用來支付早些時候花費的剩餘信用。Amazon 能夠賺取CPU積分以支付剩餘積分，讓 Amazon 能EC2夠在 24 小時內平均執行個體的CPU使用率。如果 24 小時期間的平均CPU使用量超過基準，則會按固定額外費率按每 v CPU 小時固定附加費率固計費執行個體的。

下圖顯示了一個的CPU用法`t3.large`。a 的基準CPU使用率`t3.large`為 30%。如果執行個體在 24 小時內以 30% 或更低的CPU使用率平均執行，則不會收取額外費用，因為執行個體每小時價格已涵蓋費用。



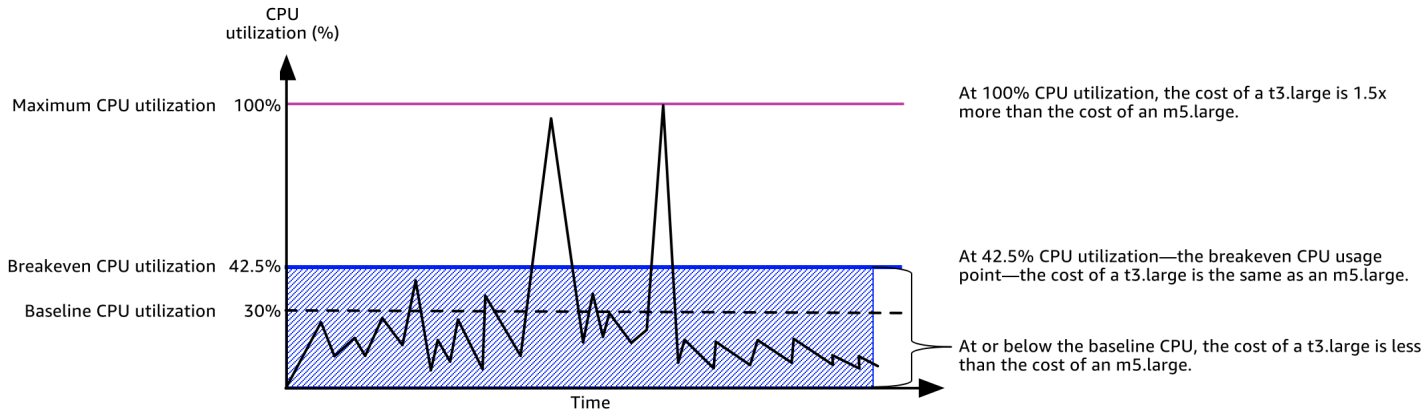
如需每個執行個體類型每 v CPU 的基準使用率，以及每個執行個體類型可獲得多少積分的詳細資訊，請參閱評分表。

何時使用無限模式與固定模式 CPU

決定是否應在unlimited模式 (例如 T3) 中使用高載效能執行個體，或是固定的效能執行個體 (例如 M5) 時，您必須判斷平衡使用量。CPU高載效能執行個體的收支平衡CPU用量是高載效能執行個體的成本與固定效能執行個體相同的時間點。盈虧平衡CPU用法可幫助您確定以下內容：

- 如果 24 小時期間的平均CPU使用量等於或低於盈虧平衡使用CPU量，請在unlimited模式中使用高載效能執行個體，以便您可以從高載效能執行個體的較低價格中獲益，同時獲得與固定效能執行個體相同的效能。
- 如果 24 小時期間的平均CPU使用量高於盈虧平衡CPU使用量，高載效能執行個體的成本將會超過相等大小的固定效能執行個體。如果 T3 執行個體持續以 100% 的速度爆發CPU，您最終支付的價格約為同等大小 M5 執行個體的 1.5 倍。

下圖顯示盈虧平衡CPU使用點，其中 a 的t3.large成本與. m5.large A 的盈虧平衡CPU使用點t3.large為 42.5%。如果平均CPU使用量為 42.5%，則執行成本與 a 相同m5.large，而且如果平均CPU使用量高於 42.5%，則會更加昂貴。t3.large如果工作負載所需的平均CPU使用量低於 42.5%，您可以享受更低的價格，同t3.large時獲得與. m5.large



下表顯示如何計算盈虧平衡使CPU用量臨界值，以便判斷在unlimited模式或固定效能執行個體中使用高載效能執行個體的成本較低的時機。資料表中的資料欄標示為 A 到 K。

執行個體類型	vCPUs	T3 價格*/小時	M5 價格*/小時	價格差異	T3 基準使用率每 v CPU (%)	剩餘積分每 v CPU 小時收費	每 v CPU 分鐘收費	每 v 可提供額外的突發分鐘數 CPU	額外可用 CPU 百分比	盈虧平衡% CPU
A	B	C	D	E = D - C	F	G	H = G / 60	I = E / H	J = (I / 60) / B	K = F + J
t3.large	2	0.0835 USD	0.096 USD	0.0125 USD	30%	0.05 USD	0.000833 USD	15	12.5%	42.5%

* 價格是以 us-east-1 和 Linux OS 為基礎。

資料表提供下列資訊：

- 資料欄 A 顯示執行個體類型，即 `t3.large`。
- 欄 B 顯示的 vCPUs 數目`t3.large`。

- 資料欄 C 顯示 `t3.large` 每小時的價格。
- 資料欄 D 顯示 `m5.large` 每小時的價格。
- 資料欄 E 顯示`t3.large`與`m5.large`的價格差異。
- 欄 F 顯示每 v CPU 的基準使用率`t3.large`，即 30%。根據基準，執行個體的小時費用涵蓋了使用費CPU用。
- G 欄顯示如果執行個體在耗盡獲得積分CPU後以 100% 爆發，則每 v CPU 小時的固定額外費率固定額外費率均一額外費率。
- https://aws.amazon.com/ec2/pricing/on-demand/#T2.2FT3.2FT4g_Unlimited_Mode_Pricing
- 第一欄顯示了每小時`t3.large`可以以 100% 的速度突發的額外分鐘數，同CPU時支付每小時相同的價格`m5.large`。
- 欄 J 顯示執行個體可以突增的額外CPU用量 (以% 為單位)，同時支付的每小時價格與`m5.large`.
- K 欄顯示了`t3.large`可以爆裂而不支付超過的盈虧平衡CPU用法（以 % 為單位）。`m5.large`除此之外，`t3.large` 的成本超過 `m5.large`。

下表顯示 T3 執行個體類型與類似大小的 M5 執行個體類型相比之下的盈虧平衡CPU用法 (以% 為單位)。

T3 執行個體類型	與 M5 相比，T3 的盈虧平衡CPU用量（以 % 為單位）
`t3.large`	42.5%
`t3.xlarge`	52.5%
`t3.2xlarge`	52.5%

剩餘額度可能會產生費用

如果執行個體的平均CPU使用率等於或低於基準，則執行個體不會產生額外費用。由於執行個體在 24 小時的期間獲得了最高額度 (例如，`t3.micro` 執行個體可以在 24 小時的期間獲得最多 288 點)，因此可以用掉等於最高額度的剩餘額度，而不會產生費用。

但是，如果CPU使用率維持在基準線之上，則執行個體無法獲得足夠的積分來支付已花費的剩餘積分。未支付的剩餘積分按每 v CPU 小時的固定額外費率收費。如需有關費率的資訊，請參閱 T2/T3/T4g 無限制模式定價。

發生下列任一情況時，將會針對先前用掉的剩餘額度來收取費用。

- 支出剩餘額度超過執行個體在 24 小時期間可獲得的最大額度數量。在小時結束時，將收取超過最大值的支出剩餘額度的費用。
- 執行個體已停止或終止。
- 執行個體從 `unlimited` 切換至 `standard`。

CloudWatch 度量會追蹤已花費的剩餘積分`CPUSurplusCreditBalance`。計費的剩餘積分會由 CloudWatch 量度追蹤`CPUSurplusCreditsCharged`。如需詳細資訊，請參閱高載效能執行個體的其他 CloudWatch 指標。

T2 無限制執行個體沒有啟動額度

T2 標準執行個體會收到啟動額度，但 T2 無限制執行個體則不會收到。T2 Unlimited 執行個體可隨時超出基準，而無需額外付費，只要其平均CPU使用率在滾動 24 小時時段或其生命週期 (以較短者為準) 內或低於基準線即可。因此，T2 Unlimited 執行個體不需要使用啟動額度，以在啟動之後立即達成高效能。

如果 T2 執行個體從 `standard` 切換為 `unlimited`，則在將剩下的 `CPUCreditBalance` 結轉過去之前，會先從 `CPUCreditBalance` 移除所有累積的啟動額度。

T4g、T3a 和 T3 執行個體永遠不會收到啟動額度，因為它們支援無限制模式。無限制模式點數組態可讓 T4G、T3A 和 T3 執行個體視需要使用，以超越基準並視需要長時間使用。CPU

啟用無限制模式

在執行中或已停止的執行個體上，可以隨時從 `unlimited` 切換為 `standardUnlimited`，也可從 `standard` 切換為 `unlimited`。如需詳細資訊，請參閱 將爆量效能執行個體啟動為無限制或標準 和 如何修改爆量效能執行個體的額度規格。

您可以在每個 AWS 區域、每個高載績效執行環境系列的帳戶層次設`unlimited`定為預設信用選項，以便使用預設信用選項來啟動帳戶中的所有新高載績效執行環境。如需詳細資訊，請參閱設定帳戶的預設額度規格。

您可以檢查高載效能執行個體是設定為`unlimited`或`standard`使用 Amazon EC2 主控台或. AWS CLI如需詳細資訊，請參閱 檢視爆量效能執行個體的額度規格 和 檢視預設額度規格。

在無限制與標準之間切換時，額度會有什麼變化

`CPUCreditBalance`是追蹤執行個體累積的學分數目的 CloudWatch 量度。`CPUSurplusCreditBalance`是追蹤執行個體花費的剩餘積分數 CloudWatch 量的量度。

當您將設定為 unlimited 的執行個體變更為 standard，會發生下列情況：

- CPUCreditBalance 值會維持不變結轉過去。
- CPUSurplusCreditBalance 值會立即收取費用。

當 standard 執行個體切換為 unlimited 時，會發生下列情況：

- 將包含累積獲得之額度的 CPUCreditBalance 值結轉過去。
- 若為 T2 Standard 執行個體，從 CPUCreditBalance 值移除所有啟動額度，然後將剩下的 CPUCreditBalance 值 (包含累積獲得的額度) 結轉過去。

監控額度用量

若要查看執行個體花費的點數是否超過基準提供的額度，您可以使用 CloudWatch 指標追蹤使用情況，也可以設定每小時警示以收到點數用量通知。如需詳細資訊，請參閱監控高CPU載執行個體的積分。

高載執行個體的無限模式範例

下列範例說明設定為 unlimited 之執行個體的額度使用。

範例

- 範例 1：說明 T3 Unlimited 的額度使用
- 範例 2：說明 T2 Unlimited 的額度使用

範例 1：說明 T3 Unlimited 的額度使用

在這個範例中，您會看到t3.nano執行個體的CPU使用率啟動unlimited，以及它如何花費獲得和剩餘積分來維持CPU使用率。

t3.nano執行個體在滾動 24 小時期間內可CPU獲得 144 個點數，可兌換 144 分鐘的 v CPU 使用時間。當它耗盡其CPU信用餘額（以 CloudWatch 指標表示CPUCreditBalance）時，它可以將剩餘的CPU信用額度（尚未賺取）花費在需要的時間內爆發。由於 t3.nano 執行個體在 24 小時期間已獲得最高的 144 點，因此可以用掉等於此最高額度的剩餘額度，而不會立即產生費用。如果花費超過 144 個CPU學分，則會在小時結束時收取差額。

此範例的目的 (如下列圖表所示)，是要說明即使用完 CPUCreditBalance，執行個體如何可利用剩餘額度來大幅提升效能。下列的工作流程參考了圖表上的編號點：

P1 – 在圖表上的第 0 小時，執行個體會啟動為 unlimited，並立即開始獲得額度。執行個體從啟動時一直處於閒置狀態 — CPU 使用率為 0%，且不會花費點數。所有為使用的額度都會累計到額度餘額。前 24 小時，CPUCreditUsage 為 0，而 CPUCreditBalance 值達到其最大值 144。

P2 — 在接下來的 12 小時內，CPU利用率為 2.5%，低於 5% 的基線。執行個體獲得的額度超出其用掉的額度，但 CPUCreditBalance 值不得超出其最大值，即 144 點。

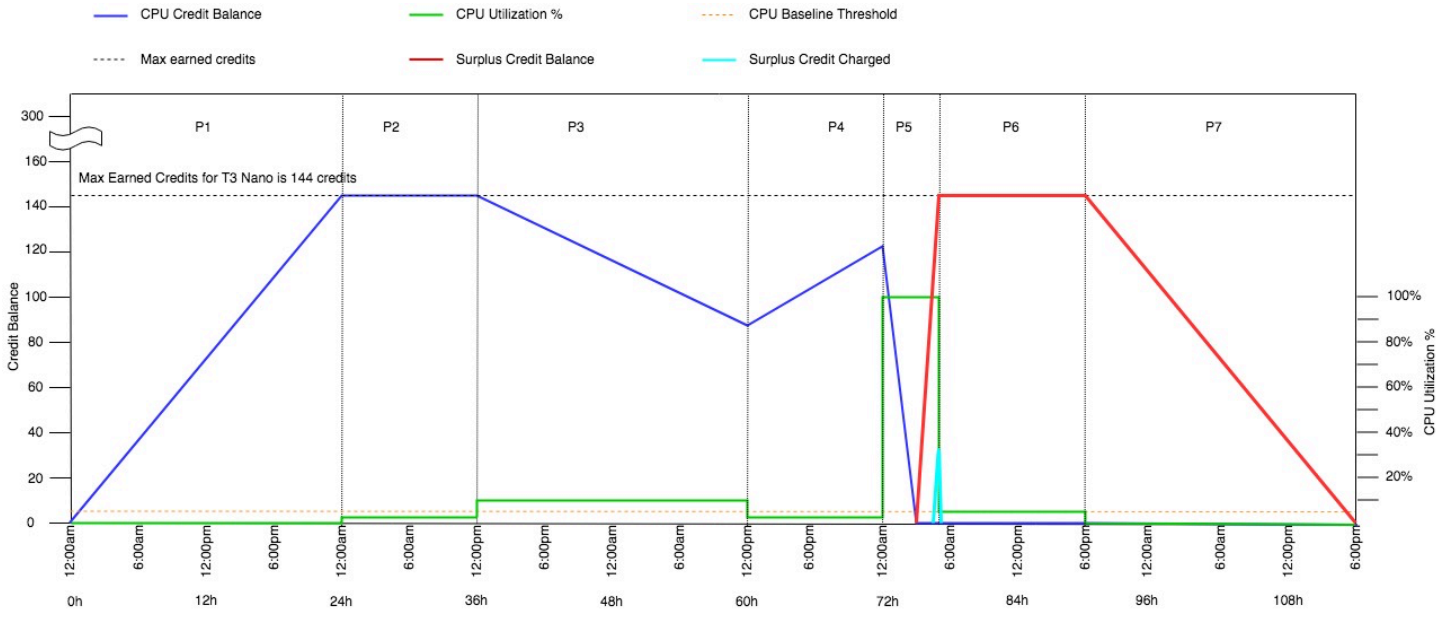
P3 — 在接下來的 24 小時內，CPU使用率為 7%（高於基準線），這需要支出 57.6 個學分。執行個體用掉的額度超過獲得的額度，而且 CPUCreditBalance 值減少至 86.4 點。

P4 — 在接下來的 12 小時內，CPU使用率會降低至 2.5% (低於基準線)，這需要花費 36 個學分。同時執行個體獲得 72 點。執行個體獲得的額度超過用掉的額度，而且 CPUCreditBalance 值增加至 122 點。

P5 — 在接下來的 5 小時內，執行個體會以 100% 的CPU使用率爆發，總共花費 570 個點數來維持爆發。在此期間大約一個小時後，執行個體耗盡了整個 CPUCreditBalance 122 個學分，並開始花費剩餘積分來維持高CPU使用率，在此期間共計 448 個剩餘積分 (570-122=448)。當CPUSurplusCreditBalance值達到 144 個CPU點數 (t3.nano執行個體在 24 小時內可獲得的最大值) 時，之後花費的任何剩餘積分都無法抵銷獲得的積分。之後用掉的剩餘額度合計達 304 點 (448-144=304)，因此在該小時結束時，會對 304 點產生少許的額外費用。

P6 — 在接下來的 13 小時內，CPU使用率為 5% (基準線)。執行個體獲得與其用掉一樣多的額度，不需額外支付 CPUSurplusCreditBalance。CPUSurplusCreditBalance 值仍為 144 點。

P7 — 在此範例中的過去 24 小時內，執行個體處於閒置狀態，CPU使用率為 0%。在此期間，執行個體獲得 144 點，用來支付 CPUSurplusCreditBalance。



範例 2：說明 T2 Unlimited 的額度使用

在這個範例中，您會看到t2.nano執行個體的CPU使用率啟動unlimited，以及它如何花費獲得和剩餘積分來維持CPU使用率。

t2.nano執行個體在滾動 24 小時期間內可CPU獲得 72 個點數，可兌換 72 分鐘的 v CPU 使用時間。當它耗盡其CPU信用餘額（以 CloudWatch 指標表示CPUCreditBalance）時，它可以將剩餘的CPU信用額度（尚未賺取）花費在需要的時間內爆發。由於 t2.nano 執行個體在 24 小時期間已獲得最高的 72 點，因此可以用掉等於此最高額度的剩餘額度，而不會立即產生費用。如果花費超過 72 個CPU學分，則會在小時結束時收取差額。

此範例的目的 (如下列圖表所示)，是要說明即使用完 CPUCreditBalance，執行個體如何可利用剩餘額度來大幅提升效能。您可以假設，在圖表中時間線的開頭，執行個體已有累積的額度餘額，此餘額等於該執行個體在 24 小時期間內可以獲得的最高額度。下列的工作流程參考了圖表上的編號點：

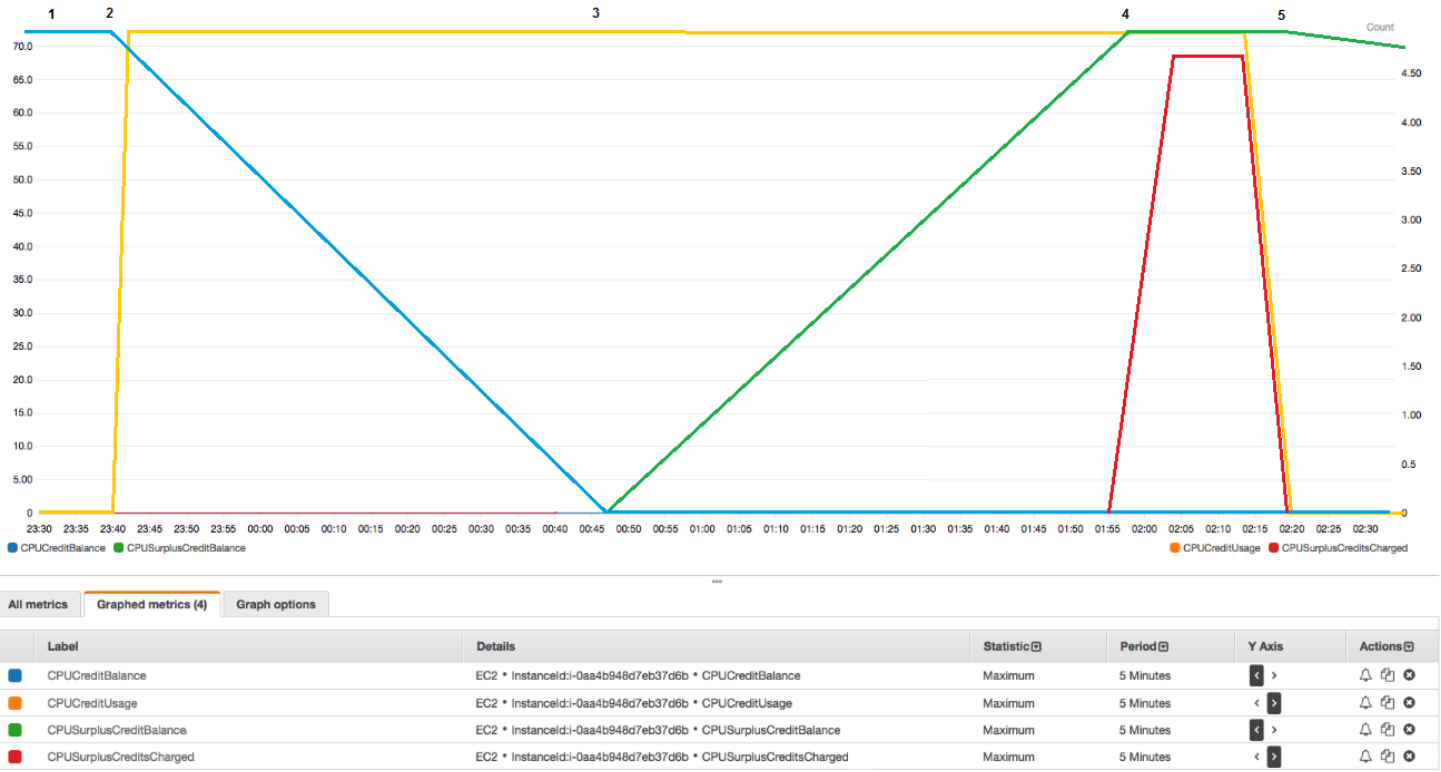
1 – 在前 10 分鐘期間，CPUCreditUsage 為 0，而 CPUCreditBalance 維持在最大值 72 點。

2 — 在 23:40，隨著CPU使用率的增加，執行個體支出CPU積分，而且值會減少。CPUCreditBalance

3 — 執行個體在 00:47 左右耗盡其整體CPUCreditBalance，並開始花費剩餘的積分來維持高使用率。CPU

4 — 剩餘學分花費到 01:55, 當CPUSurplusCreditBalance值達到 72 CPU 學分. 這等於 t2.nano 執行個體在 24 小時期間內可獲得的最高點數。之後，任何用掉的剩餘額度，都不能用 24 小時期間內所獲得的額度抵銷，因此在該小時結束時，會產生少許的額外費用。

5 – 執行個體繼續使用剩餘額度，直到大約 02:20。目前，使用CPU率低於基準，執行個體開始以每小時 3 個積分 (或每 5 分鐘 0.25 點) 賺取積分，用來支付。CPUSurplusCreditBalance在 CPUSurplusCreditBalance 值減少到 0 之後，執行個體開始以每 5 分鐘 0.25 點的速度，累積其 CPUCreditBalance 中的獲得額度。



計算賬單（Linux 實例）

剩餘點數的費用為每 v CPU 小時 0.05 美元。該實例在 01:55 到 02:20 之間花費了大約 25 剩餘積分，這相當於 0.42 伏-小時。CPU此執行個體的額外費用為 0.42 伏特小CPU時 x 每伏 0.05 美元/伏特CPU小時 = 0.021 美元，四捨五入至 0.02 美元。下列是這個 T2 Unlimited 執行個體月底的帳單：

Amazon Elastic Compute Cloud running Linux/UNIX		
$0.0058 per On Demand Linux t2.nano Instance Hour	720.000 Hrs	$4.18

Amazon Elastic Compute Cloud T2 CPU Credits		
$0.05 per vCPU-Hour of T2 CPU credits	0.420 vCPU-Hours	$0.02

計算帳單 (視窗執行個體)

剩餘積分每 v CPU 小時費用為 0.096 美元。該實例在 01:55 到 02:20 之間花費了大約 25 剩餘積分，這相當於 0.42 伏-小時。CPU此執行個體的額外費用為 0.42 伏特小CPU時 x 每伏 0.96 美元/伏特CPU小時 = 0.04032 美元，四捨五入至 0.04 美元。下列是這個 T2 Unlimited 執行個體月底的帳單：

Amazon Elastic Compute Cloud running Windows		
$0.0081 per On Demand Windows t2.nano Instance Hour	720.000 Hrs	$5.83

Amazon Elastic Compute Cloud T2 CPU Credits		
$0.096 per vCPU-Hour of T2 CPU credits	0.420 vCPU-Hours	$0.04

您可以設定帳單提醒，每小時通知任何累計的費用，並根據需要採取行動。

爆量效能執行個體的標準模式

高載效能執行個體設定`standard`為適合平均CPU使用率持續低於執行個體基準CPU使用率的工作負載。為了超出基準線，執行個體會花費在其CPU信用餘額中累積的積分。如果執行個體的累積信用額度不足，則CPU使用率會逐漸降低至基準層次，如此一來，執行個體在累積的CPU信用餘額耗盡時，就不會發生急劇的效能下降。如需詳細資訊，請參閱高載效能執行個體的關鍵概念。

目錄

- 高載執行個體的標準模式概念
  - 標準爆量效能執行個體如何運作
  - 啟動額度
  - 啟動額度限制
  - 啟動額度與獲得額度之間的差異
- 高載執行個體的標準模式範例
  - 範例 1：說明 T3 Standard 的額度使用
  - 範例 2：說明 T2 Standard 的額度使用
    - 期間 1：1 – 24 小時
    - 期間 2：25 – 36 小時
    - 期間 3：37 – 61 小時
    - 期間 4：62 – 72 小時
    - 期間 5：73 – 75 小時
    - 期間 6：76 – 90 小時
    - 期間 7：91 – 96 小時

高載執行個體的標準模式概念

standard 模式是爆量效能執行個體的組態選項。對於執行中或已停止的執行個體，隨時都可以啟用或停用它。您可以在每個 AWS 區域、每個高載績效執行環境系列的帳戶層次設standard定為預設信用選項，以便使用預設信用選項來啟動帳戶中的所有新高載績效執行環境。

標準爆量效能執行個體如何運作

當設定為 standard 的爆量效能執行個體處於執行中狀態時，每小時會持續獲得 (以毫秒級的解析度) 固定比率的額度。若為 T2 Standard，當執行個體停止時，會失去所有已累積的額度，而且其額度餘額會重設為零。當此等執行個體重新啟動時，會獲得一組新的啟動額度，並且開始累積獲得的額度。對於 T4G、T3A 和 T3 標準執行個體，在CPU執行個體停止後的 7 天內，積分餘額會遺失。如果您在七天內啟動執行個體，則不會失去任何額度。

T2 標準執行個體可獲得兩種類型的CPU積分：獲得積分和啟動積分。當 T2 Standard 執行個體處於執行狀態時，每小時會持續獲得 (以毫秒級的解析度) 固定比率的額度。一開始，執行個體尚未獲得額度，可能無法順暢啟動；因此，為了提供流暢啟動的體驗，執行個體在開始時會獲得啟動額度，可以在累積獲得額度的同時先用掉這些額度。

T4g、T3a 和 T3 執行個體不會收到啟動額度，因為它們支援無限制模式。無限制模式點數組態可讓 T4G、T3A 和 T3 執行個體視需要使用，以超越基準並視需要長時間使用。CPU

啟動額度

T2 標準執行個體在啟動或啟動CPU時每 v 可獲得 30 個啟動積分，而 T1 標準執行個體則可獲得 15 個啟動積分。例如，t2.micro執行個體有一個 v CPU 並取得 30 個啟動積分，而t2.xlarge執行個體則有四個， vCPUs 並取得 120 個啟動積分。啟動額度的設計旨在提供流暢的啟動體驗，讓執行個體能夠在尚未獲得累計的額度之前，先在啟動之後立即大幅提升效能。

啟動額度會比獲得的額度先用掉。未使用的啟動信用額度會累積在CPU信用餘額中，但不會計入CPU信用餘額限制。例如，t2.micro執行個體的CPU信用餘額限制為 144 個已獲得積分。如果它啟動並保持閒置 24 小時，則其積分餘CPU額達到 174（30 個啟動積分 + 144 個獲得的積分），超過了上限。不過，在該執行個體用掉 30 點的啟動額度之後，額度餘額不得超過 144 點。如需每個執行個體大小的CPU信用餘額限制的詳細資訊，請參閱點數表。

下表列出啟動或開始時收到的初始CPU信用分攤，以及的數目vCPUs。

執行個體類型	啟動額度	vCPUs
t1.micro	15	1

執行個體類型	啟動額度	vCPUs
t2.nano	30	1
t2.micro	30	1
t2.small	30	1
t2.medium	60	2
t2.large	60	2
t2.xlarge	120	4
t2.2xlarge	240	8

啟動額度限制

T2 Standard 執行個體可以獲得啟動額度的次數是有限的。預設的限制是所有 T2 Standard 執行個體共 100 次啟動或開始 (每個帳戶、每個區域、每個累計 24 小時期間合計)。例如，當 1 個執行個體在 24 小時期間內停止和啟動 100 次、當 100 個執行個體在 24 小時期間內啟動，或是當等於 100 次啟動的其他組合動作發生時，就會到達限制的次數。新帳戶的限制值可能較低，而根據您的使用量，此限制會隨時間提高。

> **Tip**
> 為確保工作負載隨時都能獲得所需的效能，請切換為 爆量效能執行個體的無限制模式，或考慮使用較大的執行個體大小。

啟動額度與獲得額度之間的差異

下表列出了啟動額度與獲得額度之間的差異。

	啟動額度	獲得的額度
額度獲得率	T2 標準執行個體在啟動或開始CPU時，每 v 可獲得 30 個啟動積分。	視執行個體大小而定，每個 T2 執行個體每小時持續獲得設定的CPU點數 (以毫秒級解析度)。如需每個執行個體大小

	啟動額度	獲得的額度
	如果 T2 執行個體從 `unlimited` 切換為 `standard`，在切換時不會獲得啟動額度。	獲得的CPU積分數量的詳細資訊，請參閱點數表。
額度獲得限制	獲得啟動額度的限制，是所有 T2 Standard 執行個體共 100 次啟動或開始 (每個帳戶、每個區域、每個累計 24 小時期間合計)。新帳戶的限制值可能較低，而根據您的使用量，此限制會隨時間提高。	T2 執行個體累積的積分不能超過CPU信用餘額限額。如果CPU信用餘額已達到上限，則在達到上限後獲得的任何積分都將被丟棄。啟動額度不會列入此限額的計算。如需每個 T2 執行個體大小的CPU信用餘額限制的詳細資訊，請參閱點數表。
額度使用	啟動額度會比獲得的額度先用掉。	獲得的額度只會在所有啟動額度用掉之後才使用。
額度過期	當 T2 Standard 執行個體正在執行時，啟動額度不會過期。如果 T2 Standard 執行個體停止，或是切換為 T2 Unlimited，所有的啟動額度都會消失。	當 T2 執行個體執行時，已經累績獲得的額度不會過期。當 T2 執行個體停止時，所有累績獲得的額度都會消失。

CloudWatch 指標會追蹤累積的啟動積分和累積的獲得積分數目`CPUCreditBalance`。如需詳細資訊，請參閱CloudWatch 測量結果表格`CPUCreditBalance`中的。

高載執行個體的標準模式範例

下列範例說明當執行個體設定為 `standard` 時的額度使用。

範例

- 範例 1：說明 T3 Standard 的額度使用
- 範例 2：說明 T2 Standard 的額度使用

範例 1：說明 T3 Standard 的額度使用

在此範例中，您會看到啟動為 `t3.nano` 的 `standard` 執行個體如何獲得、累積和用掉獲得的額度。您會看到額度餘額如何反映累積獲得的額度。

執行中的 t3.nano 執行個體每 24 小時會獲得 144 點。其額度餘額限制為獲得的 144 點。到達限額之後，獲得的新額度都將被捨棄。如需可獲得和累積的額度詳細資訊，請參閱額度表。

您可能會啟動 T3 Standard 執行個體，並立即使用它。或是您可能會啟動 T3 Standard 執行個體，然後讓該執行個體閒置幾天之後，才在其上執行應用程式。執行個體是否在使用中或處於閒置狀態，將決定額度是否會用掉或累積。如果執行個體在啟動後 24 小時的期間仍維持閒置，則額度餘額會達到其限制，即獲得的額度在累積後的上限。

此範例說明在啟動後 24 小時期間仍維持閒置的執行個體，並逐步說明 96 小時期間的 7 個階段、顯示獲得、累積、用掉和捨棄的比率，以及每個期間結束時的額度餘額值。

下列的工作流程參考了圖表上的編號點：

P1 – 在圖表上的第 0 小時，執行個體會啟動為 standard，並立即開始獲得額度。執行個體從啟動時一直處於閒置狀態 — CPU 使用率為 0%，且不會花費點數。所有為使用的額度都會累計到額度餘額。前 24 小時，CPUCreditUsage 為 0，而 CPUCreditBalance 值達到其最大值 144。

P2 — 在接下來的 12 小時內，CPU利用率為 2.5%，低於 5% 的基線。執行個體獲得的額度超出其用掉的額度，但 CPUCreditBalance 值不得超出其最大值，即 144 點。任何獲得的額度若超出限制，即會遭到捨棄。

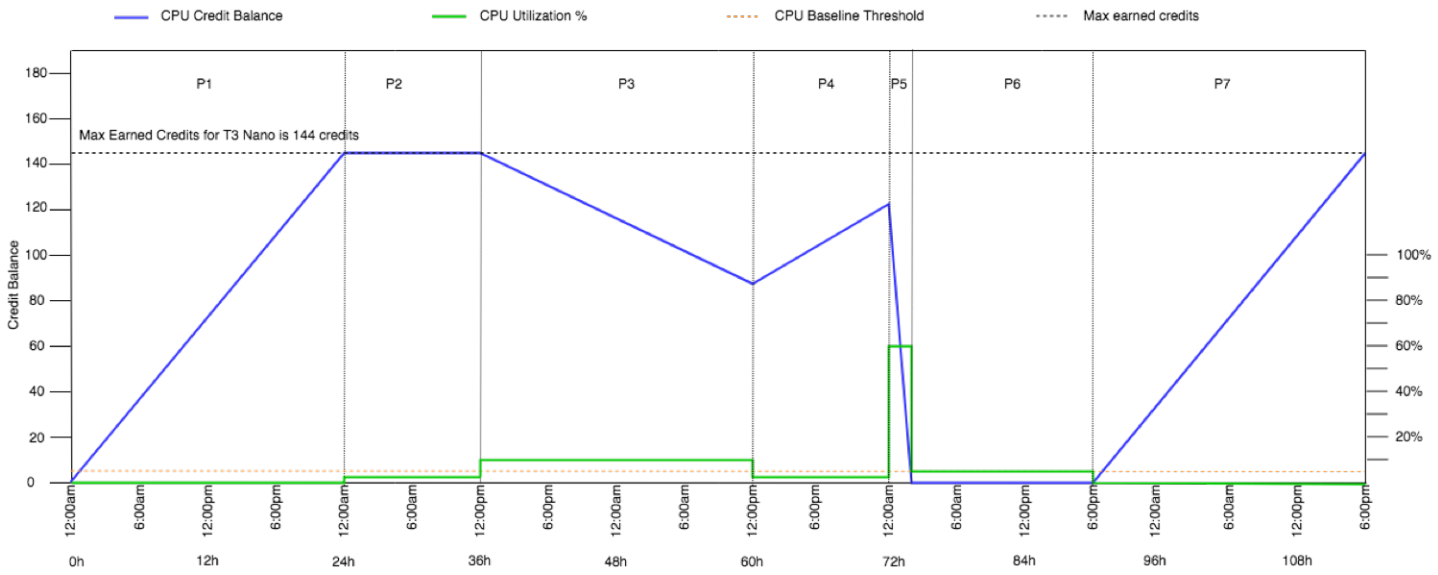
P3 — 在接下來的 24 小時內，CPU使用率為 7%（高於基準線），這需要支出 57.6 個學分。執行個體用掉的額度超過獲得的額度，而且 CPUCreditBalance 值減少至 86.4 點。

P4 — 在接下來的 12 小時內，CPU使用率會降低至 2.5% (低於基準線)，這需要花費 36 個學分。同時執行個體獲得 72 點。執行個體獲得的額度超過用掉的額度，而且 CPUCreditBalance 值增加至 122 點。

P5 — 在接下來的兩個小時內，執行個體以 60% 的CPU使用率爆發，並耗盡整個 122 個學分的CPUCreditBalance價值。在此期間結束時，CPU使用率為零CPUCreditBalance時，會強制降至 5% 的基準使用率層級。處於基準時，執行個體獲得與其用掉一樣多的額度。

P6 — 在接下來的 14 小時內，CPU使用率為 5% (基準線)。執行個體獲得與其用掉一樣多的額度。CPUCreditBalance 值仍為 0。

P7 — 在此範例中的過去 24 小時內，執行個體處於閒置狀態，CPU使用率為 0%。在此期間，執行個體獲得 144 點，其會累積在 CPUCreditBalance。



範例 2：說明 T2 Standard 的額度使用

在此範例中，會示範做為 `t2.nano` 啟動的 `standard` 執行個體，如何獲得、累積和用掉啟動與獲得額度。範例中將會顯示額度餘額不只反映累積獲得的額度，也會反映累積的啟動額度。

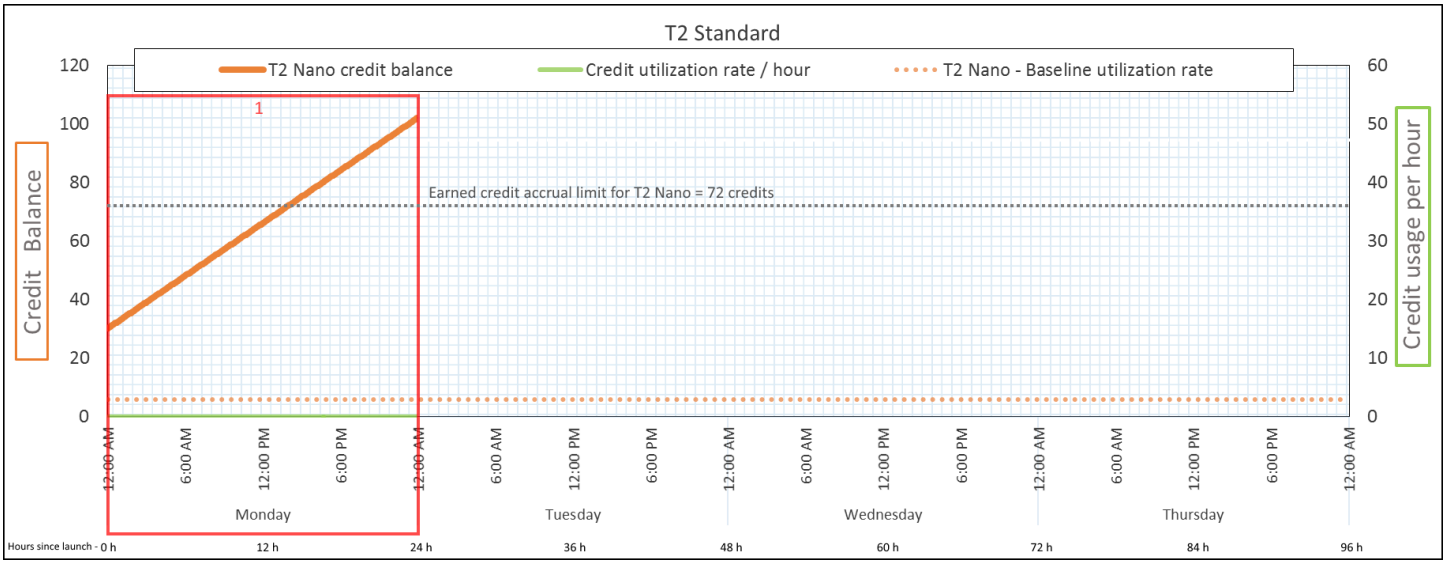
`t2.nano` 執行個體會在啟動時獲得 30 點的啟動額度，而且每 24 小時可獲得 72 點。其額度餘額上限為 72 點獲得的額度；啟動額度不會列入此限額的計算。到達限額之後，獲得的新額度都將被捨棄。如需可獲得和累積的額度詳細資訊，請參閱額度表。如需限制的詳細資訊，請參閱 啟動額度限制。

您可能會啟動 T2 Standard 執行個體，並立即使用它。或是您可能會啟動 T2 Standard 執行個體，然後讓該執行個體閒置幾天之後，才在其上執行應用程式。執行個體是否在使用中或處於閒置狀態，將決定額度是否會用掉或累積。如果執行個體在啟動後 24 小時的期間仍維持閒置，額度餘額會顯示為超過其限額，因為該餘額同時反映了累積獲得的額度和累積的啟動額度。但CPU是，在使用之後，啟動積分會先花費。之後，此限額一律反映可累積獲得額度的上限。

此範例說明在啟動後 24 小時期間仍維持閒置的執行個體，並逐步說明 96 小時期間的 7 個階段、顯示獲得、累積、用掉和捨棄的比率，以及每個期間結束時的額度餘額值。

期間 1：1 – 24 小時

在圖表上的第 0 小時，T2 執行個體會啟動為 `standard`，並立即獲得 30 點的啟動額度。該執行個體會在執行狀態中獲得額度。執行個體從啟動時一直處於閒置狀態 — CPU 使用率為 0%，且不會花費點數。所有為使用的額度都會累計到額度餘額。在啟動後大約 14 小時，其額度餘額為 72 點 (30 點的啟動額度 + 42 點獲得的額度)，等於該執行個體可以在 24 小時中所獲得的額度。在啟動 24 小時後，額度餘額超過 72 點，因為有未使用的啟動額度累計到額度餘額—額度餘額為 102 點 (30 點的啟動額度 + 72 點獲得的額度)。



CPU 額度使用率	每 24 小時 0 學分 (0% CPU 使用率)
額度獲得率	每 24 小時 72 點
額度捨棄率	每 24 小時 0 點
額度餘額	102 點 (30 點的啟動額度 + 72 點獲得的額度)

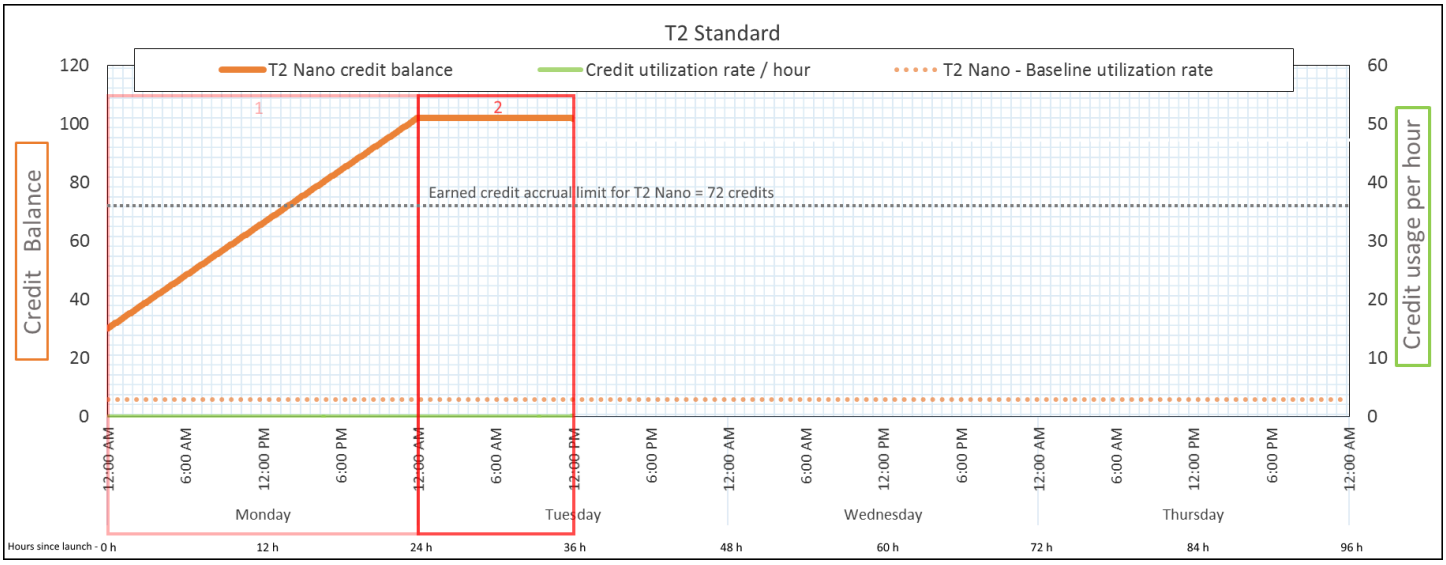
結論

如果啟動後沒有CPU使用率，則執行環境會累積比 24 小時內可獲得的額度多 (30 個啟動積分 + 72 個已認列積分 = 102 個學分)。

在實際案例中，EC2執行個體在啟動和執行時會耗用少量的點數，這樣可防止餘額達到此範例中的最大理論值。

期間 2：25 – 36 小時

在接下來的 12 小時內，該執行個體會繼續保持閒置和獲得額度，但額度餘額不會增加。餘額會穩定地保持在 102 點 (30 點的啟動額度 + 72 點獲得的額度)。額度餘額已到達其上限 (72 點獲得的額度)，因此新獲得的額度將被捨棄。



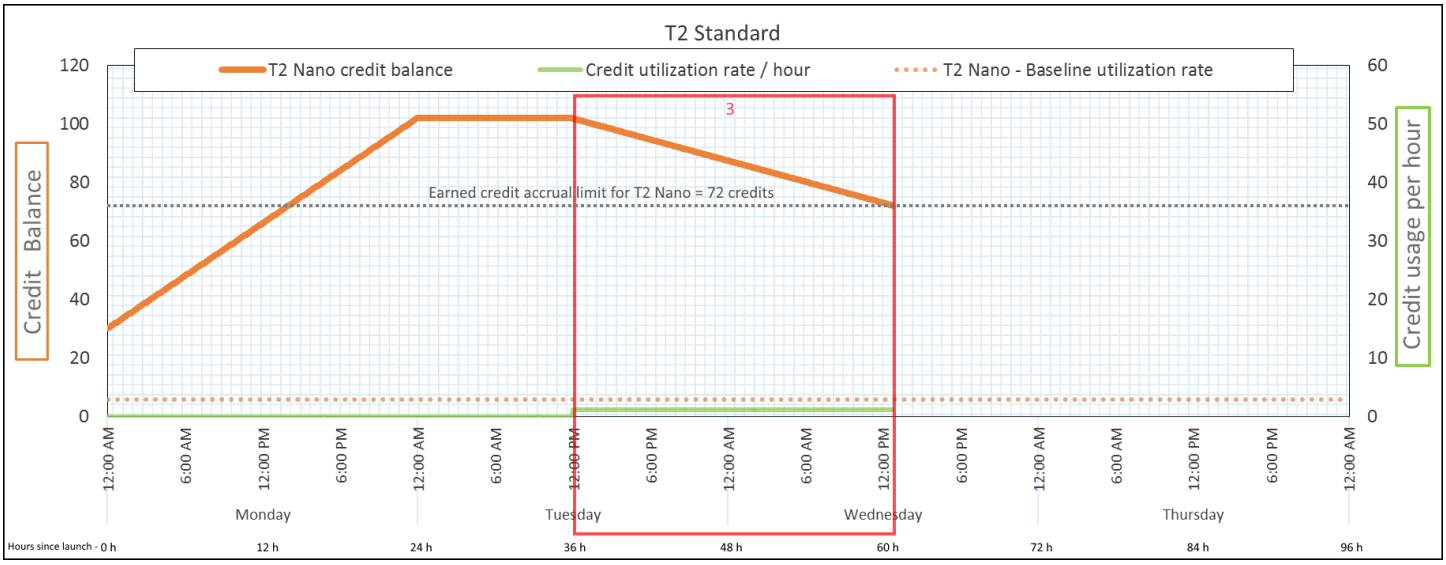
CPU 額度使用率	每 24 小時 0 學分 (0% CPU 使用率)
額度獲得率	每 24 小時 72 點 (每小時 3 點)
額度捨棄率	每 24 小時 72 點 (100% 的額度獲得率)
額度餘額	102 點 (30 點的啟動額度 + 72 點獲得的額度) – 餘額未變

結論

執行個體會持續地獲得額度，但如果額度餘額已到達其上限，就不能再累積更多獲得的額度。到達限額之後，獲得的新額度都將被捨棄。啟動額度不會列入額度餘額限額的計算。如果餘額中包含累計的啟動額度，則餘額看起來會像是超過限額。

期間 3：37 – 61 小時

在接下來的 25 小時內，執行個體使用 2%CPU，這需要 30 個點數。在同一個期間，執行個體獲得 75 點，但額度餘額減少了。餘額減少是由於累積的啟動額度會先用掉，而額度餘額已到達 72 點獲得額度的限制，因此新獲得的額度被捨棄。



CPU 額度使用率	每 24 小時 28.8 點（每小時 1.2 個點數，2% 的 CPU使用率，40% 的信用賺取率）—25 小時內 30 個積分
額度獲得率	每 24 小時 72 點
額度捨棄率	每 24 小時 72 點 (100% 的額度獲得率)
額度餘額	72 點 (30 點的啟動額度已用掉；72 點獲得的額度尚未用掉)

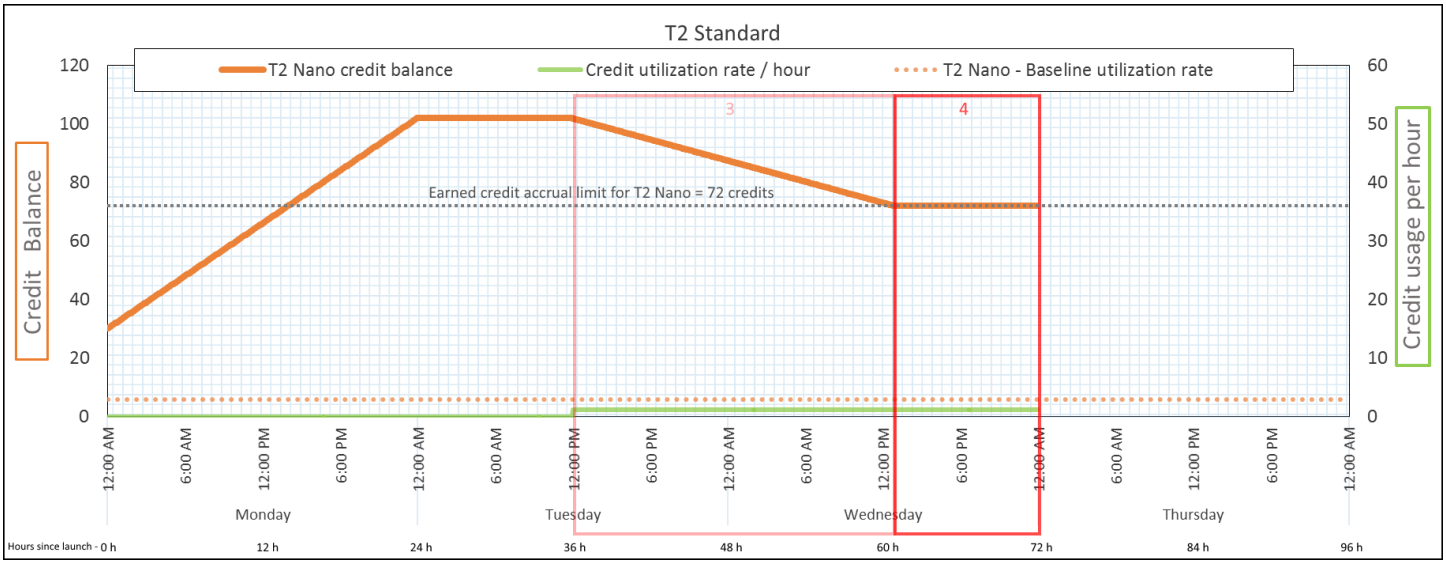
結論

執行個體會先用掉啟動額度，然後才會使用獲得的額度。啟動額度不會列入額度限額的計算。在啟動額度用掉之後，餘額就永遠不會高於 24 小時內可獲得的額度。此外，當執行個體執行時，就無法再獲得更多啟動額度。

期間 4：62 – 72 小時

在接下來的 11 小時內，執行個體使用 2%CPU，這需要 13.2 個點數。這與前期的CPU使用率相同，但餘額不會減少。而是維持在 72 點。

餘額沒有減少是因為額度獲得率高於額度使用率。在此期間執行個體用掉 13.2 點，但也獲得了 33 點。但是，餘額的上限為 72 點，因此任何獲得的額度只要超過此限額，就會被捨棄。餘額會穩定地保持在 72 點，這跟期間 2 維持在 102 點不同，因為沒有累積的啟動額度。



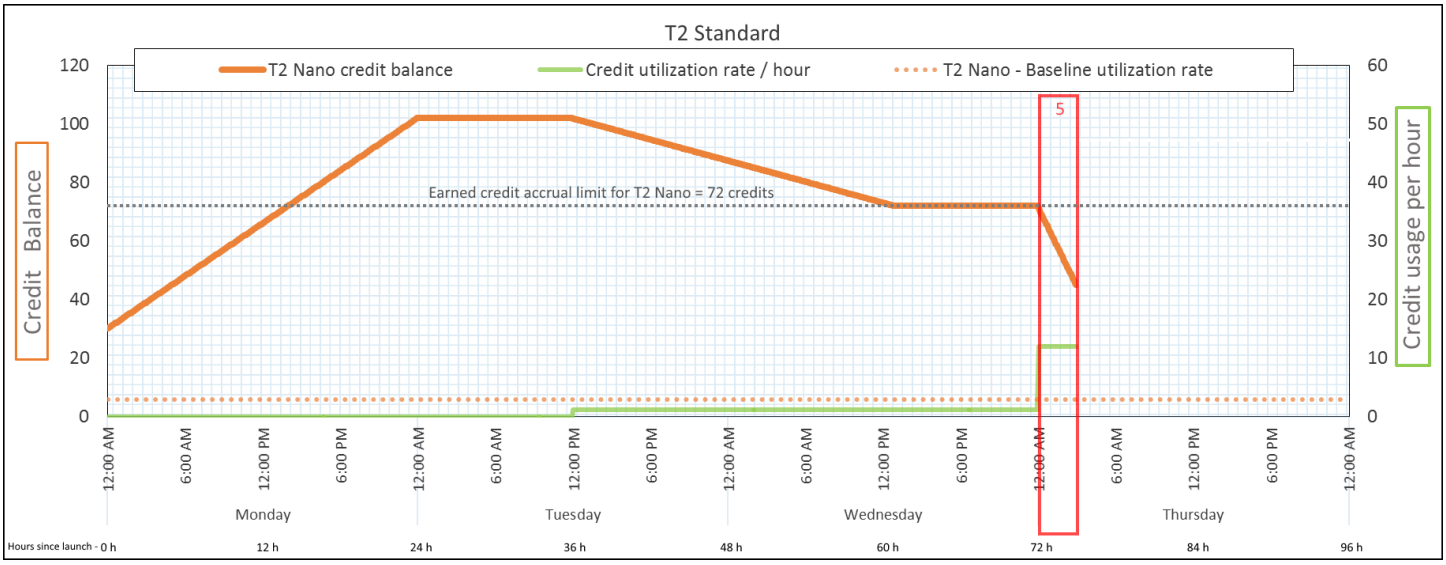
CPU 額度使用率	每 24 小時提供 28.8 點 (每小時 1.2 個點數、2% 的CPU使用率、40% 的信用賺取率) —11 小時內有 13.2 個學分
額度獲得率	每 24 小時 72 點
額度捨棄率	每 24 小時 43.2 點 (60% 的額度獲得率)
額度餘額	72 點 (0 點的啟動額度 + 72 點獲得的額度) – 餘額到達上限

結論

在啟動額度用掉之後，額度餘額的上限，會取決於執行個體在 24 小時內可獲得的額度。如果執行個體獲得的額度比用掉的更多，新獲得的額度只要超過限額，就會被捨棄。

期間 5：73 – 75 小時

在接下來的三個小時內，執行個體會以 20% 的CPU使用率爆發，這需要 36 個點數。在同樣的 3 小時期間，執行個體會獲得 9 點，這會使得額度淨餘額減少 27 點。在 3 小時期間結束時，額度餘額為 45 點累積獲得的額度。



CPU 額度使用率	每 24 小時 288 個積分（每小時 12 個學分，CPU使用率為 20%，獲得的信用額度的 400%）—3 個小時內可獲得 36 個積分
額度獲得率	每 24 小時 72 點 (3 個小時 9 點)
額度捨棄率	每 24 小時 0 點
額度餘額	45 點 (先前的餘額 (72) - 用掉的額度 (36) + 獲得的額度 (9)) – 在每 24 小時獲得 216 點的額度獲得率之下，但額度餘額減少了，額度餘額減少 (使用率 288/24 + 獲得率 72/24 = 餘額減少率 216/24)

結論

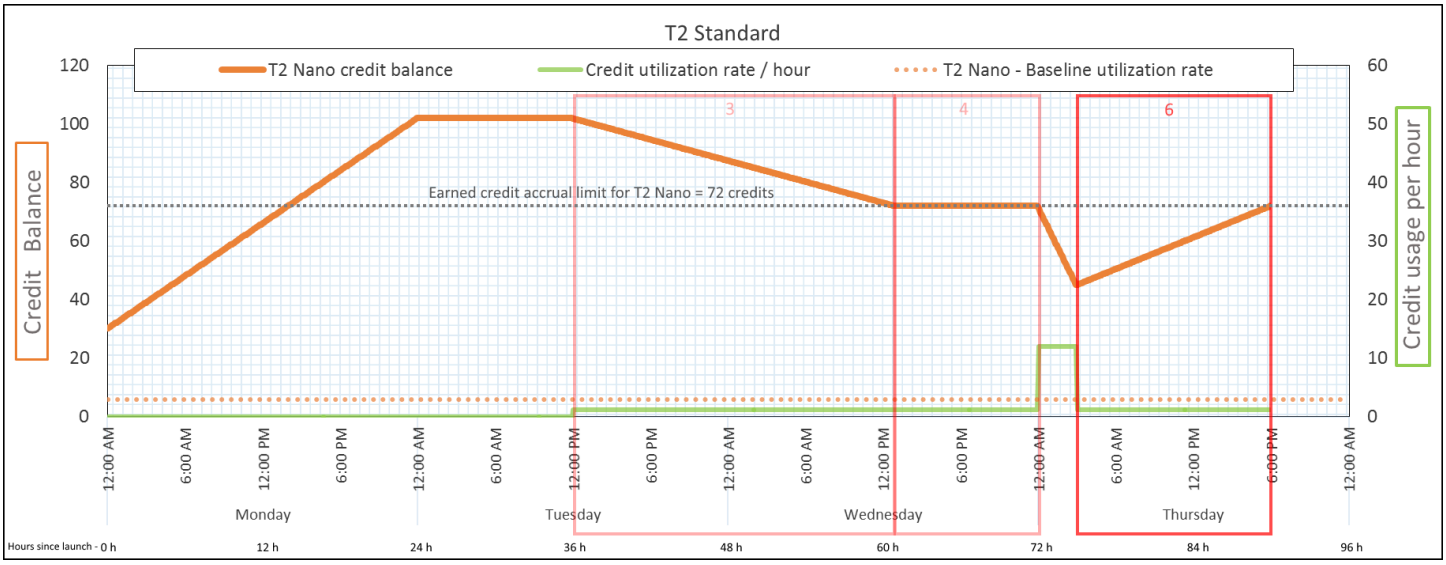
如果執行個體用掉的額度比獲得的多，其額度餘額會減少。

期間 6：76 – 90 小時

在接下來的 15 小時內，執行個體使用 2%CPU，這需要 18 個點數。此CPU使用率與「期間 3」與「期間 4」中的使用率相同。不過，餘額在此期間是增加的，而期間 3 和期間 4 的餘額則分別是減少和持平。

在期間 3 中用掉了累積的啟動額度，並捨棄超過額度限額的所有獲得額度，因此使額度餘額減少。在期間 4 中，執行個體用掉的額度比獲得的少。捨棄超過額度限額的所有獲得額度，因此額度餘額持平，維持在 72 點的最高值。

在此期間中沒有累積的啟動額度，而餘額中累積獲得的額度低於限額。沒有任何獲得的額度被捨棄。此外，執行個體獲得的額度比用掉的多，因此使額度餘額增加。



CPU 額度使用率	每 24 小時 28.8 點（每小時 1.2 個積分，2% 的 CPU使用率，40% 的信用賺取率）-15 小時內獲得 18 個積分
額度獲得率	每 24 小時 72 點 (15 個小時 45 點)
額度捨棄率	每 24 小時 0 點
額度餘額	72 點 (餘額以每小時 43.2 點的速度增加 – 變動率 = 使用率 28.8/24 + 獲得率 72/24)

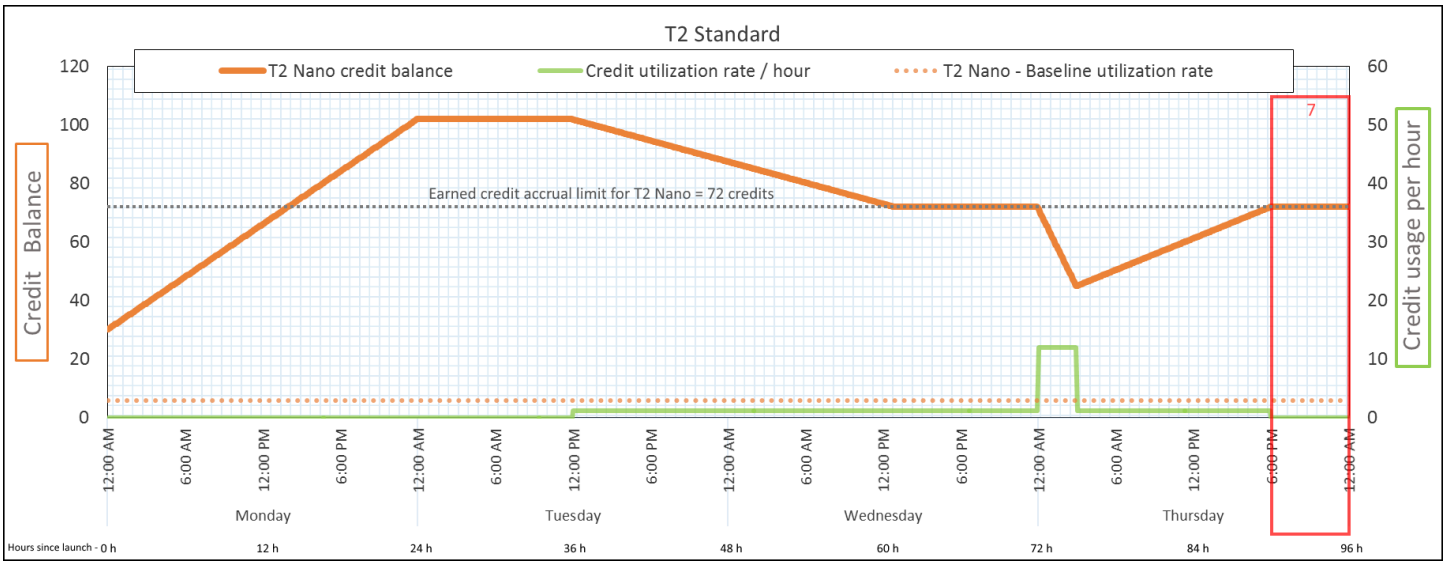
結論

如果執行個體用掉的額度比獲得的少，其額度餘額會增加。

期間 7：91 – 96 小時

在接下來的六個小時內，執行個體會保持閒置狀態，CPU使用率為 0%，且不會花費點數。此CPU使用率與「期間 2」中的使用率相同，但餘額不會達到 102 個點數 — 高原為 72 個學分，也就是執行個體的信用餘額限制。

在期間 2 中，額度餘額包含 30 點累積的啟動額度。在期間 3 中用掉了啟動額度。執行中的執行個體無法再獲得更多的啟動額度。在到達其額度餘額上限後，任何獲得的額度只要超過此限額，就會被捨棄。



CPU 額度使用率	每 24 小時 0 學分 (0% CPU 使用率)
額度獲得率	每 24 小時 72 點
額度捨棄率	每 24 小時 72 點 (100% 的額度獲得率)
額度餘額	72 點 (0 點的啟動額度、72 點獲得的額度)

結論

執行個體會持續地獲得額度，但如果額度餘額已到達其上限，就不能再累積更多獲得的額度。到達限額之後，獲得的新額度都將被捨棄。額度餘額的上限，會取決於執行個體在 24 小時內可獲得的額度。如需額度餘額上限的詳細資訊，請參閱額度表。

使用爆量效能執行個體

啟動、監視和修改高載效能執行個體 (T 執行個體) 的步驟類似。主要差異為它們啟動時的預設額度規格。

每個 T 執行個體系列都有下列預設信用規格：

- T4G、T3A 和 T3 執行個體的推出方式為 `unlimited`
- 專用執行個體中的 T3 執行個體只能以 `standard` 狀態啟動
- T2 執行個體啟動為 `standard`

可變更帳戶的預設額度規格。

目錄

- 將爆量效能執行個體啟動為無限制或標準
- 使用 Auto Scaling 群組將爆量效能執行個體啟動為無限制
- 檢視爆量效能執行個體的額度規格
- 如何修改爆量效能執行個體的額度規格
- 設定帳戶的預設額度規格
- 檢視預設額度規格

將爆量效能執行個體啟動為無限制或標準

您可以使用 Amazon EC2 主控台、命令列工具`unlimited`或`standard`使用 Auto Scaling 群組來啟動 T 執行個體。 AWS SDK

下列程序說明如何使用EC2主控台或 AWS CLI。如需有關使用「Auto Scaling」群組的資訊，請參閱使用 Auto Scaling 群組將爆量效能執行個體啟動為無限制。

Console

將 T 執行個體啟動為無限制或標準

1. 請遵循該程序來啟動執行個體。
2. 在 Instance type (執行個體類型) 中，選取 T 執行個體類型。
3. 展開 Advanced details (進階詳細資訊)，並針對 Credit specification (額度規格)，選取額度規格。如果您沒有進行選取，則會使用預設值 (適用`standard`於 T2)，也就是 T4G、T3A 和 T3。`unlimited`
4. 在 Summary (摘要) 面板中，檢閱您的執行個體組態，然後選擇 Launch instance (啟動執行個體)。如需詳細資訊，請參閱使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體。

AWS CLI

將 T 執行個體啟動為無限制或標準

使用 run-instances 命令來啟動您的執行個體。使用 `--credit-specification CpuCredits=` 參數來指定額度規格。有效的點數規格為 `unlimited` 和 `standard`。

- 對於 T4G、T3A 和 T3，如果您不包含`--credit-specification`參數，則執行個體會依`unlimited`預設啟動。
- 若為 T2，如果未包含 `--credit-specification` 參數，執行個體預設會啟動為 `standard`。

```
aws ec2 run-instances \
    --image-id ami-abc12345 \
    --count 1 \
    --instance-type t3.micro \
    --key-name MyKeyPair \
    --credit-specification "CpuCredits=unlimited"
```

使用 Auto Scaling 群組將爆量效能執行個體啟動為無限制

啟動或啟動 T 執行個體時，它們需要CPU積分才能獲得良好的啟動載入體驗。如果您使用 Auto Scaling 群組來啟動您的執行個體，我們建議將您的執行個體設定為 `unlimited`。如此在由 Auto Scaling 群組自動啟動或重新啟動時，這些執行個體即可使用剩餘額度。請使用剩餘額度來防止效能受限。

建立啟動範本

您必須使用「啟動範本」，在 Auto Scaling 群組中將執行個體啟動為 `unlimited`。啟動組態不支援將執行個體啟動為 `unlimited`。

> **Note**
>
> `unlimited` 模式不支援在專用執行個體上啟動的 T3 執行個體。

Console

建立以「無限制」狀態啟動執行個體的啟動範本

1. 遵循 Amazon EC2 Auto Scaling 使用者指南中的使用進階設定建立啟動範本程序。

2. 在啟動範本內容中，針對執行個體類型，選擇執行個體大小。
3. 若要在 Auto Scaling 群組中以 `unlimited` 的形式啟動執行個體，請在進階詳細資料下，針對額度規格選擇無限制。
4. 在您完成定義啟動範本參數時，選擇 Create launch template (建立啟動範本)。

AWS CLI

建立以「無限制」狀態啟動執行個體的啟動範本

使用指create-launch-template令並指定`unlimited`為信用規格。

- 對於 T4G、T3A 和 T3，如果您未包含該`CreditSpecification={CpuCredits=unlimited}`值，則執行個體會依`unlimited`預設啟動。
- 若為 T2，如果未包含 `CreditSpecification={CpuCredits=unlimited}` 值，執行個體預設會啟動為 `standard`。

```
aws ec2 create-launch-template \
    --launch-template-name MyLaunchTemplate \
    --version-description FirstVersion \
    --launch-template-data
 ImageId=ami-8c1be5f6,InstanceType=t3.medium,CreditSpecification={CpuCredits=unlimited}
```

使用 Auto Scaling 群組與啟動範本建立關聯

若要建立啟動範本與 Auto Scaling 群組的關聯，請使用啟動範本來建立 Auto Scaling 群組，或是將啟動範本加入現有的 Auto Scaling 群組。

Console

使用啟動範本建立 Auto Scaling 群組

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在畫面上方的導覽列中，選取與您用來建立啟動範本相同的區域。
3. 在導覽窗格中，選擇 Auto Scaling Groups (AS 群組) 與 Create Auto Scaling group (建立 AS 群組)。
4. 選取 Launch Template (啟動範本)、選取您的啟動範本，然後選取 Next Step (下一步)。

5. 填寫 Auto Scaling 群組的欄位。檢閱完您在 Review page (檢閱頁面) 上的組態設定後，請選擇 Create Auto Scaling group (建立 Auto Scaling 群組)。如需詳細資訊，請參閱 Amazon Auto Scaling 使用者指南中的使用啟動範本建立 EC2 Auto Scaling 群組。

AWS CLI

使用啟動範本建立 Auto Scaling 群組

使用指create-auto-scaling-group AWS CLI 令並指定`--launch-template`參數。

Console

將啟動範本新增至現有的 Auto Scaling 群組

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在畫面上方的導覽列中，選取與您用來建立啟動範本相同的區域。
3. 在導覽窗格中，選擇 Auto Scaling Groups (AS 安全群組)。
4. 從 Auto Scaling 群組清單中，選取 Auto Scaling 群組，然後選取 Actions (動作)、Edit (編輯)。
5. 在 Details (詳細資訊) 標籤上，對於 Launch Template (啟動範本)，選擇一個啟動範本，然後選擇 Save (儲存)。

AWS CLI

將啟動範本新增至現有的 Auto Scaling 群組

使用指update-auto-scaling-group AWS CLI 令並指定`--launch-template`參數。

檢視爆量效能執行個體的額度規格

您可以檢視執行中`unlimited`或已停止 T 執行處理的信用規格 (或`standard`)。

Console

若要檢視 T 執行環境的信用規格，請執行下

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側導覽窗格中選擇 Instances (執行個體)。

3. 選取實例。
4. 選擇詳細資訊，然後檢視額度規格欄位。此值為 `unlimited` 或 `standard`。

AWS CLI

若要說明 T 執行個體的信用規格

使用指describe-instance-credit-specifications令。如果您未指定一或多個執行環境IDs，則會傳回信用規格為的`unlimited`所有執行環境，以及先前已使`unlimited`用信用規格設定配置的執行環境。例如，如果您將 T3 執行個體調整為 M4 執行個體的大小，而其設定為`unlimited`，Amazon 會EC2傳回 M4 執行個體。

```
aws ec2 describe-instance-credit-specifications --instance-id i-1234567890abcdef0
```

範例輸出

```
{
  "InstanceCreditSpecifications": [
    {
        "InstanceId": "i-1234567890abcdef0",
        "CpuCredits": "unlimited"
    }
  ]
}
```

如何修改爆量效能執行個體的額度規格

您可以在`unlimited`和之間隨時切換執行中或已停止的 T 執行個體的信用規格`standard`。

請注意，在 `unlimited` 模式中，執行個體可能會使用剩餘抵用金，這可能會產生額外費用。如需詳細資訊，請參閱剩餘額度可能會產生費用。

Console

若要修改 T 執行環境的信用規格，請執行下

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側導覽窗格中選擇 Instances (執行個體)。

3. 選取實例。若要一次修改多個執行個體的額度規格，請選取所有適用的執行個體。
4. 選擇動作、執行個體設定、變更額度規格。只有在選取 T 例證時，才會啟用此選項。
5. 若要將額度規格變更為 `unlimited`，請選取執行個體 ID 旁的核取方塊。若要將額度規格變更為 `standard`，請清除執行個體 ID 旁的核取方塊。

AWS CLI

若要修改 T 執行環境的信用規格，請執行下

使用指modify-instance-credit-specification令。使用 `--instance-credit-specification` 參數來指定執行個體及其額度規格。有效的額度規格為 `unlimited` 和 `standard`。

```
aws ec2 modify-instance-credit-specification \
    --region us-east-1 \
    --instance-credit-specification
 "InstanceId=i-1234567890abcdef0,CpuCredits=unlimited"
```

範例輸出

```
{
  "SuccessfulInstanceCreditSpecifications": [
    {
      "InstanceId": "i- 1234567890abcdef0"
    }
  ],
  "UnsuccessfulInstanceCreditSpecifications": []
}
```

設定帳戶的預設額度規格

每個 T 執行個體系列都有預設信用規格。您可以在每個 AWS 區域的科目層次，變更每個 T 執行環境系列的預設信用規格。

如果您使用EC2主控台中的啟動執行環境精靈來啟動執行環境，則您為信用規格選取的值會覆寫帳戶層次的預設信用規格。如果您使用 AWS CLI 來啟動執行環境，則會使用預設信用規格來啟動帳戶中的所有新 T 執行環境。現有執行中或已停止執行個體的額度規格不受影響。

考量事項

執行個體系列的預設額度規格只能在連續 5 分鐘的期間內修改一次，而在連續 24 小時內最多可修改四次。

Console

在每個區域的帳戶層級設定預設額度規格

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 若要變更 AWS 區域，請使用頁面右上角的「地區」選取器。
3. 在左側導覽窗格中，選擇 [EC2儀表板]。
4. 在 Account attributes (帳戶屬性) 中，選擇 Default credit specification (預設額度規格)。
5. 選擇 Manage (管理)。
6. 針對每個執行個體系列選擇 Unlimited (無限制) 或 Standard (標準)，然後選擇 Update (更新)。

AWS CLI

在帳戶層次設定預設額度規格 (AWS CLI)

使用指modify-default-credit-specification令。使用 `--cpu-credits` 參數指定 AWS 區域、執行個體系列和預設額度規格。有效的預設額度規格為 `unlimited` 和 `standard`。

```
aws ec2 modify-default-credit-specification \
    --region us-east-1 \
    --instance-family t2 \
    --cpu-credits unlimited
```

檢視預設額度規格

您可以在每個 AWS 區域的科目層次，檢視 T 執行環境系列的預設信用規格。

Console

若要在科目層次檢視預設的信用規格，請執行下

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 若要變更 AWS 區域，請使用頁面右上角的「地區」選取器。
3. 在左側導覽窗格中，選擇 [EC2儀表板]。

4. 在 Account attributes (帳戶屬性) 中，選擇 Default credit specification (預設額度規格)。

AWS CLI

若要在科目層次檢視預設的信用規格，請執行下

使用指get-default-credit-specification令。指定 AWS 區域和執行個體系列。

```
aws ec2 get-default-credit-specification --region us-east-1 --instance-family t2
```

監控高CPU載執行個體的積分

EC2將指標發送到 Amazon CloudWatch。您可以在 CloudWatch 主控台的 Amazon EC2 每個執行個體指標中查看CPU積分指標，或使用列 AWS CLI 出每個執行個體的指標。如需詳細資訊，請參閱CloudWatch 您執行個體可用的指標。

目錄

- 高載效能執行個體的其他 CloudWatch 指標
- 計算CPU信用用量

高載效能執行個體的其他 CloudWatch 指標

高載效能執行個體具有下列額外 CloudWatch 指標，每五分鐘更新一次：

- `CPUCreditUsage`— 在測量期間花費的CPU學分數。
- `CPUCreditBalance`— 執行個體已累CPU積的學分數。當CPU爆發和CPU積分花費得比賺取的速度更快時，這種餘額就會耗盡。
- `CPUSurplusCreditBalance`— 當`CPUCreditBalance`值為零時，用於維持使CPU用率的剩餘CPU信用額度數。
- `CPUSurplusCreditsCharged`— 超過可以在 24 小時內獲得的最大CPU學分數量的剩餘CPU積分數量，從而吸引了額外的費用。

最後兩種指標只適用於設定為 `unlimited` 的執行個體。

下表說明高載效能執行處理的 CloudWatch 測量結果。如需詳細資訊，請參閱CloudWatch 您執行個體可用的指標。

指標	描述
CPUCreditUsage	執行處理CPU用於使用的CPU點數。一個點CPU數等於以 100% 使用率CPU執行一分鐘的一 vvCPUs，或同等的使用率和時間組合 (例如，一個 v 以 50% 使用率CPU執行兩分鐘，或以 25% 的使用率 vCPUs 執行兩分鐘兩分鐘)。 CPU信用指標只有五分鐘的頻率提供。如果您要指定大於 5 分鐘的期間，請使用 Sum 統計資訊代替 Average 統計資訊。 單位：學分 (v CPU-分鐘)
CPUCreditBalance	執行個體啟動或啟動後累CPU積的已獲得積分數。如果是 T2 Standard，CPUCreditBalance 也包含已產生的啟動額度。 獲得額度後，額度會在額度餘額中累積，並在支付額度時，從額度餘額中移出。額度餘額有最大值限制，它取決於執行個體大小。到達限制之後，任何獲得的新額度都會遭到捨棄。如果是 T2 Standard，啟動額度不會計入此限制。 執行個體中CPUCreditBalance 的積分可用於超出其基準CPU使用率。 當執行個體執行時，CPUCreditBalance 中的額度不會過期。當 T4G、T3A 或 T3 執行個體停止時，該CPUCredit Balance 值會持續七天。因此，所有累積的額度都會消失。當 T2 執行個體停止時，CPUCreditBalance 值不會持續保存，而且所有累積的額度都將消失。 CPU信用指標只有五分鐘的頻率提供。 單位：學分 (v CPU-分鐘)
CPUSurplusCreditBa lance	當 unlimited 執行個體的 CPUCreditBalance 值為 0 時，該執行個體已支出的剩餘額度數量。 該CPUSurplusCreditBalance 值由賺取的CPU積分支付。如果剩餘額度超過執行個體在 24 小時期間可獲得的最大額度數量，超過最大值的支出剩餘額度將必須負擔額外的費用。

指標	描述
	單位：學分 (v CPU-分鐘)
`CPUSurplusCreditsCharged`	所賺取的積分未支付的已支付剩餘CPU點數，因此會產生額外費用。 發生以下任何情況時，將收取支出剩餘額度的費用。 • 支出剩餘額度超過執行個體在 24 小時期間可獲得的最大額度數量。在小時結束時，將收取超過最大值的支出剩餘額度的費用。 • 執行個體已停止或終止。 • 執行個體從 `unlimited` 切換至 `standard`。 單位：學分 (v CPU-分鐘)

計算CPU信用用量

執行處理的點數用量是使用上表所述的執行處理 CloudWatch 測量結果來計算。CPU

Amazon EC2 將指標發送到 CloudWatch 每五分鐘一次。在任何時點參考指標的 prior 值，即表示該指標先前的值 (五分鐘前傳送的值)。

計算標準執行個體的點CPU數用量

- 如果CPU使用率低於基準，當花費的CPU信用額度少於前五分鐘間隔中獲得的積分時，則信用餘額會增加。
- 如果CPU使用率高於基準，當支出的CPU信用額度超過前五分鐘間隔中獲得的信用額度時，則信用餘額會減少。

在數學上，這可以透過下列方程式計算：

Example

```
CPUCreditBalance = prior CPUCreditBalance + [Credits earned per hour * (5/60) -
 CPUCreditUsage]
```

執行個體的大小，決定了執行個體每小時可獲得的額度，以及執行個體額度餘額可累積獲得額度的上限。如需每小時所獲得的額度、每種執行個體大小的額度餘額限額資訊，請參閱額度表。

範例

此範例使用 t3.nano 執行個體。若要計算執行個體的 CPUCreditBalance 值，請使用前述的方程式，如下所示：

- CPUCreditBalance – 要計算的最新額度餘額。
- prior CPUCreditBalance – 5 分鐘前的額度餘額。在此範例中，執行個體累積了 2 種額度。
- Credits earned per hour – 一個 t3.nano 執行個體每小時會獲得 6 點。
- 5/60— 代表 CloudWatch量度發佈之間的五分鐘間隔。將每小時獲得的額度乘上 5/60 (5 分鐘)，以算出該執行個體在過去 5 分鐘內獲得的額度。t3.nano 執行個體每 5 分鐘會獲得 0.5 點。
- CPUCreditUsage – 在過去 5 分鐘內執行個體所用掉的額度。在此範例中，執行個體過去 5 分鐘內用掉了 1 點。

使用這些值，可以計算出 CPUCreditBalance 值：

Example

```
CPUCreditBalance = 2 + [0.5 - 1] = 1.5
```

計算無限執行個體的點CPU數用量

當爆量效能執行個體的效能需要爆發超過基準時，一律會先用掉累積的額度，然後再使用剩餘額度。當它耗盡其應計的CPU信用餘額時，它可以花費剩餘的信用額度來爆CPU發，只要它需要。當CPU使用率低於基準時，一律會在執行個體累積已獲得的積分之前償還剩餘的積分。

在下列的方程式中，我們使用 Adjusted balance 一詞，來代表這個 5 分鐘間隔內所發生的動作。我們使用此值來取得CPUCreditBalance和CPUSurplusCreditBalance CloudWatch 量度的值。

Example

```
Adjusted balance = [prior CPUCreditBalance - prior CPUSurplusCreditBalance] + [Credits
 earned per hour * (5/60) - CPUCreditUsage]
```

0 的 Adjusted balance 值代表該執行個體為了突增的效能，已經用完所有獲得的額度，而且未使用剩餘額度。因此，CPUCreditBalance 和 CPUSurplusCreditBalance 都會設定為 0。

正的 Adjusted balance 值表示該執行個體已累積獲得的額度，而且已償還先前的剩餘額度 (如果有的話)。因此，Adjusted balance 值會指派給 CPUCreditBalance，而 CPUSurplusCreditBalance 會設定為 0。執行個體的大小，會決定執行個體所能累積的額度上限。

Example

```
CPUCreditBalance = min [max earned credit balance, Adjusted balance]
CPUSurplusCreditBalance = 0
```

負的 Adjusted balance 值代表執行個體已經用完所有累積獲得的額度，而且也因為突增的效能用掉了剩餘額度。因此，Adjusted balance 值會指派給 CPUSurplusCreditBalance，而 CPUCreditBalance 會設為 0。再說一次，執行個體的大小，會決定執行個體所能累積的額度上限。

Example

```
CPUSurplusCreditBalance = min [max earned credit balance, -Adjusted balance]
CPUCreditBalance = 0
```

如果用掉的剩餘額度超過執行個體可累積的最高額度，則剩餘額度餘額會設定為最高額度，如先前的方程式中所示。剩下的剩餘額度會收取費用 (以 CPUSurplusCreditsCharged 指標表示)。

Example

```
CPUSurplusCreditsCharged = max [-Adjusted balance - max earned credit balance, 0]
```

最後，當執行個體終止時，透過 CPUSurplusCreditBalance 追蹤的所有剩餘額度都會計費。如果執行個體從 unlimited 切換為 standard，也會針對所有剩下的 CPUSurplusCreditBalance 收費。

GPU執行個體的效能加速

GPU以執行個體為基礎，提NVIDIAGPUs供數千個運算核心的存取權。您可以利用CUDA或開放運算語言 (OpenCL) parallel 運算架構，使用這些執行個體來加速科學、工程和轉譯應用程式。您也可以將它們用於圖形應用程式，包括遊戲串流、3D 應用程式串流和其他圖形工作負載。

您必須先安裝適當的驅動程式，才能啟動或最佳化執行個GPU體，如下所示：

- 若要在已連接NVIDIAGPU的執行個體 (例如 P3 或 G4dn 執行個體) 上安裝NVIDIA驅動程式，請參閱。NVIDIA司機

- 若要在已連接AMDGPU的執行個體 (例如 G4ad 執行個體) 上安裝AMD驅動程式，請參閱。AMD驅動

目錄

- 在 Amazon EC2 GPU 執行個體上啟用NVIDIAGRID虛擬應用程式
- 優化 Amazon EC2 執行個體上的GPU設定
- 在 G4ad 執行個體上設定雙 4K 顯示器
- 開始使用 P5 執行個體

在 Amazon EC2 GPU 執行個體上啟用NVIDIAGRID虛擬應用程式

若要在GPU具有 NVIDIA GPUs (預設為啟用「GRID虛擬工作站」) 的執行個體上啟動「NVIDIAGRID虛擬應用程式」，您必須定義驅動程式的產品類型。您使用的程序取決於執行個體的作業系統。

Linux 執行個體

若要在 Linux 執行個體上啟動GRID虛擬應用程式

1. 從提供的範本檔案建立 `/etc/nvidia/gridd.conf` 檔案。

```
[ec2-user ~]$ sudo cp /etc/nvidia/gridd.conf.template /etc/nvidia/gridd.conf
```

2. 在您偏好的文字編輯器中開啟 `/etc/nvidia/gridd.conf` 檔案。
3. 找到 `FeatureType` 行，並將其設定為等於 `0`。然後新增一行 `IgnoreSP=TRUE`。

```
FeatureType=0 IgnoreSP=TRUE
```

4. 儲存檔案並結束。
5. 重新開機執行個體，套用新的組態。

```
[ec2-user ~]$ sudo reboot
```

Windows 執行個體

在 Windows 執行個體上啟用GRID虛擬應用程式

1. 開啟 regedit.exe 來開啟登錄編輯程式。

2. 導覽至 `HKEY_LOCAL_MACHINE\SOFTWARE\NVIDIA Corporation\Global\GridLicensing`。
3. 開啟右窗格上的內容 (按一下滑鼠右鍵) 功能表，然後選擇「新增」、DWORD。
4. 對於「名稱」，輸入FeatureType並鍵入`Enter`。
5. 開啟上下文 (按一下右鍵) 功能表，FeatureType然後選擇 「修改」。
6. 針對數值資料，`0`針對NVIDIAGRID虛擬應用程式輸入，然後選擇確定。
7. 開啟右窗格上的內容 (按一下滑鼠右鍵) 功能表，然後選擇「新增」、DWORD。
8. 針對 Name (名稱)，輸入 IgnoreSP 然後輸入 `Enter`。
9. 開啟 IgnoreSP 上的內容選單 (按一下滑鼠右鍵)，然後選擇 Modify (修改)。
10. 針對 Value data (值資料)，輸入 1 並選擇 OK (確定)。
11. 關閉登錄編輯器。

優化 Amazon EC2 執行個體上的GPU設定

您可以執行多項GPU設定最佳化，以在執行個NVIDIAGPU體上達到最佳效能。在這些執行個體類型中，NVIDIA驅動程式會使用自動加速功能，這會改變GPU時脈速度。透過停用自動加速並將GPU時脈速度設定為最大頻率，您可以持續發揮GPU執行個體的最高效能。

在 Linux 上最佳化GPU設定

1. 將設定設定GPU設定為永久性。此命令可能需要幾分鐘的執行時間。

   ```
   [ec2-user ~]$ sudo nvidia-persistenced
   ```

2. [僅限 G3 和 P2 執行個體] 停用執行個體GPUs上所有執行個體的自動加速功能。

   ```
   [ec2-user ~]$ sudo nvidia-smi --auto-boost-default=0
   ```

3. 將所有GPU時脈速度設定為最大頻率。使用下列命令中指定的記憶體和圖形時脈速度。

   某些版本的NVIDIA驅動程序不支持設置應用程序的時鐘速度，並顯示錯誤`"Setting applications clocks is not supported for GPU..."`，您可以忽略。

   - G3 執行個體：

     ```
     [ec2-user ~]$ sudo nvidia-smi -ac 2505,1177
     ```

   - G4dn 執行個體：

```
[ec2-user ~]$ sudo nvidia-smi -ac 5001,1590
```

- G5 執行個體：

```
[ec2-user ~]$ sudo nvidia-smi -ac 6250,1710
```

- G6 和 GR6 執行個體：

```
[ec2-user ~]$ sudo nvidia-smi -ac 6251,2040
```

- P2 執行個體：

```
[ec2-user ~]$ sudo nvidia-smi -ac 2505,875
```

- P3 和 P3dn 執行個體：

```
[ec2-user ~]$ sudo nvidia-smi -ac 877,1530
```

- P4d 執行個體：

```
[ec2-user ~]$ sudo nvidia-smi -ac 1215,1410
```

- P4de 執行個體：

```
[ec2-user ~]$ sudo nvidia-smi -ac 1593,1410
```

- P5 執行個體：

```
[ec2-user ~]$ sudo nvidia-smi -ac 2619,1980
```

在視窗上最佳化GPU設定

1. 開啟視 PowerShell 窗並瀏覽至NVIDIA安裝資料夾。

```
cd "C:\Windows\System32\DriverStore\FileRepository\nvgridsw_aws.inf_*\"
```

2. [僅限 G3 和 P2 執行個體] 停用執行個體GPUs上所有執行個體的自動加速功能。

```
.\nvidia-smi --auto-boost-default=0
```

3. 將所有GPU時脈速度設定為最大頻率。使用下列命令中指定的記憶體和圖形時脈速度。

   某些版本的NVIDIA驅動程序不支持設置應用程序的時鐘速度，並顯示錯誤"Setting applications clocks is not supported for GPU..."，您可以忽略。

   - G3 執行個體：

     ```
     .\nvidia-smi -ac "2505,1177"
     ```

   - G4dn 執行個體：

     ```
     .\nvidia-smi -ac "5001,1590"
     ```

   - G5 執行個體：

     ```
     .\nvidia-smi -ac "6250,1710"
     ```

   - G6 和 GR6 執行個體：

     ```
     .\nvidia-smi -ac "6251,2040"
     ```

   - P2 執行個體：

     ```
     .\nvidia-smi -ac "2505,875"
     ```

   - P3 和 P3dn 執行個體：

     ```
     .\nvidia-smi -ac "877,1530"
     ```

   - P4d 執行個體：

     ```
     [ec2-user ~]$ sudo nvidia-smi -ac 1215,1410
     ```

   - P4de 執行個體：

     ```
     [ec2-user ~]$ sudo nvidia-smi -ac 1593,1410
     ```

   - P5 執行個體：

     ```
     [ec2-user ~]$ sudo nvidia-smi -ac 2619,1980
     ```

在 G4ad 執行個體上設定雙 4K 顯示器

啟動 G4ad 執行個體之後，您可以設定雙 4K 顯示器。

安裝AMD驅動程式並設定雙螢幕

1. Connect 至您的 Linux 執行個體，以取得GPU您要以雙 4K (2x4k) 為目標的PCI匯流排位址：

   ```
   lspci -vv | grep -i amd
   ```

   您將會得到類似下列的輸出：

   ```
   00:1e.0 Display controller: Advanced Micro Devices, Inc. [*AMD*/ATI] Device 7362 (rev
    c3)
   Subsystem: Advanced Micro Devices, Inc. [AMD/ATI] Device 0a34
   ```

2. 請注意，PCI匯流排位址是 00:1 e.0 在上述輸出中。建立名為 `/etc/modprobe.d/amdgpu.conf` 的檔案並新增：

   ```
   options amdgpu virtual_display=0000:00:1e.0,2
   ```

3. 若要在 Linux 上安裝AMD驅動程式，請參閱AMDEC2執行個體的驅動程式。如果您已經安裝了AMDGPU驅動程式，則需要透過 dkms 重建 amdgpu 核心模組。
4. 使用下面的 xorg.conf 檔案定義雙 (2x4k) 畫面拓撲，並將檔案儲存在 `/etc/X11/xorg.conf`：

   ```
   ~$ cat /etc/X11/xorg.conf
   Section "ServerLayout"
       Identifier     "Layout0"
       Screen          0 "Screen0"
       Screen        1 "Screen1"
       InputDevice     "Keyboard0" "CoreKeyboard"
       InputDevice     "Mouse0" "CorePointer"
       Option          "Xinerama" "1"
   EndSection
   Section "Files"
       ModulePath "/opt/amdgpu/lib64/xorg/modules/drivers"
       ModulePath "/opt/amdgpu/lib/xorg/modules"
       ModulePath "/opt/amdgpu-pro/lib/xorg/modules/extensions"
       ModulePath "/opt/amdgpu-pro/lib64/xorg/modules/extensions"
       ModulePath "/usr/lib64/xorg/modules"
       ModulePath "/usr/lib/xorg/modules"
   EndSection
   ```

```
Section "InputDevice"
    # generated from default
    Identifier     "Mouse0"
    Driver         "mouse"
    Option         "Protocol" "auto"
    Option         "Device" "/dev/psaux"
    Option         "Emulate3Buttons" "no"
    Option         "ZAxisMapping" "4 5"
EndSection
Section "InputDevice"
    # generated from default
    Identifier     "Keyboard0"
    Driver         "kbd"
EndSection

Section "Monitor"
    Identifier     "Virtual"
    VendorName     "Unknown"
    ModelName      "Unknown"
    Option         "Primary" "true"
EndSection

Section "Monitor"
    Identifier     "Virtual-1"
    VendorName     "Unknown"
    ModelName      "Unknown"
    Option         "RightOf" "Virtual"
EndSection

Section "Device"
    Identifier     "Device0"
    Driver         "amdgpu"
    VendorName     "AMD"
    BoardName      "Radeon MxGPU V520"
    BusID          "PCI:0:30:0"
EndSection

Section "Device"
    Identifier     "Device1"
    Driver         "amdgpu"
    VendorName     "AMD"
    BoardName      "Radeon MxGPU V520"
    BusID          "PCI:0:30:0"
EndSection
```

```
Section "Extensions"
    Option         "DPMS" "Disable"
EndSection

Section "Screen"
    Identifier     "Screen0"
    Device         "Device0"
    Monitor        "Virtual"
    DefaultDepth   24
    Option         "AllowEmptyInitialConfiguration" "True"
    SubSection "Display"
        Virtual    3840 2160
        Depth      32
    EndSubSection
EndSection

Section "Screen"
    Identifier     "Screen1"
    Device         "Device1"
    Monitor        "Virtual"
    DefaultDepth   24
    Option         "AllowEmptyInitialConfiguration" "True"
    SubSection "Display"
        Virtual    3840 2160
        Depth      32
    EndSubSection
EndSection
```

5. DCV依照設定互動式桌面中的指示進行設定。
6. DCV設置完成後，重新啟動。
7. 確認驅動程式可正常運作：

```
dmesg | grep amdgpu
```

回應應如下所示：

```
Initialized amdgpu
```

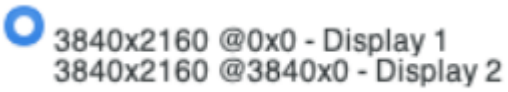
8. 您應該會在 `DISPLAY=:0 xrandr -q` 輸出中看到您已連接 2 個虛擬顯示器：

```
~$ DISPLAY=:0 xrandr -q
```

```
Screen 0: minimum 320 x 200, current 3840 x 1080, maximum 16384 x 16384
Virtual connected primary 1920x1080+0+0 (normal left inverted right x axis y axis)
 0mm x 0mm
 4096x3112  60.00
 3656x2664  59.99
 4096x2160  60.00
 3840x2160  60.00
 1920x1200  59.95
 1920x1080  60.00
 1600x1200  59.95
 1680x1050  60.00
 1400x1050  60.00
 1280x1024  59.95
 1440x900 59.99
 1280x960 59.99
 1280x854 59.95
 1280x800 59.96
 1280x720 59.97
 1152x768 59.95
 1024x768 60.00 59.95
 800x600  60.32 59.96 56.25
 848x480  60.00 59.94
 720x480  59.94
 640x480  59.94 59.94
Virtual-1 connected 1920x1080+1920+0 (normal left inverted right x axis y axis) 0mm x
 0mm
 4096x3112  60.00
 3656x2664  59.99
 4096x2160  60.00
 3840x2160  60.00
 1920x1200  59.95
 1920x1080  60.00
 1600x1200  59.95
 1680x1050  60.00
 1400x1050  60.00
 1280x1024  59.95
 1440x900 59.99
 1280x960 59.99
 1280x854 59.95
 1280x800 59.96
 1280x720 59.97
 1152x768 59.95
 1024x768 60.00 59.95
 800x600  60.32 59.96 56.25
```

```
 848x480  60.00 59.94
 720x480  59.94
640x480  59.94 59.94
```

9. 當您連線到時DCV，請將解析度變更為 2x4K，確認雙顯示器支援已由註冊。DCV



設定適用於 Linux 的互動式桌面

在您確認 Linux 執行個體已安裝AMDGPU驅動程式且 amdgpu 正在使用中之後，您就可以安裝互動式桌面管理員。我們建議使用MATE桌面環境以獲得最佳的相容性和效能。

先決條件

開啟文字編輯器，並將以下內容儲存為名為 `xorg.conf` 的檔案。你的執行個體需要這個檔案。

```
Section "ServerLayout"
Identifier     "Layout0"
Screen          0 "Screen0"
InputDevice     "Keyboard0" "CoreKeyboard"
InputDevice     "Mouse0" "CorePointer"
EndSection
Section "Files"
ModulePath "/opt/amdgpu/lib64/xorg/modules/drivers"
ModulePath "/opt/amdgpu/lib/xorg/modules"
ModulePath "/opt/amdgpu-pro/lib/xorg/modules/extensions"
ModulePath "/opt/amdgpu-pro/lib64/xorg/modules/extensions"
ModulePath "/usr/lib64/xorg/modules"
ModulePath "/usr/lib/xorg/modules"
EndSection
Section "InputDevice"
# generated from default
Identifier     "Mouse0"
Driver         "mouse"
Option         "Protocol" "auto"
Option         "Device" "/dev/psaux"
Option         "Emulate3Buttons" "no"
Option         "ZAxisMapping" "4 5"
EndSection
Section "InputDevice"
# generated from default
```

```
Identifier     "Keyboard0"
Driver         "kbd"
EndSection
Section "Monitor"
Identifier     "Monitor0"
VendorName     "Unknown"
ModelName      "Unknown"
EndSection
Section "Device"
Identifier     "Device0"
Driver         "amdgpu"
VendorName     "AMD"
BoardName      "Radeon MxGPU V520"
BusID          "PCI:0:30:0"
EndSection
Section "Extensions"
Option         "DPMS" "Disable"
EndSection
Section "Screen"
Identifier     "Screen0"
Device         "Device0"
Monitor        "Monitor0"
DefaultDepth   24
Option         "AllowEmptyInitialConfiguration" "True"
SubSection "Display"
    Virtual    3840 2160
    Depth      32
EndSubSection
EndSection
```

若要在 Amazon Linux 2 上設定互動式桌面

1. 安裝存EPEL放庫。

```
$ C:\> sudo amazon-linux-extras install epel -y
```

2. 安裝MATE桌面。

```
$ C:\> sudo amazon-linux-extras install mate-desktop1.x -y
$ C:\> sudo yum groupinstall "MATE Desktop" -y
$ C:\> sudo systemctl disable firewalld
```

3. 將 `xorg.conf` 檔案複製至 `/etc/X11/xorg.conf`。

4. 重新啟動執行個體。

```
$ C:\> sudo reboot
```

5. (選擇性) 安裝要NICEDCV做為高效能顯示通訊協定的NICEDCV伺服器，然後使用偏好的用戶端連線至NICEDCV工作階段。

在 Ubuntu 上設定互動式桌面

1. 安裝MATE桌面。

```
$ sudo apt install xorg-dev ubuntu-mate-desktop -y
$ C:\> sudo apt purge ifupdown -y
```

2. 將 `xorg.conf` 檔案複製至 `/etc/X11/xorg.conf`。
3. 重新啟動執行個體。

```
$ sudo reboot
```

4. 為 Ubuntu 的適當版本安裝AMF編碼器。

```
$ sudo apt install ./amdgpu-pro-20.20-*/amf-amdgpu-pro_20.20-*_amd64.deb
```

5. (選擇性) 安裝要NICEDCV做為高效能顯示通訊協定的NICEDCV伺服器，然後使用偏好的用戶端連線至NICEDCV工作階段。
6. DCV安裝後給用DCV戶視頻權限：

```
$ sudo usermod -aG video dcv
```

在 CentOS 上設定互動式桌面

1. 安裝存EPEL放庫。

```
$ sudo yum update -y
$ C:\> sudo yum install epel-release -y
```

2. 安裝MATE桌面。

```
$ sudo yum groupinstall "MATE Desktop" -y
```

```
$ C:\> sudo systemctl disable firewalld
```

3. 將 `xorg.conf` 檔案複製至 `/etc/X11/xorg.conf`。
4. 重新啟動執行個體。

```
$ sudo reboot
```

5. (選擇性) 安裝要NICEDCV做為高效能顯示通訊協定的NICEDCV伺服器，然後使用偏好的用戶端連線至NICEDCV工作階段。

開始使用 P5 執行個體

P5 執行個體提供 8 個上NVIDIA半球，GPUs以及 640 GB 的高頻GPU寬記憶體。它們具有第三代AMDEPYC處理器，並提供 2 TB 的系統內存，30 TB 的本地NVMe實例存儲，3,200 Gbps 聚合網絡帶寬和GPUDirectRDMA支持。P5 執行個體也支援 Amazon EC2 UltraCluster 技術，可降低延遲並改善使用EFA的網路效能。

如下資料表提供 `p5.48xlarge` 規格的摘要。

vCPUs	系統記憶體	GPUs	GPU 記憶	網路頻寬	GPUDirect RDMA	GPU 點對點	執行個體儲存體
192	2 TiB	8 NVIDIA H100 GPUs	640 GB HBM3	三百百吉比斯 EFAv2	支援	每秒 900 GB NVSw	8 x 800 GB 磁碟 NVMeSSD 區

軟體組態

開始使用 P5 執行個體的最簡單方法是使用已針對所有必要軟體預先設定的 AWS Deep Learning AMI 啟動執行個體。如需與 P5 執行個體搭配使用的最新 AWS Deep Learning AMI 資訊，請參閱AWS 深度學習基礎 GPU AMI (Ubuntu 20.04)。

如果您需要建立自AMI訂以搭配 P5 執行個體使用，建議您安裝下列最低軟體版本：

- NVIDIA驅動程式

- CUDA12.1 或更高版本
- NVIDIAGDRCopy2.3 或更高版本
- EFA安裝程式 1.24.1 或更新版本
- NCCL2.18.3 或更高版本
- aws-ofi-nccl 插件 1.7.2-AWS 或更高版本

我們也建議您將執行個體設定為不使用深層的 C-states。如需詳細資訊，請參閱 Amazon Linux 2 使用者指南中的限制更深 C 狀態的高效能和低延遲。最新的 AWS 深度學習基礎GPUAMI已預先設定為不使用更深層的 C 狀態。

Ubuntu 20.04 具體建議

下列有關 Ubuntu 20.04 的建議有助於避免開機時出現無法預期的介面命名方式：

- 確保您正在執行 `systemd 245.4-4ubuntu3.19`，或稍後使用以下命令：

```
systemd --version
```

- 確定您已設定GRUB：
  - 在文字編輯器中開啟 `/etc/default/grub` 組態檔案。
  - 編輯 `GRUB_CMDLINE_LINUX_DEFAULT` 項目，以便包含 `net.naming-scheme=v247`。
  - 執行 `sudo update-grub` 以重新啟動執行個體。

網路與EFA組態

P5 執行個體使用多重介面可提供 3200 Gbps 的網路頻寬。EFAP5 執行個體支援 32 個網路卡。我們建議您為每張EFA網路卡定義一個網路介面卡。若要在啟動時設定這些介面，我們建議您進行下列設定：

- 對於網路介面 `0`，指定裝置索引 `0`
- 對於網路介面 `1` 至 `31`，指定裝置索引 `1`

如需如何設定 P5 執行個體的詳細資訊，EFA請參閱在加速運算執行個體上最大化網路頻寬 EFA。

Amazon EC2 Mac 實例

EC2Mac 執行個體非常適合開發、建置、測試和簽署應用程式，適用於 Apple 平台 iPhoneiPad，例如、Mac、視覺專業版、Apple Watch、蘋果電視和 Safari。您可以使用SSH或 Apple 遠端桌面 (ARD) 連線至 Mac 執行個體。

> **Note**
>
> 以專用執行個體為計費單位。在該主機上執行的執行個體並無額外費用。

Amazon EC2 Mac 實例本地支持 macOS 操作系統。

- EC2x86 Mac 實例（`mac1.metal`）是建立在 2018 年 Mac 迷你硬件由 3.2 GHz 英特爾第八代（咖啡湖）酷睿 i7 處理器供電。
- EC2M1 Mac 執行個體 (`mac2.metal`) 建置於 2020 年 Mac 迷你硬體，搭載蘋果矽晶 M1 處理器。
- EC2M1 超級 Mac 執行個體 (`mac2-m1ultra.metal`) 建置於 2022 年 Mac 工作室硬體，搭載蘋果矽晶 M1 超級處理器。
- EC2M2 Mac 實例（`mac2-m2.metal`）建立在 2023 Mac 迷你硬件上，由蘋果矽 M2 處理器提供支持。
- EC2M2 Pro Mac 實例（`mac2-m2pro.metal`）建立在 2023 Mac 迷你硬件上，由蘋果矽晶 M2 Pro 處理器提供支持。

目錄

- 考量事項
- 執行個體整備
- EC2macOS AMIs
- EC2初 macOS 化
- Amazon EC2 系統監視器的 macOS
- 相關資源
- 使用或啟動 Mac 執行 AWS Management Console 個體 AWS CLI
- 使用或 Connect 至 Mac 執SSH行個體 GUI
- 更新 Mac 實例上的操作系統和軟件
- 增加 Mac 執行個體上的EBS磁碟區大小

- 停止或終止您的 Amazon EC2 Mac 實例
- 查找支持的 macOS 版本 Amazon EC2 Mac 專用主機
- 訂閱 macOS AMI 通知
- AMIIDs使用 AWS Systems Manager 參數存儲檢索 macOS API
- Amazon EC2 macOS AMIs 本注意事項

考量事項

Mac 執行個體適用下列考量：

- Mac 執行個體只能做為專用執行個體上的裸機執行個體使用，最小配置期間為 24 小時，這段期間過後您才可以發佈專用執行個體。每個 專用執行個體 可以啟動一個 Mac 執行個體。您可以與組織內的 AWS 帳戶或組織單位或整個 AWS 組 AWS 織共用專用主機。
- Mac 實例有不同的可用 AWS 區域。如需各個 Mac 執行個體可用性的清單 AWS 區域，請參閱各區域的 Amazon EC2 執行個體類型。
- Mac 執行個體只能做為 隨需執行個體 使用，不可做為 競價型執行個體 或 預留執行個體 使用。您可以購買 Savings Plan，節省 Mac 執行個體的費用。
- Mac 執行個體可以執行下列其中一個作業系統：
  - macOS Mojave (版本 10.14) (僅限 x86 Mac 執行個體)
  - macOS Catalina (版本 10.15) (僅限 x86 Mac 執行個體)
  - macOS Big Sur (版本 11) (x86 和 M1 Mac 執行個體)
  - macOS Monterey (版本 12) (x86 和 M1 Mac 執行個體)
  - macOS Ventura (13 版) (所有 Mac 執行個體、M2 和 M2 Pro Mac 執行個體支援 macOS Ventura 13.2 版或更新版本)
  - macOS Sonoma (版本 14) (所有 Mac 執行個體)
- EBS支持熱插拔。
- AWS 不管理或支持 Apple 硬件SSD上的內部。我們強烈建議您改用 Amazon EBS 卷。 EBS磁碟區在 Mac 執行個體上提供與其他執行個體相同的彈性、可用性和耐久EC2性優勢。
- 我們建議您將一般用途 SSD (gp2和gp3) 和佈建 IOPS SSD (io1和io2) 搭配 Mac 執行個體使用，以獲得最佳EBS效能。
- Mac 執行個體支援 Amazon EC2 Auto Scaling。
- 在 x86 Mac 執行個體上，自動軟體更新會停用。我們建議您在將執行個體投入實際執行環境前套用更新，並在執行個體上測試更新。如需詳細資訊，請參閱更新 Mac 實例上的操作系統和軟件。

- 當您停止或終止 Mac 執行個體時，系統會在 專用主機 上執行清理工作流程。如需詳細資訊，請參閱停止或終止您的 Amazon EC2 Mac 實例。

> **Warning**
>
> 請勿使用 FileVault。啟用 FileVault 將導致主機因為分割區被鎖定而無法開機。如果需要資料加密，請使用 Amazon EBS 加密以避免開機問題和效能影響。使用 Amazon EBS 加密，加密操作會在託管執行個體的伺服器上進行，以確保執行個體 data-at-rest 及其附加EBS儲存 data-in-transit 之間以及之間的安全性。有關更多信息，請參閱 Amazon EBS用戶指南中的 Amazon EBS 加密

執行個體整備

啟動 Mac 執行個體之後，您將需要等待執行個體準備就緒後，才能夠與其連接。對於出 AWS 售 AMI x86 Mac 執行個體或蘋果矽晶 Mac 執行個體的供應商，啟動時間可以從大約 6 分鐘到 20 分鐘不等。根據選擇的 Amazon EBS 磁碟區大小、在使用者資料中包含額外指令碼，或自訂 macOS 上其他載入的軟體AMI，啟動時間可能會增加。

您可以使用小型 shell 指令碼 (如下面的指令碼) 來輪詢執行個體 describe-instance-status API何時準備好連線到該執行個體。將以下命令中的範例執行個體 ID 取代為您自己的 ID：

```
for i in $(seq 1 200); do aws ec2 describe-instance-status --instance-
ids=i-0123456789example \
    --query='InstanceStatuses[0].InstanceStatus.Status'; sleep 5; done;
```

EC2macOS AMIs

Amazon EC2 macOS 旨在為在 Amazon EC2 Mac 執行個體上執行的開發人員工作負載提供穩定、安全且高效能的環境。EC2macOS AMIs 包含可輕鬆整合的套件 AWS，例如啟動設定工具和常用的 AWS 資料庫和工具。

如需 EC2 macOS 的詳細資訊AMIs，請參閱Amazon EC2 macOS AMIs 本注意事項。

AWS 定期提供更新AMIs的 EC2 macOS，其中包括對所擁有套件的更新，以 AWS 及最新的完整測試 macOS 版本。此外， AWS 提供最新AMIs的次要版本更新或主要版本更新，只要他們可以完全測試和審查更新。如果您不需要保留 Mac 執行個體的資料或自訂項目，您可以使用目前執行個體啟動新的執行個體，然後終止上一個執行個體AMI，以取得最新的更新。否則您可以選擇哪些更新要套用到您的 Mac 執行個體。

如需如何訂閱 macOS AMI 通知的詳細資訊，請參閱訂閱 macOS AMI 通知。

EC2初 macOS 化

EC2macOS 初始化用於在啟動時初始化 EC2 Mac 執行個體。它會使用優先順序群組，來同時執行任務的邏輯群組。

launchd plist 檔案是 `/Library/LaunchDaemons/com.amazon.ec2.macos-init.plist`。適用於 EC2 macOS 初始化的檔案位於中`/usr/local/aws/ec2-macos-init`。

如需詳細資訊，請參閱 https://github.com/aws/ec2- 澳門初始化。

Amazon EC2 系統監視器的 macOS

適用於 macOS 的 Amazon EC2 系統監視器向 Amazon CloudWatch 提供CPU利用指標 它會在 1 分鐘內 CloudWatch 透過自訂序列裝置傳送這些指標。您可以啟用或停用此代理程式，如下所示。依預設會啟用此功能。

```
sudo setup-ec2monitoring [enable | disable]
```

> **Note**
>
> 適用於 macOS 的 Amazon EC2 系統監視器目前不支持蘋果矽 Mac 實例。

相關資源

如需定價的資訊，請參閱 定價。

如需有關 Mac 執行個體的詳細資訊，請參閱 Amazon EC2 Mac 執行個體。

如需 Mac 執行個體的硬體規格和網路效能的詳細資訊，請參閱一般用途執行個體。

使用或啟動 Mac 執行 AWS Management Console 個體 AWS CLI

EC2Mac 執行個體需要專用主機。您首先需要將主機配置到您的帳戶，然後在主機上啟動執行個體。

您可以使用 AWS Management Console 或啟動 Mac 執行個體 AWS CLI。

使用主控台啟動 Mac 執行個體

在 專用主機 上啟動 Mac 實體啟動

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 配置專用執行個體，如下所示：
   a. 在導覽窗格中，選擇 專用主機。
   b. 選擇 Allocate 專用執行個體 (配置 專用執行個體)，然後執行下列動作：
      i. 對於「例如」系列，請選擇 Mac1、Mac 2、Mac2 平方米、澳門 2 平方米或澳門 2 平方米超高。如果 Instance family (執行個體系列) 未在清單中出現，則表示目前所選的區域不支援。
      ii. 針對例證類型，請根據選擇的例證系列選擇 Mac1. 金屬、Mac2. 金屬、Mac2-平方米金屬、Mac2-m2pro.metal 或 Mac2- M 1 超金屬。
      iii. 在 Availability Zone (可用區域) 中，選擇 專用執行個體 的可用區域。
      iv. 在 Quantity (數量) 中，保留 1。
      v. 選擇 Allocate (配置)。
3. 在主機上啟動執行個體，如下所示：
   a. 選取您建立的 專用執行個體，然後執行下列動作：
      i. 選擇 Actions (動作)、Launch instance(s) onto host (在主機上啟動執行個體)。
      ii. 在「應用程式和作業系統映像 (Amazon 機器映像)」下，選取 macOS AMI。
      iii. 在例證類型之下，選擇適當的例證類型 (Mac1. 金屬、Mac2. 金屬、Mac2-m2. 金屬、Mac 2-m2pro.metal 或 Mac2-M 1 超金屬)。
      iv. 在 Advanced details (進階詳細資訊) 下，確認已根據您建立的專用執行個體預先設定 Tenancy (租用)、Tenancy host by (租用執行個體者) 和 Tenancy host ID (租用執行個體 ID)。視需要更新 Tenancy affinity (租用親和性)。
      v. 完成精靈，視需要指定EBS磁碟區、安全群組和金鑰配對。
      vi. 在 Summary (摘要) 面板中，選擇 Launch instance (啟動執行個體)。
   b. 會有確認頁面讓您知道您的執行個體正在啟動。選擇 View all instances (檢視所有執行個體)，以關閉確認頁面並返回主控台。執行個體的初始狀態為 `pending`。當其狀態變更為 `running` 並通過狀態檢查時，則表示該執行狀態就緒。

啟動 Mac 執行個體，使用 AWS CLI

配置專用執行個體

使用下列配置主機命令為您的 Mac 執行個體instance-type配置專用主機mac1.metalmac2.metal，mac2-m2.metal並以適用於您環境的mac2-m1ultra.metal、、、或region和取代專availability-zone用主機。mac2-m2pro.metal

```
aws ec2 allocate-hosts --region us-east-1 --instance-type mac1.metal --availability-
zone us-east-1b --auto-placement "on" --quantity 1
```

在主機上啟動執行個體

使用下列執行個體指令來啟動 Mac 執行個體，再次instance-type以mac1.metal、、或mac2.metalmac2-m2.metalmac2-m2pro.metalmac2-m1ultra.metal、以及和之前使用的執行availability-zone個體取代。region

```
aws ec2 run-instances --region us-east-1 --instance-type mac1.metal --placement
 Tenancy=host --image-id ami_id --key-name my-key-pair
```

執行個體的初始狀態為 pending。當其狀態變更為 running 並通過狀態檢查時，則表示該執行狀態就緒。使用下列describe-instance-status指令顯示執行個體的狀態資訊。

```
aws ec2 describe-instance-status --instance-ids i-017f8354e2dc69c4f
```

以下是執行中並通過狀態檢查之執行個體的範例輸出。

```
{
    "InstanceStatuses": [
        {
            "AvailabilityZone": "us-east-1b",
            "InstanceId": "i-017f8354e2dc69c4f",
            "InstanceState": {
                "Code": 16,
                "Name": "running"
            },
            "InstanceStatus": {
                "Details": [
                    {
                        "Name": "reachability",
```

```
                        "Status": "passed"
                    }
                ],
                "Status": "ok"
            },
            "SystemStatus": {
                "Details": [
                    {
                        "Name": "reachability",
                        "Status": "passed"
                    }
                ],
                "Status": "ok"
            }
        }
    ]
}
```

使用或 Connect 至 Mac 執SSH行個體 GUI

您可以使用SSH或圖形化使用者介面 (GUI) 連線至 Mac 執行個體。

使用 SSH 連線至您的執行個體

> **Important**
>
> 多個使用者可以同時存取作業系統。由於端口 5900 上內置了屏幕共享服務，通常GUI會有 1：1 用戶：會話。在 macOS SSH 中使用支援多個工作階段，直到 sshd_config 檔案中的「工作階段上限」限制為止。

Amazon EC2 Mac 執行個體預設不允許遠端SSH根目錄。密碼驗證也將停用，以避免密碼遭到暴力破解攻擊。ec2 使用者帳戶設定為使用遠端登入。SSHec2-user 用戶也擁有 sudo 權限。連線到您的執行個體後，您可以新增其他使用者。

若要支援使用連線至執行個體SSH，請使用 key pair 組和允許SSH存取的安全性群組啟動執行個體，並確保執行個體具有網際網路連線能力。連線到執行個體時，您可以提供金鑰對的 `.pem` 檔案。

請遵循下列程序，使用用SSH戶端連線至 Mac 執行個體。如果您在嘗試連線至執行個體時遇到錯誤，請參閱 解決連線到 Amazon EC2 Linux 執行個體的問題。

若要使用連線至執行個體 SSH

1. ssh在命令列中輸入，確認您的本機電腦已安裝SSH用戶端。如果您的電腦無法辨識該命令，請搜尋作業系統的SSH用戶端並進行安裝。
2. 取得執行個體的公開DNS名稱。使用 Amazon 主EC2控台，您可以在「詳細資料」和「聯網」索引標籤上找到公開DNS名稱。使用 AWS CLI，您可以使用描述實例命令找到公用DNS名稱。
3. 找到您在啟動執行個體時所指定金鑰對的 `.pem` 檔案。
4. 使用下列ssh命令 Connect 至執行個體，並指定執行個體和`.pem`檔案的公用DNS名稱。

```
ssh -i /path/key-pair-name.pem ec2-user@instance-public-dns-name
```

Connect 至執行個體的圖形化使用者介面 (GUI)

請按照下列步驟連線至執行個體的GUI使用VNC、Apple 遠端桌面 (ARD) 或 Apple 螢幕共享應用程式 (隨附於 macOS 中)。

> **Note**
>
> macOS 10.14 及以上版本只允許控制是否透過 System Preferences (系統偏好設定) 啟用 Screen Sharing (螢幕共用)。

使用用戶端或VNC用ARD戶端連線至執行個體

1. 確認您的本機電腦已安ARD裝支援的VNC用戶ARD端或用戶端。在 macOS 上，您可以利用內建的「螢幕共用」應用程式。否則，請搜索您ARD的操作系統並進行安裝。
2. 從您的本機電腦，使用連線至您的執行個體SSH。
3. 如下所示使用 passwd 命令設定 ec2-user 帳戶的密碼。

```
[ec2-user ~]$ sudo passwd ec2-user
```

4. 使用下列命令，安裝並啟動 macOS Screen Sharing。

```
[ec2-user ~]$ sudo launchctl enable system/com.apple.screensharing
sudo launchctl load -w /System/Library/LaunchDaemons/com.apple.screensharing.plist
```

5. 輸入 exit 並按 Enter，斷開與執行個體的連線。

6. 從您的電腦，使用以下 ssh 命令連線到執行個體。除了上一節所示的選項之外，還可以使用啟用連接埠轉送選-L項，並將本機連接埠 5900 上的所有流量轉送至執行個體上的ARD伺服器。

```
ssh -L 5900:localhost:5900 -i /path/key-pair-name.pem ec2-user@instance-public-dns-name
```

7. 從您的本機電腦，使用支援連線的用VNC戶ARD端或ARD用戶端`localhost:5900`。例如，使用 macOS 上的「螢幕共用」應用程式，如下所示：

   a. 開啟 Finder，並選取開始。

   b. 選取連線至伺服器。

   c. 在伺服器地址欄位中，輸入 `vnc://localhost:5900`。

   d. 請根據提示登入，使用您**`ec2-user`**為 ec2-user 帳戶建立的使用者名稱和密碼。

修改 Mac 執行個體上的 macOS 螢幕解析度

使用ARD或支援的用VNC戶端連線至 EC2 Mac 執行個體後ARD，您可以使用任何公開可用的 macOS 工具或公用程式 (例如顯示器放置工具) 修改 macOS 環境的螢幕解析度。

若要使用 displayplacer 修改螢幕解析度

1. 安裝 displayplacer。

```
[ec2-user ~]$ brew tap jakehilborn/jakehilborn && brew install displayplacer
```

2. 顯示目前的螢幕資訊和可能的螢幕解析度。

```
[ec2-user ~]$ displayplacer list
```

3. 套用所需的螢幕解析度。

```
[ec2-user ~]$ displayplacer "id:<screenID> res:<width>x<height> origin:(0,0)
 degree:0"
```

   例如：

```
RES="2560x1600"
displayplacer "id:69784AF1-CD7D-B79B-E5D4-60D937407F68 res:${RES} scaling:off
 origin:(0,0) degree:0"
```

更新 Mac 實例上的操作系統和軟件

> **Warning**
>
> 只有蘋果矽晶片 Mac 執行個體才能安裝測試版或預覽版 macOS 版本。Amazon EC2 不符合測試版或預覽 macOS 版本的資格，且無法確保執行個體在更新至生產前 macOS 版本後仍能正常運作。
>
> 嘗試在 Amazon EC2 x86 Mac 執行個體上安裝測試版或預覽 macOS 版本會導致 Amazon EC2 Mac 專用主機在您停止或終止執行個體時降級，而且無法在該主機上啟動或啟動新執行個體。

在 x86 Mac 執行個體和 Apple 晶片 Mac 執行個體上更新軟體的步驟。

- 在 x86 Mac 執行個體上更新軟體
- 在 Apple 晶片 Mac 執行個體上更新軟體

在 x86 Mac 執行個體上更新軟體

在 x86 Mac 執行個體上，您可以使用 `softwareupdate` 指令從 Apple 安裝作業系統更新。

若要在 x86 Mac 執行個體上從 Apple 安裝作業系統更新

1. 使用以下命令列出具有可用更新的套件。

```
[ec2-user ~]$ softwareupdate --list
```

2. 安裝所有更新或僅特定更新。若要安裝特定更新，請使用下列命令。

```
[ec2-user ~]$ sudo softwareupdate --install label
```

   若要安裝所有更新，請使用下列命令。

```
[ec2-user ~]$ sudo softwareupdate --install --all --restart
```

系統管理員可以用 AWS Systems Manager 來在 x86 Mac 執行個體上推出預先核准的作業系統更新。如需詳細資訊，請參閱《AWS Systems Manager 使用者指南》https://docs.aws.amazon.com/systems-manager/latest/userguide/。

您可以使用 Homebrew 在 EC2 macOS 中安裝套件的更新AMIs，以便在執行個體上取得這些套件的最新版本。您也可以使用自製軟件在 Amazon EC2 macOS 上安裝和運行常見的 macOS 應用程序。如需詳細資訊，請參閱 Homebrew 文件。

使用 Homebrew 安裝更新

1. 使用以下命令更新 Homebrew。

```
[ec2-user ~]$ brew update
```

2. 使用以下命令列出具有可用更新的套件。

```
[ec2-user ~]$ brew outdated
```

3. 安裝所有更新或僅特定更新。若要安裝特定更新，請使用下列命令。

```
[ec2-user ~]$ brew upgrade package name
```

若要安裝所有更新，請使用下列命令。

```
[ec2-user ~]$ brew upgrade
```

在 Apple 晶片 Mac 執行個體上更新軟體

考量事項

彈性網絡適配器（ENA）驅動程序

由於網路驅動程式設定有更新，ENA驅動程式 1.0.2 版本與 macOS 13.3 或更新版本不相容。如果您想要安裝任何測試版、預覽版或生產 macOS 13.3 版或更新版本，但尚未安裝最新ENA驅動程式，請使用下列程序安裝新版本的驅動程式。

若要安裝新版本的ENA驅動程式

1. 在終端機窗口中，使用連接到蘋果矽 Mac 實例SSH。
2. 使用以下命令將ENA應用程序下載到`Applications`文件中。

```
[ec2-user ~]$ brew install amazon-ena-ethernet-dext
```

> **疑難排解秘訣**
>
> 如果您收到警告 `No available formula with the name amazon-ena-ethernet-dext`，則請執行下列命令。
>
> ```
> [ec2-user ~]$ brew update
> ```

3. 輸入 exit 並按 return，斷開與執行個體的連線。
4. 使用用VNC戶端啟動ENA應用程式。
   a. 使用設定用VNC戶端Connect 至執行個體的圖形化使用者介面 (GUI)。
   b. 使用「螢幕共用」應用程式連線至執行個體後，請前往「應用程式」資料夾並開啟應用ENA程式。
   c. 選擇啟用
   d. 若要確認驅動程式已正確啟動，請在終端視窗中執行下列命令。命令輸出顯示舊驅動程式處於終止狀態，並且新驅動程式處於啟動狀態。

      ```
      systemextensionsctl list;
      ```

   e. 重新啟動執行個體後，只會出現新的驅動程式。

Apple 晶片 Mac 執行個體上的軟體更新

在 Apple 晶片 Mac 執行個體上，您必須完成數個步驟才能執行就地作業系統更新。首先，使用與 VNC (虛擬網路運算) GUI 用戶端存取執行個體的內部磁碟。此過程使用內置VNC客戶端 macOS 屏幕共享。然後，將擁有權委派給管理使用者 (`ec2-user`)，方法是以 Amazon EBS 磁碟區`aws-managed-user`的身分登入。

當您執行此程序時，會建立兩個密碼。一個密碼用於管理使用者 (`ec2-user`)，另一個密碼用於特殊管理使用者 (`aws-managed-user`)。請記住這些密碼，因為將在執行該程序時使用這些密碼。

> **Note**
>
> 在 macOS Big Sur 上使用此程序，只能執行次要更新，如從 macOS Big Sur 11.7.3 更新為 macOS Big Sur 11.7.4。對於 macOS Monterey 或更高版本，您可以執行主要軟體更新。

存取內部磁碟

1. SSH透過本機電腦，在「終端機」中，使用下列指令連線至 Apple 矽晶 Mac 執行個體。如需詳細資訊，請參閱使用 SSH 連線至您的執行個體。

```
ssh -i /path/key-pair-name.pem ec2-user@instance-public-dns-name
```

2. 使用下列命令，安裝並啟動 macOS Screen Sharing。

```
[ec2-user ~]$ sudo launchctl enable system/com.apple.screensharing
sudo launchctl load -w /System/Library/LaunchDaemons/com.apple.screensharing.plist
```

3. 使用以下命令設定 `ec2-user` 的密碼。記住密碼，供以後使用。

```
[ec2-user ~]$ sudo /usr/bin/dscl . -passwd /Users/ec2-user
```

4. 輸入 exit 並按回車鍵，斷開與執行個體的連線。
5. 從您的本機電腦，在終端機中，使用下列指令透過SSH通道重新連線至連接VNC埠的執行個體。

```
ssh -i /path/key-pair-name.pem -L 5900:localhost:5900 ec2-user@instance-public-dns-name
```

> **Note**
>
> 在完成下列VNC連線和GUI步驟之前，請勿結束此SSH階段作業。重新啟動執行個體時，連線會自動關閉。

6. 從您的本機電腦，使用下列步驟連線到 `localhost:5900`：
   a. 開啟 Finder，並選取開始。
   b. 選取連線至伺服器。
   c. 在伺服器地址欄位中，輸入 `vnc://localhost:5900`。

7. 在 macOS 視窗中，使用您在步驟 3 中建立的密碼以 `ec2-user` 身分連線至 Apple 晶片 Mac 執行個體的遠端工作階段。
8. 使用下列其中一個選項存取名為InternalDisk的內部磁碟。
   a. 對於 macOS Ventura 或更高版本：開啟系統設定，在左窗格中選取一般，然後選取窗格右下角的啟動磁碟。
   b. 對於 macOS Monterey 或以下版本：開啟系統偏好設定，選取啟動磁碟，然後選擇視窗左下角的鎖定圖像來解鎖窗格。

> **疑難排解秘訣**
>
> 如果您需要掛載內部磁碟，則請在終端中執行下列命令。
>
> ```
> APFSVolumeName="InternalDisk" ; SSDContainer=$(diskutil list | grep
> "Physical Store disk0" -B 3 | grep "/dev/disk" | awk {'print $1'} ) ;
> diskutil apfs addVolume $SSDContainer APFS $APFSVolumeName
> ```

9. 選擇名為InternalDisk的內部磁碟，然後選取 [重新啟動]。出現提示時再次選取重新啟動。

> **Important**
>
> 如果內部磁碟的名稱為 Macintosh HD InternalDisk，則您的執行個體必須停止並重新啟動，才能更新專用主機。如需詳細資訊，請參閱停止或終止您的 Amazon EC2 Mac 實例。

使用下列程序將擁有權委派給管理使用者。當您使用重新連線至執行個體時SSH，您可以使用特殊的管理使用者 (`aws-managed-user`) 從內部磁碟開機。`aws-managed-user` 的初始密碼為空白，因此您需要在第一次連線時覆寫它。然後，由於開機磁碟區已變更，因此您需要重複以下步驟來安裝和啟動 macOS Screen Sharing。

將擁有權委派給 Amazon EBS 磁碟區上的管理員

1. 從本機電腦的「終端機」中，使用下列命令連線至 Apple 晶片 Mac 執行個體。

```
ssh -i /path/key-pair-name.pem aws-managed-user@instance-public-dns-name
```

2. 當您收到警告 `WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!` 時，請使用下列其中一個命令來解決此問題。

   a. 使用以下命令清除已知主機。然後，重複上一個步驟。

   ```
   rm ~/.ssh/known_hosts
   ```

   b. 將以下內容新增至上一個步驟中的SSH指令。

   ```
   -o UserKnownHostsFile=/dev/null -o StrictHostKeyChecking=no
   ```

3. 使用下列命令設定 `aws-managed-user` 的密碼。`aws-managed-user` 初始密碼為空白，因此您需要在第一次連線時覆寫它。

   a.
   ```
   [aws-managed-user ~]$ sudo /usr/bin/dscl . -passwd /Users/aws-managed-
   user password
   ```

   b. 當您收到 `Permission denied. Please enter user's old password:` 提示時，請按 Enter 鍵。

   > **疑難排解秘訣**
   >
   > 如果發生 `passwd: DS error: eDSAuthFailed` 錯誤，請使用下列命令。
   >
   > ```
   > [aws-managed-user ~]$ sudo passwd aws-managed-user
   > ```

4. 使用下列命令，安裝並啟動 macOS Screen Sharing。

   ```
   [aws-managed-user ~]$ sudo launchctl enable system/com.apple.screensharing
   sudo launchctl load -w /System/Library/LaunchDaemons/com.apple.screensharing.plist
   ```

5. 輸入 exit 並按回車鍵，斷開與執行個體的連線。
6. 從您的本機電腦，在終端機中，使用下列指令透過SSH通道重新連線至連接VNC埠的執行個體。

   ```
   ssh -i /path/key-pair-name.pem -L 5900:localhost:5900 aws-managed-user@instance-
   public-dns-name
   ```

7. 從您的本機電腦，使用下列步驟連線到 `localhost:5900`：

   a. 開啟 Finder，並選取開始。

b. 選取連線至伺服器。

c. 在伺服器地址欄位中，輸入 `vnc://localhost:5900`。

8. 在 macOS 視窗中，使用您在步驟 3 中建立的密碼以 `aws-managed-user` 身分連線至 Apple 晶片 Mac 執行個體的遠端工作階段。

> **Note**
>
> 當系統提示您使用 Apple ID 登入時，請選取稍後設定。

9. 使用下列其中一個選項存取 Amazon EBS 磁碟區。

   a. 對於 macOS Ventura 或更新版本：開啟系統設定，在左窗格中選取一般，然後選取窗格右下角的啟動磁碟。

   b. 對於 macOS Monterey 或較早的版本：開啟系統偏好設定，選取啟動磁碟，然後使用視窗左下角的鎖定圖像來解鎖窗格。

> **Note**
>
> 在重新開機之前，當系統提示輸入管理員密碼時，請使用您上面設定的 `aws-managed-user` 密碼。此密碼可能與您為 `ec2-user` 或執行個體上的預設管理員帳戶設定的密碼不同。下列說明指定何時使用執行個體的管理員密碼。

10. 選取 Amazon 磁EBS碟區 (未在「啟動磁碟」視窗InternalDisk中命名的磁碟區)，然後選擇「重新啟動」。

> **Note**
>
> 如果您有多個可開機的 Amazon 磁EBS碟區連接到 Apple 矽晶 Mac 執行個體，請務必為每個磁碟區使用唯一的名稱。

11. 確認重新啟動，然後在出現提示時選擇授權使用者。
12. 於在此磁碟區授權使用者窗格中，確認已選取管理使用者 (預設為 `ec2-user`)，然後選取授權。
13. 輸入您在上一個程序的步驟 3 中建立的 `ec2-user` 密碼，然後選取繼續。
14. 出現提示時，請輸入特殊管理使用者的密碼 (`aws-managed-user`)。
15. 從您的本地計算機，在終端中，使用用SSH戶名重新連接到您的實例`ec2-user`。

> **疑難排解秘訣**
>
> 如果收到警告WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!，請運行以下命令並使用重新連接到您的實例SSH。
>
> ```
> rm ~/.ssh/known_hosts
> ```

16. 若要執行軟體更新，請使用 在 x86 Mac 執行個體上更新軟體 下的命令。

增加 Mac 執行個體上的EBS磁碟區大小

您可以增加 Mac 實例上 Amazon EBS 卷的大小。如需詳細資訊，請參閱 Amazon EBS使用者指南中的 Amazon EBS 彈性磁碟區。

增加磁碟區的大小之後，您必須增加APFS容器的大小，如下所示。

使增加的磁碟空間可供使用

1. 判斷是否需要重新啟動。如果您調整執行中 Mac 執行個體上現有EBS磁碟區的大小，您必須重新啟動執行個體才能使用新的大小。如果在啟動期間修改磁碟空間，則不需要重新開機。

   檢視磁碟大小的目前狀態：

   ```
   [ec2-user ~]$  diskutil list external physical
   /dev/disk0 (external, physical):
      #:                       TYPE NAME                    SIZE       IDENTIFIER
      0:                 GUID_partition_scheme            *322.1 GB     disk0
      1:                 EFI EFI                           209.7 MB     disk0s1
      2:                 Apple_APFS Container disk2        321.9 GB     disk0s2
   ```

2. 複製並貼上下列命令。

   ```
   [ec2-user ~]$ PDISK=$(diskutil list physical external | head -n1 | cut -d" " -f1)
   APFSCONT=$(diskutil list physical external | grep "Apple_APFS" | tr -s " " | cut -
   d" " -f8)
   yes | sudo diskutil repairDisk $PDISK
   ```

3. 複製並貼上下列命令。

```
[ec2-user ~]$ sudo diskutil apfs resizeContainer $APFSCONT 0
```

停止或終止您的 Amazon EC2 Mac 實例

當您停止 Mac 執行個體時，該執行個體會在進入 `stopped` 狀態之前維持約 15 分鐘的 `stopping` 狀態。

當您停止或終止 Mac EC2 執行個體時，Amazon 會在基礎專用主機上執行清除工作流程，以清除內部SSD、清除持續性NVRAM變數，以及更新至最新的裝置韌體。這可確保 Mac 執行個體提供與其他 EC2 Nitro 執行個體相同的安全性和資料隱私權。它還允許您運行最新的 macOS AMIs。在清理工作流程期間，專用執行個體會暫時進入 pending (待定) 狀態。在 x86 Mac 執行個體上，清理工作流程最多可能需要 50 分鐘才能完成。在 Apple 晶片 Mac 執行個體上，清理工作流程最多可能需要 110 分鐘才能完成。此外，在 x86 Mac 執行個體上，如果裝置韌體需要更新，清理工作流程最多可能需要 3 個小時才能完成。

直到清理工作流程完成，也即 專用執行個體 進入 `available` 狀態，您才能啟動已停止的 Mac 執行個體或啟動新的 Mac 執行個體。

專用執行個體進入 `pending` 狀態時，計量和計費會暫停。清理工作流程期間，您不需支付費用。

為您的 Mac 執行個體釋出 專用主機

當您完成 Mac 執行個體時，您可以釋出 專用執行個體。您必須先停止或終止 Mac 執行個體，才能發佈專用執行個體。至少在配置期間經過 24 小時後才能發佈主機。

發佈專用執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體並選取 Instance state (執行個體狀態)，然後選取 Stop instance (停止執行個體) 或 Terminate instance (終止執行個體)。
4. 在導覽窗格中，選擇 專用執行個體。
5. 選取 專用執行個體 並選取 Actions (動作)，Release host (釋出主機)。
6. 出現確認提示時，請選擇 Release (釋出)。

查找支持的 macOS 版本 Amazon EC2 Mac 專用主機

您可以檢視 Amazon EC2 Mac 專用主機支援的最新 macOS 版本。透過此功能，您可以驗證專用主機是否支援使用偏好 macOS 版本的執行個體啟動。

每個 macOS 版本都需要基礎蘋果 Mac 上的最低固件版本才能成功啟動。如果分配的 Mac 專用主機長時間保持閒置狀態，或者其上有長時間運行的實例，則 Apple Mac 固件版本可能會過時。

為確保最新 macOS 版本的支援能力，您可以停止或終止已配置的 Mac 專用主機上的執行個體。這會觸發主機清理工作流程，並更新基礎 Apple Mac 上的韌體，以支援最新的 macOS 版本。當您停止或終止執行中的執行個體時，具有長時間執行執行個體的專用主機會自動更新。

如需拖曳工作流程的詳細資訊，請參閱停止或終止您的 Amazon EC2 Mac 實例。

如需啟動 Mac 執行個體的詳細資訊，請參閱使用或啟動 Mac 執行 AWS Management Console 個體 AWS CLI。

您可以使用 Amazon 主EC2控台或檢視分配的專用主機上支援的最新 macOS 版本的相關資訊 AWS CLI。

Console

使用主控台檢視專用主機韌體資訊

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 專用主機。
3. 在專用主機詳細資料頁面的「最新受支援的 macOS 版本」 下，您可以查看主機可支援的最新 macOS 版本。

AWS CLI

使用檢視專用主機韌體資訊 AWS CLI

使用該`describe-mac-hosts`命令，`region`用適當的替換 AWS 區域。

```
$ aws ec2 describe-mac-hosts --region us-east-1
  {
      "MacHosts": [
          {
```

```
                "HostId": "h-07879acf49EXAMPLE",
                "MacOSLatestSupportedVersions": [
                    "14.3",
                    "13.6.4",
                    "12.7.3"
                ]
            }
        ]
    }
```

訂閱 macOS AMI 通知

若要AMIs在最新推出或 BridgeOS 更新時收到通知，請使用 Amazon SNS 訂閱通知。

如需 EC2 macOS 的詳細資訊AMIs，請參閱Amazon EC2 macOS AMIs 本注意事項。

若要訂閱 macOS AMI 通知

1. 在 https://console.aws.amazon.com/sns/v3/ home 上打開 Amazon SNS 控制台。
2. 如有必要，請在導覽列中將「區域」變更為美國東部 (維吉尼亞北部)。您必須使用此區域，因為您訂閱的SNS通知是在此區域中建立的。
3. 在導覽窗格中，選擇訂閱。
4. 選擇 Create subscription (建立訂閱)。
5. 針對 Create subscription (建立訂閱) 對話方塊，執行下列作業：

   a. 對於主題 ARN，請複製並貼上下列其中一個 Amazon 資源名稱 (ARNs)：

      - **`arn:aws:sns:us-east-1:898855652048:amazon-ec2-macos-ami-updates`**
      - **`arn:aws:sns:us-east-1:898855652048:amazon-ec2-bridgeos-updates`**

   b. 在「通訊協定」中，選擇下列其中一項：

      - 電子郵件：

        針對 Endpoint (端點)，輸入可用於接收通知的電子郵件地址。建立訂閱後，您將收到主旨行為 `AWS Notification - Subscription Confirmation` 的確認訊息。開啟電子郵件並選擇 Confirm subscription (確認訂閱) 完成訂閱。

      - SMS:

        對於 Endpoint (端點)，輸入可用於接收通知的電子郵件地址。

- AWS Lambda，AmazonSQS，Amazon 數據 Firehose 件（通知以JSON格式提供）：

  針對端點，輸入ARN您可用來接收通知的 Lambda 函數、SQS佇列或 Firehose 串流。

c. 選擇 Create subscription (建立訂閱)。

每當 macOS 發布時，我們AMIs都會向該`amazon-ec2-macos-ami-updates`主題的訂閱者發送通知。只要 bridgeOS 有所變更，我們就會向 `amazon-ec2-bridgeos-updates` 主題的訂閱者傳送通知。如果您不想再接收這些通知，請使用下列程序來取消訂閱。

若要取消訂閱 macOS AMI 通知

1. 在 https://console.aws.amazon.com/sns/v3/ home 上打開 Amazon SNS 控制台。
2. 如有必要，請在導覽列中將「區域」變更為美國東部 (維吉尼亞北部)。您必須使用此區域，因為SNS通知是在此區域中建立的。
3. 在導覽窗格中，選擇 Subscriptions (訂閱)。
4. 選取訂閱，接著選取 Actions (動作)、Delete subscriptions (刪除訂閱)，提示確認時請選取 Delete (刪除)。

AMIIDs使用 AWS Systems Manager 參數存儲檢索 macOS API

您必須在啟動執行個體AMI時指定。特定AMI於 AWS 區域、作業系統和處理器架構。您可以AMIs在中檢視所有 macOS， AWS 區域 並AMI透過查詢「 AWS Systems Manager 參數存放區」來擷取最新的 macOS API。使用這些公開參數，您無需手動查詢 macOS AMI IDs。x86和 ARM64 macOS 均可使用公開參數AMIs，並可與現有 AWS CloudFormation 範本整合。

所需的許可

若要執行此動作，IAM主體必須具有呼叫`ssm:GetParameter`API動作的權限。

若要檢視目前所有 macOS AMIs 的清單，請 AWS 區域 使用 AWS CLI

使用下列get-parameters-by-path命令檢視目前「地區」AMIs 中所有 macOS 的清單。

```
aws ssm get-parameters-by-path --path /aws/service/ec2-macos --recursive --query
 "Parameters[].Name"
```

若要擷取最新主要 macOS 的 AMI ID，請AMI使用 AWS CLI

使用下面的獲取參數命令與子參數。`image_id`在下列範例中，請`sonoma`以 macOS 支援`x86_64_mac`的主要版本、處理器取代，以及`region-code` AWS 區域 支援的最新 macOS AMI ID。

```
aws ssm get-parameter --name /aws/service/ec2-macos/sonoma/x86_64_mac/latest/image_id
 --region region-code
```

如需詳細資訊，請參閱「AWS Systems Manager 使用手冊」中的「呼叫 macOS 的AMI公開參數」。

Amazon EC2 macOS AMIs 本注意事項

以下資訊提供 EC2 macOS 中預設包含的套件詳細資訊，AMIs並摘要說明每個 EC2 macOS AMI 版本的變更。

如需如何訂閱 macOS AMI 通知的詳細資訊，請參閱訂閱 macOS AMI 通知。

包含在 Amazon EC2 macOS 默認軟件包 AMIs

下表說明 EC2 macOS 中預設包含的套件AMIs。

套件	版本備註
EC2初 macOS 化	https://github.com/aws/ec2-macos-init/tags
EC2macOS 實用程序	https://github.com/aws/ec2-macos-utils/tags
AmazonSSM代理	https://github.com/aws/amazon-ssm-agent/releases
AWS Command Line Interface (AWS CLI) 第二版	https://raw.githubusercontent.com/aws/aws-cli/v2/CHANGELOG.rst
Xcode 的命令列工具	https://developer.apple.com/documentation/xcode-release-notes
Homebrew	https://github.com/Homebrew/brew/releases
EC2實例 Connect	https://github.com/aws/aws-ec2-instance-connect-config/releases

套件	版本備註
Safari	https://developer.apple.com/documentation/safari-release-notes

Amazon EC2 macOS AMI 更新

下表說明 EC2 macOS AMI 版本中包含的變更。請注意，某些變更會套用至所有 EC2 macOSAMIs，而有些變更僅適用於這些 macOS 的一部分AMIs。

EC2macOS AMI 更新

發行版本	改變

2024.08.20	所有 AMIs
	• 自製軟件更新至 14 年 3 月 4 日
	• 已更新`aws-cli`至 2 月 17 日
	
	蘋果系 macOS 索諾瑪 14.6.1 的發布（所有 Mac 實例）
	• 沒有已發佈的CVE項目。
	
	蘋果系 macOS 文圖拉 13.6.9 的發布（所有 Mac 實例）
	• 沒有已發佈的CVE項目。
	• 更新野生動物園至 17.6
	  • 野生動物園 17.6 的安全內容
	
	發布的 macOS 蒙特雷 12.7.6（所有 Mac 實例）
	• macOS 蒙特雷 12.7.6 的安全性內容
	• 更新野生動物園至 17.6
	  • 野生動物園 17.6 的安全內容

發行版本	改變

2024.06.07	所有 AMIs
	
	• 自製軟件更新至 4.3.1-1
	• 已更新`aws-cli`至 2 月 15 日
	• 已更新`amazon-ssm-agent` 至 3 月 3 日
	
	蘋果系 macOS 索諾瑪 14.5 的發布（所有 Mac 實例）
	
	• macOS 索諾瑪 14.5 的安全內容
	
	蘋果系 macOS 文圖拉 13.6.7 的發布（所有 Mac 實例）
	
	• macOS 文圖拉 13.6.7 的安全性內容
	• 更新野生動物園到 17.5
	  • 野生動物園 17.5 的安全內容
	
	蘋果系 macOS 蒙特雷 12.7.5 的發布（所有 Mac 實例）
	
	• macOS 蒙特雷 12.7.5 的安全性內容
	• 更新野生動物園到 17.5
	  • 野生動物園 17.5 的安全內容

發行版本	改變

2024.04.12	所有 AMIs
	• 已更新自製軟體至 4 月 16 日
	• 已更新`aws-cli`至 2 月 15 日
	
	蘋果系 macOS 索諾瑪 14.4.1 的發布（所有 Mac 實例）
	• macOS 索諾瑪 14.4.1 的安全性內容
	
	蘋果系 macOS 文圖拉 13.6.6 的發布（所有 Mac 實例）
	• macOS 文圖拉 13.6.6 的安全性內容
	• 更新野生動物園 17.4.1
	  • 野生動物園 17.4.1 的安全性內容
	
	對於 macOS 蒙特雷（所有 Mac 實例）
	• 更新野生動物園 17.4.1
	  • 野生動物園 17.4.1 的安全性內容

Amazon EBS 優化執行個體類型

Amazon 最佳EBS化執行個體使用最佳化的組態堆疊，並為 Amazon I/O 提供額外的專用頻寬，此優化可將 Amazon EBS I/O 與執行個體的其他流量之間的爭用降到最低，從而為您的EBS磁碟區提供最佳效能。EBS

連接EBS至最佳化執行個體時，一般用途 SSD (`gp2`和`gp3`) 磁碟區的設計目的是在指定年度內至少提供 99% 的佈建IOPS效能達到 99% 的時間，而佈建 IOPS SSD (`io1`和`io2`) 磁碟區則設計為在指定年度提供至少 90% 的佈建IOPS效能達到 99.9% 的時間。輸送量最佳化 HDD (`st1`) 和冷 HDD (`sc1`) 至少可提供 90% 的預期輸送量效能，達到指定年度 99% 的時間。不相容的期間約為統一分佈，目標為每小時 99% 的預期總輸送量。如需詳細資訊，請參閱 Amazon EBS使用者指南中的 Amazon EBS 磁碟區類型。

某些執行個體類型預設為EBS最佳化，因此如果您嘗試停用它，也不需要啟用它，也不會產生任何作用。其他執行個體類型可選擇支援EBS最佳化，您可以在啟動期間或之後啟用，只需支付額外的小時費用。某些執行個體類型不支援EBS最佳化。

如需詳細的執行個體類型規格和功能，請參閱 Amazon EC2 執行個體類型指南。

主題

- EBS默認優化
- EBS支援最佳化
- 獲得最大的 Amazon EBS 優化性能
- 尋找 Amazon EBS 優化的 Amazon EC2 執行個體
- 為 Amazon 執行個體啟用 EC2 Amazon EBS 優化

EBS默認優化

依預設，下列執行個體類型為EBS最佳化。如果禁用EBS優化，則無需啟用EBS優化也不會產生任何效果。

> **Important**
>
> 執行個體的EBS效能受到執行個體類型的效能限制，或其連接磁碟區的彙總效能 (以較小者為準) 所限制。若要達到最大EBS效能，執行個體必須具有連接的磁碟區，這些磁碟區必須提供等於或大於最大執行個體效能的組合效能。例如，執行個體必須至少佈建16,000IOPS每個磁碟區 (5gp35磁碟區 x 16,000 IOPS = 80,000IOPS)r6i.16xlarge，才能達成80,000IOPS對應的執行個體。我們建議您選擇提供比應用程式需求更專用 Amazon EBS 輸送量的執行個體類型；否則，Amazon EBS 和 Amazon 之間的連線EC2可能會成為效能瓶頸。

> **Note**
>
> [1] 這些執行個體至少每 24 小時可維持 30 分鐘的最大效能一次，之後恢復為基準效能。
> [2] 這些執行個體可以無限期維持其上述效能。如果您的工作負載需要維持超過 30 分鐘的最佳效能，請選取其中一個執行個體。

一般用途執行個體

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
a1.medium [1]	300	3500	37.50	437.50	2500	20000
a1.large [1]	525	3500	65.62	437.50	4000	20000
a1.xlarge [1]	800	3500	100.00	437.50	6000	20000
a1.2xlarge [1]	1750	3500	218.75	437.50	10000	20000
a1.4xlarge [2]	3500		437.5		20000	
a1.metal [2]	3500		437.5		20000	
m4.large [2]	450		56.25		3600	
m4.xlarge [2]	750		93.75		6000	
m4.2xlarge [2]	1000		125.0		8000	
m4.4xlarge [2]	2000		250.0		16000	
m4.10xlarge [2]	4000		500.0		32000	
m4.16xlarge [2]	10000		1250.0		65000	
m5.large [1]	650	4750	81.25	593.75	3600	18750

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
m5.xlarge [1]	1150	4750	143.75	593.75	6000	18750
m5.2xlarge [1]	2300	4750	287.50	593.75	12000	18750
m5.4xlarge [2]	4750		593.75		18750	
m5.8xlarge [2]	6800		850.0		30000	
m5.12xlarge [2]	9500		1187.5		40000	
m5.16xlarge [2]	13600		1700.0		60000	
m5.24xlarge [2]	19000		2375.0		80000	
m5.metal [2]	19000		2375.0		80000	
m5a.large [1]	650	2880	81.25	360.00	3600	16000
m5a.xlarge [1]	1085	2880	135.62	360.00	6000	16000
m5a.2xlarge [1]	1580	2880	197.50	360.00	8333	16000
m5a.4xlarge [2]	2880		360.0		16000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
m5a.8xlarge [2]	4750		593.75		20000	
m5a.12xlarge [2]	6780		847.5		30000	
m5a.16xlarge [2]	9500		1187.5		40000	
m5a.24xlarge [2]	13750		1718.75		60000	
m5ad.large [1]	650	2880	81.25	360.00	3600	16000
m5ad.xlarge [1]	1085	2880	135.62	360.00	6000	16000
m5ad.2xlarge [1]	1580	2880	197.50	360.00	8333	16000
m5ad.4xlarge [2]	2880		360.0		16000	
m5ad.8xlarge [2]	4750		593.75		20000	
m5ad.12xlarge [2]	6780		847.5		30000	
m5ad.16xlarge [2]	9500		1187.5		40000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
m5ad.24xlarge [2]	13750		1718.75		60000	
m5d.large [1]	650	4750	81.25	593.75	3600	18750
m5d.xlarge [1]	1150	4750	143.75	593.75	6000	18750
m5d.2xlarge [1]	2300	4750	287.50	593.75	12000	18750
m5d.4xlarge [2]	4750		593.75		18750	
m5d.8xlarge [2]	6800		850.0		30000	
m5d.12xlarge [2]	9500		1187.5		40000	
m5d.16xlarge [2]	13600		1700.0		60000	
m5d.24xlarge [2]	19000		2375.0		80000	
m5d.metal [2]	19000		2375.0		80000	
m5dn.large [1]	650	4750	81.25	593.75	3600	18750

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
m5dn.xlarge [1]	1150	4750	143.75	593.75	6000	18750
m5dn.2xlarge [1]	2300	4750	287.50	593.75	12000	18750
m5dn.4xlarge [2]	4750		593.75		18750	
m5dn.8xlarge [2]	6800		850.0		30000	
m5dn.12xlarge [2]	9500		1187.5		40000	
m5dn.16xlarge [2]	13600		1700.0		60000	
m5dn.24xlarge [2]	19000		2375.0		80000	
m5dn.metal [2]	19000		2375.0		80000	
m5n.large [1]	650	4750	81.25	593.75	3600	18750
m5n.xlarge [1]	1150	4750	143.75	593.75	6000	18750
m5n.2xlarge [1]	2300	4750	287.50	593.75	12000	18750

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
m5n.4xlarge [2]	4750		593.75		18750	
m5n.8xlarge [2]	6800		850.0		30000	
m5n.12xlarge [2]	9500		1187.5		40000	
m5n.16xlarge [2]	13600		1700.0		60000	
m5n.24xlarge [2]	19000		2375.0		80000	
m5n.metal [2]	19000		2375.0		80000	
m5zn.large [1]	800	3170	100.00	396.25	3333	13333
m5zn.xlarge [1]	1564	3170	195.50	396.25	6667	13333
m5zn.2xlarge [2]	3170		396.25		13333	
m5zn.3xlarge [2]	4750		593.75		20000	
m5zn.6xlarge [2]	9500		1187.5		40000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
m5zn.12xlarge [2]	19000		2375.0		80000	
m5zn.metal [2]	19000		2375.0		80000	
m6a.large [1]	650	10000	81.25	1250.00	3600	40000
m6a.xlarge [1]	1250	10000	156.25	1250.00	6000	40000
m6a.2xlarge [1]	2500	10000	312.50	1250.00	12000	40000
m6a.4xlarge [1]	5000	10000	625.00	1250.00	20000	40000
m6a.8xlarge [2]	10000		1250.0		40000	
m6a.12xlarge [2]	15000		1875.0		60000	
m6a.16xlarge [2]	20000		2500.0		80000	
m6a.24xlarge [2]	30000		3750.0		120000	
m6a.32xlarge [2]	40000		5000.0		160000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
m6a.48xlarge [2]	40000		5000.0		240000	
m6a.metal [2]	40000		5000.0		240000	
m6g.medium [1]	315	4750	39.38	593.75	2500	20000
m6g.large [1]	630	4750	78.75	593.75	3600	20000
m6g.xlarge [1]	1188	4750	148.50	593.75	6000	20000
m6g.2xlarge [1]	2375	4750	296.88	593.75	12000	20000
m6g.4xlarge [2]	4750		593.75		20000	
m6g.8xlarge [2]	9500		1187.5		40000	
m6g.12xlarge [2]	14250		1781.25		50000	
m6g.16xlarge [2]	19000		2375.0		80000	
m6g.metal [2]	19000		2375.0		80000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
m6gd.medium [1]	315	4750	39.38	593.75	2500	20000
m6gd.large [1]	630	4750	78.75	593.75	3600	20000
m6gd.xlarge [1]	1188	4750	148.50	593.75	6000	20000
m6gd.2xlarge [1]	2375	4750	296.88	593.75	12000	20000
m6gd.4xlarge [2]	4750		593.75		20000	
m6gd.8xlarge [2]	9500		1187.5		40000	
m6gd.12xlarge [2]	14250		1781.25		50000	
m6gd.16xlarge [2]	19000		2375.0		80000	
m6gd.metal [2]	19000		2375.0		80000	
m6i.large [1]	650	10000	81.25	1250.00	3600	40000
m6i.xlarge [1]	1250	10000	156.25	1250.00	6000	40000
m6i.2xlarge [1]	2500	10000	312.50	1250.00	12000	40000

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
m6i.4xlarge [1]	5000	10000	625.00	1250.00	20000	40000
m6i.8xlarge [2]	10000		1250.0		40000	
m6i.12xlarge [2]	15000		1875.0		60000	
m6i.16xlarge [2]	20000		2500.0		80000	
m6i.24xlarge [2]	30000		3750.0		120000	
m6i.32xlarge [2]	40000		5000.0		160000	
m6i.metal [2]	40000		5000.0		160000	
m6id.large [1]	650	10000	81.25	1250.00	3600	40000
m6id.xlarge [1]	1250	10000	156.25	1250.00	6000	40000
m6id.2xlarge [1]	2500	10000	312.50	1250.00	12000	40000
m6id.4xlarge [1]	5000	10000	625.00	1250.00	20000	40000
m6id.8xlarge [2]	10000		1250.0		40000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
m6id.12xlarge [2]	15000		1875.0		60000	
m6id.16xlarge [2]	20000		2500.0		80000	
m6id.24xlarge [2]	30000		3750.0		120000	
m6id.32xlarge [2]	40000		5000.0		160000	
m6id.metal [2]	40000		5000.0		160000	
m6idn.large [1]	1562	25000	195.31	3125.00	6250	100000
m6idn.xlarge [1]	3125	25000	390.62	3125.00	12500	100000
m6idn.2xlarge [1]	6250	25000	781.25	3125.00	25000	100000
m6idn.4xlarge [1]	12500	25000	1562.50	3125.00	50000	100000
m6idn.8xlarge [2]	25000		3125.0		100000	
m6idn.12xlarge [2]	37500		4687.5		150000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
m6idn.16xlarge [2]	50000		6250.0		200000	
m6idn.24xlarge [2]	75000		9375.0		300000	
m6idn.32xlarge [2]	100000		12500.0		400000	
m6idn.metal [2]	100000		12500.0		400000	
m6in.large [1]	1562	25000	195.31	3125.00	6250	100000
m6in.xlarge [1]	3125	25000	390.62	3125.00	12500	100000
m6in.2xlarge [1]	6250	25000	781.25	3125.00	25000	100000
m6in.4xlarge [1]	12500	25000	1562.50	3125.00	50000	100000
m6in.8xlarge [2]	25000		3125.0		100000	
m6in.12xlarge [2]	37500		4687.5		150000	
m6in.16xlarge [2]	50000		6250.0		200000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
m6in.24xlarge [2]	75000		9375.0		300000	
m6in.32xlarge [2]	100000		12500.0		400000	
m6in.metal [2]	100000		12500.0		400000	
m7a.medium [1]	325	10000	40.62	1250.00	2500	40000
m7a.large [1]	650	10000	81.25	1250.00	3600	40000
m7a.xlarge [1]	1250	10000	156.25	1250.00	6000	40000
m7a.2xlarge [1]	2500	10000	312.50	1250.00	12000	40000
m7a.4xlarge [1]	5000	10000	625.00	1250.00	20000	40000
m7a.8xlarge [2]	10000		1250.0		40000	
m7a.12xlarge [2]	15000		1875.0		60000	
m7a.16xlarge [2]	20000		2500.0		80000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
m7a.24xlarge [2]	30000		3750.0		120000	
m7a.32xlarge [2]	40000		5000.0		160000	
m7a.48xlarge [2]	40000		5000.0		240000	
m7a.metal-48xl [2]	40000		5000.0		240000	
m7g.medium [1]	315	10000	39.38	1250.00	2500	40000
m7g.large [1]	630	10000	78.75	1250.00	3600	40000
m7g.xlarge [1]	1250	10000	156.25	1250.00	6000	40000
m7g.2xlarge [1]	2500	10000	312.50	1250.00	12000	40000
m7g.4xlarge [1]	5000	10000	625.00	1250.00	20000	40000
m7g.8xlarge [2]	10000		1250.0		40000	
m7g.12xlarge [2]	15000		1875.0		60000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
m7g.16xlarge [2]	20000		2500.0		80000	
m7g.metal [2]	20000		2500.0		80000	
m7gd.medium [1]	315	10000	39.38	1250.00	2500	40000
m7gd.large [1]	630	10000	78.75	1250.00	3600	40000
m7gd.xlarge [1]	1250	10000	156.25	1250.00	6000	40000
m7gd.2xlarge [1]	2500	10000	312.50	1250.00	12000	40000
m7gd.4xlarge [1]	5000	10000	625.00	1250.00	20000	40000
m7gd.8xlarge [2]	10000		1250.0		40000	
m7gd.12xlarge [2]	15000		1875.0		60000	
m7gd.16xlarge [2]	20000		2500.0		80000	
M7 克. 金屬	20000		2500.0		80000	
m7i.large [1]	650	10000	81.25	1250.00	3600	40000

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
m7i.xlarge [1]	1250	10000	156.25	1250.00	6000	40000
m7i.2xlarge [1]	2500	10000	312.50	1250.00	12000	40000
m7i.4xlarge [1]	5000	10000	625.00	1250.00	20000	40000
m7i.8xlarge [2]	10000		1250.0		40000	
m7i.12xlarge [2]	15000		1875.0		60000	
m7i.16xlarge [2]	20000		2500.0		80000	
m7i.24xlarge [2]	30000		3750.0		120000	
m7i.48xlarge [2]	40000		5000.0		240000	
m7i.metal-24xl [2]	30000		3750.0		120000	
m7i.metal-48xl [2]	40000		5000.0		240000	
m7i-flex.large [1]	312	10000	39.06	1250.00	2500	40000

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
m7i-flex.xlarge [1]	625	10000	78.12	1250.00	3600	40000
m7i-flex.2xlarge [1]	1250	10000	156.25	1250.00	6000	40000
m7i-flex.4xlarge [1]	2500	10000	312.50	1250.00	12000	40000
m7i-flex.8xlarge [1]	5000	10000	625.00	1250.00	20000	40000
mac1.metal [2]	14000		1750.0		80000	
mac2.metal [2]	10000		1250.0		55000	
超微金屬	10000		1250.0		55000	
mac2-m2.metal [2]	8000		1000.0		55000	
mac2-m2pro.metal [2]	8000		1000.0		55000	
t3.nano [1]	43	2085	5.38	260.62	250	11800
t3.micro [1]	87	2085	10.88	260.62	500	11800
t3.small [1]	174	2085	21.75	260.62	1000	11800
t3.medium [1]	347	2085	43.38	260.62	2000	11800

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
t3.large [1]	695	2780	86.88	347.50	4000	15700
t3.xlarge [1]	695	2780	86.88	347.50	4000	15700
t3.2xlarge [1]	695	2780	86.88	347.50	4000	15700
t3a.nano [1]	45	2085	5.62	260.62	250	11800
t3a.micro [1]	90	2085	11.25	260.62	500	11800
t3a.small [1]	175	2085	21.88	260.62	1000	11800
t3a.medium [1]	350	2085	43.75	260.62	2000	11800
t3a.large [1]	695	2780	86.88	347.50	4000	15700
t3a.xlarge [1]	695	2780	86.88	347.50	4000	15700
t3a.2xlarge [1]	695	2780	86.88	347.50	4000	15700
t4g.nano [1]	43	2085	5.38	260.62	250	11800
t4g.micro [1]	87	2085	10.88	260.62	500	11800
t4g.small [1]	174	2085	21.75	260.62	1000	11800
t4g.medium [1]	347	2085	43.38	260.62	2000	11800
t4g.large [1]	695	2780	86.88	347.50	4000	15700
t4g.xlarge [1]	695	2780	86.88	347.50	4000	15700

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
t4g.2xlarge [1]	695	2780	86.88	347.50	4000	15700

運算最佳化

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
c4.large [2]	500		62.5		4000	
c4.xlarge [2]	750		93.75		6000	
c4.2xlarge [2]	1000		125.0		8000	
c4.4xlarge [2]	2000		250.0		16000	
c4.8xlarge [2]	4000		500.0		32000	
c5.large [1]	650	4750	81.25	593.75	4000	20000
c5.xlarge [1]	1150	4750	143.75	593.75	6000	20000
c5.2xlarge [1]	2300	4750	287.50	593.75	10000	20000
c5.4xlarge [2]	4750		593.75		20000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
c5.9xlarge [2]	9500		1187.5		40000	
c5.12xlarge [2]	9500		1187.5		40000	
c5.18xlarge [2]	19000		2375.0		80000	
c5.24xlarge [2]	19000		2375.0		80000	
c5.metal [2]	19000		2375.0		80000	
c5a.large [1]	200	3170	25.00	396.25	800	13300
c5a.xlarge [1]	400	3170	50.00	396.25	1600	13300
c5a.2xlarge [1]	800	3170	100.00	396.25	3200	13300
c5a.4xlarge [1]	1580	3170	197.50	396.25	6600	13300
c5a.8xlarge [2]	3170		396.25		13300	
c5a.12xlarge [2]	4750		593.75		20000	
c5a.16xlarge [2]	6300		787.5		26700	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
c5a.24xlarge [2]	9500		1187.5		40000	
c5ad.large [1]	200	3170	25.00	396.25	800	13300
c5ad.xlarge [1]	400	3170	50.00	396.25	1600	13300
c5ad.2xlarge [1]	800	3170	100.00	396.25	3200	13300
c5ad.4xlarge [1]	1580	3170	197.50	396.25	6600	13300
c5ad.8xlarge [2]	3170		396.25		13300	
c5ad.12xlarge [2]	4750		593.75		20000	
c5ad.16xlarge [2]	6300		787.5		26700	
c5ad.24xlarge [2]	9500		1187.5		40000	
c5d.large [1]	650	4750	81.25	593.75	4000	20000
c5d.xlarge [1]	1150	4750	143.75	593.75	6000	20000
c5d.2xlarge [1]	2300	4750	287.50	593.75	10000	20000

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
c5d.4xlarge [2]	4750		593.75		20000	
c5d.9xlarge [2]	9500		1187.5		40000	
c5d.12xlarge [2]	9500		1187.5		40000	
c5d.18xlarge [2]	19000		2375.0		80000	
c5d.24xlarge [2]	19000		2375.0		80000	
c5d.metal [2]	19000		2375.0		80000	
c5n.large [1]	650	4750	81.25	593.75	4000	20000
c5n.xlarge [1]	1150	4750	143.75	593.75	6000	20000
c5n.2xlarge [1]	2300	4750	287.50	593.75	10000	20000
c5n.4xlarge [2]	4750		593.75		20000	
c5n.9xlarge [2]	9500		1187.5		40000	
c5n.18xlarge [2]	19000		2375.0		80000	
c5n.metal [2]	19000		2375.0		80000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
c6a.large [1]	650	10000	81.25	1250.00	3600	40000
c6a.xlarge [1]	1250	10000	156.25	1250.00	6000	40000
c6a.2xlarge [1]	2500	10000	312.50	1250.00	12000	40000
c6a.4xlarge [1]	5000	10000	625.00	1250.00	20000	40000
c6a.8xlarge [2]	10000		1250.0		40000	
c6a.12xlarge [2]	15000		1875.0		60000	
c6a.16xlarge [2]	20000		2500.0		80000	
c6a.24xlarge [2]	30000		3750.0		120000	
c6a.32xlarge [2]	40000		5000.0		160000	
c6a.48xlarge [2]	40000		5000.0		240000	
c6a.metal [2]	40000		5000.0		240000	
c6g.medium [1]	315	4750	39.38	593.75	2500	20000
c6g.large [1]	630	4750	78.75	593.75	3600	20000

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
c6g.xlarge [1]	1188	4750	148.50	593.75	6000	20000
c6g.2xlarge [1]	2375	4750	296.88	593.75	12000	20000
c6g.4xlarge [2]	4750		593.75		20000	
c6g.8xlarge [2]	9500		1187.5		40000	
c6g.12xlarge [2]	14250		1781.25		50000	
c6g.16xlarge [2]	19000		2375.0		80000	
c6g.metal [2]	19000		2375.0		80000	
c6gd.medium [1]	315	4750	39.38	593.75	2500	20000
c6gd.large [1]	630	4750	78.75	593.75	3600	20000
c6gd.xlarge [1]	1188	4750	148.50	593.75	6000	20000
c6gd.2xlarge [1]	2375	4750	296.88	593.75	12000	20000
c6gd.4xlarge [2]	4750		593.75		20000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
c6gd.8xlarge [2]	9500		1187.5		40000	
c6gd.12xlarge [2]	14250		1781.25		50000	
c6gd.16xlarge [2]	19000		2375.0		80000	
c6gd.metal [2]	19000		2375.0		80000	
c6gn.medium [1]	760	9500	95.00	1187.50	2500	40000
c6gn.large [1]	1235	9500	154.38	1187.50	5000	40000
c6gn.xlarge [1]	2375	9500	296.88	1187.50	10000	40000
c6gn.2xlarge [1]	4750	9500	593.75	1187.50	20000	40000
c6gn.4xlarge [2]	9500		1187.5		40000	
c6gn.8xlarge [2]	19000		2375.0		80000	
c6gn.12xlarge [2]	28500		3562.5		120000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
c6gn.16xlarge [2]	38000		4750.0		160000	
c6i.large [1]	650	10000	81.25	1250.00	3600	40000
c6i.xlarge [1]	1250	10000	156.25	1250.00	6000	40000
c6i.2xlarge [1]	2500	10000	312.50	1250.00	12000	40000
c6i.4xlarge [1]	5000	10000	625.00	1250.00	20000	40000
c6i.8xlarge [2]	10000		1250.0		40000	
c6i.12xlarge [2]	15000		1875.0		60000	
c6i.16xlarge [2]	20000		2500.0		80000	
c6i.24xlarge [2]	30000		3750.0		120000	
c6i.32xlarge [2]	40000		5000.0		160000	
c6i.metal [2]	40000		5000.0		160000	
c6id.large [1]	650	10000	81.25	1250.00	3600	40000
c6id.xlarge [1]	1250	10000	156.25	1250.00	6000	40000

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
c6id.2xlarge [1]	2500	10000	312.50	1250.00	12000	40000
c6id.4xlarge [1]	5000	10000	625.00	1250.00	20000	40000
c6id.8xlarge [2]	10000		1250.0		40000	
c6id.12xlarge [2]	15000		1875.0		60000	
c6id.16xlarge [2]	20000		2500.0		80000	
c6id.24xlarge [2]	30000		3750.0		120000	
c6id.32xlarge [2]	40000		5000.0		160000	
c6id.metal [2]	40000		5000.0		160000	
c6in.large [1]	1562	25000	195.31	3125.00	6250	100000
c6in.xlarge [1]	3125	25000	390.62	3125.00	12500	100000
c6in.2xlarge [1]	6250	25000	781.25	3125.00	25000	100000
c6in.4xlarge [1]	12500	25000	1562.50	3125.00	50000	100000

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
c6in.8xlarge [2]	25000		3125.0		100000	
c6in.12xlarge [2]	37500		4687.5		150000	
c6in.16xlarge [2]	50000		6250.0		200000	
c6in.24xlarge [2]	75000		9375.0		300000	
c6in.32xlarge [2]	100000		12500.0		400000	
c6in.metal [2]	100000		12500.0		400000	
c7a.medium [1]	325	10000	40.62	1250.00	2500	40000
c7a.large [1]	650	10000	81.25	1250.00	3600	40000
c7a.xlarge [1]	1250	10000	156.25	1250.00	6000	40000
c7a.2xlarge [1]	2500	10000	312.50	1250.00	12000	40000
c7a.4xlarge [1]	5000	10000	625.00	1250.00	20000	40000
c7a.8xlarge [2]	10000		1250.0		40000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
c7a.12xlarge [2]	15000		1875.0		60000	
c7a.16xlarge [2]	20000		2500.0		80000	
c7a.24xlarge [2]	30000		3750.0		120000	
c7a.32xlarge [2]	40000		5000.0		160000	
c7a.48xlarge [2]	40000		5000.0		240000	
c7a.metal-48xl [2]	40000		5000.0		240000	
c7g.medium [1]	315	10000	39.38	1250.00	2500	40000
c7g.large [1]	630	10000	78.75	1250.00	3600	40000
c7g.xlarge [1]	1250	10000	156.25	1250.00	6000	40000
c7g.2xlarge [1]	2500	10000	312.50	1250.00	12000	40000
c7g.4xlarge [1]	5000	10000	625.00	1250.00	20000	40000
c7g.8xlarge [2]	10000		1250.0		40000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
c7g.12xlarge [2]	15000		1875.0		60000	
c7g.16xlarge [2]	20000		2500.0		80000	
c7g.metal [2]	20000		2500.0		80000	
c7gd.medium [1]	315	10000	39.38	1250.00	2500	40000
c7gd.large [1]	630	10000	78.75	1250.00	3600	40000
c7gd.xlarge [1]	1250	10000	156.25	1250.00	6000	40000
c7gd.2xlarge [1]	2500	10000	312.50	1250.00	12000	40000
c7gd.4xlarge [1]	5000	10000	625.00	1250.00	20000	40000
c7gd.8xlarge [2]	10000		1250.0		40000	
c7gd.12xlarge [2]	15000		1875.0		60000	
c7gd.16xlarge [2]	20000		2500.0		80000	
金屬 2	20000		2500.0		80000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
c7gn.medium [1]	521	10000	65.12	1250.00	2083	40000
c7gn.large [1]	1042	10000	130.25	1250.00	4167	40000
c7gn.xlarge [1]	2083	10000	260.38	1250.00	8333	40000
c7gn.2xlarge [1]	4167	10000	520.88	1250.00	16667	40000
c7gn.4xlarge [1]	8333	10000	1041.62	1250.00	33333	40000
c7gn.8xlarge [1]	16667	20000	2083.38	2500.00	66667	80000
c7gn.12xlarge [1]	25000	30000	3125.00	3750.00	100000	120000
c7gn.16xlarge [1]	33333	40000	4166.62	5000.00	133333	160000
金屬 [1]	33333	40000	4166.62	5000.00	133333	160000
c7i.large [1]	650	10000	81.25	1250.00	3600	40000
c7i.xlarge [1]	1250	10000	156.25	1250.00	6000	40000
c7i.2xlarge [1]	2500	10000	312.50	1250.00	12000	40000
c7i.4xlarge [1]	5000	10000	625.00	1250.00	20000	40000

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
c7i.8xlarge [2]	10000		1250.0		40000	
c7i.12xlarge [2]	15000		1875.0		60000	
c7i.16xlarge [2]	20000		2500.0		80000	
c7i.24xlarge [2]	30000		3750.0		120000	
c7i.48xlarge [2]	40000		5000.0		240000	
c7i.metal-24xl [2]	30000		3750.0		120000	
c7i.metal-48xl [2]	40000		5000.0		240000	
C7-柔性. 大 1	312	10000	39.06	1250.00	2500	40000
C7-柔性. 大 1	625	10000	78.12	1250.00	3600	40000
C7-柔性版 .2 倍大 1	1250	10000	156.25	1250.00	6000	40000
C7-柔性 .4 倍大 1	2500	10000	312.50	1250.00	12000	40000
C7-柔性 .8 倍大 1	5000	10000	625.00	1250.00	20000	40000

記憶體最佳化

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
r4.large [2]	425		53.125		3000	
r4.xlarge [2]	850		106.25		6000	
r4.2xlarge [2]	1700		212.5		12000	
r4.4xlarge [2]	3500		437.5		18750	
r4.8xlarge [2]	7000		875.0		37500	
r4.16xlarge [2]	14000		1750.0		75000	
r5.large [1]	650	4750	81.25	593.75	3600	18750
r5.xlarge [1]	1150	4750	143.75	593.75	6000	18750
r5.2xlarge [1]	2300	4750	287.50	593.75	12000	18750
r5.4xlarge [2]	4750		593.75		18750	
r5.8xlarge [2]	6800		850.0		30000	
r5.12xlarge [2]	9500		1187.5		40000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
r5.16xlarge [2]	13600		1700.0		60000	
r5.24xlarge [2]	19000		2375.0		80000	
r5.metal [2]	19000		2375.0		80000	
r5a.large [1]	650	2880	81.25	360.00	3600	16000
r5a.xlarge [1]	1085	2880	135.62	360.00	6000	16000
r5a.2xlarge [1]	1580	2880	197.50	360.00	8333	16000
r5a.4xlarge [2]	2880		360.0		16000	
r5a.8xlarge [2]	4750		593.75		20000	
r5a.12xlarge [2]	6780		847.5		30000	
r5a.16xlarge [2]	9500		1187.5		40000	
r5a.24xlarge [2]	13570		1696.25		60000	
r5ad.large [1]	650	2880	81.25	360.00	3600	16000

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
r5ad.xlarge [1]	1085	2880	135.62	360.00	6000	16000
r5ad.2xlarge [1]	1580	2880	197.50	360.00	8333	16000
r5ad.4xlarge [2]	2880		360.0		16000	
r5ad.8xlarge [2]	4750		593.75		20000	
r5ad.12xlarge [2]	6780		847.5		30000	
r5ad.16xlarge [2]	9500		1187.5		40000	
r5ad.24xlarge [2]	13570		1696.25		60000	
r5b.large [1]	1250	10000	156.25	1250.00	5417	43333
r5b.xlarge [1]	2500	10000	312.50	1250.00	10833	43333
r5b.2xlarge [1]	5000	10000	625.00	1250.00	21667	43333
r5b.4xlarge [2]	10000		1250.0		43333	
r5b.8xlarge [2]	20000		2500.0		86667	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
r5b.12xlarge [2]	30000		3750.0		130000	
r5b.16xlarge [2]	40000		5000.0		173333	
r5b.24xlarge [2]	60000		7500.0		260000	
r5b.metal [2]	60000		7500.0		260000	
r5d.large [1]	650	4750	81.25	593.75	3600	18750
r5d.xlarge [1]	1150	4750	143.75	593.75	6000	18750
r5d.2xlarge [1]	2300	4750	287.50	593.75	12000	18750
r5d.4xlarge [2]	4750		593.75		18750	
r5d.8xlarge [2]	6800		850.0		30000	
r5d.12xlarge [2]	9500		1187.5		40000	
r5d.16xlarge [2]	13600		1700.0		60000	
r5d.24xlarge [2]	19000		2375.0		80000	
r5d.metal [2]	19000		2375.0		80000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
r5dn.large [1]	650	4750	81.25	593.75	3600	18750
r5dn.xlarge [1]	1150	4750	143.75	593.75	6000	18750
r5dn.2xlarge [1]	2300	4750	287.50	593.75	12000	18750
r5dn.4xlarge [2]	4750		593.75		18750	
r5dn.8xlarge [2]	6800		850.0		30000	
r5dn.12xlarge [2]	9500		1187.5		40000	
r5dn.16xlarge [2]	13600		1700.0		60000	
r5dn.24xlarge [2]	19000		2375.0		80000	
r5dn.metal [2]	19000		2375.0		80000	
r5n.large [1]	650	4750	81.25	593.75	3600	18750
r5n.xlarge [1]	1150	4750	143.75	593.75	6000	18750
r5n.2xlarge [1]	2300	4750	287.50	593.75	12000	18750

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
r5n.4xlarge [2]	4750		593.75		18750	
r5n.8xlarge [2]	6800		850.0		30000	
r5n.12xlarge [2]	9500		1187.5		40000	
r5n.16xlarge [2]	13600		1700.0		60000	
r5n.24xlarge [2]	19000		2375.0		80000	
r5n.metal [2]	19000		2375.0		80000	
r6a.large [1]	650	10000	81.25	1250.00	3600	40000
r6a.xlarge [1]	1250	10000	156.25	1250.00	6000	40000
r6a.2xlarge [1]	2500	10000	312.50	1250.00	12000	40000
r6a.4xlarge [1]	5000	10000	625.00	1250.00	20000	40000
r6a.8xlarge [2]	10000		1250.0		40000	
r6a.12xlarge [2]	15000		1875.0		60000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
r6a.16xlarge [2]	20000		2500.0		80000	
r6a.24xlarge [2]	30000		3750.0		120000	
r6a.32xlarge [2]	40000		5000.0		160000	
r6a.48xlarge [2]	40000		5000.0		240000	
r6a.metal [2]	40000		5000.0		240000	
r6g.medium [1]	315	4750	39.38	593.75	2500	20000
r6g.large [1]	630	4750	78.75	593.75	3600	20000
r6g.xlarge [1]	1188	4750	148.50	593.75	6000	20000
r6g.2xlarge [1]	2375	4750	296.88	593.75	12000	20000
r6g.4xlarge [2]	4750		593.75		20000	
r6g.8xlarge [2]	9500		1187.5		40000	
r6g.12xlarge [2]	14250		1781.25		50000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
r6g.16xlarge [2]	19000		2375.0		80000	
r6g.metal [2]	19000		2375.0		80000	
r6gd.medium [1]	315	4750	39.38	593.75	2500	20000
r6gd.large [1]	630	4750	78.75	593.75	3600	20000
r6gd.xlarge [1]	1188	4750	148.50	593.75	6000	20000
r6gd.2xlarge [1]	2375	4750	296.88	593.75	12000	20000
r6gd.4xlarge [2]	4750		593.75		20000	
r6gd.8xlarge [2]	9500		1187.5		40000	
r6gd.12xlarge [2]	14250		1781.25		50000	
r6gd.16xlarge [2]	19000		2375.0		80000	
r6gd.metal [2]	19000		2375.0		80000	
r6i.large [1]	650	10000	81.25	1250.00	3600	40000
r6i.xlarge [1]	1250	10000	156.25	1250.00	6000	40000

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
r6i.2xlarge [1]	2500	10000	312.50	1250.00	12000	40000
r6i.4xlarge [1]	5000	10000	625.00	1250.00	20000	40000
r6i.8xlarge [2]	10000		1250.0		40000	
r6i.12xlarge [2]	15000		1875.0		60000	
r6i.16xlarge [2]	20000		2500.0		80000	
r6i.24xlarge [2]	30000		3750.0		120000	
r6i.32xlarge [2]	40000		5000.0		160000	
r6i.metal [2]	40000		5000.0		160000	
r6idn.large [1]	1562	25000	195.31	3125.00	6250	100000
r6idn.xlarge [1]	3125	25000	390.62	3125.00	12500	100000
r6idn.2xlarge [1]	6250	25000	781.25	3125.00	25000	100000
r6idn.4xlarge [1]	12500	25000	1562.50	3125.00	50000	100000

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
r6idn.8xlarge [2]	25000		3125.0		100000	
r6idn.12xlarge [2]	37500		4687.5		150000	
r6idn.16xlarge [2]	50000		6250.0		200000	
r6idn.24xlarge [2]	75000		9375.0		300000	
r6idn.32xlarge [2]	100000		12500.0		400000	
r6idn.metal [2]	100000		12500.0		400000	
r6in.large [1]	1562	25000	195.31	3125.00	6250	100000
r6in.xlarge [1]	3125	25000	390.62	3125.00	12500	100000
r6in.2xlarge [1]	6250	25000	781.25	3125.00	25000	100000
r6in.4xlarge [1]	12500	25000	1562.50	3125.00	50000	100000
r6in.8xlarge [2]	25000		3125.0		100000	
r6in.12xlarge [2]	37500		4687.5		150000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
r6in.16xlarge [2]	50000		6250.0		200000	
r6in.24xlarge [2]	75000		9375.0		300000	
r6in.32xlarge [2]	100000		12500.0		400000	
r6in.metal [2]	100000		12500.0		400000	
r6id.large [1]	650	10000	81.25	1250.00	3600	40000
r6id.xlarge [1]	1250	10000	156.25	1250.00	6000	40000
r6id.2xlarge [1]	2500	10000	312.50	1250.00	12000	40000
r6id.4xlarge [1]	5000	10000	625.00	1250.00	20000	40000
r6id.8xlarge [2]	10000		1250.0		40000	
r6id.12xlarge [2]	15000		1875.0		60000	
r6id.16xlarge [2]	20000		2500.0		80000	
r6id.24xlarge [2]	30000		3750.0		120000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
r6id.32xlarge [2]	40000		5000.0		160000	
r6id.metal [2]	40000		5000.0		160000	
r7a.medium [1]	325	10000	40.62	1250.00	2500	40000
r7a.large [1]	650	10000	81.25	1250.00	3600	40000
r7a.xlarge [1]	1250	10000	156.25	1250.00	6000	40000
r7a.2xlarge [1]	2500	10000	312.50	1250.00	12000	40000
r7a.4xlarge [1]	5000	10000	625.00	1250.00	20000	40000
r7a.8xlarge [2]	10000		1250.0		40000	
r7a.12xlarge [2]	15000		1875.0		60000	
r7a.16xlarge [2]	20000		2500.0		80000	
r7a.24xlarge [2]	30000		3750.0		120000	
r7a.32xlarge [2]	40000		5000.0		160000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
r7a.48xlarge [2]	40000		5000.0		240000	
r7a.metal-48xl [2]	40000		5000.0		240000	
r7g.medium [1]	315	10000	39.38	1250.00	2500	40000
r7g.large [1]	630	10000	78.75	1250.00	3600	40000
r7g.xlarge [1]	1250	10000	156.25	1250.00	6000	40000
r7g.2xlarge [1]	2500	10000	312.50	1250.00	12000	40000
r7g.4xlarge [1]	5000	10000	625.00	1250.00	20000	40000
r7g.8xlarge [2]	10000		1250.0		40000	
r7g.12xlarge [2]	15000		1875.0		60000	
r7g.16xlarge [2]	20000		2500.0		80000	
r7g.metal [2]	20000		2500.0		80000	
r7gd.medium [1]	315	10000	39.38	1250.00	2500	40000

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
r7gd.large [1]	630	10000	78.75	1250.00	3600	40000
r7gd.xlarge [1]	1250	10000	156.25	1250.00	6000	40000
r7gd.2xlarge [1]	2500	10000	312.50	1250.00	12000	40000
r7gd.4xlarge [1]	5000	10000	625.00	1250.00	20000	40000
r7gd.8xlarge [2]	10000		1250.0		40000	
r7gd.12xlarge [2]	15000		1875.0		60000	
r7gd.16xlarge [2]	20000		2500.0		80000	
金屬 [2]	20000		2500.0		80000	
r7i.large [1]	650	10000	81.25	1250.00	3600	40000
r7i.xlarge [1]	1250	10000	156.25	1250.00	6000	40000
r7i.2xlarge [1]	2500	10000	312.50	1250.00	12000	40000
r7i.4xlarge [1]	5000	10000	625.00	1250.00	20000	40000
r7i.8xlarge [2]	10000		1250.0		40000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
r7i.12xlarge [2]	15000		1875.0		60000	
r7i.16xlarge [2]	20000		2500.0		80000	
r7i.24xlarge [2]	30000		3750.0		120000	
r7i.48xlarge [2]	40000		5000.0		240000	
r7i.metal-24xl [2]	30000		3750.0		120000	
r7i.metal-48xl [2]	40000		5000.0		240000	
r7iz.large [1]	792	10000	99.00	1250.00	3600	40000
r7iz.xlarge [1]	1584	10000	198.00	1250.00	6667	40000
r7iz.2xlarge [1]	3168	10000	396.00	1250.00	13333	40000
r7iz.4xlarge [1]	5000	10000	625.00	1250.00	20000	40000
r7iz.8xlarge [2]	10000		1250.0		40000	
r7iz.12xlarge [2]	19000		2375.0		76000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
r7iz.16xlarge [2]	20000		2500.0		80000	
r7iz.32xlarge [2]	40000		5000.0		160000	
r7iz.metal-16xl [2]	20000		2500.0		80000	
r7iz.metal-32xl [2]	40000		5000.0		160000	
均衡。中 1	315	10000	39.38	1250.00	2500	40000
大 1	630	10000	78.75	1250.00	3600	40000
大尺寸 1	1250	10000	156.25	1250.00	6000	40000
大尺寸 1 个	2500	10000	312.50	1250.00	12000	40000
大尺寸 1 只	5000	10000	625.00	1250.00	20000	40000
8 克大尺寸 2	10000		1250.0		40000	
大尺寸	15000		1875.0		60000	
六克大	20000		2500.0		80000	
8 克. 全天候大 2	30000		3750.0		120000	
大尺寸	40000		5000.0		240000	
金屬 -24 克	30000		3750.0		120000	
金屬 -48 克	40000		5000.0		240000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
u-3tb1.56xlarge [2]	19000		2375.0		80000	
u-6tb1.56xlarge [2]	38000		4750.0		160000	
u-6tb1.112xlarge [2]	38000		4750.0		160000	
u-6tb1.metal [2]	38000		4750.0		160000	
u-9tb1.112xlarge [2]	38000		4750.0		160000	
u-9tb1.metal [2]	38000		4750.0		160000	
u-12tb1.112xlarge [2]	38000		4750.0		160000	
u-12tb1.metal [2]	38000		4750.0		160000	
u-18tb1.112xlarge [2]	38000		4750.0		160000	
u-18tb1.metal [2]	38000		4750.0		160000	
u-24tb1.112xlarge [2]	38000		4750.0		160000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
u-24tb1.metal [2]	38000		4750.0		160000	
二十四千克	60000		7500.0		420000	
超大	100000		12500.0		420000	
二十四分之二	100000		12500.0		420000	
二十四分之二	100000		12500.0		420000	
x1.16xlarge [2]	7000		875.0		40000	
x1.32xlarge [2]	14000		1750.0		80000	
x2gd.medium [1]	315	4750	39.38	593.75	2500	20000
x2gd.large [1]	630	4750	78.75	593.75	3600	20000
x2gd.xlarge [1]	1188	4750	148.50	593.75	6000	20000
x2gd.2xlarge [1]	2375	4750	296.88	593.75	12000	20000
x2gd.4xlarge [2]	4750		593.75		20000	
x2gd.8xlarge [2]	9500		1187.5		40000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
x2gd.12xlarge [2]	14250		1781.25		60000	
x2gd.16xlarge [2]	19000		2375.0		80000	
x2gd.metal [2]	19000		2375.0		80000	
x2idn.16xlarge [2]	40000		5000.0		173333	
x2idn.24xlarge [2]	60000		7500.0		260000	
x2idn.32xlarge [2]	80000		10000.0		260000	
x2idn.metal [2]	80000		10000.0		260000	
x2iedn.xlarge [1]	2500	20000	312.50	2500.00	8125	65000
x2iedn.2xlarge [1]	5000	20000	625.00	2500.00	16250	65000
x2iedn.4xlarge [1]	10000	20000	1250.00	2500.00	32500	65000
x2iedn.8xlarge [2]	20000		2500.0		65000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
x2iedn.16xlarge [2]	40000		5000.0		130000	
x2iedn.24xlarge [2]	60000		7500.0		195000	
x2iedn.32xlarge [2]	80000		10000.0		260000	
x2iedn.metal [2]	80000		10000.0		260000	
x2iezn.2xlarge [2]	3170		396.25		13333	
x2iezn.4xlarge [2]	4750		593.75		20000	
x2iezn.6xlarge [2]	9500		1187.5		40000	
x2iezn.8xlarge [2]	12000		1500.0		55000	
x2iezn.12xlarge [2]	19000		2375.0		80000	
x2iezn.metal [2]	19000		2375.0		80000	
x1e.xlarge [2]	500		62.5		3700	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
x1e.2xlarge [2]	1000		125.0		7400	
x1e.4xlarge [2]	1750		218.75		10000	
x1e.8xlarge [2]	3500		437.5		20000	
x1e.16xlarge [2]	7000		875.0		40000	
x1e.32xlarge [2]	14000		1750.0		80000	
z1d.large [1]	800	3170	100.00	396.25	3333	13333
z1d.xlarge [1]	1580	3170	197.50	396.25	6667	13333
z1d.2xlarge [2]	3170		396.25		13333	
z1d.3xlarge [2]	4750		593.75		20000	
z1d.6xlarge [2]	9500		1187.5		40000	
z1d.12xlarge [2]	19000		2375.0		80000	
z1d.metal [2]	19000		2375.0		80000	

儲存最佳化

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
d2.xlarge [2]	750		93.75		6000	
d2.2xlarge [2]	1000		125.0		8000	
d2.4xlarge [2]	2000		250.0		16000	
d2.8xlarge [2]	4000		500.0		32000	
d3.xlarge [1]	850	2800	106.25	350.00	5000	15000
d3.2xlarge [1]	1700	2800	212.50	350.00	10000	15000
d3.4xlarge [2]	2800		350.0		15000	
d3.8xlarge [2]	5000		625.0		30000	
d3en.xlarge [1]	850	2800	106.25	350.00	5000	15000
d3en.2xlarge [1]	1700	2800	212.50	350.00	10000	15000
d3en.4xlarge [2]	2800		350.0		15000	
d3en.6xlarge [2]	4000		500.0		25000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
d3en.8xlarge [2]	5000		625.0		30000	
d3en.12xlarge [2]	7000		875.0		40000	
h1.2xlarge [2]	1750		218.75		12000	
h1.4xlarge [2]	3500		437.5		20000	
h1.8xlarge [2]	7000		875.0		40000	
h1.16xlarge [2]	14000		1750.0		80000	
i3.large [2]	425		53.125		3000	
i3.xlarge [2]	850		106.25		6000	
i3.2xlarge [2]	1700		212.5		12000	
i3.4xlarge [2]	3500		437.5		16000	
i3.8xlarge [2]	7000		875.0		32500	
i3.16xlarge [2]	14000		1750.0		65000	
i3.metal [2]	19000		2375.0		80000	
i3en.large [1]	576	4750	72.10	593.75	3000	20000

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
i3en.xlarge [1]	1153	4750	144.20	593.75	6000	20000
i3en.2xlarge [1]	2307	4750	288.39	593.75	12000	20000
i3en.3xlarge [1]	3800	4750	475.00	593.75	15000	20000
i3en.6xlarge [2]	4750		593.75		20000	
i3en.12xlarge [2]	9500		1187.5		40000	
i3en.24xlarge [2]	19000		2375.0		80000	
i3en.metal [2]	19000		2375.0		80000	
i4g.large [1]	625	10000	78.12	1250.00	2500	40000
i4g.xlarge [1]	1250	10000	156.25	1250.00	5000	40000
i4g.2xlarge [1]	2500	10000	312.50	1250.00	10000	40000
i4g.4xlarge [1]	5000	10000	625.00	1250.00	20000	40000
i4g.8xlarge [2]	10000		1250.0		40000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
i4g.16xlarge [2]	20000		2500.0		80000	
i4i.large [1]	625	10000	78.12	1250.00	2500	40000
i4i.xlarge [1]	1250	10000	156.25	1250.00	5000	40000
i4i.2xlarge [1]	2500	10000	312.50	1250.00	10000	40000
i4i.4xlarge [1]	5000	10000	625.00	1250.00	20000	40000
i4i.8xlarge [2]	10000		1250.0		40000	
i4i.12xlarge [2]	15000		1875.0		60000	
i4i.16xlarge [2]	20000		2500.0		80000	
i4i.24xlarge [2]	30000		3750.0		120000	
i4i.32xlarge [2]	40000		5000.0		160000	
i4i.metal [2]	40000		5000.0		160000	
im4gn.large [1]	1250	10000	156.25	1250.00	5000	40000
im4gn.xlarge [1]	2500	10000	312.50	1250.00	10000	40000

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
im4gn.2xlarge [1]	5000	10000	625.00	1250.00	20000	40000
im4gn.4xlarge [2]	10000		1250.0		40000	
im4gn.8xlarge [2]	20000		2500.0		80000	
im4gn.16xlarge [2]	40000		5000.0		160000	
is4gen.medium [1]	625	10000	78.12	1250.00	2500	40000
is4gen.large [1]	1250	10000	156.25	1250.00	5000	40000
is4gen.xlarge [1]	2500	10000	312.50	1250.00	10000	40000
is4gen.2xlarge [1]	5000	10000	625.00	1250.00	20000	40000
is4gen.4xlarge [2]	10000		1250.0		40000	
is4gen.8xlarge [2]	20000		2500.0		80000	

加速運算

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
dl1.24xlarge [2]	19000		2375.0		80000	
dl2q.24xlarge [2]	19000		2375.0		80000	
f1.2xlarge [2]	1700		212.5		12000	
f1.4xlarge [2]	3500		437.5		44000	
f1.16xlarge [2]	14000		1750.0		75000	
g3.4xlarge [2]	3500		437.5		20000	
g3.8xlarge [2]	7000		875.0		40000	
g3.16xlarge [2]	14000		1750.0		80000	
g4ad.xlarge [1]	400	3170	50.00	396.25	1700	13333
g4ad.2xlarge [1]	800	3170	100.00	396.25	3400	13333
g4ad.4xlarge [1]	1580	3170	197.50	396.25	6700	13333
g4ad.8xlarge [2]	3170		396.25		13333	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
g4ad.16xlarge [2]	6300		787.5		26667	
g4dn.xlarge [1]	950	3500	118.75	437.50	3000	20000
g4dn.2xlarge [1]	1150	3500	143.75	437.50	6000	20000
g4dn.4xlarge [2]	4750		593.75		20000	
g4dn.8xlarge [2]	9500		1187.5		40000	
g4dn.12xlarge [2]	9500		1187.5		40000	
g4dn.16xlarge [2]	9500		1187.5		40000	
g4dn.metal [2]	19000		2375.0		80000	
g5.xlarge [1]	700	3500	87.50	437.50	3000	15000
g5.2xlarge [1]	850	3500	106.25	437.50	3500	15000
g5.4xlarge [2]	4750		593.75		20000	
g5.8xlarge [2]	16000		2000.0		65000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
g5.12xlarge [2]	16000		2000.0		65000	
g5.16xlarge [2]	16000		2000.0		65000	
g5.24xlarge [2]	19000		2375.0		80000	
g5.48xlarge [2]	19000		2375.0		80000	
g5g.xlarge [1]	1188	4750	148.50	593.75	6000	20000
g5g.2xlarge [1]	2375	4750	296.88	593.75	12000	20000
g5g.4xlarge [2]	4750		593.75		20000	
g5g.8xlarge [2]	9500		1187.5		40000	
g5g.16xlarge [2]	19000		2375.0		80000	
g5g.metal [2]	19000		2375.0		80000	
大 [1]	1000	5000	125.00	625.00	4000	20000
大尺寸	2000	5000	250.00	625.00	8000	20000
大尺寸	8000		1000.0		32000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
大尺寸	16000		2000.0		64000	
大尺寸	20000		2500.0		80000	
大尺寸	20000		2500.0		80000	
大尺寸	30000		3750.0		120000	
大尺寸	60000		7500.0		240000	
大尺寸 1	1000	5000	125.00	625.00	4000	20000
G6. 2 大 1 號	2000	5000	250.00	625.00	8000	20000
G6. 4 倍大 2	8000		1000.0		32000	
G6. 8 倍大 2	16000		2000.0		64000	
G6. 12 倍大	20000		2500.0		80000	
G6. 16 大	20000		2500.0		80000	
G6. 全天候 2	30000		3750.0		120000	
G6. 48 倍大 2	60000		7500.0		240000	
玻璃大 2	8000		1000.0		32000	
大尺寸 2	16000		2000.0		64000	
inf1.xlarge [1]	1190	4750	148.75	593.75	4000	20000
inf1.2xlarge [1]	1190	4750	148.75	593.75	6000	20000

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
inf1.6xlarge [2]	4750		593.75		20000	
inf1.24xlarge [2]	19000		2375.0		80000	
inf2.xlarge [1]	1250	10000	156.25	1250.00	6000	40000
inf2.8xlarge [2]	10000		1250.0		40000	
inf2.24xlarge [2]	30000		3750.0		120000	
inf2.48xlarge [2]	60000		7500.0		240000	
p2.xlarge [2]	750		93.75		6000	
p2.8xlarge [2]	5000		625.0		32500	
p2.16xlarge [2]	10000		1250.0		65000	
p3.2xlarge [2]	1750		218.75		10000	
p3.8xlarge [2]	7000		875.0		40000	
p3.16xlarge [2]	14000		1750.0		80000	

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
p3dn.24xlarge [2]	19000		2375.0		80000	
p4d.24xlarge [2]	19000		2375.0		80000	
p4de.24xlarge [2]	19000		2375.0		80000	
p5.48xlarge [2]	80000		10000.0		260000	
trn1.2xlarge [1]	5000	20000	625.00	2500.00	16250	65000
trn1.32xlarge [2]	80000		10000.0		260000	
trn1n.32xlarge [2]	80000		10000.0		260000	
vt1.3xlarge [1]	2375	4750	296.88	593.75	10000	20000
vt1.6xlarge [2]	4750		593.75		20000	
vt1.24xlarge [2]	19000		2375.0		80000	

高效能運算

執行個體大小	基準頻寬 (Mbps)	最高頻寬 (Mbps)	基準輸送量 (MB/s, 128 KB I/O)	最大傳輸量 (MB/s、128 KiB I/O)	基準線 IOPS (16 KiB I/O)	最大值 IOPS (16 KiB I/O)
hpc6a.48xlarge [1]	87	2085	10.88	260.62	500	11000
hpc6id.32xlarge [1]	87	2085	10.88	260.62	500	11000
hpc7a.12xlarge [1]	87	2085	10.88	260.62	500	11000
hpc7a.24xlarge [1]	87	2085	10.88	260.62	500	11000
hpc7a.48xlarge [1]	87	2085	10.88	260.62	500	11000
hpc7a.96xlarge [1]	87	2085	10.88	260.62	500	11000
hpc7g.4xlarge [1]	87	2085	10.88	260.62	500	11000
hpc7g.8xlarge [1]	87	2085	10.88	260.62	500	11000
hpc7g.16xlarge [1]	87	2085	10.88	260.62	500	11000

EBS支援最佳化

下列執行個體類型支援EBS最佳EBS化，但預設不會啟用最佳化。您必須在啟動期間或之後支付額外的小時費用啟用EBS最佳化，才能達到所述EBS效能等級。

執行個體大小	最高頻寬 (Mbps)	最大傳輸量 (MB/s、128 KiB I/O)	最大值 IOPS (16 KiB I/O)
c1.xlarge	1000	125.0	8000
c3.xlarge	500	62.5	4000
c3.2xlarge	1000	125.0	8000
c3.4xlarge	2000	250.0	16000
i2.xlarge	500	62.5	4000
i2.2xlarge	1000	125.0	8000
i2.4xlarge	2000	250.0	16000
m1.large	500	62.5	4000
m1.xlarge	1000	125.0	8000
m2.2xlarge	500	62.5	4000
m2.4xlarge	1000	125.0	8000
m3.xlarge	500	62.5	4000
m3.2xlarge	1000	125.0	8000
r3.xlarge	500	62.5	4000
r3.2xlarge	1000	125.0	8000
r3.4xlarge	2000	250.0	16000

Note

`i2.8xlarge`、`c3.8xlarge`、和`r3.8xlarge`執行個體沒有專用EBS頻寬，因此不提供EBS最佳化功能。在這些執行個體上，網路流量和 Amazon EBS 流量共用相同的 10 GB 網路界面。

獲得最大的 Amazon EBS 優化性能

執行個體的EBS效能受到執行個體類型的效能限制，或其連接磁碟區的彙總效能 (以較小者為準) 所限制。若要達到最大EBS效能，執行個體必須具有連接的磁碟區，這些磁碟區必須提供等於或大於最大執行個體效能的組合效能。例如，執行個體必須至少佈建16,000IOPS每個磁碟區 (5gp35磁碟區 x 16,000 IOPS = 80,000IOPS)r6i.16xlarge，才能達成80,000IOPS對應的執行個體。我們建議您選擇提供比應用程式需求更專用 Amazon EBS 輸送量的執行個體類型；否則，Amazon EBS 和 Amazon 之間的連線EC2可能會成為效能瓶頸。

您可以使用 EBSIOBalance% 和 EBSByteBalance% 指標來協助您判斷執行個體的大小是否正確。您可以在 CloudWatch 主控台中檢視這些指標，並設定根據您指定的閾值觸發的警示。這些指標以百分比表示。維持低平衡百分比的執行個體視為有待擴展規模。平衡百分比從未低於 100% 的執行個體視為有待縮減規模。如需詳細資訊，請參閱使用監控執行個體 CloudWatch。

記憶體增強型執行個體專為在雲端執行大型記憶體內資料庫而設計，包括SAPHANA記憶體內資料庫的生產部署。若要最大化EBS效能，請使用具有偶數io1或具有相同佈建效能的io2磁碟區的高記憶體執行個體。例如，對於IOPS繁重的工作負載，請使用四個io1或已佈建 40,000 個io2磁碟區IOPS來取得最大 160,000 個執行IOPS個體。同樣地，對於輸送量繁重的工作負載，請使用 6 個io1或已佈建 48,000 的io2磁碟區，IOPS以取得最大 4,750 MB/s 輸送量。如需其他建議，請參閱的儲存區組態SAPHANA。

考量事項

- 在 2020 年 2 月 26 日之後推出的 G4dn、i3en、資料 1、M5a、R5a、R5a、T3、T3A 和 Z1D 執行個體可提供最大的最佳化效能。EBS若要從 2020 年 2 月 26 日之前啟動的執行個體獲得最大效能，請停止並啟動該執行個體。
- 在 2019 年 12 月 3 日之後推出的 C5、C5d、C5n、M5、M5d、M5n、M5n、M5n、M5n、M5n、M5n、M5n、M5n、M5dn 執行個體可提供最大的最佳化效能。EBS若要從 2019 年 12 月 3 日之前啟動的執行個體獲得最大效能，請停止並啟動該執行個體。
- u-6tb1.metalu-9tb1.metal、和在 2020 年 3 月 12 日之後啟動的u-12tb1.metal執行個體可提供最大的最EBS佳化效能。2020 年 3 月 12 日之前推出的這些類型執行個體可能會提供較低的效能。若要從 2020 年 3 月 12 日之前啟動的執行個體獲得最大效能，請聯絡您的客戶團隊以免費升級該執行個體。

尋找 Amazon EBS 優化的 Amazon EC2 執行個體

您可以使用檢 AWS CLI 視目前區域中支援EBS最佳化的執行個體類型。

尋找預設為 Amazon EBS 優化的執行個體類型

使用下列 describe-instance-types 命令。如果在 Windows 命令提示字元中執行此命令，請以 ^ 字元取代\ 行接續字元。

```
aws ec2 describe-instance-types \
--query 'InstanceTypes[].{InstanceType:InstanceType,"MaxBandwidth(Mb/
s)":EbsInfo.EbsOptimizedInfo.MaximumBandwidthInMbps,MaxIOPS:EbsInfo.EbsOptimizedInfo.MaximumIop
s)":EbsInfo.EbsOptimizedInfo.MaximumThroughputInMBps}' \
--filters Name=ebs-info.ebs-optimized-support,Values=default --output=table
```

`eu-west-1` 的輸出範例：

```
-----------------------------------------------------------------------------
|                            DescribeInstanceTypes                          |
+-------------------+---------------------+-----------+-----------------------+
|   InstanceType    | MaxBandwidth(Mb/s)  |  MaxIOPS  |  MaxThroughput(MB/s)  |
+-------------------+---------------------+-----------+-----------------------+
|  m5dn.8xlarge     |  6800               |  30000    |  850.0                |
|  m6gd.xlarge      |  4750               |  20000    |  593.75               |
|  c4.4xlarge       |  2000               |  16000    |  250.0                |
|  r4.16xlarge      |  14000              |  75000    |  1750.0               |
|  m5ad.large       |  2880               |  16000    |  360.0                |
...
```

尋找選擇性支援 Amazon EBS 最佳化的執行個體類型

使用下列 describe-instance-types 命令。

```
aws ec2 describe-instance-types \
--query 'InstanceTypes[].{InstanceType:InstanceType,"MaxBandwidth(Mb/
s)":EbsInfo.EbsOptimizedInfo.MaximumBandwidthInMbps,MaxIOPS:EbsInfo.EbsOptimizedInfo.MaximumIop
s)":EbsInfo.EbsOptimizedInfo.MaximumThroughputInMBps}' \
--filters Name=ebs-info.ebs-optimized-support,Values=supported --output=table
```

`eu-west-1` 的輸出範例：

```
---------------------------------------------------------------------------
|                         DescribeInstanceTypes                           |
+--------------+---------------------+----------+-----------------------+
| InstanceType | MaxBandwidth(Mb/s)  | MaxIOPS  |  MaxThroughput(MB/s)  |
```

```
+--------------+----------------------+----------+-----------------------+
|  i2.2xlarge  |  1000                |  8000    |  125.0                |
|  m2.4xlarge  |  1000                |  8000    |  125.0                |
|  m2.2xlarge  |  500                 |  4000    |  62.5                 |
|  c1.xlarge   |  1000                |  8000    |  125.0                |
|  i2.xlarge   |  500                 |  4000    |  62.5                 |
|  m3.xlarge   |  500                 |  4000    |  62.5                 |
|  m1.xlarge   |  1000                |  8000    |  125.0                |
|  r3.4xlarge  |  2000                |  16000   |  250.0                |
|  r3.2xlarge  |  1000                |  8000    |  125.0                |
|  c3.xlarge   |  500                 |  4000    |  62.5                 |
|  m3.2xlarge  |  1000                |  8000    |  125.0                |
|  r3.xlarge   |  500                 |  4000    |  62.5                 |
|  i2.4xlarge  |  2000                |  16000   |  250.0                |
|  c3.4xlarge  |  2000                |  16000   |  250.0                |
|  c3.2xlarge  |  1000                |  8000    |  125.0                |
|  m1.large    |  500                 |  4000    |  62.5                 |
+--------------+----------------------+----------+-----------------------+
```

為 Amazon 執行個體啟用 EC2 Amazon EBS 優化

您只能EBS針對選擇性支援 Amazon EBS 優化但預設不是 Amazon EBS 優化的執行個體類型手動啟用 Amazon 優化。對於這些執行個體類型，您可以在啟動期間或之後啟用 Amazon EBS 優化，但需額外支付小時費用。

Console

在啟動期間啟用 Amazon EBS 優化

在 [啟動執行個體] 精靈中，選取所需的執行個體類型。展開 [進階詳細資料] 區段，然後針對EBS最佳化執行個體選取 [啟用]。

如果選取的執行個體類型不支援 Amazon EBS 優化，則會停用下拉式清單。如果執行個體類型預設為 Amazon EBS 最佳化，則表示已選取「啟用」。

啟動後啟用 Amazon EBS 優化

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選取 Instances (執行個體)，然後選取該執行個體。
3. 停止執行個體。依序選擇 Actions (動作)、Instance state (執行個體狀態)、Stop instance (停止執行個體)。

> **Warning**
>
> 停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。若要保留執行個體存放區磁碟區的資料，請將資料備份至持久性儲存。

4. 在仍然選取執行個體的情況下，選取 Actions (動作)、Instance settings (執行個體設定)、Change instance type (變更執行個體類型)。
5. 選取 [EBS最佳化]，然後選擇 [套用]。

   如果執行個體類型預設為 Amazon EBS 優化，或者不支援 Amazon EBS 優化，則會停用該核取方塊。
6. 重新啟動執行個體。選擇 Instance state (執行個體狀態)、Start instance (啟動執行個體)。

Command line

在啟動期間啟用 Amazon EBS 優化

您可以使用下列其中一個命令，搭配對應的選項。

- run-instances 搭配 `--ebs-optimized` (AWS CLI)
- New-EC2Instance與`-EbsOptimized`（AWS Tools for Windows PowerShell）

啟動後啟用 Amazon EBS 優化

1. 如果執行個體正在執行，請使用下列其中一個指令來停止執行個體。
   - stop-instances (AWS CLI)
   - Stop-EC2Instance (AWS Tools for Windows PowerShell)

> **Warning**
>
> 停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。若要保留執行個體存放區磁碟區的資料，請將資料備份至持久性儲存。

2. 使用下列其中一個指令搭配對應選項來啟用EBS最佳化：
   - modify-instance-attribute與`--ebs-optimized`（AWS CLI）

- Edit-EC2InstanceAttribute與`-EbsOptimized`（AWS Tools for Windows PowerShell）

CPUAmazon EC2 實例的選項

許多 Amazon EC2 執行個體都支援同步多執行緒 (SMT)，可讓多個執行緒在單CPU一核心上同時執行。每個執行緒在執行個體上都會以虛擬 CPU (vCPU) 表示。執行個體具有預設的CPU核心數量，根據執行個體類型而有所不同。例如，`m5.xlarge`執行個體類型預設為每個CPU核心兩個核心和兩個執行緒，總共有四 vCPUs 個。

> **Note**
>
> 每個 v 都CPU是CPU核心的執行緒，但 T2 執行個體、M7a 執行個體、Apple 晶片 Mac 執行個體，以及 64 位元ARM平台，例如採用 AWS Graviton 處理器驅動的執行個體除外。

在大多數情況下，Amazon EC2 執行個體類型具有記憶體和數量的組合，可滿足您 vCPUs 的工作負載需求。不過，您可以在執行個體啟動期間指定下列CPU選項，以針對特定工作負載或業務需求最佳化執行個體：

- CPU核心數：您可以自訂執行個體的CPU核心數目。如果執行個體具有足夠數量處理記憶體密集型工作負載但核心較少CPU的執行個體，您可能會這麼做，將軟體的RAM授權成本最佳化。
- 每個核心的執行緒：您可以SMT透過指定每個CPU核心的單一執行緒來停用。您可以針對特定工作負載執行此動作，例如高效能運算 (HPC) 工作負載。

定價

指定CPU選項無需額外或減少費用。您的費用與使用預設CPU選項啟動的執行個體相同。

目錄

- 為 Amazon EC2 執行個體指定CPU選項的規則
- Amazon EC2 執行個體類型支援的CPU選項
- 指CPU定 Amazon EC2 執行個體的選項
- 檢視 Amazon EC2 執行個體的執行CPU緒和核心

為 Amazon EC2 執行個體指定CPU選項的規則

若要指定執行個體的CPU選項，請注意下列規則：

- 您無法指定裸機執行個體的CPU選項。
- CPU選項只能在執行個體啟動期間指定，而且在啟動後無法修改。
- 啟動執行個體時，您必須同時指定要求中每個CPU核心的核心數目和執行緒數目。如需請求範例，請參閱指CPU定 Amazon EC2 執行個體的選項。
- 執行個體 vCPUs 的數目是核心數CPU乘以每個核心的執行緒。若要指定的自訂數目vCPUs，您必須為執行個體類型指定每個CPU核心的有效核心和執行緒數目。您不能超過執行個體的 vCPUs 預設數目。如需詳細資訊，請參閱Amazon EC2 執行個體類型支援的CPU選項。
- 若要停用同步多執行緒 (SMT) (也稱為「超執行緒」)，請為每個核心指定一個執行緒。
- 當您變更現有例證的例證類型時，選CPU項會自動變更為新例證類型的預設CPU選項。
- 在您停止、啟動或重新啟動執行個體之後，指定的CPU選項會持續存在。

Amazon EC2 執行個體類型支援的CPU選項

下表列出支援指定CPU選項的執行個體類型。

目錄

- 一般用途執行個體
- 運算最佳化執行個體
- 記憶體最佳化執行個體
- 儲存最佳化執行個體
- 加速運算執行個體
- 高效能運算執行個體

一般用途執行個體

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m2.xlarge	2	2	1	1、2	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m2.2xlarge	4	4	1	1、2、3、4	1
m2.4xlarge	8	8	1	1、2、3、4、5、6、7、8	1
m3.large	2	1	2	1	1、2
m3.xlarge	4	2	2	1、2	1、2
m3.2xlarge	8	4	2	1、2、3、4	1、2
m4.large	2	1	2	1	1、2
m4.xlarge	4	2	2	1、2	1、2
m4.2xlarge	8	4	2	1、2、3、4	1、2
m4.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
m4.10xlarge	40	20	2	2、4、6、8、10、12、14、16、18、20	1、2
m4.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
m5.large	2	1	2	1	1、2
m5.xlarge	4	2	2	2	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m5.2xlarge	8	4	2	2、4	1、2
m5.4xlarge	16	8	2	2、4、6、8	1、2
m5.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
m5.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
m5.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
m5.24xlarge	96	48	2	4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
m5a.large	2	1	2	1	1、2
m5a.xlarge	4	2	2	2	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m5a.2xlarge	8	4	2	2、4	1、2
m5a.4xlarge	16	8	2	2、4、6、8	1、2
m5a.8xlarge	32	16	2	4、6、8、10、12、14、16	1、2
m5a.12xlarge	48	24	2	6、12、18、24	1、2
m5a.16xlarge	64	32	2	8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
m5a.24xlarge	96	48	2	12、18、24、36、48	1、2
m5ad.large	2	1	2	1	1、2
m5ad.xlarge	4	2	2	2	1、2
m5ad.2xlarge	8	4	2	2、4	1、2
m5ad.4xlarge	16	8	2	2、4、6、8	1、2
m5ad.8xlarge	32	16	2	4、6、8、10、12、14、16	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m5ad.12xlarge	48	24	2	6、12、18、24	1、2
m5ad.16xlarge	64	32	2	8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
m5ad.24xlarge	96	48	2	12、18、24、36、48	1、2
m5d.large	2	1	2	1	1、2
m5d.xlarge	4	2	2	2	1、2
m5d.2xlarge	8	4	2	2、4	1、2
m5d.4xlarge	16	8	2	2、4、6、8	1、2
m5d.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
m5d.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m5d.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
m5d.24xlarge	96	48	2	4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
m5dn.large	2	1	2	1	1、2
m5dn.xlarge	4	2	2	1、2	1、2
m5dn.2xlarge	8	4	2	2、4	1、2
m5dn.4xlarge	16	8	2	2、4、6、8	1、2
m5dn.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
m5dn.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m5dn.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
m5dn.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
m5n.large	2	1	2	1	1、2
m5n.xlarge	4	2	2	1、2	1、2
m5n.2xlarge	8	4	2	2、4	1、2
m5n.4xlarge	16	8	2	2、4、6、8	1、2
m5n.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
m5n.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m5n.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
m5n.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
m5zn.large	2	1	2	1	1、2
m5zn.xlarge	4	2	2	1、2	1、2
m5zn.2xlarge	8	4	2	2、4	1、2
m5zn.3xlarge	12	6	2	2、4、6	1、2
m5zn.6xlarge	24	12	2	2、4、6、8、10、12	1、2
m5zn.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m6a.large	2	1	2	1	1、2
m6a.xlarge	4	2	2	1、2	1、2
m6a.2xlarge	8	4	2	1、2、3、4	1、2
m6a.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
m6a.8xlarge	32	16	2	4、6、8、10、12、14、16	1、2
m6a.12xlarge	48	24	2	一、二、三、四、五、六、七、八、十六	1、2
m6a.16xlarge	64	32	2	四、六、八、十、十、十、十四、十六、三	1、2
m6a.24xlarge	96	48	2	四、六、八、十、十、十、十四、十六、三十八	1、2
m6a.32xlarge	128	64	2	八、十六、二十、二十、二十、二十四	1、2
m6a.48xlarge	192	96	2	八、十六、二十、二十、二十、二、三十六	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m6g.large	2	2	1	1、2	1
m6g.xlarge	4	4	1	1、2、3、4	1
m6g.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1
m6g.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
m6g.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m6g.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m6g.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
m6gd.large	2	2	1	1、2	1
m6gd.xlarge	4	4	1	1、2、3、4	1
m6gd.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m6gd.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
m6gd.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1
m6gd.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m6gd.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
m6i.large	2	1	2	1	1、2
m6i.xlarge	4	2	2	1、2	1、2
m6i.2xlarge	8	4	2	2、4	1、2
m6i.4xlarge	16	8	2	2、4、6、8	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m6i.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
m6i.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
m6i.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
m6i.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m6i.32xlarge	128	64	2	二、四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四十六、四十、四、五十五、五十五、五十五、五十六、六十六、六十六、64	1、2
m6id.large	2	1	2	1	1、2
m6id.xlarge	4	2	2	1、2	1、2
m6id.2xlarge	8	4	2	2、4	1、2
m6id.4xlarge	16	8	2	2、4、6、8	1、2
m6id.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m6id.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
m6id.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
m6id.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m6id.32xlarge	128	64	2	二、四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四十六、四十、四、五十五、五十五、五十五、五十六、六十六、六十六、64	1、2
m6idn.large	2	1	2	1	1、2
m6idn.xlarge	4	2	2	1、2	1、2
m6idn.2xlarge	8	4	2	1、2、3、4	1、2
m6idn.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
m6idn.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m6idn.12xlarge	48	24	2	一、二、三、四、五、六、七、八、九、十、十、十、十、十、十、十、十、十、十、十、十、十、十、十、二、二、二、二、二、二、24	1、2
m6idn.16xlarge	64	32	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1、2
m6idn.24xlarge	96	48	2	4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m6idn.32xlarge	128	64	2	四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四十、四十、四十、五十五、五十五、五十五、五十六、六、六、六、六、64	1、2
m6in.large	2	1	2	1	1、2
m6in.xlarge	4	2	2	1、2	1、2
m6in.2xlarge	8	4	2	1、2、3、4	1、2
m6in.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
m6in.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m6in.12xlarge	48	24	2	一、二、三、四、五、六、七、八、九、十、十、十、十、十、十、十、十、十、十、十、十、十、十、十、二、二、二、二、二、二、24	1、2
m6in.16xlarge	64	32	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1、2
m6in.24xlarge	96	48	2	4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m6in.32xlarge	128	64	2	四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四十、四十、四十、五十五、五十五、五十五、五十六、六、六、六、六、64	1、2
m7a.large	2	2	1	1、2	1
m7a.xlarge	4	4	1	1、2、3、4	1
m7a.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1
m7a.4xlarge	16	16	1	一、二、四、六、八、十、十、十、十、十四、十六	1
m7a.8xlarge	32	32	1	1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m7a.12xlarge	48	48	1	一、二、三、四、五、六、十二、二、二、二、三十、四、四、四十八	1
m7a.16xlarge	64	64	1	一、二、三、四、五、六、七、八、十六、二、三十、四十、四十八、五十六	1
m7a.24xlarge	96	96	1	一、二、三、四、五、六、七、八、九、十、十、十、十、二、二、二、三十六、六、七、八四、九六	1
m7a.32xlarge	128	128	1	四、六、八、十、十、十、十四、十六、四、四、六、八、八	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m7a.48xlarge	192	192	1	四、六、八、十、十、十、十四、十六、十、二十、二十、二十、二、六、四、168、192	1
m7g.large	2	2	1	1、2	1
m7g.xlarge	4	4	1	1、2、3、4	1
m7g.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1
m7g.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
m7g.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m7g.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m7g.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
m7gd.large	2	2	1	1、2	1
m7gd.xlarge	4	4	1	1、2、3、4	1
m7gd.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m7gd.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
m7gd.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1
m7gd.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m7gd.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
m7i.large	2	1	2	1	1、2
m7i.xlarge	4	2	2	1、2	1、2
m7i.2xlarge	8	4	2	1、2、3、4	1、2
m7i.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m7i.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2
m7i.12xlarge	48	24	2	一、二、三、四、五、六、七、八、九、十、十、十、十、十、十、十、十、十、十、十、十、十、十、十、二、二、二、二、二、二、24	1、2
m7i.16xlarge	64	32	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m7i.24xlarge	96	48	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m7i.48xlarge	192	96	2	四、六、八、十、十、十四、十六、二十、二十、二十、二、二、二、三十、三十、三十、四、四、四、四、四、五十、五十五、五十六、六、六、六、六、六、六、六、六、六、七、七、七、六、七、六、七、六、七、六、七、六、七、六、七、六、七、六、七、六、七、六、七、七、六、七、六、七、七、六、七、六	1、2
m7i-flex.large	2	1	2	1	1、2
m7i-flex.xlarge	4	2	2	1、2	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
m7i-flex.2xlarge	8	4	2	1、2、3、4	1、2
m7i-flex.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
m7i-flex.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2
t3.nano	2	1	2	1	1、2
t3.micro	2	1	2	1	1、2
t3.small	2	1	2	1	1、2
t3.medium	2	1	2	1	1、2
t3.large	2	1	2	1	1、2
t3.xlarge	4	2	2	2	1、2
t3.2xlarge	8	4	2	2、4	1、2
t3a.nano	2	1	2	1	1、2
t3a.micro	2	1	2	1	1、2
t3a.small	2	1	2	1	1、2
t3a.medium	2	1	2	1	1、2
t3a.large	2	1	2	1	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
t3a.xlarge	4	2	2	2	1、2
t3a.2xlarge	8	4	2	2、4	1、2
t4g.nano	2	2	1	1、2	1
t4g.micro	2	2	1	1、2	1
t4g.small	2	2	1	1、2	1
t4g.medium	2	2	1	1、2	1
t4g.large	2	2	1	1、2	1
t4g.xlarge	4	4	1	1、2、3、4	1
t4g.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1

運算最佳化執行個體

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c3.large	2	1	2	1	1、2
c3.xlarge	4	2	2	1、2	1、2
c3.2xlarge	8	4	2	1、2、3、4	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c3.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
c3.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
c4.large	2	1	2	1	1、2
c4.xlarge	4	2	2	1、2	1、2
c4.2xlarge	8	4	2	1、2、3、4	1、2
c4.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
c4.8xlarge	36	18	2	2、4、6、8、10、12、14、16、18	1、2
c5.large	2	1	2	1	1、2
c5.xlarge	4	2	2	2	1、2
c5.2xlarge	8	4	2	2、4	1、2
c5.4xlarge	16	8	2	2、4、6、8	1、2
c5.9xlarge	36	18	2	2、4、6、8、10、12、14、16、18	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c5.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
c5.18xlarge	72	36	2	4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36	1、2
c5.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
c5a.large	2	1	2	1	1、2
c5a.xlarge	4	2	2	1、2	1、2
c5a.2xlarge	8	4	2	1、2、3、4	1、2
c5a.4xlarge	16	8	2	1, 2, 3, 4, 8	1、2
c5a.8xlarge	32	16	2	1, 2, 3, 4, 8, 12, 16	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c5a.12xlarge	48	24	2	1, 2, 3, 4, 8, 12, 16, 20, 24	1、2
c5a.16xlarge	64	32	2	1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32	1、2
c5a.24xlarge	96	48	2	1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48	1、2
c5ad.large	2	1	2	1	1、2
c5ad.xlarge	4	2	2	1、2	1、2
c5ad.2xlarge	8	4	2	1、2、3、4	1、2
c5ad.4xlarge	16	8	2	1, 2, 3, 4, 8	1、2
c5ad.8xlarge	32	16	2	1, 2, 3, 4, 8, 12, 16	1、2
c5ad.12xlarge	48	24	2	1, 2, 3, 4, 8, 12, 16, 20, 24	1、2
c5ad.16xlarge	64	32	2	1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c5ad.24xlarge	96	48	2	1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48	1、2
c5d.large	2	1	2	1	1、2
c5d.xlarge	4	2	2	2	1、2
c5d.2xlarge	8	4	2	2、4	1、2
c5d.4xlarge	16	8	2	2、4、6、8	1、2
c5d.9xlarge	36	18	2	2、4、6、8、10、12、14、16、18	1、2
c5d.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
c5d.18xlarge	72	36	2	4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c5d.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
c5n.large	2	1	2	1	1、2
c5n.xlarge	4	2	2	2	1、2
c5n.2xlarge	8	4	2	2、4	1、2
c5n.4xlarge	16	8	2	2、4、6、8	1、2
c5n.9xlarge	36	18	2	2、4、6、8、10、12、14、16、18	1、2
c5n.18xlarge	72	36	2	4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36	1、2
c6a.large	2	1	2	1	1、2
c6a.xlarge	4	2	2	1、2	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c6a.2xlarge	8	4	2	1、2、3、4	1、2
c6a.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
c6a.8xlarge	32	16	2	4、6、8、10、12、14、16	1、2
c6a.12xlarge	48	24	2	一、二、三、四、五、六、七、八、十六	1、2
c6a.16xlarge	64	32	2	四、六、八、十、十、十、十四、十六、三	1、2
c6a.24xlarge	96	48	2	四、六、八、十、十、十、十四、十六、三十八	1、2
c6a.32xlarge	128	64	2	八、十六、二十、二十、二十、二十四	1、2
c6a.48xlarge	192	96	2	八、十六、二十、二十、二十、二、三十六	1、2
c6g.large	2	2	1	1、2	1
c6g.xlarge	4	4	1	1、2、3、4	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c6g.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1
c6g.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
c6g.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c6g.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c6g.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
c6gd.large	2	2	1	1、2	1
c6gd.xlarge	4	4	1	1、2、3、4	1
c6gd.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c6gd.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
c6gd.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1
c6gd.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c6gd.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
c6gn.medium	1	1	1	1	1
c6gn.large	2	2	1	1、2	1
c6gn.xlarge	4	4	1	1、2、3、4	1
c6gn.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c6gn.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
c6gn.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1
c6gn.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c6gn.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
c6i.large	2	1	2	1	1、2
c6i.xlarge	4	2	2	1、2	1、2
c6i.2xlarge	8	4	2	2、4	1、2
c6i.4xlarge	16	8	2	2、4、6、8	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c6i.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
c6i.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
c6i.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
c6i.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c6i.32xlarge	128	64	2	二、四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四十六、四十、四、五十五、五十五、五十五、五十六、六十六、六十六、64	1、2
c6id.large	2	1	2	1	1、2
c6id.xlarge	4	2	2	1、2	1、2
c6id.2xlarge	8	4	2	2、4	1、2
c6id.4xlarge	16	8	2	2、4、6、8	1、2
c6id.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c6id.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
c6id.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
c6id.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c6id.32xlarge	128	64	2	二、四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四十六、四十、四、五十五、五十五、五十五、五十六、六十六、六十六、64	1、2
c6in.large	2	1	2	1	1、2
c6in.xlarge	4	2	2	1、2	1、2
c6in.2xlarge	8	4	2	1、2、3、4	1、2
c6in.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
c6in.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c6in.12xlarge	48	24	2	一、二、三、四、五、六、七、八、九、十、十、十、十、十、十、十、十、十、十、十、十、十、十、十、二、二、二、二、二、二、24	1、2
c6in.16xlarge	64	32	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1、2
c6in.24xlarge	96	48	2	4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c6in.32xlarge	128	64	2	四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四十、四十、四十、五十五、五十五、五十五、五十六、六、六、六、六、64	1、2
c7a.large	2	2	1	1、2	1
c7a.xlarge	4	4	1	1、2、3、4	1
c7a.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1
c7a.4xlarge	16	16	1	一、二、四、六、八、十、十、十、十、十四、十六	1
c7a.8xlarge	32	32	1	1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c7a.12xlarge	48	48	1	一、二、三、四、五、六、十二、二、二、二、三十、四、四、四十八	1
c7a.16xlarge	64	64	1	一、二、三、四、五、六、七、八、十六、二、三十、四十、四十八、五十六	1
c7a.24xlarge	96	96	1	一、二、三、四、五、六、七、八、九、十、十、十、十、二、二、二、三十六、六、七、八四、九六	1
c7a.32xlarge	128	128	1	四、六、八、十、十、十、十四、十六、四、四、六、八、八	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c7a.48xlarge	192	192	1	四、六、八、十、十、十、十四、十六、十、二十、二十、二十、二、六、四、168、192	1
c7g.large	2	2	1	1、2	1
c7g.xlarge	4	4	1	1、2、3、4	1
c7g.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1
c7g.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
c7g.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c7g.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c7g.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
c7gd.large	2	2	1	1、2	1
c7gd.xlarge	4	4	1	1、2、3、4	1
c7gd.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c7gd.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
c7gd.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1
c7gd.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c7gd.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
c7gn.large	2	2	1	1、2	1
c7gn.xlarge	4	4	1	1、2、3、4	1
c7gn.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c7gn.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
c7gn.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1
c7gn.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c7gn.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
c7i.large	2	1	2	1	1、2
c7i.xlarge	4	2	2	1、2	1、2
c7i.2xlarge	8	4	2	1、2、3、4	1、2
c7i.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c7i.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2
c7i.12xlarge	48	24	2	一、二、三、四、五、六、七、八、九、十、十、十、十、十、十、十、十、十、十、十、十、十、十、十、二、二、二、二、二、二、24	1、2
c7i.16xlarge	64	32	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c7i.24xlarge	96	48	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c7i.48xlarge	192	96	2	四、六、八、十、十、十四、十六、二十、二十、二十、二、二、二、三十、三十、三十、四、四、四、四、四、四、五十、五十五、五十六、六、六、六、六、六、六、六、六、六、六、七、七、七、六、七、六、七、六、七、六、七、六、七、六、七、六、七、六、七、六、七、六、七、六、七、七、六、七、六、七、七、六、七、六	1、2
c7i-flex.large	2	1	2	1	1、2
c7i-flex.xlarge	4	2	2	1、2	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
c7i-flex.2xlarge	8	4	2	1、2、3、4	1、2
c7i-flex.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
c7i-flex.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2

記憶體最佳化執行個體

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r3.large	2	1	2	1	1、2
r3.xlarge	4	2	2	1、2	1、2
r3.2xlarge	8	4	2	1、2、3、4	1、2
r3.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
r3.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
r4.large	2	1	2	1	1、2
r4.xlarge	4	2	2	1、2	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r4.2xlarge	8	4	2	1、2、3、4	1、2
r4.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
r4.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2
r4.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
r5.large	2	1	2	1	1、2
r5.xlarge	4	2	2	2	1、2
r5.2xlarge	8	4	2	2、4	1、2
r5.4xlarge	16	8	2	2、4、6、8	1、2
r5.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
r5.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r5.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
r5.24xlarge	96	48	2	4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
r5a.large	2	1	2	1	1、2
r5a.xlarge	4	2	2	2	1、2
r5a.2xlarge	8	4	2	2、4	1、2
r5a.4xlarge	16	8	2	2、4、6、8	1、2
r5a.8xlarge	32	16	2	4、6、8、10、12、14、16	1、2
r5a.12xlarge	48	24	2	6、12、18、24	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r5a.16xlarge	64	32	2	8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
r5a.24xlarge	96	48	2	12、18、24、36、48	1、2
r5ad.large	2	1	2	1	1、2
r5ad.xlarge	4	2	2	2	1、2
r5ad.2xlarge	8	4	2	2、4	1、2
r5ad.4xlarge	16	8	2	2、4、6、8	1、2
r5ad.8xlarge	32	16	2	4、6、8、10、12、14、16	1、2
r5ad.12xlarge	48	24	2	6、12、18、24	1、2
r5ad.16xlarge	64	32	2	8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
r5ad.24xlarge	96	48	2	12、18、24、36、48	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r5b.large	2	1	2	1	1、2
r5b.xlarge	4	2	2	1、2	1、2
r5b.2xlarge	8	4	2	2、4	1、2
r5b.4xlarge	16	8	2	2、4、6、8	1、2
r5b.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
r5b.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
r5b.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
r5b.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r5d.large	2	1	2	1	1、2
r5d.xlarge	4	2	2	2	1、2
r5d.2xlarge	8	4	2	2、4	1、2
r5d.4xlarge	16	8	2	2、4、6、8	1、2
r5d.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
r5d.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
r5d.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
r5d.24xlarge	96	48	2	4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r5dn.large	2	1	2	1	1、2
r5dn.xlarge	4	2	2	1、2	1、2
r5dn.2xlarge	8	4	2	2、4	1、2
r5dn.4xlarge	16	8	2	2、4、6、8	1、2
r5dn.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
r5dn.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
r5dn.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
r5dn.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r5n.large	2	1	2	1	1、2
r5n.xlarge	4	2	2	1、2	1、2
r5n.2xlarge	8	4	2	2、4	1、2
r5n.4xlarge	16	8	2	2、4、6、8	1、2
r5n.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
r5n.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
r5n.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
r5n.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r6a.large	2	1	2	1	1、2
r6a.xlarge	4	2	2	1、2	1、2
r6a.2xlarge	8	4	2	1、2、3、4	1、2
r6a.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
r6a.8xlarge	32	16	2	4、6、8、10、12、14、16	1、2
r6a.12xlarge	48	24	2	一、二、三、四、五、六、七、八、十六	1、2
r6a.16xlarge	64	32	2	四、六、八、十、十、十、十四、十六、三	1、2
r6a.24xlarge	96	48	2	四、六、八、十、十、十、十四、十六、三十八	1、2
r6a.32xlarge	128	64	2	八、十六、二十、二十、二十、二十四	1、2
r6a.48xlarge	192	96	2	八、十六、二十、二十、二十、二、三十六	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r6g.large	2	2	1	1、2	1
r6g.xlarge	4	4	1	1、2、3、4	1
r6g.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1
r6g.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
r6g.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r6g.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r6g.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
r6gd.large	2	2	1	1、2	1
r6gd.xlarge	4	4	1	1、2、3、4	1
r6gd.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r6gd.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
r6gd.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1
r6gd.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r6gd.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
r6i.large	2	1	2	1	1、2
r6i.xlarge	4	2	2	1、2	1、2
r6i.2xlarge	8	4	2	2、4	1、2
r6i.4xlarge	16	8	2	2、4、6、8	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r6i.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
r6i.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
r6i.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
r6i.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r6i.32xlarge	128	64	2	二、四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四十六、四十、四、五十五、五十五、五十五、五十六、六十六、六十六、64	1、2
r6idn.large	2	1	2	1	1、2
r6idn.xlarge	4	2	2	1、2	1、2
r6idn.2xlarge	8	4	2	1、2、3、4	1、2
r6idn.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
r6idn.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r6idn.12xlarge	48	24	2	一、二、三、四、五、六、七、八、九、十、十、十、十、十、十、十、十、十、十、十、十、十、十、十、二、二、二、二、二、二、24	1、2
r6idn.16xlarge	64	32	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1、2
r6idn.24xlarge	96	48	2	4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r6idn.32xlarge	128	64	2	四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四十、四十、四十、五十五、五十五、五十五、五十六、六、六、六、六、64	1、2
r6in.large	2	1	2	1	1、2
r6in.xlarge	4	2	2	1、2	1、2
r6in.2xlarge	8	4	2	1、2、3、4	1、2
r6in.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
r6in.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r6in.12xlarge	48	24	2	一、二、三、四、五、六、七、八、九、十、十、十、十、十、十、十、十、十、十、十、十、十、十、十、二、二、二、二、二、二、24	1、2
r6in.16xlarge	64	32	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1、2
r6in.24xlarge	96	48	2	4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r6in.32xlarge	128	64	2	四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四十、四十、四十、五十五、五十五、五十五、五十六、六、六、六、六、64	1、2
r6id.large	2	1	2	1	1、2
r6id.xlarge	4	2	2	1、2	1、2
r6id.2xlarge	8	4	2	2、4	1、2
r6id.4xlarge	16	8	2	2、4、6、8	1、2
r6id.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
r6id.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r6id.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
r6id.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
r6id.32xlarge	128	64	2	二、四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四十六、四十、四、五十五、五十五、五十五、五十六、六十六、六十六、64	1、2
r7a.large	2	2	1	1、2	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r7a.xlarge	4	4	1	1、2、3、4	1
r7a.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1
r7a.4xlarge	16	16	1	一、二、四、六、八、十、十、十、十、十四、十六	1
r7a.8xlarge	32	32	1	1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32	1
r7a.12xlarge	48	48	1	一、二、三、四、五、六、十二、二、二、二、三十、四、四、四十八	1
r7a.16xlarge	64	64	1	一、二、三、四、五、六、七、八、十六、二、三十、四十、四十八、五十六	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r7a.24xlarge	96	96	1	一、二、三、四、五、六、七、八、九、十、十、十、十、二、二、二、三十六、六、七、八四、九六	1
r7a.32xlarge	128	128	1	四、六、八、十、十、十、十四、十六、四、四、六、八、八	1
r7a.48xlarge	192	192	1	四、六、八、十、十、十、十四、十六、十、二十、二十、二十、二、六、四、168、192	1
r7g.large	2	2	1	1、2	1
r7g.xlarge	4	4	1	1、2、3、4	1
r7g.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r7g.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
r7g.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1
r7g.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r7g.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
r7gd.large	2	2	1	1、2	1
r7gd.xlarge	4	4	1	1、2、3、4	1
r7gd.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r7gd.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
r7gd.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1
r7gd.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r7gd.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
r7i.large	2	1	2	1	1、2
r7i.xlarge	4	2	2	1、2	1、2
r7i.2xlarge	8	4	2	1、2、3、4	1、2
r7i.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r7i.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2
r7i.12xlarge	48	24	2	一、二、三、四、五、六、七、八、九、十、十、十、十、十、十、十、十、十、十、十、十、十、十、十、二、二、二、二、二、二、24	1、2
r7i.16xlarge	64	32	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r7i.24xlarge	96	48	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r7i.48xlarge	192	96	2	四、六、八、十、十、十四、十六、二十、二十、二十、二、二、二、三十、三十、三十、四、四、四、四、四、五十、五十五、五十六、六、六、六、六、六、六、六、六、六、七、七、七、六、七、六、七、六、七、六、七、六、七、六、七、六、七、六、七、六、七、六、七、六、七、七、六、七、六、七、七、六、七、六	1、2
r7iz.large	2	1	2	1	1、2
r7iz.xlarge	4	2	2	1、2	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r7iz.2xlarge	8	4	2	1、2、3、4	1、2
r7iz.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
r7iz.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2
r7iz.12xlarge	48	24	2	一、二、三、四、五、六、七、八、九、十、十、十、十、十、十、十、十、十、十、十、十、十、十、十、二、二、二、二、二、二、24	1、2
r7iz.16xlarge	64	32	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r7iz.32xlarge	128	64	2	四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四、四、四十、四十、四十、五十五、五十五、五十五、五十六、六、六、六、六、64	1、2
r8g.large	2	2	1	1、2	1
r8g.xlarge	4	4	1	1、2、3、4	1
r8g.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1
r8g.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r8g.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1
r8g.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r8g.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r8g.24xlarge	96	96	1	1、2、3、4、5、6、7、8、9、十、十、十、十、十、十、十、十、十、十五、十、十、十、十、三、二、二、二、二、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四 49、50、五十、五十二、五十五、五十五、五十五、五十六、六十六、六十六、六十六、六十六、六十六、六十六、六十六、六十六、七、七、七、七、七、九十九、八、八、八、八、八、八 96	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
r8g.48xlarge	192	192	1	四、六、八、十、十、十四、十六、二十、二十、二十、二十六、二十、三十、三十、三十、三十、四、四、四、四、四、五十、五十五、五十六、六、六、六、六、六、六、六、六、六、六、七、七、七、六、七、七、六、七、六、七、六、七、六、七、六、七、六、七、六、七、六、七、六、七、六、七、六、七、七、六、七、七、六100、102、104、106、108、110、112、116、一百二十八、十二、二、二、	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
				二、二、二、二十二、十二、十二、一三、三、三、三八、一四、十四、四四、八五、六、六六、六、六、六八、六、六、六、六、七八、六八、六八、六	
u-3tb1.56xlarge	224	112	2	四、八、十二、二十、二十、二、二、三、三十、四十、四十、四十、五十六、六十六、六、六、七、七、八、八、八、八、九二、九六、百、一八、112	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
u-6tb1.56xlarge	224	224	1	八、十六、二十四、四十、五十六、六十二、八十八、九六、九六、十二、一二、一二、一八、一六、六、六、六、六、六、八、二千八、二千八、二千二、216、224	1
u-6tb1.112xlarge	448	224	2	八、十六、二十四、四十、五十六、六十二、八十八、九六、九六、十二、一二、一二、一八、一六、六、六、六、六、六、八、二千八、二千八、二千二、216、224	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
u-9tb1.112xlarge	448	224	2	八、十六、二十四、四十、五十六、六十二、八十八、九六、九六、十二、一二、一二、一八、一六、六、六、六、六、六、八、二千八、二千八、二千二、216、224	1、2
u-12tb1.112xlarge	448	224	2	八、十六、二十四、四十、五十六、六十二、八十八、九六、九六、十二、一二、一二、一八、一六、六、六、六、六、六、八、二千八、二千八、二千二、216、224	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
u-18tb1.112xlarge	448	224	2	八、十六、二十四、四十、五十六、六十二、八十八、九六、九六、十二、一二、一二、一八、一六、六、六、六、六、六、八、二千八、二千八、二千二、216、224	1、2
u-24tb1.112xlarge	448	224	2	八、十六、二十四、四十、五十六、六十二、八十八、九六、九六、十二、一二、一二、一八、一六、六、六、六、六、六、八、二千八、二千八、二千二、216、224	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
u7i-12tb.224xlarge	896	448	2	十六、二十、三十、四十八、六、七、八、八、九六、九六、十二、一二、十二、十二、六、六、六、六、八、八、二、二、二、三六、３５２、３５２、３５２、３６５２、２６２、２６２、２６２、２８２，384、392、四百、408、416、424、432、440、448	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
u7in-16tb.224xlarge	896	448	2	十六、二十、三十、四十八、六、七、八、八、九六、九六、十二、一二、十二、十二、六、六、六、六、八、八、二、二、二、三六、３５２、３５２、３５２、３６５２、２６２、２６２、２６２、２８２，384、392、四百、408、416、424、432、440、448	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
u7in-24tb.224xlarge	896	448	2	十六、二十、三十、四十八、六、七、八、八、九六、九六、十二、一二、十二、十二、六、六、六、六、八、八、二、二、二、三六、３５２、３５２、３５２、３６５２、２６２、２６２、２６２、２８２，384、392、四百、408、416、424、432、440、448	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
u7in-32tb.224xlarge	896	448	2	十六、二十、三十、四十八、六、七、八、八、九六、九六、十二、一二、十二、十二、六、六、六、六、八、八、二、二、二、三六、３５２、３５２、３５２、３６５２、２６２、２６２、２６２、２８２，384、392、四百、408、416、424、432、440、448	1、2
x1.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
x1.32xlarge	128	64	2	4、8、12、16、20、24、28、32、36、40、44、48、52、56、60、64	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
x2gd.large	2	2	1	1、2	1
x2gd.xlarge	4	4	1	1、2、3、4	1
x2gd.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1
x2gd.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
x2gd.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
x2gd.12xlarge	48	48	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
x2gd.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
x2idn.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
x2idn.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
x2idn.32xlarge	128	64	2	二、四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四十六、四十、四、五十五、五十五、五十五、五十六、六十六、六十六、64	1、2
x2iedn.xlarge	4	2	2	1、2	1、2
x2iedn.2xlarge	8	4	2	2、4	1、2
x2iedn.4xlarge	16	8	2	2、4、6、8	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
x2iedn.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
x2iedn.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
x2iedn.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
x2iedn.32xlarge	128	64	2	二、四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四十六、四十、四、五十五、五十五、五十五、五十六、六十六、六十六、64	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
x2iezn.2xlarge	8	4	2	2、4	1、2
x2iezn.4xlarge	16	8	2	2、4、6、8	1、2
x2iezn.6xlarge	24	12	2	2、4、6、8、10、12	1、2
x2iezn.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
x2iezn.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
x1e.xlarge	4	2	2	1、2	1、2
x1e.2xlarge	8	4	2	1、2、3、4	1、2
x1e.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
x1e.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
x1e.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
x1e.32xlarge	128	64	2	4、8、12、16、20、24、28、32、36、40、44、48、52、56、60、64	1、2
z1d.large	2	1	2	1	1、2
z1d.xlarge	4	2	2	1、2	1、2
z1d.2xlarge	8	4	2	2、4	1、2
z1d.3xlarge	12	6	2	2、4、6	1、2
z1d.6xlarge	24	12	2	2、4、6、8、10、12	1、2
z1d.12xlarge	48	24	2	4、6、8、10、12、14、16、18、20、22、24	1、2

儲存最佳化執行個體

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
d2.xlarge	4	2	2	1、2	1、2
d2.2xlarge	8	4	2	1、2、3、4	1、2
d2.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
d2.8xlarge	36	18	2	2、4、6、8、10、12、14、16、18	1、2
d3.xlarge	4	2	2	1、2	1、2
d3.2xlarge	8	4	2	2、4	1、2
d3.4xlarge	16	8	2	2、4、6、8	1、2
d3.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
d3en.xlarge	4	2	2	1、2	1、2
d3en.2xlarge	8	4	2	2、4	1、2
d3en.4xlarge	16	8	2	2、4、6、8	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
d3en.6xlarge	24	12	2	2、4、6、8、10、12	1、2
d3en.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
d3en.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
h1.2xlarge	8	4	2	1、2、3、4	1、2
h1.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
h1.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2
h1.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
i2.xlarge	4	2	2	1、2	1、2
i2.2xlarge	8	4	2	1、2、3、4	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
i2.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
i2.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
i3.large	2	1	2	1	1、2
i3.xlarge	4	2	2	1、2	1、2
i3.2xlarge	8	4	2	1、2、3、4	1、2
i3.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
i3.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2
i3.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
i3en.large	2	1	2	1	1、2
i3en.xlarge	4	2	2	1、2	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
i3en.2xlarge	8	4	2	2、4	1、2
i3en.3xlarge	12	6	2	2、4、6	1、2
i3en.6xlarge	24	12	2	2、4、6、8、10、12	1、2
i3en.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
i3en.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
i4g.large	2	2	1	1、2	1
i4g.xlarge	4	4	1	1、2、3、4	1
i4g.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
i4g.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
i4g.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
i4g.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
i4i.large	2	1	2	1	1、2
i4i.xlarge	4	2	2	1、2	1、2
i4i.2xlarge	8	4	2	1、2、3、4	1、2
i4i.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
i4i.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2
i4i.12xlarge	48	24	2	一、二、三、四、五、六、七、八、九、十、十、十、十、十、十、十、十、十、十、十、十、十、十、十、二、二、二、二、二、二、24	1、2
i4i.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
i4i.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
i4i.32xlarge	128	64	2	二、四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四十六、四十、四、五十五、五十五、五十五、五十六、六十六、六十六、64	1、2
im4gn.large	2	2	1	1、2	1
im4gn.xlarge	4	4	1	1、2、3、4	1
im4gn.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1
im4gn.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
im4gn.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1
im4gn.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
`is4gen.medium`	1	1	1	1	1
`is4gen.large`	2	2	1	1、2	1
`is4gen.xlarge`	4	4	1	1、2、3、4	1
`is4gen.2xlarge`	8	8	1	1、2、3、4、5、6、7、8	1
`is4gen.4xlarge`	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1
`is4gen.8xlarge`	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1

加速運算執行個體

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
dl1.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
dl2q.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
f1.2xlarge	8	4	2	1、2、3、4	1、2
f1.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
f1.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
g3.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
g3.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2
g3.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
g4ad.xlarge	4	2	2	2	1、2
g4ad.2xlarge	8	4	2	2、4	1、2
g4ad.4xlarge	16	8	2	2、4、8	1、2
g4ad.8xlarge	32	16	2	2、4、8、16	1、2
g4ad.16xlarge	64	32	2	2、4、8、16、32	1、2
g4dn.xlarge	4	2	2	2	1、2
g4dn.2xlarge	8	4	2	2、4	1、2
g4dn.4xlarge	16	8	2	2、4、6、8	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
g4dn.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
g4dn.12xlarge	48	24	2	2、4、6、8、10、12、14、16、18、20、22、24	1、2
g4dn.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
g5g.xlarge	4	4	1	1、2、3、4	1
g5g.2xlarge	8	8	1	1、2、3、4、5、6、7、8	1
g5g.4xlarge	16	16	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
g5g.8xlarge	32	32	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32	1
g5g.16xlarge	64	64	1	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16、17、18、19、20、21、22、23、24、25、26、27、28、29、30、31、32、33、34、35、36、37、38、39、40、41、42、43、44、45、46、47、48、49、50、51、52、53、54、55、56、57、58、59、60、61、62、63、64	1
g6.xlarge	4	2	2	1、2	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
g6.2xlarge	8	4	2	1、2、3、4	1、2
g6.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
g6.8xlarge	32	16	2	一、二、四、六、八、十、十、十、十、十四、十六	1、2
g6.12xlarge	48	24	2	一、二、三、六、九、十二、十五、十八、二十一、二十四	1、2
g6.16xlarge	64	32	2	1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32	1、2
g6.24xlarge	96	48	2	一、二、三、四、五、六、十二、二、二、二、三十、四、四、四十八	1、2
g6.48xlarge	192	96	2	四、六、八、十、十、二、二十四、四十、六十、七、八四、九六	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
g6e.xlarge	4	2	2	1、2	1、2
g6e.2xlarge	8	4	2	1、2、3、4	1、2
g6e.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2
g6e.8xlarge	32	16	2	2、4、6、8、10、12、14、16	1、2
g6e.12xlarge	48	24	2	三、六、九、十二、十五、十八、二十四	1、2
g6e.16xlarge	64	32	2	四、八、十二、十六、二十、二十	1、2
g6e.24xlarge	96	48	2	六、十二、十八、二十四、三十、四十八	1、2
g6e.48xlarge	192	96	2	12、24、36、48、六十、七十二、八四、九六	1、2
gr6.4xlarge	16	8	2	1、2、3、4、5、6、7、8	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
gr6.8xlarge	32	16	2	一、二、四、六、八、十、十、十、十、十四、十六	1、2
inf1.xlarge	4	2	2	2	1、2
inf1.2xlarge	8	4	2	2、4	1、2
inf1.6xlarge	24	12	2	2、4、6、8、10、12	1、2
inf1.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
inf2.xlarge	4	2	2	1、2	1、2
inf2.8xlarge	32	16	2	4、6、8、10、12、14、16	1、2
inf2.24xlarge	96	48	2	四、六、八、十、十、十、十四、十六、三十八	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
inf2.48xlarge	192	96	2	四、八、十二、十六、二十、二十、二、二、六	1、2
p2.xlarge	4	2	2	1、2	1、2
p2.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2
p2.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2
p3.2xlarge	8	4	2	1、2、3、4	1、2
p3.8xlarge	32	16	2	1、2、3、4、5、6、7、8、9、10、11、12、13、14、15、16	1、2
p3.16xlarge	64	32	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
p3dn.24xlarge	96	48	2	4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
p4d.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
p4de.24xlarge	96	48	2	2、4、6、8、10、12、14、16、18、20、22、24、26、28、30、32、34、36、38、40、42、44、46、48	1、2
p5.48xlarge	192	96	2	12、24、36、48、六十、七十二、八四、九六	1、2
trn1.2xlarge	8	4	2	2、4	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
trn1.32xlarge	128	64	2	二、四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四十六、四十、四、五十五、五十五、五十五、五十六、六十六、六十六、64	1、2
trn1n.32xlarge	128	64	2	二、四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四十六、四十、四、五十五、五十五、五十五、五十六、六十六、六十六、64	1、2
vt1.3xlarge	12	6	2	6	1、2

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
vt1.6xlarge	24	12	2	六、十二	1、2
vt1.24xlarge	96	48	2	六、十二	1、2

高效能運算執行個體

執行個體類型	預設 vCPUs	預設CPU核心	預設每核心執行緒	有效的CPU核心	每個核心的有效執行緒
hpc6id.32xlarge	64	64	1	四、六、八、十、十、十、十四、十六、十六、四、四、四、四、四、四、四、四、四、四、四、四、四、四、四十、四十、四十、五十五、五十五、五十五、五十六、六、六、六、六、64	1

指CPU定 Amazon EC2 執行個體的選項

您可以在執行個體啟動期間指定CPU選項。

下列範例說明在主控台和 run-instance 命令中使用啟動執行個體精靈時，如何指定CPU選項，以及在EC2主控台和 AWS CLI 命令中指定 [建立啟動範本] 頁面。EC2 create-launch-template AWS CLI 對於「EC2叢集」或「競價型叢集」，您必須在啟動範本中指定CPU選項。

以下範圍適用於 `r5.4xlarge` 執行個體類型，其預設值如下：

- 預設CPU核心：8
- 預設每核心執行緒：2
- 預設值vCPUs：16 (8 個 * 2)
- CPU核心數目有效：2、4、6、8
- 有效的每核心執行緒數量：1、2

停用同時多執行緒

若要停用同步多執行緒 (SMT) (也稱為「超執行緒」)，請為每個核心指定 1 個執行緒。

Console

SMT在執行個體啟動期間停用

1. 遵照 使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體 程序並視需要設定執行個體。
2. 展開 [進階詳細資料]，並選取 [指定CPU選項] 核取方塊。
3. 對於「核心數」，請選擇所需的CPU核心數目。在此範例中，若要指定`r5.4xlarge`執行個體的預設CPU核心計數，請選擇8。
4. 若要停用SMT，對於每個核心的執行緒，請選擇 1。
5. 在 Summary (摘要) 面板中，檢閱您的執行個體組態，然後選擇 Launch instance (啟動執行個體)。如需詳細資訊，請參閱使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體。

AWS CLI

SMT在執行個體啟動期間停用

使用 run-instances AWS CLI 命令，為 1 參數的 `ThreadsPerCore` 指定值 `--cpu-options`。對於`CoreCount`，指定CPU核心的數目。在此範例中，若要指定`r5.4xlarge`執行個體的預設CPU核心計數，請將值指定為8。

```
aws ec2 run-instances \
    --image-id ami-1a2b3c4d \
    --instance-type r5.4xlarge \
```

```
    --cpu-options "CoreCount=8,ThreadsPerCore=1" \
    --key-name MyKeyPair
```

指定啟動 vCPUs 時的自訂數目

您可以自訂執行個體每個CPU核心的核心和執行緒數目。

下列範例會啟動具有 4 的`r5.4xlarge`執行個體vCPUs。

Console

若要在執行個體啟動 vCPUs 期間指定自訂數目

1. 遵照 使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體 程序並視需要設定執行個體。
2. 展開 [進階詳細資料]，並選取 [指定CPU選項] 核取方塊。
3. 若要取得 4 個vCPUs，請為每個核心指定 2 CPU 個核心和 2 個執行緒，如下所示：
   - 針對「核心數」，選擇「2」。
   - 針對 Threads per core (每核心執行緒)，選擇 2。
4. 在 Summary (摘要) 面板中，檢閱您的執行個體組態，然後選擇 Launch instance (啟動執行個體)。如需詳細資訊，請參閱使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體。

AWS CLI

若要在執行個體啟動 vCPUs 期間指定自訂數目

使用 run-instance 指 AWS CLI 令，並指定參數中的CPU核心數和執行緒數目`--cpu-options`。您可以為每個CPU核心指定 2 個核心和 2 個執行緒以獲得 4 個vCPUs。

```
aws ec2 run-instances \
    --image-id ami-1a2b3c4d \
    --instance-type r5.4xlarge \
    --cpu-options "CoreCount=2,ThreadsPerCore=2" \
    --key-name MyKeyPair
```

或者，每個核心指定 4 CPU 個核心和 1 個執行緒 (停用SMT) 以取得 4 個vCPUs：

```
aws ec2 run-instances \
    --image-id ami-1a2b3c4d \
    --instance-type r5.4xlarge \
    --cpu-options "CoreCount=4,ThreadsPerCore=1" \
    --key-name MyKeyPair
```

在啟動範本 vCPUs 中指定自訂數目

您可以自訂啟動範本中執行個體每個核心的核心和執行緒數量。CPU

下列範例會建立啟動範本，指定具有 4 的`r5.4xlarge`執行個體組態vCPUs。

Console

若要在啟動範本 vCPUs 中指定自訂數目

1. 遵照 透過指定參數建立啟動範本 程序並視需要設定啟動範本。
2. 展開 [進階詳細資料]，並選取 [指定CPU選項] 核取方塊。
3. 若要取得 4 個vCPUs，請為每個核心指定 2 CPU 個核心和 2 個執行緒，如下所示：
   - 針對「核心數」，選擇「2」。
   - 針對 Threads per core (每核心執行緒)，選擇 2。
4. 在摘要面板中，檢閱執行個體組態，然後選擇建立啟動範本。如需詳細資訊，請參閱在 Amazon 啟動範本中存放執行個體EC2啟動參數。

AWS CLI

若要在啟動範本 vCPUs 中指定自訂數目

使用指create-launch-template AWS CLI 令並指定參數中的CPU核心數和執行緒`CpuOptions`數目。您可以為每個CPU核心指定 2 個核心和 2 個執行緒以獲得 4 個vCPUs。

```
aws ec2 create-launch-template \
    --launch-template-name TemplateForCPUOptions \
    --version-description CPUOptionsVersion1 \
    --launch-template-data file://template-data.json
```

下列範例JSON檔案包含啟動範本資料，其中包含此範例中執行個體組態的CPU選項。

```
{
    "NetworkInterfaces": [{
        "AssociatePublicIpAddress": true,
        "DeviceIndex": 0,
        "Ipv6AddressCount": 1,
        "SubnetId": "subnet-7b16de0c"
    }],
    "ImageId": "ami-8c1be5f6",
    "InstanceType": "r5.4xlarge",
    "TagSpecifications": [{
        "ResourceType": "instance",
        "Tags": [{
            "Key":"Name",
            "Value":"webserver"
        }]
    }],
    "CpuOptions": {
        "CoreCount":2,
        "ThreadsPerCore":2
    }
}
```

或者，每個核心指定 4 CPU 個核心和 1 個執行緒 (停用SMT) 以取得 4 個vCPUs：

```
{
    "NetworkInterfaces": [{
        "AssociatePublicIpAddress": true,
        "DeviceIndex": 0,
        "Ipv6AddressCount": 1,
        "SubnetId": "subnet-7b16de0c"
    }],
    "ImageId": "ami-8c1be5f6",
    "InstanceType": "r5.4xlarge",
    "TagSpecifications": [{
        "ResourceType": "instance",
        "Tags": [{
            "Key":"Name",
            "Value":"webserver"
        }]
    }],
    "CpuOptions": {
        "CoreCount":4,
        "ThreadsPerCore":1
```

```
    }
}
```

檢視 Amazon EC2 執行個體的執行CPU緒和核心

您可以在 Amazon EC2 主控台中檢視現有執行個體的CPU選項，或使用說明執行個體 AWS CLI。

Console

使用主控台檢視執行個體的CPU選項

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側的導覽窗格中，選取 Instances (執行個體)，然後選取執行個體。
3. 在 [詳細資料] 索引標籤的 [主機和置放群組] 下，找到 [數目] vCPUs。

AWS CLI

若要檢視例證的CPU選項 (AWS CLI)

使用 describe-instances 命令。

```
aws ec2 describe-instances --instance-ids i-123456789abcde123
```

```
...
    "Instances": [
        {
            "Monitoring": {
                "State": "disabled"
            },
            "PublicDnsName": "ec2-198-51-100-5.eu-central-1.compute.amazonaws.com",
            "State": {
                "Code": 16,
                "Name": "running"
            },
            "EbsOptimized": false,
            "LaunchTime": "2018-05-08T13:40:33.000Z",
            "PublicIpAddress": "198.51.100.5",
            "PrivateIpAddress": "172.31.2.206",
            "ProductCodes": [],
            "VpcId": "vpc-1a2b3c4d",
```

```
            "CpuOptions": {
                "CoreCount": 34,
                "ThreadsPerCore": 1
            },
            "StateTransitionReason": "",
            ...
        }
    ]
...
```

在傳回的輸出中，`CoreCount` 欄位顯示執行個體的核心數量。`ThreadsPerCore` 欄位顯示每核心的執行緒數量。

或者，若要檢視CPU資訊，您可以連線至執行個體，並使用下列其中一種系統工具：

- 視窗執行個體`Task Manager`上的視窗
- Linux 執行個體上的lscpu指令

您可以用 AWS Config 來記錄、評估、稽核和評估執行個體的組態變更，包括已終止的執行個體。如需詳細資訊，請參閱《AWS Config 開發人員指南》中的開始使用 AWS Config。

AMDSEV-SNP 適用於 Amazon EC2 實例

AMD安全加密虛擬化-安全巢狀分頁 (AMDSEV-SNP) 是提供下列屬性的CPU功能：

- 驗證 — AMD SEV-可SNP讓您擷取已簽署的驗證報告，其中包含可用來驗證執行個體狀態和識別的密碼編譯測量，以及執行個體是否在正版硬體上執行。AMD如需詳細資訊，請參閱 用 AMD SEV-證明一個 Amazon EC2 實例-SNP。
- 記憶體加密 — 從 AMD EPYC (米蘭)、 AWS 引力通 2 和 Intel Xeon 可擴充 (冰湖) 處理器開始，執行個體記憶體一律會加密。啟用的執行個體 AMD SEV-SNP 使用執行個體特定金鑰進行記憶體加密。

主題

- 概念和術語
- 要求
- 考量事項
- 定價

- 檢查 AMD SEV-對 Amazon EC2 實例的SNP支持
- 用 AMD SEV-證明一個 Amazon EC2 實例-SNP

概念和術語

在開始使用 AMD SEV-之前SNP，請確定您已熟悉下列概念和術語。

AMDSEV-SNP 驗證報告

AMDSEV-SNP 驗證報告是執行個體可從中要求的文件。CPUAMDSEV-驗SNP證報告可用來驗證執行個體的狀態和識別，以及確認AMD執行個體是否在已驗證的環境中執行。報告包含啟動測量值，這是執行個體初始啟動狀態的密碼編譯雜湊，包括其初始執行個體記憶體內容和. vCPUs AMDSEV-SNP 驗證報告會使用可鏈結回信任AMD根的簽VLEK章來簽署。

VLEK

版本化的已載入背書金鑰 (VLEK) 是版本化的簽署金鑰，由認證AMD並由用於簽署 AMD SEV-SNP 驗證報告。AMD CPUVLEK簽名可以使用提供的證書進行驗證AMD。

OVMF二進制

開啟虛擬機器韌體 (OVMF) 是用來提供執行個體UEFI環境的早期開機程式碼。早期開機程式碼會在啟動程式碼之前執行。AMIOVMF也會尋找並執行中提供的開機載入程式AMI。如需詳細資訊，請參閱存OVMF放庫。

要求

要使用 AMD SEV-SNP，您必須執行以下操作：

- 使用下列其中一個支援的執行個體類型：
  - 一般用途：`m6a.large` | `m6a.xlarge` | `m6a.2xlarge` | `m6a.4xlarge` | `m6a.8xlarge`
  - 運算優化：`c6a.large` | `c6a.xlarge` | `c6a.2xlarge` | `c6a.4xlarge` | `c6a.8xlarge` | `c6a.12xlarge` | `c6a.16xlarge`
  - 記憶體優化：`r6a.large` | `r6a.xlarge` | `r6a.2xlarge` | `r6a.4xlarge`
- 在支援的情況下啟動執行個體 AWS 區域。目前僅支援美國東部 (俄亥俄) 和歐洲 (愛爾蘭)。
- 使用AMI帶`uefi`或`uefi-preferred`啟動模式以及支持 AMD SEV-的操作系統SNP。如需有關AMDSEV作業系統SNP支援的詳細資訊，請參閱個別作業系統的說明文件。對於 AWS，AMDSEV-SNP 在 AL2 023、RHEL 9.3 SP4、SLES 15 和更新版本上支援。

考量事項

您只能在啟動執行個體SNP時啟用 AMD SEV-。啟用執行個體啟動時 AMDSEV，會套用下列規則。SNP

- 一旦啟用，AMDSEV-不SNP能被禁用。它在整個例證生命週期中保持啟用狀態
- 您只能將例證類型變更為支援 AMD SEV-的其他例證類型SNP。
- 不支援休眠和硝基飛地。
- 不支援專用主機。
- 如果執行個體的基礎主機已排程進行維護，您將在事件發生前 14 天收到排定的事件通知。您必須手動停止或重新啟動執行個體，才能將其移至新主機。

定價

當您SNP啟用 AMD SEV-的 Amazon EC2 執行個體時，系統會向您收取額外的小時使用費，相當於所選執行個體類型隨需小時費率的 10%。

這 AMD SEV-SNP 使用費是對您的 Amazon EC2 執行個體使用量另外收取費用。預留執行個體、Savings Plans 和作業系統用量不會影響此費用。

如果您將競價型執行個體設定為SNP啟用 AMDSEV- 啟用，則需要支付額外的小時使用費，相當於所選執行個體類型隨需小時費率的 10%。如果配置策略使用價格作為輸入，則 Spot 機群不包含此額外費用；僅使用 Spot 價格。

檢查 AMD SEV-對 Amazon EC2 實例的SNP支持

主題

- 尋找支援的 Amazon EC2 執行個體類型 AMD SEV-SNP
- 檢查 Amazon 實EC2例是否已啟用 AMD SEV-SNP

尋找支援的 Amazon EC2 執行個體類型 AMD SEV-SNP

您可以使用 AWS CLI 來尋找支援 AMD SEV-的執行個體類型SNP。

若要尋找支援 AMD SEV-SNP 使用的執行個體類型 AWS CLI，請使用下列 describe-instance-types指令。

```
$ aws ec2 describe-instance-types \
```

```
--filters Name=processor-info.supported-features,Values=amd-sev-snp \
--query 'InstanceTypes[*].InstanceType'
```

範例輸出。

```
[
 "r6a.2xlarge",
 "m6a.large",
 "m6a.2xlarge",
 "r6a.xlarge",
 "c6a.16xlarge",
 "c6a.8xlarge",
 "m6a.4xlarge",
 "c6a.12xlarge",
 "r6a.4xlarge",
 "c6a.xlarge",
...
]
```

檢查 Amazon 實EC2例是否已啟用 AMD SEV-SNP

您可以使用下列其中一種方法來檢查 AMD SEV-的狀態SNP。

AWS CLI

若要使用檢查執行個體SNP是否啟用了 AMD SEV- AWS CLI，請使用describe-instances指令。對於 `--instance-ids`，指定要檢查的執行個體 ID。

```
$ aws ec2 describe-instances --instance-ids instance_id
```

在指令輸出中，`AmdSevSnp`in 的值`CpuOptions`表示啟SNP用或停用 AMD SEV-。

AWS CloudTrail

在執行個體啟動要求的情況下，值`"cpuOptions": {"AmdSevSnp": enabled}`表示執行SNP個體已啟用 AMD SEV-。 AWS CloudTrail

用 AMD SEV-證明一個 Amazon EC2 實例-SNP

認證是一種程序，可讓您的執行個體證明其狀態和身分。當您SNP為執行個體啟用AMDSEV時，您可以向基礎處理器要求 AMD SEV-SNP 驗證報告。AMDSEV-SNP 驗證報告包含初始客體記憶體內容和初始 v 狀態的加密雜湊 (稱為啟動測量)。CPU驗證報告會以鏈結回信任AMD根的簽VLEK章簽署。您可

以使用驗證報告中包含的啟動測量來驗證執行個體是否在正版AMD環境中執行，並驗證用於啟動執行個體的初始啟動程式碼。

若要使用 AMD SEV-執行驗證SNP，請完成以下步驟。

主題

- 步驟 1：啟用 AMD SEV-SNP 在執行個體啟動期間
- 步驟 2：取得驗證報告
- 步驟 3：驗證報告簽名

步驟 1：啟用 AMD SEV-SNP 在執行個體啟動期間

您可以使用啟 AWS CLI 用 AMD SEV-SNP 啟用的執行個體。

若要在啟用 AMD SEV-的情況下SNP啟動執行個體，您必須使用 AWS CLI. 使用指run-instances令並包含選`--cpu-options AmdSevSnp=enabled`項。對於`--image-id`，指定AMI具有`uefi`或`uefi-prefered`啟動模式以及支援 AMD SEV-的作業系統SNP。對於 `--instance-type`，指定支援的執行個體類型。

```
$ aws ec2 run-instances \
--image-id supported_ami_id \
--instance-type supported_instance_type \
--key-name key_pair_name \
--subnet-id subnet_id \
--cpu-options AmdSevSnp=enabled
```

步驟 2：取得驗證報告

在此步驟中，您要安裝並建置公用`snpguest`程式，然後使用它來要求 AMD SEV-SNP 驗證報告和憑證。

1. 執行下列命令以從建置`snpguest`公用程式snpguest repository。

   ```
   $ git clone https://github.com/virtee/snpguest.git
   $ cd snpguest
   $ cargo build -r
   $ cd target/release
   ```

2. 產生驗證報告的請求。公用程式會向主機要求驗證報告，並將其寫入包含所提供請求資料的二進位檔案。

下列範例會建立隨機要求字串，並使用它做為要求檔案 (`request-file.txt`)。當指令傳回驗證報告時，它會儲存在您指定的檔案路徑 () `report.bin` 中。在此情況下，公用程式會將報告儲存在目前目錄中。

```
$ ./snpguest report report.bin request-file.txt --random
```

3. 從主機記憶體要求憑證，並將其儲存為PEM檔案。下列範例會將檔案儲存在與`snpguest`公用程式相同的目錄中。如果指定目錄中已存在憑證，則會覆寫這些憑證。

```
$ ./snpguest certificates PEM ./
```

步驟 3：驗證報告簽名

驗證報告使用稱為版本化的載入背書金鑰 (VLEK) 進行簽署的憑證簽署，此憑證由. AMD AWS在此步驟中，您可以驗證VLEK憑證是由發行者AMD，以及驗證報告是否由該VLEK憑證簽署。

1. 從官方AMD網站下載信任VLEK根證書到當前目錄。

```
$ sudo curl --proto '=https' --tlsv1.2 -sSf https://kdsintf.amd.com/vlek/v1/Milan/
cert_chain -o ./cert_chain.pem
```

2. 用`openssl`於驗證VLEK憑證是否由信任憑證的AMD根簽署。

```
$ sudo openssl verify --CAfile ./cert_chain.pem vlek.pem
```

預期的輸出結果：

```
certs/vcek.pem: OK
```

3. 使用此公用`snpguest`程式來驗證證明報告是否由憑證簽署。VLEK

```
$ ./snpguest verify attestation ./ report.bin
```

預期的輸出結果。

```
Reported TCB Boot Loader from certificate matches the attestation report.
Reported TCB TEE from certificate matches the attestation report.
Reported TCB SNP from certificate matches the attestation report.
```

```
Reported TCB Microcode from certificate matches the attestation report.
VEK signed the Attestation Report!
```

Amazon EC2 Linux 實例的處理器狀態控制

C 狀態控制核心閒置時可以進入的睡眠等級。C-state 從 C0 (最淺閒置狀態，此時核心處於喚醒狀態並執行指令) 開始編號，最高可到 C6 (最深閒置狀態，此時核心會關閉)。

P 狀態從核心控制所需的效能 (以CPU頻率表示)。P-state 從 P0 (最高效能設定，此時核心可視需要使用 Intel Turbo Boost 技術來提高頻率) 開始編號，之後則從 P1 (請求最大基準頻率的 P-state) 到 P15 (最低適用頻率)。

> **Note**
>
> AWS Graviton 處理器具有內建省電模式，並以固定頻率運作。因此，它們不能為作業系統提供控制 C-state 和 P-states 的能力。

C-state 和 P-state

下列執行個體類型可供作業系統控制處理器的 C-state 和 P-state：

- 一般用途：`m4.10xlarge`| `m4.16xlarge`
- 運算最佳化：`c4.8xlarge`
- 記憶體最佳化：`r4.8xlarger4.16xlargex1.16xlarge`| `x1.32xlarge` | `x1e.8xlarge` | `x1e.16xlarge` | `x1e.32xlarge`
- 儲存裝置最佳化：`d2.8xlargei3.8xlargei3.16xlarge`| `i3en.24xlarge` | `h1.8xlarge` | `h1.16xlarge`
- 加速運算：`f1.16xlargeg3.16xlarge`| | `p2.16xlarge` | `p3.16xlarge`
- 裸機：搭載 Intel 與AMD處理器的所有裸機執行個體

僅 C-state

下列執行個體類型可供作業系統控制處理器的 C-state：

- 一般用途：`m5.12xlargem5.24xlargem5d.12xlargem5d.24xlarge`| `m5n.12xlarge` `m5n.24xlarge` | `m5dn.12xlarge` | `m5dn.24xlarge` | `m5zn.6xlarge` | `m5zn.12xlarge`

m6a.24xlarge | m6a.48xlarge | m6i.16xlarge | m6i.32xlarge | m6id.16xlarge | m6id.32xlarge m6idn.16xlarge | m6in.16xlarge | m6in.32xlarge | m7a.medium | m7a.large | m7a.xlarge m7a.2xlarge | m7a.4xlarge | m7a.8xlarge | m7a.12xlarge | m7a.16xlarge m7a.24xlarge | m7a.32xlarge | m7a.48xlarge | m7i.large | m7i.xlarge | m7i.2xlarge m7i.4xlarge | m7i.8xlarge | m7i.12xlarge |m7i.16xlargem7i.24xlarge| m7i.48xlarge

- 計算最佳化：c5.9xlargec5.12xlargec5.18xlargec5.24xlargec5a.24xlargec5ad.24xlarge| c5d.9xlarge c5d.12xlarge | c5d.18xlarge | c5d.24xlarge | c5n.9xlarge | c5n.18xlarge c6a.24xlarge | c6a.32xlarge | c6a.48xlarge | c6i.16xlarge c6i.32xlarge | c6id.24xlarge | c6id.32xlarge | c6in.32xlarge c7a.medium | c7a.large | c7a.xlarge | c7a.2xlarge | c7a.4xlarge c7a.8xlarge | c7a.12xlarge | c7a.16xlarge | c7a.24xlarge c7a.32xlarge | c7a.48xlarge | c7i.large | c7i.xlarge c7i.2xlarge | c7i.4xlarge | c7i.8xlarge |c7i.12xlargec7i.16xlarge| c7i.24xlarge | c7i.48xlarge
- 記憶體最佳化：r5.12xlarger5.24xlarger5b.12xlarger5d.12xlarger5d.24xlarger5n.12xlarger5n.2 r6i.32xlarge r6id.16xlarge | r6id.32xlarge r6in.16xlarge | r6in.32xlarge r7a.medium | r7a.large r7a.xlarge | r7a.2xlarge r7a.4xlarge | r7a.8xlarge | r7a.12xlarge r7a.16xlarge | r7a.24xlarge r7a.32xlarge | r7a.48xlarge r7i.large | r7i.xlarge r7i.2xlarge | r7i.4xlarge r7i.8xlarge | r7i.12xlarge r7i.16xlarge | r7i.24xlarge |r7i.48xlarge| r7iz.large || r7iz.xlarge | r7iz.2xlarge | r7iz.4xlarge | r7iz.8xlarge || r7iz.12xlarge | r7iz.16xlarge | r7iz.32xlarge | u-3tb1.56xlarge || u-6tb1.56xlarge | u-6tb1.112xlarge | u-9tb1.112xlarge | u-12tb1.112xlarge || u-18tb1.112xlarge | u-24tb1.112xlarge | u7i-12tb.224xlarge | u7in-16tb.224xlarge || u7in-24tb.224xlarge | u7in-32tb.224xlarge | x2idn.32xlarge | x2iedn.16xlarge || x2iezn.12xlarge | z1d.6xlarge | z1d.12xlarge
- 儲存最佳化：d3en.12xlarge | dl1.24xlarge | i3en.12xlarge | i3en.24xlarge | i4i.16xlarge | r5b.12xlarge | r5b.24xlarge
- 加速運算：dl1.24xlargeg5.24xlargeg5.48xlargeg6.24xlargeg6.48xlargeg6e.12xlargeg6e.24x p5.24xlarge trn1.32xlarge | vt1.24xlarge

建議您變更 C-state 或 P-state 設定來提高處理器效能穩定性、減少延遲或針對特定工作負載微調執行個體。預設 C-state 和 P-state 設定提供最大效能，適合多數工作負載使用。然而，若您的應用程式更適合犧牲較高的單核心或雙核心頻率以降低延遲，或在較低頻率更能展現穩定效能 (而不適合使用突增 Turbo Boost 頻率)，請考慮嘗試調整這些執行個體的 C-state 或 P-state 設定。

如需有關不同處理器組態以及如何監控 Amazon Linux 組態效果的詳細資訊，請參閱 Amazon Linux 2 使用者指南中的 EC2 Amazon Linux 執行個體的處理器狀態控制。這些程序是針對 Amazon Linux 撰寫的，並適用於 Amazon Linux；不過，它們也可能適用於其他 Linux 核心為 3.9 或更新版本的 Linux 發行版。如需其他 Linux 發行版本和處理器狀態控制的詳細資訊，請參閱您的系統特定文件。

Amazon EC2 帳單和購買選項

您可以使用下列選項來優化 Amazon 的成本EC2：

- 隨需執行個體 – 對於啟動的執行個體，按秒數付費。
- S@@ avings Plan s — 承諾以USD每小時為 1 年或 3 年的一致使用量，以降低 Amazon EC2 成本。
- 預留執行個體 — 承諾使用一致的執行個體組態 (包括執行個體類型和區域)，期限為 1 或 3 年，以降低 Amazon EC2 成本。
- 競價型執行個體 — 請求未使用的EC2執行個體，這可大幅EC2降低 Amazon 成本。
- 專用執行個體 – 付費使用專供您執行執行個體的實體主機，並使用您現有的每個通訊端、每個核心或每個 VM 軟體的授權，以降低成本。
- 專用執行個體 – 依時數為單一租戶硬體上執行的執行個體付費。
- 容量保留 — 為特定可用區域中的EC2執行個體保留容量。

如果您無法對特定執行個體組態做出承諾，但可以承諾使用量，請購買 Savings Plans 以降低隨需執行個體的成本。如果您需要容量預留，請針對特定可用區域購買預留執行個體或容量預留。容量區塊可用於保留執行個GPU體叢集。如能彈性決定應用程式執行的時機以及應用程式是否可以中斷，則 Spot 執行個體會是一項具成本效益的選擇。專用執行個體或專用執行個體可協助您處理合規性要求，並使用您現有的伺服器軟體授權來減少成本。

如需詳細資訊，請參閱 Amazon EC2 定價。

為 Amazon 購買隨需實例 EC2

透過隨需執行個體，您只需要按秒數支付運算容量開銷，無需簽訂長期合約。您可以完整控制執行個體的生命週期 — 您決定何時啟動、停止、休眠、開始、重新開機或終止它。

購買隨需執行個體時，無需簽訂長期合約。您僅需按隨需執行個體處於 `running` 狀態的秒數付費，最少 60 秒。執行中隨需執行個體的每秒價格是固定的，並列在 Amazon 定EC2價、隨需定價頁面 上。

我們建議您將隨需執行個體用於具有短期、不定期但無法中斷之工作負載的應用程式。

若要大幅節省隨需執行個體，請使用 AWS Savings Plans、Spot 執行個體 或 Amazon 預留執行個體 EC2概觀。

內容

- 隨需執行個體配額
  - 監控隨需執行個體配額和用量
  - 請求提高配額
- 查詢隨需執行個體的價格

隨需執行個體配額

每 AWS 帳戶 個區域的執行隨需執行個體數量都有配額。隨需執行個體配額是根據執行中的隨需執行個體使用的虛擬中央處理單元 (vCPUs) 數量來管理，而不論執行個體類型為何。每個配額類型都會指定一或多個執 vCPUs 行個體系列的數目上限。

您的帳戶包含下列隨需執行個體配額。處於擱置中、停止、已停止和休眠狀態的執行個體不會計入隨需執行個體配額中。容量保留會計入您的隨需執行個體配額，即使它們未使用也一樣。

名稱	預設	可調整
執行中隨需 DL 執行個體	0	是
執行中隨需 F 執行個體	0	是
執行中隨需 G 和 VT 執行個體	0	是
運行隨需HPC實例	0	是
執行中隨需記憶體密集型執行個體	0	是
執行中隨需 Inf 執行個體	0	是
執行中隨需 P 執行個體	0	是

名稱	預設	可調整
執行中隨需的標準 (A、C、D、H、I、M、R、T、Z) 執行個體	5	是
執行中隨需 Trn 執行個體	0	是
執行中隨需 X 執行個體	0	是

如需不同執行個體系列、世代和大小的相關資訊，請參閱 Amazon EC2 執行個體類型指南。

您可以啟動任何符合您不斷變化的應用程式需求的執行個體類型組合，只要的數量 vCPUs 不超過您的帳戶配額即可。例如，如果標準執行個體配額為 256vCPUs，您可以啟動 32 個`m5.2xlarge`執行個體 (32 x 8vCPUs) 或 16 個`c5.4xlarge`執行個體 (16 x 16vCPUs)。如需詳細資訊，請參閱EC2隨需執行個體限制。

任務

- 監控隨需執行個體配額和用量
- 請求提高配額

監控隨需執行個體配額和用量

可以使用下列方法檢視和管理每個區域的隨需執行個體配額。

使用 Service Quotas 主控台檢視目前的配額

1. 在「家庭服務 /ec2/Quotas https://console.aws.amazon.com/servicequotas//」開啟「Service Quotas」主控台。
2. 從導覽列上，選取一個區域。
3. 在篩選條件欄位中，輸入 **`On-Demand`**。
4. 套用配額值欄會顯示您帳戶中每個隨需執行個體配額類型的最大數目。 vCPUs

使用 AWS Trusted Advisor 主控台檢視目前的配額

在 AWS Trusted Advisor 主控台中開啟 [服務限制] 頁面。

若要設定 CloudWatch 鬧鐘

透過 Amazon CloudWatch 指標整合，您可以根據配額監控EC2用量。您也可以設定警示，在接近配額時提出警告。如需詳細資訊，請參閱 Ser vice Quotas 使用指南中的 Service Quotas 和 Amazon CloudWatch 警示。

請求提高配額

即使 Amazon EC2 會根據您的用量自動增加隨需執行個體配額，但您也可以視需要申請增加配額。例如，如果您打算啟動的執行個體超過目前配額允許的數量，則可以使用或 Amazon EC2 服務配額 中所述的 Service Quotas 主控台來請求提高配額。

查詢隨需執行個體的價格

您可以使用價目表服務API或 AWS 價目表API來查詢隨需執行個體的價格。如需詳細資訊，請參閱《使用指南》API中的「使AWS Billing 用 AWS 價目表」。

Amazon 預留執行個體EC2概觀

> **Important**
>
> 我們建議 Savings Plans 而非預留執行個體 如同預留執行個體一樣，儲蓄計劃是最簡單且最靈活的方式，可節省 AWS 運算成本並提供更低的價格 (最高可享隨需定價 72% 的費用)。但是，Savings Plans 與預留執行個體不同。使用預留執行個體時，您可以承諾使用特定的執行個體組態，而 Savings Plans 則可以彈性使用最符合您需求的執行個體組態。要使用 Savings Plans，您需要承諾一致的使用量，以USD每小時為單位。如需詳細資訊，請參閱 AWS Savings Plans 使用者指南。

與隨需執行個體定價相比，預留執行個體可大幅節省 Amazon EC2 成本。預留執行個體並非實體執行個體，而是一種套用到您帳戶中隨需執行個體用量的計費折扣。這些 隨需執行個體 必須符合特定的屬性，像是執行個體類型和區域，才能受益於計費折扣。

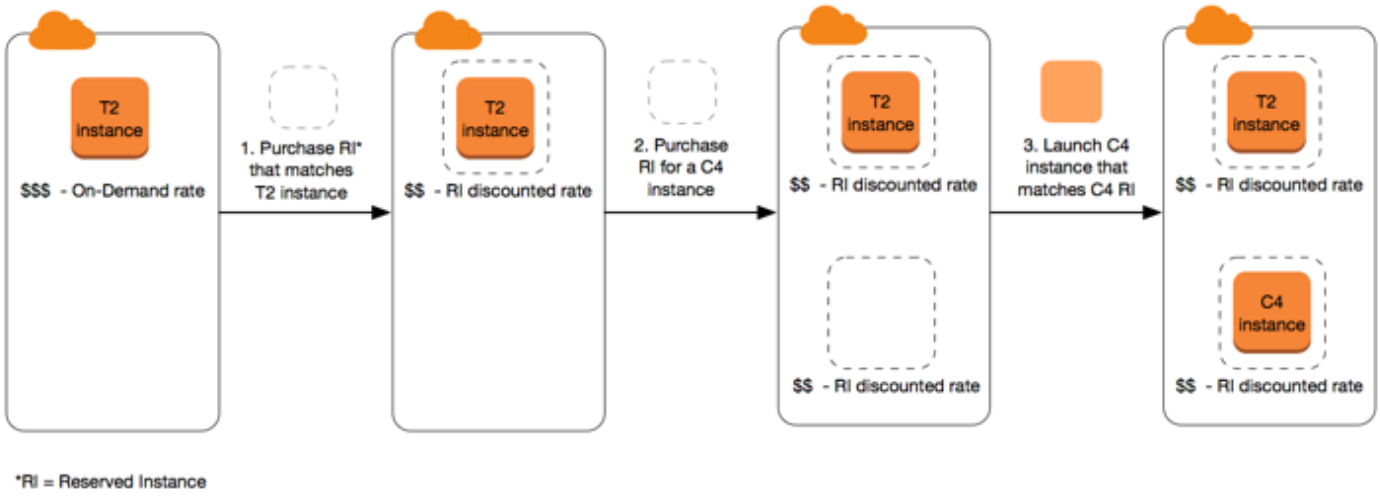
預留執行個體主題

- 預留執行個體案例
- 決定預留執行個體定價的重要變數
- 地區和區域預留執行個體 (範圍)
- 預留執行個體類型 (優惠類別)
- 預留執行個體折扣的套用方式
- 使用您的 預留執行個體

- 預留執行個體的計費方式
- 購買 Amazon 預留執行個體 EC2
- 在預留執行個體 Marketplace 場EC2中為 Amazon 銷售預留執行個體
- 修改 預留執行個體
- 交換 可轉換預留執行個體
- 預留執行個體配額

預留執行個體案例

下圖顯示購買和使用預留執行個體的基本案例。



在此案例中，您的帳戶中會有一個執行中且目前以隨需費率付款的隨需執行個體 (T2)。您購買符合您執行中執行個體屬性的預留執行個體，即會立即套用計費利益。接下來，您會為 C4 執行個體購買預留執行個體。您的帳戶中沒有任何符合此預留執行個體屬性的執行中執行個體。在最終步驟中，您會啟動符合 C4 預留執行個體屬性的執行個體，並會立即套用計費利益。

決定預留執行個體定價的重要變數

預留執行個體定價取決於以下重要變數。

執行個體屬性

預留執行個體有四個執行個體屬性可決定其價格。

- 執行個體類型：例如，`m4.large`。這是由執行個體系列 (例如 `m4`) 和執行個體大小 (例如 `large`) 所組成。

- 區域：購買 預留執行個體 的區域。
- 租用：您的執行個體是在共享 (預設) 或是單一租戶 (專用) 硬體上執行。如需詳細資訊，請參閱Amazon EC2 專用實例。
- 平台：作業系統，例如 Windows 或 Linux/Unix。如需詳細資訊，請參閱選擇平台。

期限承諾

您可以購買一年或三年承諾的預留執行個體，三年承諾提供更大折扣。

- 一年：一年定義為 31536000 秒 (365 天)。
- 三年：三年定義為 94608000 秒 (1095 天)。

預留執行個體不會自動續約；當預留執行個體到期時，您可以繼續使用EC2執行個體而不會中斷，但需支付隨需費率。在以上的範例中，當涵蓋 T2 和 C4 執行個體的預留執行個體到期時，您會重新開始支付隨需費率，直到您終止執行個體或購買符合執行個體屬性的新預留執行個體。

> **Important**
> 在您購買預留執行個體之後，您便無法取消購買。但是，若您需要變更，您可以修改、交換或販售您的預留執行個體。

付款選項

預留執行個體有下列付款選項：

- 全預付款：期間開始時便支付完整的款項，並在期間的剩餘部分不會產生其他成本或額外的每小時費用，無論使用多少小時。
- 部分預付款：必須預先支付其中一部分的成本，期間內剩餘的小時數則會以折扣後的每小時費率計費，無論是否有使用預留執行個體。
- 無預付款：您在期間中的每個小時都會以折扣後的每小時費率計費，無論是否有使用預留執行個體。不需要預付款。

> **Note**
> 無預付款預留執行個體是以合約義務為基礎，針對保留的完整期間每月付款。基於此原因，在您購買無預付款預留執行個體前，將需要成功的帳單歷史記錄。

一般來說，為預留執行個體支付較高的預付款可以節省更多費用。您也可以在預留執行個體 Marketplace 上尋找由第三方提供、價格較低且有效期間長度較短的預留執行個體。如需詳細資訊，請參閱在預留執行個體 Marketplace 場EC2中為 Amazon 銷售預留執行個體。

方案類別

若您的運算需求變更，根據優惠類別，您可能可以修改或交換您的 預留執行個體。

- 標準：這些提供更多的折扣，但只能修改。標準版 預留執行個體 無法交換。
- 可轉換：這些提供比標準預留執行個體更少的折扣，但可以與具有不同執行個體屬性的另一個可轉換預留執行個體交換。可轉換預留執行個體亦可修改。

如需詳細資訊，請參閱預留執行個體類型 (優惠類別)。

> **Important**
>
> 在您購買預留執行個體之後，您便無法取消購買。但是，若您需要變更，您可以修改、交換或販售您的預留執行個體。

如需詳細資訊，請參閱 Amazon EC2 預留執行個體定價頁面 。

地區和區域預留執行個體 (範圍)

您購買預留執行個體時會決定預留執行個體的範圍。範圍是地區或區域。

- 地區：當您為某一地區購買預留執行個體時，它便稱為地區預留執行個體。
- 區域：當您為某一特定可用區域購買預留執行個體時，它便稱為區域預留執行個體。

範圍不影響價格。您為區域或區域 預留執行個體 支付相同的價格。如需預留執行個體定價的詳細資訊，請參閱決定預留執行個體定價的重要變數和 Amazon EC2 預留執行個體定價。

如需有關如何指定預留執行個體範圍的詳細資訊，請參閱 RI 屬性，特別是可用區域項目符號。

地區與區域預留執行個體之間的差異

下表特別列出地區預留執行個體與區域預留執行個體之間的一些主要差異：

	地區預留執行個體	區域預留執行個體

	地區預留執行個體	區域預留執行個體
能夠預留容量	區域預留執行個體不會預留容量。	指定可用區域中的區域預留執行個體會預留容量。
可用區域彈性	預留執行個體折扣會套用到指定區域內任何可用區域中的執行個體用量。	無可用區域彈性 – 預留執行個體折扣僅套用到指定可用區域中的執行個體用量。
執行個體大小彈性	預留執行個體折扣會套用到執行個體系列中的執行個體用量，而無論大小。 只在使用預設租用的 Amazon Linux/Unix 預留執行個體上才支援。如需詳細資訊，請參閱由正規化因素決定的執行個體大小彈性。	無執行個體大小彈性 – 預留執行個體折扣僅套用到指定之執行個體類型和大小的執行個體用量。
將購買項目放入佇列	您可以將區域預留執行個體的購買項目放入佇列。	您無法將區域預留執行個體的購買項目放入佇列。

如需詳細資訊和範例，請參閱 預留執行個體折扣的套用方式。

預留執行個體類型 (優惠類別)

預留執行個體 的產品類別為標準版或可轉換版。標準版的 預留執行個體 提供的折扣比可轉換版的 預留執行個體 更多，但您無法交換標準版的 預留執行個體。您可以交換可轉換版的 預留執行個體。您可以修改標準版和可轉換版的 預留執行個體。

預留執行個體 的組態會在期間內包含單一執行個體類型、平台、範圍和租用。若您的運算需求變更，您可能可以修改或交換您的 預留執行個體。

標準版與可轉換版 預留執行個體 的差異

以下為標準版和可轉換版 預留執行個體 的不同。

	標準預留執行個體	Convertible Reserved Instance
修改 預留執行個體	可以修改某些屬性。如需詳細資訊，請參閱修改 預留執行個體。	可以修改某些屬性。如需詳細資訊，請參閱 修改 預留執行個體。
交換預留執行個體	無法交換。	可使用新的屬性 (包含執行個體系列、執行個體類型、平台、範圍或租用) 在另一個 可轉換預留執行個體 的期間內交換。如需詳細資訊，請參閱 交換 可轉換預留執行個體。
在預留執行個體 Marketplace 銷售	可在預留執行個體 Marketplace 销售。	不可在預留執行個體 Marketplace 销售。
在預留執行個體 Marketplace 購買	可在預留執行個體 Marketplace 購買。	不可在預留執行個體 Marketplace 購買。

預留執行個體折扣的套用方式

預留執行個體並非實體執行個體，而是一種套用到您帳戶中執行中隨需執行個體的計費折扣。隨需執行個體必須符合預留執行個體的特定規格，才能得利於計費折扣。

如果您購買預留執行個體並且已有符合預留執行個體規格的執行中預留執行個體，即會立即並自動套用計費折扣。您不需要重新啟動您的執行個體。如果您沒有符合資格的執行中隨需執行個體，請啟動具有與預留執行個體相同規格的隨需執行個體。如需詳細資訊，請參閱 使用您的 預留執行個體。

預留執行個體的方案類別 (標準或可轉換) 不會影響計費折扣的套用方式。

主題

- 區域預留執行個體的套用方式
- 區域預留執行個體的套用方式
- 執行個體大小彈性
- 套用預留執行個體的範例

區域預留執行個體的套用方式

為了在特定可用區域預留容量而購買的預留執行個體稱為區域預留執行個體。

- 預留執行個體折扣會套用到該可用區域中相符的執行個體用量。
- 執行中執行個體的屬性 (租用、平台、可用區域、執行個體類型和執行個體大小) 必須符合該預留執行個體的屬性。

例如，如果您為可用區域 us-east-1a 購買兩個 `c4.xlarge` 預設租用 Linux/Unix 標準預留執行個體，則最多能有兩個在可用區域 us-east-1a 中執行的 `c4.xlarge` 預設租用 Linux/Unix 執行個體獲得預留執行個體折扣帶來的利益。

區域預留執行個體的套用方式

為區域購買的預留執行個體稱為區域預留執行個體，並會提供可用區域和執行個體大小彈性。

- 預留執行個體折扣會套用到該區域內任何可用區域中的執行個體用量。
- 預留執行個體折扣會套用到執行個體系列中的執行個體用量，而無論大小—這稱為執行個體大小彈性。

執行個體大小彈性

由於執行個體大小彈性的因素，執行個體專屬折扣適用於具相同系列、世代和屬性的執行個體使用。根據正規化因素，預留執行個體先套用至執行個體系列中最小的執行個體，接著逐步套用至最大的執行個體。有關如何套用預留執行個體折扣的範例，請參閱 案例 2：使用正規化因素的單個帳戶中的預留執行個體。

限制

- 支援：僅區域預留執行個體支援執行個體大小彈性。
- 不支援：下列預留執行個體不支援執行個體大小彈性：
  - 為特定可用區域購買的預留執行個體 (區域預留執行個體)
  - G4ad、G4 DN、G5、G5G、G6、G6、G6e、GR6、HPC7a、P5、資訊 1 和資訊 2 執行個體的預留執行個體
  - 適用於 Windows 伺服器、含SQL標準的 Windows 伺服器、含伺服器企業的 Windows 伺服器、含SQL伺服器網頁的 Windows 伺服器RHEL，以及 SUSE Linux 企業伺SQL服器的預留執行個體
  - 搭配專用租用的預留執行個體

由正規化因素決定的執行個體大小彈性

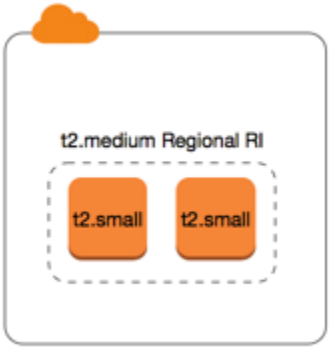
執行個體大小彈性取決於執行個體大小的正規化因素。在區域內的任何可用區域中，折扣可完整或部分套用到相同執行個體系列的執行中執行個體，而這取決於保留的執行個體大小而定。只要執行個體系列、租用和平台三個屬性符合即可。

下表列出執行個體系列中不同的大小，以及其對應的正規化因素。此規模會用於將預留執行個體折扣費率套用到執行個體系列的標準化用量。

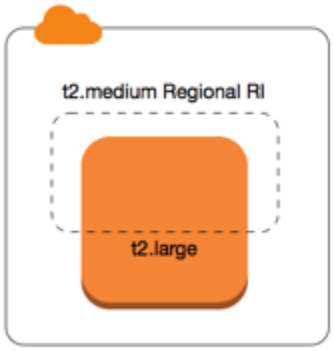
執行個體大小	正規化因素
nano	0.25
micro	0.5
small	1
medium	2
large	4
xlarge	8
2xlarge	16
3xlarge	24
4xlarge	32
6xlarge	48
8xlarge	64
9xlarge	72
10xlarge	80
12xlarge	96
16xlarge	128
18xlarge	144

執行個體大小	正規化因素
24xlarge	192
32xlarge	256
48xlarge	384
56xlarge	448
112xlarge	896

舉例而言，`t2.medium` 執行個體的正規化因素為 2。若您在US East (N. Virginia) 中購買 `t2.medium` 預設租用 Amazon Linux/Unix 預留執行個體，且在該區域中您的帳戶內有兩個執行中的 `t2.small` 執行個體，計費利益便會完整套用到兩個執行個體。



或者，若您在US East (N. Virginia) 區域中的帳戶有一個執行中的 `t2.large` 執行個體，計費利益便會套用到執行個體的 50% 用量。



正規化因數也會在修改預留執行個體時套用。如需詳細資訊，請參閱修改 預留執行個體。

裸機執行個體的正規化因素

執行個體大小彈性也套用到執行個體系列中的裸機執行個體。如果您在裸機執行個體上有搭配共用租用的地區 Amazon Linux/Unix 預留執行個體，則可享受在相同執行個體系列中節省預留執行個體。反之亦然：如果您在與裸機執行個體相同的系列中有搭配共用租用的地區 Amazon Linux/Unix 預留執行個體，則可享受在裸機執行個體系列中節省預留執行個體。

`metal` 執行個體大小沒有單一正規化因素。裸機執行個體的正規化因素與相同執行個體系列中的對等虛擬化執行個體大小相同。例如，`i3.metal` 執行個體具有與 `i3.16xlarge` 執行個體相同的正規化因素。

執行個體大小	正規化因素
`a1.metal`	32
`m5zn.metal` | `x2iezn.metal` `z1d.metal`	96
`c6g.metal` | `c6gd.metal` | `i3.metal` | `m6g.metal` | `m6gd.metal` | `r6g.metal` | `r6gd.metal` | `x2gd.metal`	128
`c5n.metal`	144
`c5.metal` | `c5d.metal` | `i3en.metal` | `m5.metal` | `m5d.metal` | `m5dn.metal` | `m5n.metal` | `r5.metal` | `r5b.metal` | `r5d.metal` | `r5dn.metal` | `r5n.metal`	192
`c6i.metal` | `c6id.metal` | `m6i.metal` | `m6id.metal` | `r6d.metal` | `r6id.metal`	256
`u-*.metal`	896

例如，`i3.metal` 執行個體的正規化因素為 128。如果您在US East (N. Virginia) 購買 `i3.metal` 預設租用 Amazon Linux/Unix 預留執行個體，則計費利益可套用如下：

- 如果您在該區域中的帳戶有一個執行中的 `i3.16xlarge`，計費利益會完全套用到 `i3.16xlarge` 執行個體 (`i3.16xlarge` 正規化因素 = 128)。
- 或者，如果您在該區域中的帳戶有兩個執行中的 `i3.8xlarge` 執行個體，計費利益會完全套用到這兩個 `i3.8xlarge` 執行個體 (`i3.8xlarge` 正規化因素 = 64)。

- 或者，如果您在該區域中的帳戶有四個執行中的 `i3.4xlarge` 執行個體，計費利益會完全套用到所有這四個 `i3.4xlarge` 執行個體 (`i3.4xlarge` 正規化因素 = 32)。

反之亦然。例如，若您在US East (N. Virginia) 購買兩個 `i3.8xlarge` 預設租用 Amazon Linux/Unix 預留執行個體，而且您在該區域中有一個執行中的 `i3.metal` 執行個體，計費利益會完全套用到 `i3.metal` 執行個體。

套用預留執行個體的範例

下列案例涵蓋套用預留執行個體的方式。

- 案例 1：單一帳戶中的預留執行個體
- 案例 2：使用正規化因素的單個帳戶中的預留執行個體
- 案例 3：連結帳戶中的區域預留執行個體
- 案例 4：連結帳戶中的區域預留執行個體

案例 1：單一帳戶中的預留執行個體

您正在帳戶 A 中執行下列隨需執行個體：

- 4 個 `m3.large` Linux，位於 us-east-1a 可用區域中的預設租用執行個體
- 2 個 `m4.xlarge` Amazon Linux，位於 us-east-1b 可用區域中的預設租用執行個體
- 1 個 `c4.xlarge` Amazon Linux，位於 us-east-1c 可用區域中的預設租用執行個體

您在帳戶 A 中購買下列預留執行個體：

- 4 個 `m3.large` Linux，位於 us-east-1a 可用區域中的預設租用預留執行個體 (容量已預留)
- 4 個 `m4.large` Amazon Linux，位於 us-east-1 區域中的預設租用預留執行個體
- 1 個 `c4.large` Amazon Linux，位於 us-east-1 區域中的預設租用預留執行個體

預留執行個體利益會以下列方式套用：

- 四個 `m3.large` 執行個體會使用四個 `m3.large` 區域預留執行個體的折扣和容量保留，因為它們之間的屬性 (執行個體大小、區域、平台、租用) 相符。
- `m4.large` 區域預留執行個體可提供可用區域和執行個體大小彈性，因為它們是使用預設租用的區域 Amazon Linux 預留執行個體。

m4.large 相當於 4 個標準化單位/小時。

您已購買四個 m4.large 區域預留執行個體，總共等於 16 個標準化單位/小時 (4x4)。帳戶 A 擁有兩個執行中的 m4.xlarge 執行個體，相當於 16 個標準化單位/小時 (2x8)。在此案例中，四個 m4.large 區域預留執行個體可為兩個 m4.xlarge 執行個體提供完整用量的計費利益。

- 位於 us-east-1 中的 c4.large 區域預留執行個體可提供可用區域和執行個體大小彈性，因為它是使用預設租用的區域 Amazon Linux 預留執行個體，並會套用到 c4.xlarge 執行個體。c4.large 執行個體相當於 4 個標準化單位/小時，c4.xlarge 則相當於 8 個標準化單位/小時。

  在此案例中，c4.large 區域預留執行個體會為 c4.xlarge 用量提供部分利益。這是因為 c4.large 預留執行個體相當於 4 個標準化單位/小時用量，但 c4.xlarge 執行個體需要 8 個標準化單位/小時。因此，c4.large 預留執行個體計費折扣會套用到 50% 的 c4.xlarge 用量。剩餘的 c4.xlarge 用量會以隨需費率收費。

案例 2：使用正規化因素的單個帳戶中的預留執行個體

您正在帳戶 A 中執行下列隨需執行個體：

- 2 個 m3.xlarge Amazon Linux，位於可用區域 us-east-1a 中的預設租用執行個體
- 2 個 m3.large Amazon Linux，位於 us-east-1b 可用區域中的預設租用執行個體

您在帳戶 A 中購買下列預留執行個體：

- 1 個 m3.2xlarge Amazon Linux，位於 us-east-1 區域中的預設租用預留執行個體

預留執行個體利益會以下列方式套用：

- 位於 us-east-1 中的 m3.2xlarge 區域預留執行個體可提供可用區域和執行個體大小彈性，因為它是使用預設租用的區域 Amazon Linux 預留執行個體。它首先套用至 m3.large 執行個體，然後套用至 m3.xlarge 執行個體，因為它會根據正規化因素，先套用至執行個體系列中最小的執行個體，接著逐步套用至最大的執行個體。

  m3.large 執行個體相當於 4 個標準化單位/小時。

  m3.xlarge 執行個體相當於 8 個標準化單位/小時。

  m3.2xlarge 執行個體相當於 16 個標準化單位/小時。

此利益適用於如下情況：

m3.2xlarge 區域預留執行個體為 2 x m3.large 用量提供全部利益，因為這些執行個體共佔用 8 個標準化單位/小時。這將留下 8 個標準化單位/小時來套用至 m3.xlarge 執行個體。

藉助剩餘的 8 個標準化單位/小時，m3.2xlarge 區域預留執行個體可為 1 x m3.xlarge 用量提供全部利益，因為每個 m3.xlarge 執行個體相當於 8 個標準化單位/小時。剩餘的 m3.xlarge 用量會以隨需費率收費。

案例 3：連結帳戶中的區域預留執行個體

預留執行個體會先套用到購買帳戶中的用量，之後則是組織內任何其他帳戶內的合格用量。如需詳細資訊，請參閱預留執行個體 與合併帳單。針對提供執行個體大小彈性的區域預留執行個體，利益會先套用至執行個體系列中最小的執行個體，接著逐步套用至最大的執行個體。

您正在帳戶 A (購買帳戶) 中執行下列隨需執行個體：

- 2 個 m4.xlarge Linux，位於 us-east-1a 可用區域中的預設租用執行個體
- 1 個 m4.2xlarge Linux，位於 us-east-1b 可用區域中的預設租用執行個體
- 2 個 c4.xlarge Linux，位於 us-east-1a 可用區域中的預設租用執行個體
- 1 個 c4.2xlarge Linux，位於 us-east-1b 可用區域中的預設租用執行個體

另一個客戶正在帳戶 B – 一個連結帳戶中執行下列隨需執行個體：

- 2 個 m4.xlarge Linux，位於 us-east-1a 可用區域中的預設租用執行個體

您在帳戶 A 中購買下列區域預留執行個體：

- 4 個 m4.xlarge Linux，位於 us-east-1 區域中的預設租用預留執行個體
- 2 個 c4.xlarge Linux，位於 us-east-1 區域中的預設租用預留執行個體

地區預留執行個體利益會以下列方式套用：

- 帳戶 A (購買帳戶) 中的兩個 m4.xlarge 執行個體及單一 m4.2xlarge 執行個體會使用四個 m4.xlarge 預留執行個體的折扣。三個執行個體全部符合屬性 (執行個體系列、區域、平台、租

用)。扣折會先套用到購買帳戶 (帳戶 A) 中的執行個體，即使帳戶 B (連結帳戶) 有兩個也符合預留執行個體的 `m4.xlarge` 也一樣。沒有容量保留，因為預留執行個體是地區預留執行個體。

- 兩個 `c4.xlarge` 預留執行個體的折扣會套用至兩個 `c4.xlarge` 執行個體，因為這兩個執行個體的大小小於 `c4.2xlarge` 執行個體。沒有容量保留，因為預留執行個體是地區預留執行個體。

案例 4：連結帳戶中的區域預留執行個體

一般而言，某帳戶擁有的預留執行個體會先套用到該帳戶中的用量。但是，若組織中其他帳戶內有合格且未使用的特定可用區域預留執行個體 (區域性預留執行個體)，便會在套用該帳戶擁有的區域預留執行個體前套用到帳戶。這是為了確保最高的預留執行個體使用率和較低的帳單。基於計費目的，組織中的所有帳戶都會視為一個帳戶處理。以下範例可協助說明。

您正在帳戶 A (購買帳戶) 中執行下列隨需執行個體：

- 1 個位於 us-east-1a 可用區域中的 `m4.xlarge` Linux 預設租用執行個體

一名客戶正在連結帳戶 B 中執行下列隨需執行個體：

- 1 個位於 us-east-1b 可用區域中的 `m4.xlarge` Linux 預設租用執行個體

您在帳戶 A 中購買下列區域預留執行個體：

- 1 個 `m4.xlarge` Linux，位於 us-east-1 區域中的預設租用預留執行個體

一名客戶也在連結帳戶 C 中購買下列區域預留執行個體：

- 1 個 `m4.xlarge` Linux，位於 us-east-1a 可用區域中的預設租用預留執行個體

預留執行個體利益會以下列方式套用：

- 帳戶 C 擁有的 `m4.xlarge` 區域性預留執行個體折扣會套用到帳戶 A 中的 `m4.xlarge` 用量。
- 帳戶 A 擁有的 `m4.xlarge` 區域預留執行個體折扣會套用到帳戶 B 中的 `m4.xlarge` 用量。
- 若帳戶 A 擁有的區域預留執行個體先套用到帳戶 A 的用量，則帳戶 C 擁有的區域性預留執行個體便會處於未使用的狀態，且帳戶 B 的用量會以隨需費率收費。

如需詳細資訊，請參閱中的瞭解您的保留項目 AWS Cost and Usage Report。

> **Note**
>
> 區域預留執行個體預留容量僅適用於擁有的帳戶，無法與其他 AWS 帳戶共用。如果您需要與其他 AWS 帳戶共用容量，請使用 透過隨需容量保留來保留運算容量。

使用您的 預留執行個體

預留執行個體會自動套用至執行中的隨需執行個體，只要其規格相符。若您沒有符合您預留執行個體規格的執行中隨需執行個體，預留執行個體會處於未使用的狀態，直到您使用必要的規格啟動執行個體為止。

如果您要啟動隨需執行個體，以利用預留執行個體的計費利益，請確定您在設定隨需執行個體時指定下列資訊：

平台

您必須指定與預留執行個體的平台 (AMI產品說明) 相符的 Amazon 機器映像 ()。例如，如果您`Linux/UNIX`為預留執行個體指定，則可以從 Amazon Linux AMI 或 Ubuntu 啟動執行個體AMI。

執行個體類型

如果購買了區域預留執行個體，您必須指定與預留執行個體相同的執行個體類型，例如 `t3.large`。如需詳細資訊，請參閱 區域預留執行個體的套用方式。

如果購買了區域預留執行個體，您必須從預留執行個體之執行個體的同一執行個體系列指定執行個體類型。例如，如果針對預留執行個體指定了 `t3.xlarge`，您必須從 T3 系列啟動執行個體，但可以指定任何大小，例如 `t3.medium`。如需詳細資訊，請參閱 區域預留執行個體的套用方式。

可用區域

如果針對特定可用區域購買了區域預留執行個體，您必須將執行個體啟動到相同的可用區域。

如果購買了區域預留執行個體，您可以將執行個體啟動到區域中針對預留執行個體指定的任何可用區域。

租用

執行個體的租用 (`dedicated` 或 `shared`) 必須符合預留執行個體的租用。如需詳細資訊，請參閱 Amazon EC2 專用實例。

如需預留執行個體如何套用到執行中隨需執行個體的範例，請參閱預留執行個體折扣的套用方式。如需詳細資訊，請參閱為什麼我的 Amazon 預EC2留執行個體無法按照我預期的方式套用到我的 AWS 帳單？

您可以使用各種方法，啟動使用預留執行個體折扣的隨需執行個體。如需不同啟動方法的詳細資訊，請參閱 啟動 Amazon EC2 實例。您也可以使用 Amazon EC2 Auto Scaling 來啟動執行個體。如需詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南。

預留執行個體的計費方式

所有預留執行個體都可提供您相較於隨需定價的折扣。您可透過預留執行個體，為整個期間支付費用 (無論實際用量為何)。您可以選擇為您的預留執行個體預付款項、部分預付款項或每月付款，取決於為預留執行個體指定的付款選項。

預留執行個體到期時，系統會向您收取EC2執行個體用量的隨需費率。最早可提前三年將 預留執行個體 放入佇列供購買。這可協助您確保您的涵蓋範圍不會間斷。如需詳細資訊，請參閱將您的購買項目放入佇列。

可 AWS 免費方案 用於新的 AWS 帳戶。如果您使用執行 Amazon EC2 執行個體，並購買預留執行個體，則 AWS 免費方案 需支付標準定價的費用。如需相關資訊，請參閱 AWS 免費方案。

目錄

- 用量帳單
- 檢視您的帳單
- 預留執行個體 與合併帳單
- 預留執行個體折扣定價方案

用量帳單

預留執行個體會在您選取的期間內以每時鐘小時計價，無論執行個體是否正在執行中。每個鐘點小時都會於標準 24 小時制的整點 (每小時的零分和零秒) 開始。例如，1:00:00 至 1:59:59 是一點鐘。如需執行個體狀態的詳細資訊，請參閱Amazon EC2 實例狀態更改。

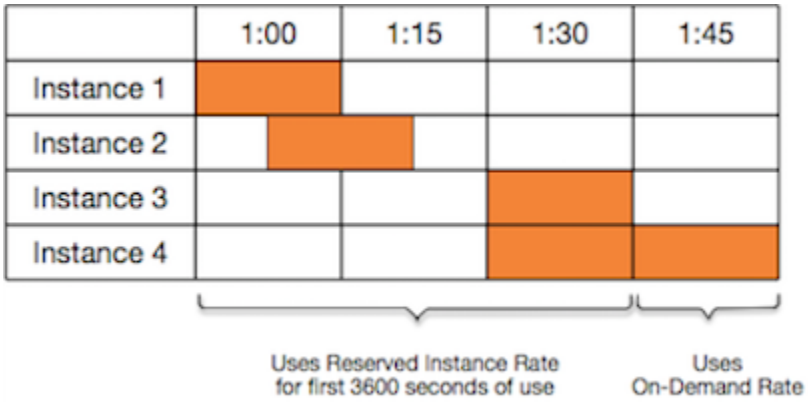
預留執行個體計費利益可以每秒套用到執行中的執行個體。以秒計費適用於使用開放原始碼 Linux 發行版本 (例如 Amazon Linux 和 Ubuntu) 的執行個體。每小時計費是用於商業 Linux 發行版，例如 RHEL 和 SUSE Linux 企業伺服器。

預留執行個體計費利益每個時鐘小時可套用到最多 3600 秒 (一小時) 的執行個體用量。您可以同時執行多個執行個體，但每個時鐘小時最多只能接收到 3600 秒的預留執行個體折扣利益；時鐘小時內超過 3600 秒的執行個體用量會以隨需費率計費。

例如，若您購買一個 `m4.xlarge` 預留執行個體，並同時執行四個 `m4.xlarge` 執行個體一個小時，一個執行個體會以預留執行個體用量收取一個小時的費用，其餘三個執行個體則會以隨需用量收取三個小時的費用。

但是，若您購買一個 `m4.xlarge` 預留執行個體，並在同一個小時內執行四個 `m4.xlarge` 執行個體，每一個執行 15 分鐘 (900 秒)，執行個體的總執行時間便為一個小時，其結果便是一個小時的預留執行個體用量和 0 個小時的隨需用量。

若同時執行多個符合資格的執行個體，預留執行個體計費利益會在一個時鐘小時內同時套用到所有執行個體，最多達 3600 秒；在這之後，便會套用隨需費率。



Billing and Cost Management 主控台上的 Cost Explorer 可讓您根據執行中的隨需執行個體分析節省費用。預留執行個體FAQ包括清單值計算範例。

如果您關閉 AWS 帳戶，資源的隨需計費就會停止。但是，若您的帳戶中有任何預留執行個體，您會繼續收到其帳單，直到它們到期為止。

檢視您的帳單

您可以透過檢視 AWS Billing and Cost Management 主控台了解您帳戶所需支付的費用。

- Dashboard (儀表板) 會顯示您帳戶的支出摘要。
- 在 Bills (帳單) 頁面上，於 Details (詳細資訊) 下方，展開 Elastic Compute Cloud 區段及區域，以取得預留執行個體的帳單資訊。

您可以在線查看費用，也可以下載CSV文件。

您也可以使用 AWS 成本和用量報告追蹤預留執行個體使用率。如需詳細資訊，請參閱 AWS Billing 使用者指南中『成本及用量報告』下的預留執行個體。

預留執行個體 與合併帳單

若購買帳戶是在單一合併帳單付款人帳戶下一組帳戶的一部分，則會共用預留執行個體的計費利益。所有成員帳戶的執行個體用量每個月都會彙總到付款人的帳戶中。這通常對有不同功能團隊或群組的公司來說非常有用；然後，會套用標準預留執行個體邏輯計算帳單。如需詳細資訊，請參閱 AWS Organizations的合併帳單。

如果您關閉購買預留執行個體的帳戶，系統會繼續為預留執行個體對付款人帳戶計費，直到預留執行個體到期。在關閉的帳戶於 90 天後永久刪除後，會員帳戶無法再從預留執行個體帳單折扣中受益。

> **Note**
>
> 區域預留執行個體預留容量僅適用於擁有的帳戶，無法與其他 AWS 帳戶共用。如果您需要與其他 AWS 帳戶共用容量，請使用 透過隨需容量保留來保留運算容量。

預留執行個體折扣定價方案

若您的帳戶符合折扣定價方案的資格，從您在該方案層級中購買預留執行個體的時間點開始，預付及執行個體使用費便會自動接收到折扣。若要符合 discount 資格，該區域中預留執行個體的定價值必須為 500,000 美元USD以上。

適用的規定如下：

- 定價層級和相關折扣僅適用於購買 Amazon EC2 標準預留執行個體。
- 定價層不適用於具有SQL伺服器標準、伺服器 Web 和SQL伺服SQL器企業的 Windows 的預留執行個體。
- 定價層不適用於具有SQL伺服器標準、伺服器 Web 和SQL伺SQL服器企業的 Linux 預留執行個體。
- 定價等級折扣僅適用於從購買的商品 AWS。它們不會套用到購買的第三方預留執行個體。
- 折扣定價方案目前無法套用到可轉換預留執行個體購買。

主題

- 計算 預留執行個體 定價折扣
- 使用折扣方案購買
- 跨定價方案
- 定價方案的合併帳單

計算 預留執行個體 定價折扣

您可以藉由計算您在區域中所有預留執行個體的清單值，來決定帳戶的定價方案。將每個保留每小時的經常性價格，乘上期間內的總小時數，並加上在購買時間點的未折扣預付價格 (也稱為固定價格)。因為清單值是以未折扣 (公有) 定價為基礎，無論您是否符合批發折扣，或是價格是否在您購買預留執行個體之後降低，都不會受到影響。

```
List value = fixed price + (undiscounted recurring hourly price * hours in term)
```

例如，針對 1 年部分預付款的 `t2.small` 預留執行個體，假設預付價格為 60.00 USD，每小時的費率為 0.007 USD。這會提供清單值 121.32 USD。

```
121.32 = 60.00 + (0.007 * 8760)
```

New console

使用 Amazon EC2 主控台檢視預留執行個體的固定價格值

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)。
3. 要顯示預付價格列，請選擇右上角的設置 (⚙)，打開預付價格，然後選擇確認。

Old console

使用 Amazon EC2 主控台檢視預留執行個體的固定價格值

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)。

3. 要顯示預付價格列，請選擇右上角的設置（ ），選擇預付價格，然後選擇關閉。

使用命令列檢視預留執行個體的固定價格值

- describe-reserved-instances (AWS CLI)
- Get-EC2ReservedInstance (AWS Tools for Windows PowerShell)
- DescribeReservedInstances(Amazon EC2API)

使用折扣方案購買

當您購買預留執行個體時，Amazon EC2 會自動將任何 discount 套用到您購買折扣定價層級內的部分。您不需要以不同的方式執行任何操作，而且可以使用任何 Amazon EC2 工具購買預留執行個體。如需詳細資訊，請參閱購買 Amazon 預留執行個體 EC2。

當您在區域中的作用中預留執行個體清單值進入折扣定價方案後，日後在該區域購買的任何預留執行個體都會以折扣費率計費。若您在區域中購買的單一預留執行個體超過折扣方案的閾值，則該購買超過價格閾值的部分便會以折扣費率收費。如需在購買程序期間建立IDs之暫時預留執行個體的詳細資訊，請參閱跨定價方案。

若您的清單值低於該折扣定價方案的價格點 (例如：若您的某些預留執行個體到期)，則日後在該區域中購買的預留執行個體將不會有折扣。但是，您原先在折扣定價方案內購買的預留執行個體，仍會繼續套用折扣。

當您購買預留執行個體時，便會發生下列四種可能案例中的其中一種：

- 無折扣 – 您在區域內的購買仍低於折扣閾值。
- 部分折扣 – 您在區域內的購買已超過第一個折扣方案的閾值。不會套用折扣到一或多個保留，剩餘的保留則會套用折扣費率。
- 完整折扣 – 您在區域內的整個購買都已符合單一折扣方案，並已取得適當折扣。
- 雙折扣費率 – 您在區域內的購買已從較低的折扣方案升等至更高的折扣方案。會根據兩種不同費率向您收費：一或多個保留會以較低的折扣費率計費，其餘保留則會以更高的折扣費率計費。

跨定價方案

若您的購買進入折扣定價方案，您會看到多個該購買的項目：其中一個為該購買中以標準價格收費的部分，另外一個則為該購買中以適用折扣費率收費的部分。

預留執行個體服務會產生數個預留執行個體，IDs因為您的購買是從未折扣方案或從一個折扣方案轉移到另一個方案。方案中每一組保留都會有一個 ID。因此，您購買CLI命令或API動作傳回的 ID 與新預留執行個體的實際 ID 不同。

定價方案的合併帳單

合併帳單帳戶會彙總區域內成員帳戶的清單值。當合併帳單帳戶所有作用中預留執行個體的清單值達到折扣定價方案，合併帳單帳戶的任何成員帳戶在之後購買的預留執行個體都會以折扣費率計費 (只要合併帳戶的清單值持續高於折扣定價方案的閾值)。如需詳細資訊，請參閱預留執行個體 與合併帳單。

購買 Amazon 預留執行個體 EC2

若要為 Amazon 購買預留執行個SDK體EC2，您可以使用 Amazon EC2 主控台、命令列工具或搜尋來自 AWS 和第三方賣家的預留執行個體產品，調整搜尋參數，直到找到所需的完全相符項目為止。

當您在尋找要購買的預留執行個體時，您會收到傳回優惠的成本報價。當您繼續進行購買時， AWS 會自動對購買價格設定限制價格。您預留執行個體的總成本不會超過向您報價的金額。

若價格因為任何原因產生變更，購買便無法完成。當您從 Amazon 預留執行個體 Marketplace 購買第三方賣家的EC2預留執行個體時，如果有與您選擇類似的產品，但預付價格較低，則會以較低的前期價格向您 AWS 銷售產品。

在您確認購買前，請檢閱您計劃購買之預留執行個體的詳細資訊，並確認所有的參數都正確。在您購買預留執行個體之後 (無論是從預留執行個體 Marketplace 的第三方賣家或從 AWS)，您就無法取消購買。您可以將購買項目排入 future 日期，並在排定的時間之前取消已排入佇列的購買項目。

若要購買和修改預留執行個體，請確保使用者具有適當的許可，例如描述可用區域的能力。如需詳細資訊，請參閱 the section called “使用 預留執行個體” (API) 或 the section called “使用 預留執行個體” (主控台)。

主題

- 選擇平台
- 將您的購買項目放入佇列
- 購買標準版 預留執行個體

- 購買 可轉換預留執行個體
- 從預留執行個體 Marketplace 購買
- 檢視您的 預留執行個體
- 取消佇列購買
- 續約 預留執行個體

選擇平台

Amazon 針對預留執行個體EC2支援下列平台：

- Linux/ UNIX
- Linux 與SQL伺服器標準
- Linux 與網頁SQL伺服器
- Linux 與企業SQL伺服器
- SUSELinux
- Red Hat Enterprise Linux
- 帶 HA 的 Red Hat Enterprise Linux
- Windows
- 帶SQL伺服器標準的視窗
- 帶有 Web SQL 服務器的窗口
- 視窗與企業SQL伺服器

當您購買預留執行個體時，您必須選擇代表您執行個體作業系統之平台的優惠。

Linux 執行個體

- 對於 SUSE RHEL Linux 和發行版，您必須為這些特定平台選擇供應項目，例如 SUSELinux 或 RHEL 平台的供應項目。
- 對於所有其他 Linux 發行版本 (包括 Ubuntu)，請選擇適用於 Linux /平UNIX台的供應項目。
- 如果您使用現有的RHEL訂閱，您必須選擇 Linux/ UNIX 平台的供應項目，而不是 RHEL 平台的供應項目。

Windows 執行個體

- 對於含SQL標準版的視窗、含SQL伺服器企業版的 Windows，以及具有SQL伺服器網頁版的 Windows，您必須選擇這些特定平台的供應項目。
- 針對所有其他的 Windows 版本，請選擇 Windows 平台的優惠。

> **Note**
>
> Ubuntu Pro 不可作為預留執行個體使用。與隨需執行個體定價相比，可節省大量成本，建議您搭配使用 Ubuntu Pro 與 Savings Plans。如需詳細資訊，請參閱 Savings Plans 使用者指南。

> **Important**
>
> 如果您計劃購買預留執行個體以套用至從啟動的隨需執行個體 AWS Marketplace AMI，請先檢查的`PlatformDetails`欄位AMI。此 `PlatformDetails` 欄位會指出要購買的 預留執行個體 項目。的平台詳細資訊AMI必須與預留執行個體的平台相符，否則預留執行個體將不會套用至隨需執行個體。如需有關如何檢視的平台詳細資訊AMI，請參閱了解 AMI 帳單資訊。

將您的購買項目放入佇列

根據預設，當您購買 預留執行個體 時，購買會立即進行。或者，您可以將購買放入未來日期和時間的佇列。例如，您可以將購買放入約為現有 預留執行個體 到期時的佇列。這可協助您確保您的涵蓋範圍不會間斷。

您可以將購買放入地區 預留執行個體 的佇列，但不是其他賣方的區域 預留執行個體 或 預留執行個體。最早可提前三年將購買放入佇列。在排程日期和時間，購買會依據預設的付款方式進行。付款成功之後，計費優勢便會套用。

您可以在 Amazon EC2 主控台中為已排入佇列的購買項目設定日期，購買項目會排入佇列，直到此 UTC日期的 00:00 為止。若要為排入佇列的購買指定不同的時間，請使用 AWS SDK或指令行工具。

您可以在 Amazon EC2 主控台中檢視已排入佇列的購買項目。佇列購買的狀態為 queued (佇列)。在排程時間之前，您可以隨時取消佇列購買。如需詳細資訊，請參閱 取消佇列購買。

購買標準版 預留執行個體

您可以在特定可用區域中購買標準預留執行個體，取得容量保留。或者，您可以放棄容量保留，購買地區標準預留執行個體。

New console

使用主控台購買標準預留執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)，然後選擇 Purchase 預留執行個體 (購買 預留執行個體)。
3. 針對 Offering class (優惠類別)，選擇 Standard (標準) 以顯示標準 預留執行個體。
4. 若要購買容量保留，請在購買畫面的右上角開啟 Only show offerings that reserve capacity (只顯示預留容量的優惠)。當您切換此設定時，會出現 Availability Zone (可用區域) 欄位。

   若要購買區域 預留執行個體，請關閉此設定。當您關閉此設定時，Availability Zone (可用區域) 欄位即會消失。
5. 視需要選取其他組態，然後選取 Search (搜尋)。
6. 在您想要購買的每一個 預留執行個體 中，輸入所需數量，然後選擇 Add to cart (新增到購物車)。

   若要從預留執行個體市場購買標準預留執行個體，請在搜尋結果中的 Seller (賣方) 列中尋找 3rd party (第三方)。Term (期間) 資料行會顯示非標準期間。如需詳細資訊，請參閱從預留執行個體 Marketplace 購買。
7. 若要查看您選取之預留執行個體的摘要，請選取 View cart (檢視購物車)。
8. 如果 Order on (訂購日期) 為 Now (立即)，則在您選擇 Order all (全部訂購) 之後會立即完成購買。若要將購買加入佇列，請選取 Now (立即) 並選取日期。您可以為購物車中每項合格產品選取不同的日期。UTC在所選日期的 00:00 之前，購買將排入佇列。
9. 若要完成訂單，請選擇 Order all (全部訂購)。

   如果在下訂單時，有類似於您選擇的產品，但價格較低，則會以較低的價格向您 AWS 銷售產品。
10. 選擇 Close (關閉)。

    訂單的狀態會列在 State (狀態) 資料行中。當您的訂單完成時，State (狀態) 的值便會從 `Payment-pending` 變更為 `Active`。當預留執行個體為 `Active` 時，即已準備好可供使用。

> **Note**
>
> 如果狀態顯示為`Retired`， AWS 可能尚未收到您的付款。

Old console

使用主控台購買標準預留執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)，然後選擇 Purchase 預留執行個體 (購買 預留執行個體)。
3. 針對 Offering Class (優惠類別)，選擇 Standard (標準) 以顯示標準預留執行個體。
4. 若要購買容量保留，請在購買畫面的右上角選擇 Only show offerings that reserve capacity (只顯示預留容量的優惠)。若要購買地區預留執行個體，請讓核取方塊保持在未勾選的狀態。
5. 視需要選取其他組態，然後選取 Search (搜尋)。

   若要從預留執行個體市場購買標準預留執行個體，請在搜尋結果中的 Seller (賣方) 列中尋找 3rd Party (第三方)。Term (期間) 資料行會顯示非標準期間。
6. 在您想要購買的每一個 預留執行個體 中，輸入數量，然後選擇 Add to Cart (新增到購物車)。
7. 若要查看您選取之預留執行個體的摘要，請選取 View Cart (檢視購物車)。
8. 如果 Order On (訂購時間) 為 Now (立即)，便會立刻完成購買。若要將購買加入佇列，請選取 Now (立即) 並選取日期。您可以為購物車中每項合格產品選取不同的日期。UTC在所選日期的 00:00 之前，購買將排入佇列。
9. 若要完成訂單，請選擇 Order (訂購)。

   如果在下訂單時，有類似於您選擇的產品，但價格較低，則會以較低的價格向您 AWS 銷售產品。
10. 選擇 Close (關閉)。

    訂單的狀態會列在 State (狀態) 資料行中。當您的訂單完成時，State (狀態) 的值便會從 `payment-pending` 變更為 `active`。當預留執行個體為 `active` 時，即已準備好可供使用。

> **Note**
>
> 如果狀態顯示為retired， AWS 可能尚未收到您的付款。

若要使用 AWS CLI

1. 使用describe-reserved-instances-offerings命令尋找可用的預留執行個體。為 `--offering-class` 參數指定 `standard`，以僅傳回標準預留執行個體。您可以套用其他參數來縮小結果範圍。例如，若您希望購買使用預設租用 `t2.large` 為期僅一年的區域 `Linux/UNIX` 預留執行個體：

```
aws ec2 describe-reserved-instances-offerings \
    --instance-type t2.large \
    --offering-class standard \
    --product-description "Linux/UNIX" \
    --instance-tenancy default \
    --filters Name=duration,Values=31536000 Name=scope,Values=Region
```

   若只尋找預留執行個體 Marketplace 上的預留執行個體，請使用 `marketplace` 篩選條件，並且不要在請求中指定有效期間，因為有效期間可能會短於 1 年或 3 年。

```
aws ec2 describe-reserved-instances-offerings \
    --instance-type t2.large \
    --offering-class standard \
    --product-description "Linux/UNIX" \
    --instance-tenancy default \
    --filters Name=marketplace,Values=true
```

   當您找到符合您需求的預留執行個體時，請記下產品 ID。例如：

```
"ReservedInstancesOfferingId": "bec624df-a8cc-4aad-a72f-4f8abc34caf2"
```

2. 使用指purchase-reserved-instances-offering令購買預留執行個體。您必須指定您在先前步驟中取得的預留執行個體優惠 ID，並指定保留的執行個體數目。

```
aws ec2 purchase-reserved-instances-offering \
    --reserved-instances-offering-id bec624df-a8cc-4aad-a72f-4f8abc34caf2 \
    --instance-count 1
```

依預設，購買會立刻完成。或者，若要將購買加入佇列，則在先前的呼叫中新增下列參數。

```
--purchase-time "2020-12-01T00:00:00Z"
```

3. 使用命describe-reserved-instances令取得預留執行個體的狀態。

```
aws ec2 describe-reserved-instances
```

或者，使用下列 AWS Tools for Windows PowerShell 指令：

- Get-EC2ReservedInstancesOffering
- New-EC2ReservedInstance
- Get-EC2ReservedInstance

購買完成之後，若您已有符合預留執行個體規格的執行中執行個體，即會立即套用計費優勢。您不需要重新啟動您的執行個體。若您沒有適合的執行中執行個體，請啟動執行個體，確保符合您在預留執行個體中指定的條件。如需詳細資訊，請參閱使用您的 預留執行個體。

如需預留執行個體套用到您執行中執行個體方式的範例，請參閱預留執行個體折扣的套用方式。

購買 可轉換預留執行個體

您可以在特定可用區域中購買可轉換預留執行個體，取得容量保留。或者，您可以放棄容量保留並購買地區可轉換預留執行個體。

New console

使用主控台購買可轉換預留執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)，然後選擇 Purchase 預留執行個體 (購買 預留執行個體)。
3. 針對 Offering class (優惠類別)，選擇 Convertible (可轉換) 以顯示標準 可轉換預留執行個體。
4. 若要購買容量保留，請在購買畫面的右上角開啟 Only show offerings that reserve capacity (只顯示預留容量的優惠)。當您切換此設定時，會出現 Availability Zone (可用區域) 欄位。

若要購買區域 預留執行個體，請關閉此設定。當您關閉此設定時，Availability Zone (可用區域) 欄位即會消失。

5. 視需要選取其他組態，然後選取 Search (搜尋)。
6. 在您想要購買的每一個 可轉換預留執行個體 中，輸入數量，然後選擇 Add to cart (新增到購物車)。
7. 若要查看您選取項目的摘要，請選取 View cart (檢視購物車)。
8. 如果 Order on (訂購日期) 為 Now (立即)，則在您選擇 Order all (全部訂購) 之後會立即完成購買。若要將購買加入佇列，請選取 Now (立即) 並選取日期。您可以為購物車中每項合格產品選取不同的日期。UTC在所選日期的 00:00 之前，購買將排入佇列。
9. 若要完成訂單，請選擇 Order all (全部訂購)。

   如果在下訂單時，有類似於您選擇的產品，但價格較低，則會以較低的價格向您 AWS 銷售產品。
10. 選擇 Close (關閉)。

    訂單的狀態會列在 State (狀態) 資料行中。當您的訂單完成時，State (狀態) 的值便會從 `Payment-pending` 變更為 `Active`。當預留執行個體為 `Active` 時，即已準備好可供使用。

> **Note**
>
> 如果狀態顯示為`Retired`， AWS 可能尚未收到您的付款。

Old console

使用主控台購買可轉換預留執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)，然後選擇 Purchase 預留執行個體 (購買 預留執行個體)。
3. 針對 Offering Class (優惠類別)，選擇 Convertible (可轉換) 以顯示標準可轉換預留執行個體。
4. 若要購買容量保留，請在購買畫面的右上角選擇 Only show offerings that reserve capacity (只顯示預留容量的優惠)。若要購買地區預留執行個體，請讓核取方塊保持在未勾選的狀態。
5. 視需要選取其他組態，然後選取 Search (搜尋)。

6. 在您想要購買的每一個 可轉換預留執行個體 中，輸入數量，然後選擇 Add to Cart (新增到購物車)。
7. 若要查看您選取項目的摘要，請選取 View Cart (檢視購物車)。
8. 如果 Order On (訂購時間) 為 Now (立即)，便會立刻完成購買。若要將購買加入佇列，請選取 Now (立即) 並選取日期。您可以為購物車中每項合格產品選取不同的日期。UTC在所選日期的 00:00 之前，購買將排入佇列。
9. 若要完成訂單，請選擇 Order (訂購)。

   如果在下訂單時，有類似於您選擇的產品，但價格較低，則會以較低的價格向您 AWS 銷售產品。
10. 選擇 Close (關閉)。

    訂單的狀態會列在 State (狀態) 資料行中。當您的訂單完成時，State (狀態) 的值便會從 `payment-pending` 變更為 `active`。當預留執行個體為 `active` 時，即已準備好可供使用。

> **Note**
>
> 如果狀態顯示為`retired`， AWS 可能尚未收到您的付款。

使用購買可轉換預留執行個體 AWS CLI

1. 使用describe-reserved-instances-offerings命令尋找可用的預留執行個體。為 `--offering-class` 參數指定 `convertible`，以僅傳回可轉換預留執行個體。您可以套用額外的參數縮減結果，例如，若您希望購買使用預設租用 `t2.large` 的區域 `Linux/UNIX` 預留執行個體：

```
aws ec2 describe-reserved-instances-offerings \
    --instance-type t2.large \
    --offering-class convertible \
    --product-description "Linux/UNIX" \
    --instance-tenancy default \
    --filters Name=scope,Values=Region
```

   當您找到符合您需求的預留執行個體時，請記下產品 ID。例如：

```
"ReservedInstancesOfferingId": "bec624df-a8cc-4aad-a72f-4f8abc34caf2"
```

2. 使用指purchase-reserved-instances-offering令購買預留執行個體。您必須指定您在先前步驟中取得的預留執行個體優惠 ID，並指定保留的執行個體數目。

```
aws ec2 purchase-reserved-instances-offering \
    --reserved-instances-offering-id bec624df-a8cc-4aad-a72f-4f8abc34caf2 \
    --instance-count 1
```

依預設，購買會立刻完成。或者，若要將購買加入佇列，則在先前的呼叫中新增下列參數。

```
--purchase-time "2020-12-01T00:00:00Z"
```

3. 使用命describe-reserved-instances令取得預留執行個體的狀態。

```
aws ec2 describe-reserved-instances
```

或者，使用下列 AWS Tools for Windows PowerShell 指令：

- Get-EC2ReservedInstancesOffering
- New-EC2ReservedInstance
- Get-EC2ReservedInstance

若您已有符合預留執行個體規格的執行中執行個體，即會立即套用計費利益。您不需要重新啟動您的執行個體。若您沒有適合的執行中執行個體，請啟動執行個體，確保符合您在預留執行個體中指定的條件。如需詳細資訊，請參閱使用您的 預留執行個體。

如需預留執行個體套用到您執行中執行個體方式的範例，請參閱預留執行個體折扣的套用方式。

從預留執行個體 Marketplace 購買

您可以從第三方賣方購買預留執行個體，他們擁有預留執行個體 Marketplace 不再需要的預留執行個體。您可以使用 Amazon EC2 主控台或命令列工具執行此操作。此程序類似於從中購買預留執行個體 AWS。如需詳細資訊，請參閱購買標準版 預留執行個體。

在預留執行個體 Marketplace 場購買的預留執行個體與直接從下列地點購買的預留執行個體有一些差異 AWS：

- 有效期間 – 您從第三方賣方購買的預留執行個體具有低於完整標準有效期間的剩餘時間。從 AWS 運行一年或三年的完整標準條款。

- 預付價格 – 第三方預留執行個體可使用不同的預付價格出售。使用費或經常性費用仍保持不變，等同於最初從 AWS購買預留執行個體時所設定的費用。
- 預留執行個體類型 — 只能從預留執行個體 Marketplace 購買 Amazon EC2 標準預留執行個體。可轉換預留執行個體RDS、Amazon 和 Amazon ElastiCache 預留執行個體無法在預留執行個體 Marketplace 上購買。

賣家會與賣家分享你的基本資料，例如你的ZIP代碼和國家/地區資料。

此資訊可讓賣方計算任何需要匯款給政府的必要交易稅 (例如營業稅或加值稅)，並會做為收款報告提供。在極少數情況下，賣家可 AWS 能必須向賣家提供你的電郵地址，以便他們就銷售相關問題（例如稅務問題）與你聯絡。

出於類似原因，在買方的購買發票上 AWS 共享賣家的法定實體名稱。若您因為稅務或其他相關原因，需要賣方的額外資訊，請聯絡 AWS Support。

檢視您的 預留執行個體

您可以使用 Amazon EC2 主控台或命令列工具檢視已購買的預留執行個體。

在主控台中檢視您的預留執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)。
3. 您的已排入佇列、作用中和已淘汰 預留執行個體 會在此列出。State (狀態) 資料行會顯示狀態。
4. 若您是預留執行個體 Marketplace 中的賣方，My Listings (我的清單) 標籤會顯示在 Reserved Instance Marketplace (預留執行個體 Marketplace) 中列出的預留狀態。如需詳細資訊，請參閱預留執行個體清單狀態。

使用命令列檢視您的預留執行個體

- describe-reserved-instances (AWS CLI)
- Get-EC2ReservedInstance(視窗工具 PowerShell)

取消佇列購買

最早可提前三年將購買放入佇列。在排程時間之前，您可以隨時取消佇列購買。

取消佇列購買

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)。
3. 選取一或多個 預留執行個體。
4. 選擇 Actions (動作)、Delete queued Reserved Instances (刪除佇列預留執行個體)。
5. 出現確認提示時，依序選擇 Delete (刪除)、Close (關閉)。

使用命令列取消佇列的購買

- delete-queued-reserved-instances (AWS CLI)
- Remove-EC2QueuedReservedInstance(視窗工具 PowerShell)

續約 預留執行個體

您可以在預留執行個體排定到期前續約。續約預留執行個體會將具有相同預留執行個體組態的購買排入佇列，直到當前預留執行個體到期為止。

使用主控台使用排入佇列的購買來續約預留執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)。
3. 選取要續約的預留執行個體。
4. 選擇 Actions (動作)、Renew Reserved Instances (續約預留執行個體)。
5. 若要完成訂單，請選擇 Order all (全部訂購)，然後選擇 Close (關閉)。

在預留執行個體 Marketplace 場EC2中為 Amazon 銷售預留執行個體

Amazon EC2 預留執行個體 Marketplace 是一個平台，可協助 AWS 客戶和第三方賣家銷售未使用的標準預留執行個體。這些預留執行個體的期限長度和定價選項可能有所不同。當您不再需要預留執行個體時，您可能會想要銷售預留執行個體，例如當您將執行個體移至新執行個體時 AWS 區域、變更為不同的執行個體類型、在預留執行個體期限到期前完成專案、業務需求變更，或者您的容量過多。

在預留執行個體 Marketplace 中列出您的預留執行個體後，它們便可供潛在買方尋找。所有預留執行個體都會根據剩餘期間的長度和每小時的價格分組。

若要滿足買方透過預留執行個體 Marketplace 購買第三方賣家預留執行個體的要求，請 AWS 先以指定群組中最低的預付價格出售預留執行個體。然後，以下一個最低的價格 AWS 出售預留執行個體，直到買方的整筆訂單出貨為止。 AWS 然後處理交易，並將預留執行個體的擁有權移轉給買方。

直到賣出之前，您都會擁有您的預留執行個體。在銷售之後，您便已放棄容量保留和折扣後的經常性費用。若您繼續使用您的執行個體， AWS 會從您賣出預留執行個體的時間開始，以隨需價格向您收費。

若您希望在預留執行個體 Marketplace 中銷售您未使用的預留執行個體，您必須符合特定適用性條件。

關於在預留執行個體 Marketplace 上購買預留執行個體的資訊，請參閱 從預留執行個體 Marketplace 購買。

內容

- 法規與限制
- 註冊為賣家
- 用於收款的銀行帳戶
- 稅務資訊
- 為您的 預留執行個體 定價
- 列出您的 預留執行個體
- 預留執行個體清單狀態
- 清單的生命週期
- 在您的預留執行個體售出後
- 收取款項
- 與買方共享的資訊

法規與限制

在您販售未使用的預留前，您必須先在預留執行個體 Marketplace 中註冊為賣方。如需相關資訊，請參閱 註冊為賣家。

以下限制會在販售預留執行個體時適用：

- 只有 Amazon EC2 標準區域和區域預留執行個體可以在預留執行個體 Marketplace 銷售。
- Amazon 可EC2轉換預留執行個體無法在預留執行個體 Marketplace 銷售。
- 其他 AWS 服務的預留執行個體 (例如 Amazon RDS 和 Amazon ElastiCache) 無法在預留執行個體 Marketplace 銷售。

- 標準預留執行個體的剩餘期間至少要一個月。
- 您無法在預設停用的區域銷售標準預留執行個體。
- 預留執行個體 Marketplace 允許的最低價為 0.00 美元。
- 您可以在預留執行個體 Marketplace 中銷售無預留、部分預付或全部預付等執行個體，只要其在您的帳戶中處於活動狀態至少 30 天。此外，如果預留執行個體需要預付款，則只有在收到預付款後 AWS 才能出售該筆預付款。
- 如果您使用大量 discount 購買預留執行個體，就無法在預留執行個體 Marketplace 中銷售預留執行個體。
- 您無法直接在預留執行個體 Marketplace 中修改您的清單。但是，您可以藉由先取消它，然後再使用新的參數建立另一個清單，來變更您的清單。如需相關資訊，請參閱 為您的 預留執行個體 定價。您也可以在將它們列出之前修改您的預留執行個體。如需相關資訊，請參閱「修改 預留執行個體」。
- AWS 收取您在預留執行個體 Marketplace 銷售的每個標準預留執行個體預付總價的 12% 的服務費。預付價格為賣方向標準預留執行個體收取的價格。
- 當您註冊成為賣方時，您指定的銀行必須具有美國地址。如需詳細資訊，請參閱 AWS Marketplace 賣方指南中的付費產品的其他賣方需求。
- Amazon Web Services 印度私人有限公司 (AWS 印度) 客戶無法在預留執行個體 Marketplace 銷售預留執行個體，即使他們擁有美國的銀行帳戶也一樣。如需詳細資訊，請參閱AWS 帳戶 和 AWS India 帳戶之間有何差異？

註冊為賣家

> **Note**
>
> 只有 AWS 帳戶根使用者 可以註冊一個帳戶作為賣家。

若要在預留執行個體 Marketplace 中進行銷售，您必須先註冊為賣方。在註冊期間，您會提供下列資訊：

- 銀行信息 —AWS 必須擁有您的銀行信息，以便在您出售預訂時支付收集的資金。您指定的銀行必須具有美國地址。如需詳細資訊，請參閱用於收款的銀行帳戶。
- 稅務資訊 – 所有賣方都必須完成稅務資訊面試，才能判斷任何必要的稅務報告義務。如需詳細資訊，請參閱稅務資訊。

AWS 收到完成的賣家註冊後，您會收到一封確認註冊的電子郵件，並通知您可以開始在預留執行個體 Marketplace 銷售產品。

用於收款的銀行帳戶

AWS 必須擁有您的銀行資訊，才能支付銷售預留執行個體時所收集的資金。您指定的銀行必須具有美國地址。如需詳細資訊，請參閱 AWS Marketplace 賣方指南中的付費產品的其他賣方需求。

註冊預設銀行帳戶以用於接收款項

1. 開啟預留執行個體市場賣方註冊頁面，然後使用您的 AWS 登入資料登入。
2. 在 Manage Bank Account (管理銀行帳戶) 頁面上，提供下列關於接收款項銀行的資訊：
   - 銀行帳戶持有人名稱
   - 路由號碼
   - 帳戶號碼
   - 銀行帳戶類型

> **Note**
> 若您使用的是企業銀行帳戶，您會收到提示，請您透過傳真 (1-206-765-3424) 傳送銀行帳戶的資訊。

在註冊之後，提供的銀行帳戶便會設為預設，等待銀行進行驗證。驗證新的銀行帳戶最多可能需要兩週的時間，在此期間您無法收到款項。針對已建立的帳戶，支付款項通常需要兩天的時間才能完成。

變更接收款項的預設銀行帳戶

1. 在預留執行個體市場賣方註冊頁面上，使用您用來註冊的帳戶登入。
2. 在 Manage Bank Account (管理銀行帳戶) 頁面上，新增銀行帳戶或視需要修改銀行帳戶。

稅務資訊

您販售預留執行個體的行為必須遵守交易式稅務的規範，例如營業稅或加值稅。建議您和您公司的稅務、法律、財務或會計部門確認，以判斷是否適用於交易式的稅務。您必須負責收集和將交易式的稅務傳送給適當的稅務主管機關。

做為賣方註冊程序的一部分，您必須在賣方註冊入口網站中完成稅務面試。面談會收集您的稅務資訊，並填入 W-9、W-8 或 W BEN -8-E IRS 表格BEN，該表格用於確定任何必要的稅務申報義務。

您做為稅務面試一部分所輸入的稅務資訊可能會有所不同，取決於您是否是以個人或公司運作，以及您或您的公司是否為美國或非美國的個人或實體。當您填寫稅務面試時，請謹記下列幾點：

- 提供的資訊 (包括本主題中的資訊) 並不構成稅務、法律或其他專業建議。 AWS若要瞭解IRS報告要求可能會對您的業務造成什麼影響，或者您有其他問題，請聯絡您的稅務、法律或其他專業顧問。
- 為了盡可能有效地滿足IRS報告要求，請回答所有問題並輸入面試期間要求的所有信息。
- 檢查您的答案。避免拼字錯誤或輸入不正確的稅務識別號碼。這可能會導致無效的稅務表單。

根據您的稅務面試回應和IRS報告閾值，Amazon 可能會提交 1099-K 表格。 Amazon 在您的稅務帳戶達到閾值水平的翌年 1 月 31 日或之前郵寄您的 1099-K 表的副本。例如，若您的帳戶在 2018 年達到閾值，則您的 1099-K 表單會在 2019 年 1 月 31 日或之前寄送給您。

有關IRS要求和 1099-K 表的更多信息，請訪問網站。IRS

為您的 預留執行個體 定價

設定預留執行個體的價格時，請考慮下列事項：

- 預付價格 - 預付價格是您可以為要販售之預留執行個體指定的唯一價格。預付價格是買方購買預留執行個體時所需支付的一次性價格。

  由於預留執行個體的值會隨著時間的推移而減少，因此依預設， AWS 可以將價格設定為每月以相等的遞增方式降低。但是，您可以根據販售您保留的時間，設定不同的預付價格。例如，若您的預留執行個體具有九個月的剩餘期間，您可以指定客戶購買具有九個月剩餘期間的該預留執行個體時，您所能接受的價格。您可以為剩餘期間五個月設定另一個價格，以及為剩餘期間一個月設定另一個價格。

  預留執行個體 Marketplace 允許的最低價為 0.00 美元。
- 限制 - 銷售預留執行個體的以下限制適用於 AWS 帳戶的生命週期。這不是年度限制。
  - 您可以在預留執行個體賣出最多 50,000 美元。
  - 您最多可以賣 5,000 個預留執行個體

  這些限制通常無法增加，但會case-by-case 根據要求進行評估。如需請求提高限制，請完成提高服務限制表單。針對限制類型，選擇EC2預留執行個體銷售。
- 無法修改 - 您無法直接修改清單。但是，您可以藉由先取消它，然後再使用新的參數建立另一個清單，來變更您的清單。

- 可以取消 – 只要清單處於 `active` 狀態，即可隨時取消您的清單。若清單已相符或正在為銷售進行處理，您便無法取消清單。若您清單中的某些執行個體已相符且您取消了清單，則只有剩餘未相符的執行個體會從清單中移除。

列出您的 預留執行個體

身為已註冊賣方，您可以選擇銷售一或多個預留執行個體。您可以選擇一次出售全部或部分出售。此外，您可以使用任何執行個體類型、平台和範圍組態，列出預留執行個體。

主控台會決定建議價格。它會檢查符合 預留執行個體 的優惠，並匹配最低價格的產品。否則會根據 預留執行個體 剩餘時間的成本來計算建議價格。如果計算的值低於 1.01 USD，則建議價格為 1.01 USD。

若您取消您的清單，但該清單中的一部分已賣出，則取消對已經賣出的部分不會有任何效果。只有清單中尚未賣出的部分才會無法在預留執行個體 Marketplace 中取得。

若要在預留執行個體 Marketplace 中使用 AWS Management Console

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)。
3. 選取要列出的預留執行個體，然後選取 Actions (動作)、Sell 預留執行個體 (販售 預留執行個體)。
4. 在 Configure Your 預留執行個體 Listing (設定您的預留執行個體清單) 頁面上，在相關資料行中設定要販售的執行個體數目，以及剩餘期間的預付價格。選取 Months Remaining (剩餘月數) 資料行，查看您保留的值隨剩餘期間變化的方式。
5. 若您是進階使用者希望自訂定價，您可以為後續月份輸入不同的值。若要返回預設的線性價格下降，請選擇 Reset (重設)。
6. 當您完成設定清單後，請選擇 Continue (繼續)。
7. 在 Confirm Your 預留執行個體 Listing (確認您的預留執行個體清單) 頁面上確認您清單的詳細資訊，若您已感到滿意，請選擇 List Reserved Instance (列出預留執行個體)。

在主控台中檢視您的清單

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)。
3. 選取您已列出的預留執行個體，並選取頁面底部附近的 My Listings (我的清單) 索引標籤。

使用以管理預留執行個體 Marketplace 中的預留執行個體 AWS CLI

1. 使用describe-reserved-instances命令取得預留執行個體的清單。
2. 請記下您要列出並呼叫的預留執行個體 ID create-reserved-instances-listing。您必須指定預留執行個體的 ID、執行個體數目，以及定價排程。
3. 若要檢視刊登物品，請使用describe-reserved-instances-listings指令。
4. 如要取消刊登物品，請使用cancel-reserved-instances-listings指令。

預留執行個體清單狀態

預留執行個體頁面的 My Listings (我的清單) 標籤上的 Listing State (清單狀態) 會顯示您清單目前的狀態：

Listing State (清單狀態) 顯示的資訊與預留執行個體市場中的清單狀態有關。它與 Reserved Instances (預留執行個體) 中 State (狀態) 資料行顯示的狀態資訊不同。此 State (狀態) 資訊與您的保留有關。

- active (作用中) – 清單可供購買。
- canceled (已取消) - 清單已取消，無法在預留執行個體市場中購買。
- closed (已關閉) – 預留執行個體並未列出。預留執行個體可能會因為清單的銷售已完成而為 `closed` 狀態。

清單的生命週期

當您清單中的所有執行個體都已符合並售出，My Listings (我的清單) 標籤便會顯示 Total instance count (總執行個體計數) 符合 Sold (已售出) 下方列出的計數。此外，您的清單中將不會有任何剩餘 Available (可用) 的執行個體，且其 Status (狀態) 為 `closed`。

當您的清單中只有一部分銷售時，會 AWS 淘汰清單中的預留執行個體，並建立與計數中剩餘的預留執行個體相等的預留執行個體數量。因此，清單 ID 和其代表的清單 (目前具有較少銷售中的保留) 仍會處於作用中。

任何之後此清單內發生的預留執行個體銷售都會使用相同的方式處理。出售清單中的所有預留執行個體後，將清單 AWS 標示為`closed`。

例如，您建立 預留執行個體 listing ID 5ec28771-05ff-4b9b-aa31-9e57dexample，其清單計數為 5。

Reserved Instance (預留執行個體) 主控台頁面中的 My Listings (我的清單) 標籤會以此方式顯示清單：

Reserved Instance listing ID 5ec28771-05ff-4b9b-aa31-9e57dexample

- 總保留計數 = 5
- 售出 = 0
- 可用 = 5
- 狀態 = 作用中

買方購買兩個保留，使剩餘可供銷售的保留計數為 3。由於此部分銷售，因此 AWS 會建立一個計數為三的新保留項目，以代表剩餘仍待售的保留項目。

這是在 My Listings (我的清單) 標籤中您清單的外觀：

Reserved Instance listing ID 5ec28771-05ff-4b9b-aa31-9e57dexample

- 總保留計數 = 5
- 售出 = 2
- 可用 = 3
- 狀態 = 作用中

若您取消您的清單，但該清單中的一部分已賣出，則取消對已經賣出的部分不會有任何效果。只有清單中尚未賣出的部分才會無法在預留執行個體 Marketplace 中取得。

在您的預留執行個體售出後

售出預留執行個體時， AWS 會傳送電子郵件通知給您。每天若有任何類型的活動，您都會接收到一封電子郵件通知，擷取一天內所有的活動。您可以在建立或出售刊登物品時，或是 AWS 將資金匯入您的帳戶時進行的活動。

在主控台追蹤預留執行個體清單的狀態

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)。
3. 選擇 My Listings (我的清單) 索引標籤。

   My Listings (我的清單) 標籤包含 Listing State (清單狀態) 值。它同時也包含期間、清單價格，清單中有多少可用、擱置中、已售出和已取消執行個體解析的相關資訊。

您也可以使用describe-reserved-instances-listings指令搭配適當的篩選器，以取得刊登物品的相關資訊。

收取款項

一旦 AWS 收到買方的款項，系統就會向已售出預留執行個體的註冊擁有者帳戶電子郵件傳送訊息。

AWS 發送自動結算所（ACH）電匯到您指定的銀行帳戶。通常，這種匯款會在賣出您預留執行個體的一到三天後發生。每天會進行一次匯款。您將收到一封電子郵件，含有釋出基金後的收款報告。請記住，在收到銀行的驗證之前，您將無法 AWS 收到支付款項。這可能需要兩週的時間。

您賣出的預留執行個體會繼續在您描述您的預留執行個體時出現。

您可以透過電匯直接存入您的銀行帳戶，為預留執行個體收到現金支付。 AWS 收取您在預留執行個體 Marketplace 銷售的每個預留執行個體預付總價格的 12% 的服務費。

與買方共享的資訊

當您在預留執行個體 Marketplace 銷售產品時， AWS 會根據美國法規在買方聲明上提供貴公司的法定名稱。此外，若買方因為需要聯絡您取得發票或其他與稅務相關的原因而呼叫 AWS Support ， AWS 可能會需要向買方提供您的電子郵件地址，以便買方能夠直接聯絡您。

基於類似原因，買家代ZIP碼和國家/地區資訊會在支付報告中提供給賣家。做為賣方，您可能需要此資訊伴隨任何您匯款給政府的必要交易稅 (例如營業稅和加值稅)。

AWS 無法提供稅務建議，但如果您的稅務專家認為您需要特定的額外資訊，請聯絡 AWS Support。

修改 預留執行個體

當您的需求變更時，您可以修改您的標準或可轉換預留執行個體，並繼續獲得計費利益。您可以修改預留執行個體的屬性，例如可用區域、執行個體大小 (位於相同執行個體系列和層代內) 和範圍。

> **Note**
>
> 您也可以將可轉換預留執行個體與另一個具有不同組態的可轉換預留執行個體交換。如需詳細資訊，請參閱交換 可轉換預留執行個體。

您也可以修改所有預留執行個體或子集。您可以將原始預留執行個體分隔為兩個或多個新的預留執行個體。例如，若您在 `us-east-1a` 中有 10 個執行個體的保留，並決定將 5 個執行個體移動到 `us-east-1b`，修改請求結果會導致兩個新的保留：其中一個保留針對位於 `us-east-1a` 中的 5 個執行個體，另外一個保留則針對位於 `us-east-1b` 的 5 個執行個體。

您也可以將兩個或多個預留執行個體合併為單一預留執行個體。例如，若您有四個 `t2.small` 預留執行個體，每一個都具有一個執行個體，您可以合併它們來建立一個 `t2.large` 預留執行個體。如需詳細資訊，請參閱支援執行個體大小的修改。

在修改後，預留執行個體的利益僅會套用到符合新參數的執行個體。例如，若您變更保留的可用區域，容量保留和定價利益會自動套用到新可用區域中的執行個體用量。不再符合新參數的執行個體會以隨需費率計費，除非您的帳戶有其他適用的保留。

若您的修改請求成功：

- 修改後的保留會立即生效，定價利益會在修改請求的小時內開始套用到新的執行個體。例如，若您在下午 9:15 成功修改您的保留，定價利益會在下午 9:00 傳送到您新的執行個體。您可以使用describe-reserved-instances命令取得已修改預留執行個體的有效日期。
- 原始的保留會淘汰。其結束日期便是新保留的開始日期，新保留的結束日期則與原始預留執行個體的結束日期相同。若您修改具有 16 個月剩餘期間的三年保留，其結果產生的修改後保留便是 16 個月的保留，其結束日期與原先的保留相同。
- 修改後的保留列出的固定價格為 0 USD，而非原始保留的固定價格。
- 修改後保留的固定價格不會影響套用到您帳戶的折扣定價方案計算，因為該計算是以原始保留的固定價格為基礎。

若您的修改請求失敗，您的預留執行個體將會維持其原始的組態，並可立即進行另一次的修改請求。

修改無須支付任何費用，您不會收到任何新的帳單或發票。

您可以無限次數的修改您的保留，但在您提交後，您便無法變更或取消擱置中的修改請求。在修改成功完成後，您可以提交另一個修改請求，轉返任何您進行的變更 (若需要的話)。

內容

- 修改的要求和限制
- 支援執行個體大小的修改
- 提交修改請求
- 故障診斷修改請求

修改的要求和限制

您可以修改這些屬性如下。

可修改屬性	支援平台	限制及考量
變更相同區域內的可用區域	Linux 與 Windows	-
將範圍從「可用區域」變更為「區域」，反之亦然	Linux 與 Windows	區域性預留執行個體的範圍限定為可用區域，並在該可用區域中預留容量。若您將範圍從可用區域變更為區域 (換言之，從區域到地區)，您將會喪失容量保留利益。 地區性預留執行個體的範圍限定為區域。您的預留執行個體折扣可套用到該區域內任何可用區域中正在執行的執行個體。此外，預留執行個體折扣適用於選定執行個體系列中所有大小的執行個體用量。若您將範圍從區域變更為可用區域 (換言之，從地區到區域)，您將會喪失可用區域彈性和執行個體大小彈性 (若適用)。 如需詳細資訊，請參閱預留執行個體折扣的套用方式。
在相同執行個體系列和層代內變更執行個體大小	僅限 Linux/ UNIX 其他平台上的預留執行個體無法使用執行個體大小彈性，包括 Linux 與SQL伺服器標準版、Linux 與SQLSQL伺服器網頁版、RHEL 企業版 Linux、SUSE Linux、Windows SQL 標準版視窗、含SQL伺服器企業版的 Windows，	保留必須使用預設租用。某些執行個體系列不受支援，因為沒有其他可用的大小。如需詳細資訊，請參閱 支援執行個體大小的修改

可修改屬性	支援平台	限制及考量
	以及帶SQL伺服器 Web 的 Windows。	

要求

如果您的新組態有足夠的容量 (如果適用)，且符合下列條件，Amazon 會EC2處理您的修改請求：

- 預留執行個體在您購買之前或當時無法修改
- 預留執行個體必須為作用中
- 不能有擱置的修改請求
- 預留執行個體 Marketplace 中未列出預留執行個體。
- 原始保留的執行個體大小使用量和新組態必須相符。如需詳細資訊，請參閱支援執行個體大小的修改。
- 原始 預留執行個體 全部為標準版 預留執行個體 或全部為 可轉換預留執行個體，而非每個類型各有一些
- 原始 預留執行個體 如果是標準版 預留執行個體，則必須在同一小時內過期
- 預留執行個體必須支援執行個體大小彈性。如需不支援執行個體大小彈性的預留執行個體清單，請參閱執行個體大小彈性。

支援執行個體大小的修改

若符合下列要求，您可以修改預留執行個體的執行個體大小。

要求

- 該平台是 Linux UNIX 的。
- 您必須選取相同執行個體系列 (用字母表示，例如 T) 和層代 (用數字表示，例如 2) 中的其它執行個體大小。

  例如，您可以將預留執行個體從 `t2.small` 修改為 `t2.large`，因為它們都位於相同的 T2 系列和層代中。但是您無法將預留執行個體從 T2 修改為 M2，或從 T2 修改為 T3，因為在這兩個範例中，目標執行個體系列和層代與原始預留執行個體不同。
- 只有在預留執行個體大小彈性支援執行個體大小時，才能修改預留執行個體的執行個 如需不支援執行個體大小彈性的預留執行個體清單，請參閱執行個體大小彈性。

- 您無法修改執行個體預留執行個體的`t1.micro`執行個體大小，因為只`t1.micro`有一個大小。
- 原始和新的 預留執行個體 必須具有相同的執行個體大小使用量。

內容

- 執行個體大小使用量
- 裸機執行個體的正規化因素

執行個體大小使用量

每一個預留執行個體都具有執行個體大小使用量，由執行個體大小的正規化因素和保留中的執行個體數目決定。當您修改 預留執行個體 中的執行個體大小時，新組態的使用量必須符合原始組態的使用量，否則不會處理修改請求。

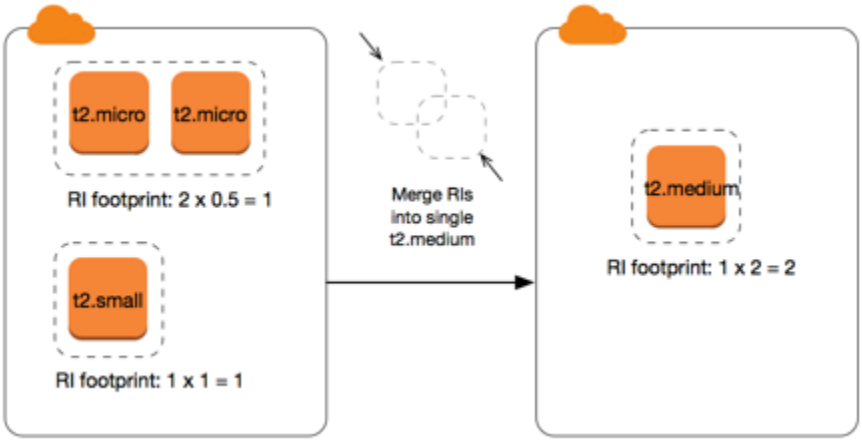
若要計算預留執行個體的執行個體大小使用量，請將執行個體的數目乘上正規化因素。在 Amazon 主EC2控台中，標準化係數以單位測量。下表說明執行個體系列中執行個體大小的正規化因素。例如，`t2.medium` 的正規化因素為 2，因此四個 `t2.medium` 執行個體的保留具有 8 個單位的使用量。

執行個體大小	正規化因素
nano	0.25
micro	0.5
small	1
medium	2
large	4
xlarge	8
2xlarge	16
3xlarge	24
4xlarge	32
6xlarge	48

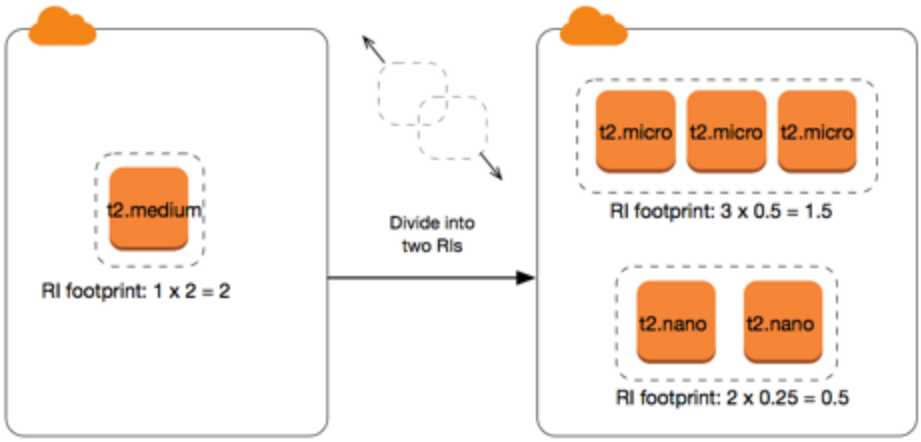
執行個體大小	正規化因素
8xlarge	64
9xlarge	72
10xlarge	80
12xlarge	96
16xlarge	128
18xlarge	144
24xlarge	192
32xlarge	256
48xlarge	384
56xlarge	448
112xlarge	896

您可以將您的保留在相同執行個體系列中配置到不同的執行個體大小，只要您保留的執行個體大小使用量維持不變。例如，您可以將一個 `t2.large` (1 @ 4 個單位) 執行個體的保留分割為四個 `t2.small` (4 @ 1 個單位) 執行個體。同樣地，您可以將四個 `t2.small` 執行個體的保留合併為一個 `t2.large` 執行個體。不過，您無法將兩個 `t2.small` 執行個體的保留變更為一個 `t2.large` 執行個體，因為新保留的使用量 (4 個單位) 大於原始保留的使用量 (2 個單位)。

在下列範例中，您有兩個 `t2.micro` 執行個體的保留 (1 個單位)，以及一個 `t2.small` 執行個體的保留 (1 個單位)。如果您將這兩個保留合併為具有一個 `t2.medium` 執行個體 (2 個單位) 的單一保留，新保留的使用量等於合併保留的使用量。



您也可以修改保留，將其分割為兩個或多個保留。在下列範例中，您有一個具有 `t2.medium` 執行個體 (2 個單位) 的保留。您可以將保留分割為兩個保留，一個具有兩個 `t2.nano` 執行個體 (5 個單位)，另一個具有三個 `t2.micro` 執行個體 (1.5 個單位)。



裸機執行個體的正規化因素

您可以使用相同執行個體系列內的其他大小，修改具有 `metal` 執行個體的保留。同樣地，您可以使用相同執行個體系列內的 `metal` 大小，修改具有裸機執行個體之外的執行個體的保留。一般而言，裸機執行個體的大小與相同執行個體系列中的最大可用執行個體大小相同。例如，`i3.metal` 執行個體的大小與 `i3.16xlarge` 執行個體相同，因此有相同的正規化因素。

下表說明具有裸機執行個體之執行個體系列中，裸機執行個體大小的正規化因素。與其他執行個體大小不同的是，`metal` 執行個體的正規化因素取決於執行個體系列。

執行個體大小	正規化因素
`a1.metal`	32

執行個體大小	正規化因素
m5zn.metal | x2iezn.metal z1d.metal	96
c6g.metal | c6gd.metal | i3.metal | m6g.metal | m6gd.metal | r6g.metal | r6gd.metal | x2gd.metal	128
c5n.metal	144
c5.metal | c5d.metal | i3en.metal | m5.metal | m5d.metal | m5dn.metal | m5n.metal | r5.metal | r5b.metal | r5d.metal | r5dn.metal | r5n.metal	192
c6i.metal | c6id.metal | m6i.metal | m6id.metal | r6d.metal | r6id.metal	256
u-*.metal	896

例如，i3.metal 執行個體的正規化因素為 128。如果您購買 i3.metal 預設租用 Amazon Linux/Unix 預留執行個體，您可以將保留分割如下：

- i3.16xlarge 的大小與 i3.metal 執行個體相同，因此正規化因素為 128 (128/1)。一個 i3.metal 執行個體的保留可以修改為一個 i3.16xlarge 執行個體。
- i3.8xlarge 的大小是 i3.metal 執行個體的一半，因此正規化因素為 64 (128/2)。一個 i3.metal 執行個體的保留可以分割成兩個 i3.8xlarge 執行個體。
- i3.4xlarge 的大小是 i3.metal 執行個體的四分之一，因此正規化因素為 32 (128/4)。一個 i3.metal 執行個體的保留可以分割成四個 i3.4xlarge 執行個體。

提交修改請求

在修改預留執行個體之前，請確定您已閱讀適用的限制。在修改執行個體大小之前，請計算要修改之原始保留項目的總執行個體大小佔用空間，並確保其與新規劃的執行個體大小總佔用空間相符。

若要使用修改預留執行個體 AWS Management Console

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在 Reserved Instances (預留執行個體) 頁面上，選取一或多個要修改的預留執行個體，然後選取 Actions (動作)，Modify Reserved Instances (修改預留執行個體)。

> **Note**
>
> 若您的預留執行個體並未處於作用中的狀態或無法修改，Modify 預留執行個體 (修改預留執行個體) 便會停用。

3. 修改表格的第一個項目會顯示選取 預留執行個體 的屬性，以及至少一個位於下方的目標組態。Units (單位) 資料行會顯示總執行個體大小使用量。針對每個要新增的新組態，選擇 Add (新增)。視需要修改每個組態的屬性。
   - Scope (範圍)：選擇組態是否套用到可用區域或是整個區域。
   - Availability Zone (可用區域)：選擇必要的可用區域。不適用於區域預留執行個體。
   - 執行個體類型：選取所需的執行個體類型。結合的組態必須等於原始組態的執行個體大小使用量。
   - Count (計數)：指定執行個體的數目。若要將 預留執行個體 分割為多個組態，請減少計數，選擇 Add (新增)，然後指定額外組態的計數。例如，如果您有一個計數為 10 的組態，您可以將其計數變更為 6，然後新增計數為 4 的組態。此程序會在新的 預留執行個體 啟動之後重試原始的預留執行個體。
4. 選擇 Continue (繼續)。
5. 若要在完成指定您的目標組態後確認您的修改選擇，請選擇 Submit modifications (提交修改)。
6. 您可以透過查看預留執行個體畫面中的 State (狀態) 資料行，判斷您修改請求的狀態。以下為可能的狀態。
   - 作用中 (待定修改) — 原始預留執行個體的轉移狀態
   - 已淘汰 (待定修改) – 建立新的預留執行個體時，原始預留執行個體的轉移狀態
   - 淘汰 – 已成功修改並取代預留執行個體
   - 作用中 – 下列其中一項：
     - 已成功從修改請求建立新的預留執行個體
     - 修改請求失敗後的原始預留執行個體

使用命令列檢視您的預留執行個體

1. 若要修改您的預留執行個體，您可以使用下列其中一個命令：
   - modify-reserved-instances (AWS CLI)
   - Edit-EC2ReservedInstance (AWS Tools for Windows PowerShell)

2. 若要取得修改狀態的狀態 (`processing`、`fulfilled` 或 `failed`)，請使用下列其中一個命令：
   - describe-reserved-instances-modifications (AWS CLI)
   - Get-EC2ReservedInstancesModification (AWS Tools for Windows PowerShell)

故障診斷修改請求

若您請求的目標組態設定為獨特，您會收到訊息，告知您請求正在處理中。此時，Amazon EC2 僅確定修改請求的參數有效。您的修改請求仍然可能會在處理過程中因為不可用的容量而失敗。

在某些情況下，您可能會收到訊息，表示修改請求未完成或失敗，而非確認。請使用此類訊息中的資訊，做為重新提交另一個修改請求的出發點。確認您已在提交請求前詳閱適用的限制。

並非所有選取的預留執行個體都可進行修改處理

Amazon 會EC2識別並列出無法修改的預留執行個體。如果您收到類似這樣的訊息，請前往 Amazon EC2 主控台中的預留執行個體頁面，查看預留執行個體的資訊。

處理您的修改請求時發生錯誤

您提交一或多個預留執行個體進行修改，但您的請求全部都無法處理。取決於您要修改的保留數目，您可能會收到不同版本的訊息。

Amazon EC2 顯示了無法處理您的請求的原因。例如，您可能為一或多個您要修改的預留執行個體子集指定了相同的目標組態 (可用區域和平台的組合)。請嘗試再次提交修改請求，但請確認保留的執行個體詳細資訊確實符合，且每個要修改子集的目標組態都是唯一的。

交換 可轉換預留執行個體

您可以將一或多個可轉換預留執行個體與另一個具有不同組態的可轉換預留執行個體交換，包括執行個體系列、作業系統和租用。執行交換的次數沒有限制，只要新可轉換預留執行個體的值等於或高於您要交換的可轉換預留執行個體即可。

當您交換您的可轉換預留執行個體時，您目前保留中的執行個體數目會和涵蓋新可轉換預留執行個體組態相同或更高值的執行個體數目交換。Amazon EC2 會計算您在交換後可獲得的預留執行個體數量。

您無法交換標準版 預留執行個體，但您可以修改它們。如需詳細資訊，請參閱 修改 預留執行個體。

內容

- 交換可轉換預留執行個體的需求

- 計算 可轉換預留執行個體 交換
- 合併 可轉換預留執行個體
- 交換 可轉換預留執行個體 的一部分
- 提交交換請求

交換可轉換預留執行個體的需求

如果符合以下條件，Amazon 會EC2處理您的換貨請求。您的可轉換預留執行個體必須是：

- 作用中
- 並未擱置先前的修改請求
- 在到期前至少還剩 24 小時

適用的規定如下：

- 可轉換預留執行個體只能與另一個目前由 AWS提供的可轉換預留執行個體交換。
- 可轉換預留執行個體與特定區域相關聯，該區域在保留的期間內固定不變。您無法和位於不同區域的可轉換預留執行個體交換可轉換預留執行個體。
- 您只能一次為一個可轉換預留執行個體交換一或多個可轉換預留執行個體。
- 若要交換可轉換預留執行個體的一部分，您可以修改它，將它分成兩個或多個保留，然後為新的可轉換預留執行個體交換一或多個保留。如需詳細資訊，請參閱交換 可轉換預留執行個體 的一部分。如需修改預留執行個體的詳細資訊，請參閱修改 預留執行個體。
- 所有預付可轉換預留執行個體都可和部分預付可轉換預留執行個體交換，反之亦然。

> **Note**
>
> 如果交換所需的預付款總額 (實付成本) 低於 $0.00， AWS 會自動在可轉換預留執行個體中提供一定數量的執行個體，以確保實際成本為 $0.00 或更多。

> **Note**
>
> 如果新可轉換預留執行個體的總價值 (前期價格 + 小時價格 * 剩餘小時數) 小於交換的可轉換預留執行個體的總價值，則 AWS 會自動為您提供可轉換預留執行個體中的執行個體數量，以確保總值與交換的可轉換預留執行個體相同或高於該值。

- 若要從更佳的定價獲得利益，您可以將無預付款可轉換預留執行個體交換為全預付款或部分預付款可轉換預留執行個體。
- 您無法以全預付款及部分預付款可轉換預留執行個體交換無預付款可轉換預留執行個體。
- 只有在新可轉換預留執行個體的每小時價格等於或高於交換可轉換預留執行個體的每小時價格時，您才能以無預付款可轉換預留執行個體交換另一個無預付款可轉換預留執行個體。

> **Note**
>
> 若新可轉換預留執行個體的總值 (每小時價格 * 剩餘小時數) 低於交換可轉換預留執行個體的總值， AWS 會自動在可轉換預留執行個體中給予您執行個體的數量，確保總值等於或高於交換可轉換預留執行個體的值。

- 若您交換多個具有不同過期日期的可轉換預留執行個體，新可轉換預留執行個體的過期日期將會是距離現在最遠的日期。
- 若您交換單一可轉換預留執行個體，它必須與新可轉換預留執行個體具有相同的期間 (1 年或 3 年)。若您合併多個具有不同期間長度的可轉換預留執行個體，新的可轉換預留執行個體將具有 3 年的期間。如需詳細資訊，請參閱合併 可轉換預留執行個體。
- Amazon EC2 交換可轉換預留執行個體時，會淘汰相關的保留，並將結束日期轉移到新的保留項目。交換後，Amazon EC2 將舊預訂的結束日期和新預訂的開始日期設定為等於交換日期。例如，若您交換具有 16 個月剩餘期間的三年預留，新預留是 16 個月預留，其結束日期與您交換的可轉換預留執行個體的預留相同。

計算 可轉換預留執行個體 交換

交換可轉換預留執行個體是免費的。但是，您可能需要支付校正成本，即您擁有的原始可轉換預留執行個體與您從交換中獲得新可轉換預留執行個體之間的差異，依比例分配的預付成本。

每個可轉換預留執行個體都具有清單值。此清單值會和您希望取得的可轉換預留執行個體清單值進行比較，以判斷您可以從交換中接收到多少執行個體保留。

例如：您有 1 個清單值為 35 USD 的可轉換預留執行個體，希望和清單值為 10 USD 的執行個體類型交換。

```
$35/$10 = 3.5
```

您可以使用三個 10 USD 的可轉換預留執行個體交換您的可轉換預留執行個體。因為您無法購買半個保留，因此您必須購買額外的可轉換預留執行個體才能涵蓋剩餘的部分：

```
3.5 = 3 whole Convertible Reserved Instances + 1 additional Convertible Reserved
 Instance
```

第四個可轉換預留執行個體具有和其他三個可轉換預留執行個體相同的結束日期。若您要交換部分或全預付可轉換預留執行個體，您必須為第四個保留支付校正成本。若您 可轉換預留執行個體 剩餘的預付成本為 500 USD，則依比例分配，新保留通常會需要花費 600 USD。您將需要支付 100 USD 的費用。

```
$600 prorated upfront cost of new reservations - $500 remaining upfront cost of old
 reservations = $100 difference
```

合併 可轉換預留執行個體

若您合併兩個或多個可轉換預留執行個體，新可轉換預留執行個體的期間必須和舊可轉換預留執行個體相同，或為舊可轉換預留執行個體中最高的值。新可轉換預留執行個體的過期日期將會是距離現在最遠的日期。

例如，您的帳戶中有下列可轉換預留執行個體：

Reserved Instance ID	術語	過期日期
aaaa1111	1 年	2018-12-31
bbbb2222	1 年	2018-07-31
cccc3333	3 年	2018-06-30
dddd4444	3 年	2019-12-31

- 您可以合併 aaaa1111 和 bbbb2222，並以它們交換 1 年的可轉換預留執行個體。您無法以它們交換 3 年的可轉換預留執行個體。新可轉換預留執行個體的過期日期為 2018-12-31。
- 您可以合併 bbbb2222 和 cccc3333，並以它們交換 3 年的可轉換預留執行個體。您無法以它們交換 1 年的可轉換預留執行個體。新可轉換預留執行個體的過期日期為 2018-07-31。
- 您可以合併 cccc3333 和 dddd4444，並以它們交換 3 年的可轉換預留執行個體。您無法以它們交換 1 年的可轉換預留執行個體。新可轉換預留執行個體的過期日期為 2019-12-31。

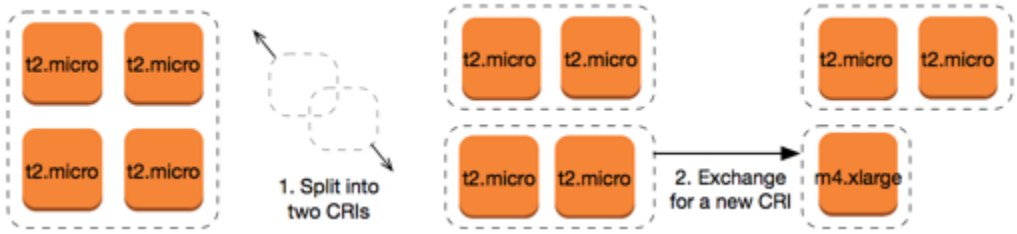
交換 可轉換預留執行個體 的一部分

您可以使用修改程序將您的可轉換預留執行個體分割為較小的保留，然後以一或多個新保留交換新的可轉換預留執行個體。下列範例示範執行此作業的方式。

Example 範例：具有多個執行個體的可轉換預留執行個體

在此範例中，您有一個保留中有四個執行個體的 `t2.micro` 可轉換預留執行個體。若要以兩個 `t2.micro` 執行個體交換一個 `m4.xlarge` 執行個體：

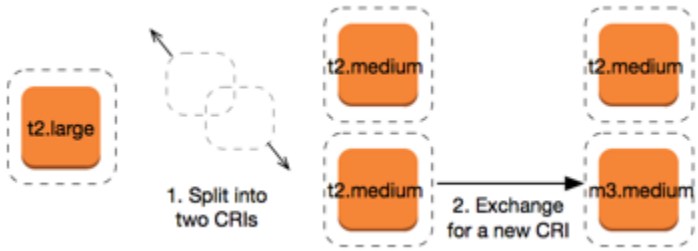
1. 修改 `t2.micro` 可轉換預留執行個體，方式為將其分割為兩個 `t2.micro` 可轉換預留執行個體，各自具有兩個執行個體。
2. 以其中一個新的 `t2.micro` 可轉換預留執行個體交換 `m4.xlarge` 可轉換預留執行個體。



Example 範例：具有單一執行個體的可轉換預留執行個體

在此範例中，您有一個 `t2.large` 可轉換預留執行個體。若要將其變更為較小的 `t2.medium` 執行個體和 `m3.medium` 執行個體：

1. 修改 `t2.large` 可轉換預留執行個體，方式為將其分割為兩個 `t2.medium` 可轉換預留執行個體。執行個體大小使用量與兩個 `t2.large` 執行個體相同的單一 `t2.medium` 執行個體。
2. 以其中一個新的 `t2.medium` 可轉換預留執行個體交換 `m3.medium` 可轉換預留執行個體。



如需詳細資訊，請參閱 支援執行個體大小的修改 和 提交交換請求。

提交交換請求

您可以使用 Amazon EC2 主控台或命令列工具交換可轉換預留執行個體。

使用主控台交換 可轉換預留執行個體

您可以搜尋可轉換預留執行個體優惠，然後從提供的選取中選取您的新組態。

使用 Amazon EC2 主控台交換可轉換預留執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 選取 Reserved Instances (預留執行個體)、選取要交換的可轉換預留執行個體，然後選取 Actions (動作)，Exchange 預留執行個體 (交換預留執行個體)。
3. 選取所需組態的屬性，然後選取 Find offering (尋找優惠)。
4. 選取新的 可轉換預留執行個體。在畫面底部，您可以檢視兌換所收到的 預留執行個體 數量，以及任何額外費用。
5. 當您選取符合您需求的 可轉換預留執行個體 時，請選取 Review (檢閱)。
6. 選擇 Exchange (交換)，然後選擇 Close (關閉)。

已交換的預留執行個體會淘汰，而新的預留執行個體則會顯示在 Amazon EC2 主控台中。此程序需要幾分鐘的時間散佈。

使用命令列介面交換 可轉換預留執行個體

若要交換 可轉換預留執行個體，請先尋找符合您需求的新 可轉換預留執行個體：

- describe-reserved-instances-offerings (AWS CLI)
- Get-EC2ReservedInstancesOffering(視窗工具 PowerShell)

取得交換的報價，其中包含您可以從交換中取得的預留執行個體數目，以及交換的校正成本：

- get-reserved-instances-exchange-報價（ ）AWS CLI
- 獲取 EC2-ReservedInstancesExchangeQuote（視窗工具 PowerShell）

最後，執行交換：

- accept-reserved-instances-exchange-報價（ ）AWS CLI
- Approve-EC2ReservedInstancesExchangeQuote(視窗工具 PowerShell)

預留執行個體配額

您可以每月購買新的預留執行個體。您每月可購買的新預留執行個體數量取決於您的每月配額，如下所示：

配額說明	預設配額
新的區域預留執行個體	每月每區域 20 個
新的地區預留執行個體	每月每可用區域 20 個

例如，在具有 3 個可用區域的區域中，預設配額為每月 80 個新的預留執行個體 (計算方式如下)：

- 該區域的 20 個區域預留執行個體
- 加上 60 個地區預留執行個體 (三個可用區域各 20 個)

`running`狀態中的執行個體會計入您的配額中。位於`pending`、`stopping`、`stopped`和`hibernated`狀態的執行個體不會計入您的配額中。

檢視您已購買的預留執行個體數

您購買的預留執行個體數量由 Instance count (執行個體計數) 欄位 (主控台) 或 `InstanceCount` 參數 (AWS CLI) 來指示。當您購買新的預留執行個體時，會根據執行個體總數來測量配額。例如，如果您購買執行個體數量為 10 的單一預留執行個體組態，則該購買所佔配額將為 10，而非 1。

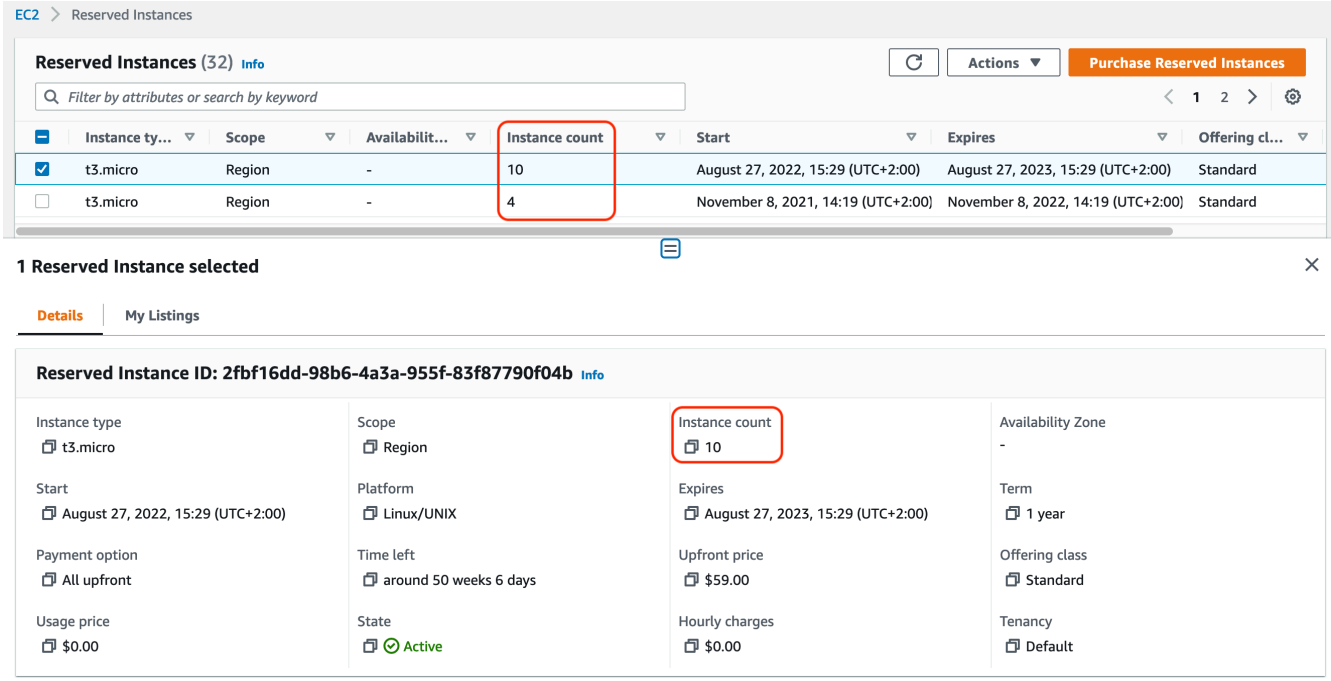
您可以使用 Amazon EC2 或查看已購買的預留執行個體數量 AWS CLI。

Console

若要檢視您已購買的預留執行個體數

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Reserved Instances (預留執行個體)。
3. 從資料表中選取預留執行個體組態，然後勾選 Instance count (執行個體計數) 欄位。

   在下列螢幕擷取畫面中，所選行表示 `t3.micro` 執行個體類型的單一預留執行個體組態。資料表檢視中的 Instance count (執行個體計數) 欄和詳細資訊檢視中的 Instance count (執行個體計數) 欄位指示此組態有 10 個預留執行個體。



AWS CLI

若要檢視您已購買的預留執行個體數

使用命describe-reserved-instancesCLI令並指定預留執行個體組態的 ID。

```
aws ec2 describe-reserved-instances \
    --reserved-instances-ids a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --output table
```

輸出範例 - `InstanceCount` 欄位指示此組態有 10 個預留執行個體。

```
-----------------------------------------------------------------
|                    DescribeReservedInstances                    |
+-----------------------------------------------------------------+
||                       ReservedInstances                       ||
|+----------------------+----------------------------------------+|
||  CurrencyCode        |  USD                                   ||
||  Duration            |  31536000                              ||
||  End                 |  2023-08-27T13:29:44+00:00             ||
||  FixedPrice          |  59.0                                  ||
||  InstanceCount       |  10                                    ||
||  InstanceTenancy     |  default                               ||
||  InstanceType        |  t3.micro                              ||
```

```
||  OfferingClass       |  standard                                ||
||  OfferingType        |  All Upfront                             ||
||  ProductDescription  |  Linux/UNIX                              ||
||  ReservedInstancesId |  a1b2c3d4-5678-90ab-cdef-EXAMPLE11111    ||
||  Scope               |  Region                                  ||
||  Start               |  2022-08-27T13:29:45.938000+00:00        ||
||  State               |  active                                  ||
||  UsagePrice          |  0.0                                     ||
|+---------------------+-----------------------------------------+|
|||                      RecurringCharges                        |||
||+---------------------------------+-------------------------+||
|||  Amount                         |  0.0                    |||
|||  Frequency                      |  Hourly                 |||
||+---------------------------------+-------------------------+||
```

PowerShell

若要檢視您已購買的預留執行個體數

使用指Get-EC2ReservedInstance令程式並指定預留執行個體組態的識別碼。

```
Get-EC2ReservedInstance -ReservedInstancesId a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

輸出範例 - `InstanceCount` 欄位指示此組態有 10 個預留執行個體。

```
AvailabilityZone    :
CurrencyCode        : USD
Duration            : 31536000
End                 : 1/12/2017 8:57:08 PM
FixedPrice          : 0
InstanceCount       : 10
InstanceTenancy     : default
InstanceType        : t3.medium
OfferingClass       : standard
OfferingType        : All Upfront
ProductDescription  : Windows
RecurringCharges    : {}
ReservedInstancesId : a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
Scope               : Region
Start               : 10/12/2016 4:00:00 PM
State               : active
Tags                : {}
UsagePrice          : 0
```

考量事項

區域預留執行個體會將折扣套用至執行中隨需執行個體。預設隨需執行個體限制為 20。您無法透過購買區域預留執行個體來超出執行中隨需執行個體限制。例如，如果您已有 20 個執行中隨需執行個體，而且您購買 20 個區域預留執行個體，則 20 個區域預留執行個體是用來將折扣套用至 20 個執行中隨需執行個體。如果您購買更多的區域預留執行個體，將無法啟動更多的執行個體，因為您已達到隨需執行個體限制。

在購買地區預留執行個體之前，請確認您的隨需執行個體限制符合或超過您想要擁有的地區預留執行個體數。如有必要，請務必先請求調高您的隨需執行個體限制，再購買更多地區預留執行個體。

區域性預留執行個體 (針對特定可用區域購買的預留執行個體) 會提供容量保留以及折扣。您可以透過購買區域性預留執行個體來超出執行中隨需執行個體限制。例如，如果您已有 20 個執行中隨需執行個體，而且購買 20 個區域性預留執行個體，則您可以另外啟動 20 個符合區域性預留執行個體之規格的隨需執行個體，如此總共有 40 個執行中執行個體。

檢視預留執行個體配額並請求增加配額

Amazon 主EC2控台提供配額資訊。您還可以請求增加您的配額。如需詳細資訊，請參閱 檢視您目前的配額 和 請求增加。

Spot 執行個體

Spot 執行個體是一種使用備用EC2容量的執行個體，可用的價格低於隨需價格。由於 Spot 執行個體可讓您以極大的折扣申請未使用的EC2執行個體，因此您可以大幅EC2降低 Amazon 成本。Spot 執行個體的每小時價格稱為 Spot 價格。每個可用區域中每個執行個體類型的 Spot 價格由 Amazon 設定EC2，並根據 Spot 執行個體的長期供需逐步調整。只要有容量可用，您的 Spot 執行個體便會執行。

如能彈性決定應用程式執行的時機以及應用程式是否可以中斷，則 競價型執行個體會是一項具成本效益的選擇。例如，競價型執行個體非常適合用於資料分析、批次處理任務、背景處理和選擇性的任務。如需詳細資訊，請參閱 Amazon EC2 競價型執行個體。

如需執行個EC2體不同購買選項的比較，請參閱Amazon EC2 帳單和購買選項。

概念

在開始使用Spot 執行個體之前，您應該先熟悉以下概念：

- 競價型容量集區 — 一組具有相同EC2執行個體類型 (例如`m5.large`) 和可用區域的未使用執行個體。
- Spot 價格 – Spot 執行個體目前的每小時價格。

- Spot 執行個體請求 – 請求 Spot 執行個體。當容量可用時，Amazon 會EC2滿足您的要求。Spot 執行個體請求為一次性或持久性。Amazon EC2 會在與請求相關聯的競價型執行個體中斷後自動重新提交永久性 Spot 執行個體請求。
- EC2執行個體重新平衡建議 — Amazon EC2 會發出執行個體重新平衡建議信號，通知您 Spot 執行個體有較高的中斷風險。此訊號讓您有機會在現有或新的 Spot 執行個體上主動重新平衡工作負載，而無需等待兩分鐘的 Spot 執行個體中斷通知。
- 競價型執行個體中斷 — 當 Amazon EC2 需要恢復容量時，Amazon EC2 會終止、停止或休眠您的競價型執行個體。Amazon EC2 提供 Spot 執行個體中斷通知，該通知會在執行個體中斷前兩分鐘發出警告。

Spot 執行個體和隨需執行個體之間

下表列出了 Spot 執行個體與隨需執行個體之間的主要差異。

	Spot Instances	On-Demand Instances
啟動時間	只有當 Spot 執行個體請求正在作用中，而且有可用的容量時，才能立即啟動。	只有在進行手動啟動請求，而且有可用的容量時，才能立即啟動。
可用容量	如果沒有可用的容量，則 Spot 執行個體請求會繼續自動發出啟動請求，直到有可用的容量出現。	如果在您提出啟動要求時無法使用容量，您會收到容量不足的錯誤訊息 (ICE)。
每小時價格	Spot 執行個體的每小時價格會隨長期的供給和需求而有所不同。	隨需執行個體 的每小時價格為靜態定價。
重新平衡建議	當執行個EC2體存在中斷風險增加時，Amazon 為執行中的 Spot 執行個體發出的信號。	您可以決定中斷 (停止、休眠或終止) 隨需執行個體 的時間點。
執行個體中斷	您可以停止並啟動 Amazon EBS 支援的競價型執行個體。此外，如果容量不再EC2可用，Amazon 可以中斷個別競價型執行個體。	您可以決定中斷 (停止、休眠或終止) 隨需執行個體 的時間點。

定價與節省

您需要支付 Spot 執行個體的 Spot 價格，該價格由 Amazon 設定，EC2並根據競價型執行個體的長期供需逐步調整。您的競價型執行個體會一直執行，直到您終止它們、容量不再可用，或者您的 Amazon EC2 Auto Scaling 群組會在擴展期間終止這些執行個體。

如果您或 Amazon 中EC2斷執行中的競價型執行個體，則會根據使用的作業系統和中斷 Spot 執行個體的使用者向您收取使用秒數或整個小時的費用，否則您不會收取任何費用。如需詳細資訊，請參閱針對已中斷的 競價型執行個體收費。

Savings Plans 不涵蓋 Spot 執行個體。如果您有 Savings Plan，除了使用 Spot 執行個體所節省的成本之外，它不會提供額外的節省成本。此外，您在 Spot 執行個體上的支出不適用於 Compute Savings Plans 中的承諾。

檢視價格

若要檢視每個執行個體類型目前 (每 AWS 區域 五分鐘更新一次) 的最低競價型價格，請參閱 Amazon EC2 Spot 執行個體定價頁面。

若要檢視過去三個月的 Spot 價格歷史記錄，請使用 Amazon EC2 主控台或describe-spot-price-history指令 (AWS CLI)。如需詳細資訊，請參閱Spot 執行個體定價歷史記錄。

我們獨立地將可用區域映射到每個代碼 AWS 帳戶。如此，不同帳戶之間的相同可用區域代碼 (例如 `us-west-2a`)，就會得到不同的結果。

檢視節省

您可以檢視針對單一 Spot 機群或所有 Spot 執行個體使用 Spot 執行個體所節省的成本。您可以檢視過去一小時或過去三天所節省的成本，也可以檢視每 v CPU 小時和每個記憶體 (GiB) 小時的平均成本。節省是估計值，因為不含用量的帳單調整，所以會與實際的節省有所出入。如需檢視節省資訊的詳細資訊，請參閱 購買 競價型執行個體省下的成本。

檢視帳單

您的帳單會提供您服務用量的詳細資訊。如需詳細資訊，請參閱 AWS Billing 使用者指南中的檢視您的帳單。

Amazon EC2 現貨的最佳實踐

Amazon EC2 可 AWS 雲端 透過 Spot 執行個體存取備用EC2運算容量，相較於隨需價格，最多可節省 90%。隨需執行個體和競價型執行個體之間的唯一區別是，如果 Amazon EC2 需要回收容量

EC2，Amazon 可能會中斷 Spot 執行個體，在兩分鐘的時間內通知。為了確保使用 Spot 執行個體的最佳體驗，請務必瞭解並套用最佳做法以供其使用。

競價型執行個體 建議用於無狀態、容錯、靈活的應用程式。例如，Spot 執行個體適用於大數據、容器化工作負載、CI/CD、無狀態 Web 伺服器、高效能運算 (HPC) 和轉譯工作負載。

執行時，競價型執行個體 與 隨需執行個體 完全相同。但是，Spot 並無法保證您可以保持執行中的執行個體有足夠時間來完成工作負載。Spot 也無法保證您可以立即取得所尋找執行個體的可用性，也無法保證您一律可以取得所請求的彙總容量。此外，Spot 執行個體中斷和容量可能會隨著時間而改變，因為 Spot 執行個體可用性會因供給和需求而有所不同，而過去的效能並不能保證未來的結果。

競價型執行個體 不適合執行個體節點之間不靈活、可設定狀態、錯誤不容忍或緊密結合的工作負載。對於無法容忍整個目標容量無法完全可用的偶爾期間的工作負載，我們不建議使用 Spot 執行個體。儘管遵循 Spot 最佳實務可靈活調整執行個體類型，而可用區域可提供高可用性的最佳機會，但無法保證容量可用，因為隨需執行個體的需求激增可能會中斷 Spot 執行個體上的工作負載。

我們強烈建議將 Spot 執行個體用於這些工作負載，或嘗試容錯移轉到隨需執行個體以處理中斷或無法使用的期間。容錯移轉到隨需執行個體可能會無意中導致其他 Spot 執行個體的中斷。此外，如果執行個體類型和可用區域組合的 Spot 執行個體遭到中斷，您可能很難獲得具有相同組合的隨需執行個體。

無論您是經驗豐富的 Spot 使用者或 Spot 執行個體的新使用者，如果您目前遇到 Spot 執行個體中斷或可用性問題，建議您遵循這些最佳做法，以獲得使用 Spot 服務的最佳體驗。

Spot 最佳實務

- 針對中斷準備個別執行個體
- 對執行個體類型和可用區域具有彈性
- 使用屬性型執行個體類型選取範圍
- 使用 Spot 放置分數來識別最佳區域和可用區域
- 使用 EC2 Auto Scaling 群組或EC2叢集來管理您的彙總容量
- 使用價格和容量最佳化分配策略
- 使用整合式 AWS 服務來管理 Spot 執行個體
- 使用哪種 Spot 請求方法最好？

針對中斷準備個別執行個體

從容處理 Spot 執行個體中斷的最佳方法，就是建立容錯的應用程式架構。若要完成此操作，您可以利用EC2執行個體重新平衡建議和 Spot 執行個體中斷通知。

EC2執行個體重新平衡建議是一種訊號，會在 Spot 執行個體有中斷風險增加時通知您。該訊號使您有機會在兩分鐘 Spot 執行個體中斷通知之前主動管理 Spot 執行個體。您可以決定將工作負載重新平衡至未處於提高之中斷風險的新的或現有 競價型執行個體。我們使用 Auto Scaling 群組和EC2叢集中的容量重新平衡功能，讓您輕鬆使用此訊號。

Spot 執行個體中斷通知是在 Amazon 中EC2斷 Spot 執行個體前兩分鐘發出的警告。如果您的工作負載具有「時間彈性」，當執行個體中斷時，您可以將它們設定為停止或休眠，而非終止。Amazon EC2會在中斷時自動停止或休眠您的 Spot 執行個體，並在我們有可用容量時自動恢復執行個體。

我們建議您在 Amazon EventBridge 中建立擷取重新平衡建議和中斷通知的規則，然後觸發檢查點以檢查工作負載的進度，或妥善處理中斷。如需詳細資訊，請參閱監控重新平衡建議訊號。如需逐步引導您如何建立和使用事件規則的詳細範例，請參閱利用 Amazon EC2 Spot 執行個體中斷通知。

如需詳細資訊，請參閱 EC2例項重新平衡建議 和 Spot 執行個體中斷。

對執行個體類型和可用區域具有彈性

競價型容量集區是具有相同EC2執行個體類型 (例如`m5.large`) 和可用區域 (例如 us-east-1a) 的一組未使用執行個體。您應該對於請求的執行個體類型，以及可在其中部署工作負載的可用區域具有彈性。這讓 Spot 有更好的機會找到並配置您所需的運算容量。例如，不要只要求 `c5.large` 是否願意使用來自 c4、m5 和 m4 系列的 large。

根據您的特定需求，您可以評估哪些執行個體類型可以具彈性，以滿足您的運算需求。如果工作負載可以垂直擴展，則應在請求中包含較大的執行個體類型 (更多 vCPUs 和記憶體)。如果您只能水平擴展，則應納入舊一代執行個體類型，因為隨需客戶的需求較少。

一個很好的經驗法則是針對每個工作負載的至少 10 個執行個體類型上具有彈性。此外，請確定所有可用區域均已設定為在您的工作負載中使用，VPC並為您的工作負載選取。

使用屬性型執行個體類型選取範圍

透過以屬性為vCPUs基礎的執行個體類型選取，您可以為要執行的工作負載指定執行個體屬性，例如記憶體和儲存體。EC2然後，Auto Scaling 或EC2叢集將自動識別並啟動符合您指定屬性的執行個體。這樣就省去了手動選擇特定執行個體類型所需的工作量，這需要深入了解每個執行個體類型的產品。

此外，以屬性為基礎的例證類型選取可讓您在新發行的例證類型可用時自動使用它們。這可確保無縫存取日益廣泛的競價型執行個體容量。

基於屬性的執行個體類型選擇非常適合工作負載和架構，這些工作負載和架構可靈活運用於執行的執行個體類型，例如高效能運算 (HPC) 和大數據工作負載。

如需詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南和指定EC2叢集或 Spot 叢集的執行個體類型選取屬性本指南中的使用以屬性為基礎的執行個體類型選擇建立混合執行個體群組。

使用 Spot 放置分數來識別最佳區域和可用區域

競價型執行個體是未使用的EC2容量，而且此容量會根據EC2供給和需求而波動。因此，您可能無法始終在特定時間在特定位置獲得所需的確切 Spot 容量。若要降低這種不可預測性，您可以使用競價位置分數功能。此功能針對較有可能具有足夠容量以滿足 Spot 容量需求的區域或可用區域提供建議，而不需要您先在這些位置啟動競價型執行個體。

Spot 放置分數最適合用於可彈性執行個體類型及其可使用的區域或可用區域或可用區域的工作負載。您只需指定所需的競價型容量、執行個體類型需求，以及是否要針對區域或可用區域提供建議。作為回報，您會收到每個區域或可用區域的分數介於 1 到 10 之間，指出在該位置成功佈建請求的 Spot 容量的可能性。分數為 10 表示您的 Spot 請求很有可能成功。

請務必注意，Spot 放置分數是一項 point-in-time 建議，因為容量可能會隨時間而有所不同。它不保證可用容量或預測中斷的風險。

您可以使用 Amazon EC2 主控台中的競價位置分數功能 AWS CLI，或使用SDK. 如需詳細資訊，請參閱Spot 配置分數。

使用 EC2 Auto Scaling 群組或EC2叢集來管理您的彙總容量

Spot 可讓您以彙總容量 (包括記憶體vCPUs、儲存或網路傳輸量為單位) 進行思考，而不是以個別執行個體來思考。Auto Scaling 群組和EC2叢集可讓您啟動和維護目標容量，並自動請求資源以取代任何中斷或手動終止的容量。設定 Auto Scaling 群組或EC2叢集時，只需根據應用程式需求指定執行個體類型和目標容量。如需詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南和本使用者指南建立EC2艦隊中的自動擴展群組。

使用價格和容量最佳化分配策略

Auto Scaling 群組中的分配策略可協助您佈建目標容量，而無需手動尋找具有備用容量的 Spot 容量集區。建議使用 `price-capacity-optimized` 策略，因為此策略會自動從最可用且價格最低的 Spot 容量集區佈建執行個體。您也可以利用EC2艦隊中的`price-capacity-optimized`配置策略。由於您的 Spot 執行個體容量來自具有最佳容量的集區，因此可降低回收 Spot 執行個體的可能性。如需有關配置策略的詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南中的 Spot 執行個體和本使用者指南當工作負載具有較高的中斷成本時中的。

使用整合式 AWS 服務來管理 Spot 執行個體

其他 AWS 服務與 Spot 整合，可降低整體運算成本，而無需管理個別執行個體或叢集。我們建議您針對適用的工作負載考慮下列解決方案：Amazon EMR、Amazon 彈性容器服務 AWS Batch、Amazon

Elastic Kubernetes Service SageMaker AWS Elastic Beanstalk、Amazon 和 Amazon。 GameLift若要進一步了解這些服務的競價型最佳實務，請參閱 Amazon S EC2 pot 執行個體研討會網站。

使用哪種 Spot 請求方法最好？

請使用下表決定請求 Spot 執行個體時要使用哪API個。

API	何時使用？	使用案例	我應該使用這個API嗎？
CreateAutoScalingGroup	• 您需要具有單一組態或混合組態的多個執行個體。 • 您想要透過可設定的自動化生命週期管理API。	建立 Auto Scaling 群組，管理執行個體的生命週期，同時維護所需的執行個體數量。支援在指定的最小和最大限制之間水平擴展 (新增更多執行個體)。	是
CreateFleet	• 您需要具有單一組態或混合組態的多個執行個體。 • 您希望自行管理您的執行個體生命週期。 • 如果您不需要自動擴展，建議您使用 `instant` 類型機群。	在單一請求中建立隨需執行個體和 Spot 執行個體的叢集，並具有多個啟動規格，這些規格會因執行個體類型AMI、可用區域或子網路而異。Spot 執行個體分配策略預設為每單位 `lowest-price` ，但您可以將其變更為 `price-capacity-optimized` 、`capacity-optimized` 或 `diversified` 。	是 – 在 `instant` 模式中 (如果您不需要自動擴展)
RunInstances	•		

API	何時使用？	使用案例	我應該使用這個API嗎？
	您已經在使用啟動隨需執行個體，而且只想變更單一參數來變更為啟動 Spot 執行個體。RunInstances API • 您不需要具有不同執行個體類型的多個執行個體。	使用AMI和一個執行個體類型啟動指定數量的執行個體。	否 — 因為 RunInstances 不允許在單一要求中混合執行個體類型
RequestSpotFleet	• 我們強烈不鼓勵使用， RequestSpotFleet API因為它是沒有計劃投資API的遺產。 • 如果您想要管理執行個體生命週期，請使用 CreateFleet API. • 如果您不想管理執行個體生命週期，請使用 CreateAutoScalingGroup API.	做NOTUSE。RequestSpotFleet 是遺產，沒API有計劃投資。	否

API	何時使用？	使用案例	我應該使用這個API嗎？
RequestSpotInstances	• 我們強烈不鼓勵使用，RequestSpotInstances API因為它是沒有計劃投資API的遺產。	做NOTUSE。RequestSpotInstances是遺產，沒API有計劃投資。	否

競價型執行個體的運作方式

若要啟動競價型執行個體，您可以建立競價型執行個體請求，或 Amazon 代表您EC2建立競價型執行個體請求。Spot 執行個體請求完成後，會啟動 Spot 執行個體。

您可以使用數個不同的服務來啟動 Spot 執行個體。如需詳細資訊，請參閱開始使用 Amazon EC2 Spot 執行個體。在本使用者指南中，我們說明了以下使用啟動 Spot 執行個體的方法EC2：

- 您可以使用 Amazon EC2 主控台中的啟動執行個體精靈或執行執行個體命令來建立競價型執行個體 AWS CLI 請求。如需詳細資訊，請參閱管理 Spot 執行個體。
- 您可以建立EC2叢集，在其中指定所需的 Spot 執行個體數量。Amazon 代表您為EC2叢集中指定的每個競價型執行個體EC2建立 Spot 執行個體請求。如需詳細資訊，請參閱建立EC2艦隊。
- 您可以建立 Spot Fleet 請求，在其中指定所需數量的 Spot 執行個體。Amazon 代表您EC2為 Spot 叢集請求中指定的每個競價型執行個體建立競價型執行個體請求。如需詳細資訊，請參閱建立 Spot 機群。

如果有可用容量，您的 Spot 執行個體就會啟動。您的競價型執行個體會一直執行，直到您停止或終止，或者直到 Amazon 將其中EC2斷 (稱為 Spot 執行個體中斷) 為止。Amazon EC2 可以在 Spot 執行個體中斷時停止、終止或休眠。

使用 Spot 執行個體時，您必須對中斷做好準備。當 Spot 執行個體的需求增加或 Spot 執行個體的供應減少時，Amazon EC2 可能會中斷您的 Spot 執行個體。Amazon 中EC2斷 Spot 執行個體時，會提供競價型執行個體中斷通知，在 Amazon 中EC2斷執行個體前兩分鐘發出警告。您無法為 競價型執行個體啟用終止保護。如需詳細資訊，請參閱Spot 執行個體中斷。

目錄

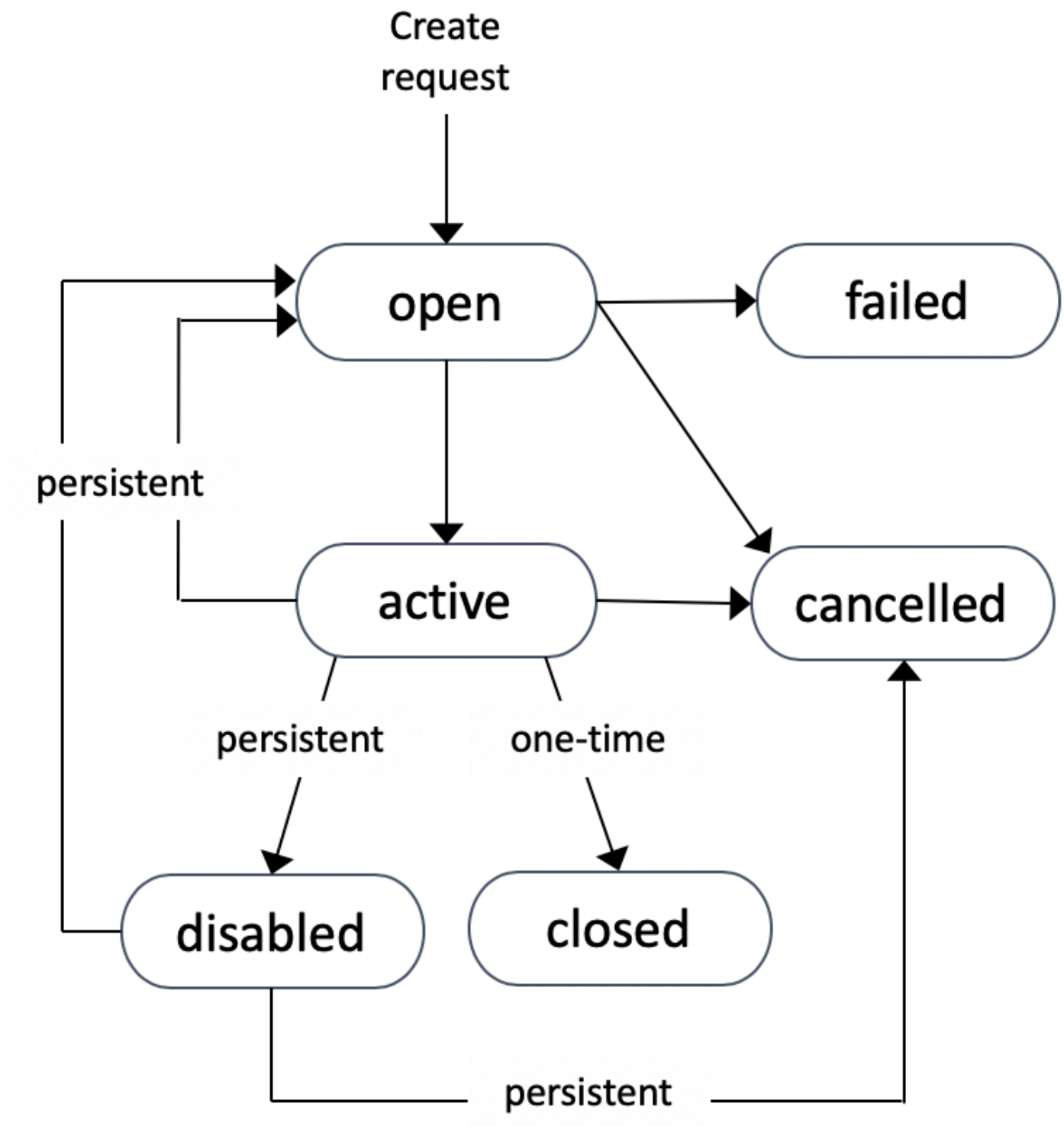
- Spot 執行個體請求狀態
- 在啟動群組中啟動 競價型執行個體
- 在可用區域群組中啟動 競價型執行個體
- 啟動競價型執行個體 VPC
- 啟動高載效能執行個體
- 在單租戶硬體上啟動

Spot 執行個體請求狀態

Spot 執行個體請求可以是下列任一種狀態：

- `open` – 請求正等待完成。
- `active` – 請求已完成，並具有關聯的 Spot 執行個體。
- `failed` – 請求具有一個或多個無效的參數。
- `closed` – Spot 執行個體已中斷或終止。
- `disabled` – 您已停止 Spot 執行個體。
- `cancelled` – 您已取消請求，或請求已過期。

下圖顯示了請求狀態之間的轉換。請注意，轉換取決於請求的類型 (一次性或持久性)。



在 Amazon EC2 啟動競價型執行個體、請求到期或您取消請求之前，一次性競價型執行個體請求會保持有效。如果沒有容量可用，則 Spot 執行個體會終止，而且 Spot 執行個體請求也會結束。

即使請求已完成，持久性的 Spot 執行個體請求在過期或是您取消它之前，都會繼續維持有效運作。如果沒有容量可用，則 Spot 執行個體會中斷。在執行個體中斷後，當容量可再次提供使用時，會啟動停止或恢復休眠的 Spot 執行個體。如果有容量可用，您可以停止 Spot 執行個體並將其重新啟動。如果 Spot 執行個體終止 (無論競價型執行個體處於停止或執行中狀態)，則會再次開啟競價型執行個體請求，Amazon 會EC2啟動新的 Spot 執行個體。如需詳細資訊，請參閱 停止 Spot 執行個體、啟動 Spot 執行個體 及 終止 Spot 執行個體。

您可以透過狀態，追蹤 Spot 執行個體請求的狀態和已啟動的 Spot 執行個體的狀態。如需詳細資訊，請參閱取得 Spot 執行個體請求的狀態。

在啟動群組中啟動 競價型執行個體

在 Spot 執行個體請求中指定一個啟動群組，告知 Amazon 只有在能EC2夠全部啟動 Spot 執行個體的情況下才啟動一組 Spot 執行個體。此外，如果 Spot 服務必須終止啟動群組內的其中一個執行個體，則必須終止群組中所有的執行個體。但是，如果您終止啟動群組中的一或多個執行個體，Amazon EC2 不會終止啟動群組中剩餘的執行個體。

雖然此選項很實用，但加上此限制可能會降低完成 Spot 執行個體請求的機率，並提高終止 Spot 執行個體的機率。例如，您的啟動群組包含多個可用區域的執行個體。如果其中一個可用區域的容量減少且無法再使用，Amazon 就EC2會終止該啟動群組的所有執行個體。

如果建立另一個成功的 Spot 執行個體請求，它指定與先前的成功請求相同的 (現有的) 啟動群組，則新的執行個體會加入該啟動群組。之後，如果這個啟動群組中的執行個體遭到終止，則該啟動群組中的所有執行個體都會終止，這也包括由第一個和第二個請求啟動的執行個體。

在可用區域群組中啟動 競價型執行個體

在您的競價型執行個體請求中指定一個可用區域群組，EC2以告訴 Amazon 在同一可用區域中啟動一組 Spot 執行個體。Amazon 不EC2需要同時中斷可用區域群組中的所有執行個體。如果 Amazon EC2 必須中斷可用區域群組中的其中一個執行個體，則其他執行個體仍在執行。

雖然此選項很實用，但加上這項限制可能會降低完成 Spot 執行個體請求的機率。

如果指定了可用區域群組，但未在 Spot 執行個體請求中指定可用區域，則結果取決於您指定的網路。

預設值 VPC

Amazon 針對指定的子網路EC2使用可用區域。如未指定子網，它會替您選取一個可用區域及其預設子網，但不一定是最低價的區域。如果您刪除可用區域適用的預設子網，則必須指定不同的子網。

非預設值 VPC

Amazon 針對指定的子網路EC2使用可用區域。

啟動競價型執行個體 VPC

您可以用為 競價型執行個體指定子網的同一種方法，來為 隨需執行個體指定子網。

- [預設VPC] 如果您希望 Spot 執行個體在特定的低價可用區域中啟動，則必須在 Spot 執行個體請求中指定對應的子網路。如果您未指定子網路，Amazon 會為您EC2選取一個子網路，而此子網路的可用區域可能不會有最低的 Spot 價格。
- [非預設VPC] 您必須為 Spot 執行個體指定子網路。

啟動高載效能執行個體

T 執行個體類型是爆量效能執行個體。如果您使用高載效能執行個體類型啟動 Spot 執行個體，並且計劃立即使用高載效能 Spot 執行個體，而且沒有閒置時間來累積積CPU分，建議您以標準模式啟動它們，以避免支付更高的費用。如果您以無限制模式啟動高載效能 Spot 執行個體並CPU立即爆發，您將花費剩餘的點數進行爆量。如果您在短時間內使用執行個體，則執行個體沒有時間累積積CPU分來支付剩餘積分，而且在您終止執行個體時，系統會向您收取剩餘積分的費用。

無限制模式僅適用於高載效能 Spot 執行個體的執行時間足以累積積CPU分以進行爆發的 Spot 執行個體。否則，支付剩餘額度會使爆量效能 競價型執行個體比使用其他執行個體更為昂貴。如需詳細資訊，請參閱何時使用無限模式與固定模式 CPU。

T2 執行個體在標準模式中設定時，可取得啟動額度。T2 執行個體是唯一可獲得啟動額度的爆量效能執行個體。啟動額度旨在透過提供足夠的運算資源來設定執行個體，創造 T2 執行個體具有生產力的初始啟動體驗。不允許重複啟動 T2 執行個體來獲得新的啟動額度。如果您需要持續CPU，您可以賺取積分(透過閒置一段時間)、對 T2 Spot 執行個體使用無限制模式，或使用具有專CPU用的執行個體類型。

在單租戶硬體上啟動

您可以在單一租戶的硬體上執行 Spot 執行個體。專用 Spot 執行個體與屬於其他 AWS 帳戶的執行個體實體隔離。如需詳細資訊，請參閱Amazon EC2 專用實例和 Amazon EC2 專用執行個體。

若要執行專用 Spot 執行個體，請進行下列其中一項動作：

- 在建立 Spot 執行個體請求時指定 `dedicated` 的租用。如需詳細資訊，請參閱管理 Spot 執行個體。
- 請求執行個體租VPC用為的競價型執行個體`dedicated`。如需詳細資訊，請參閱使用預設租用將專用執行VPC個體啟動到。如果您在執行個體租用中提出申請，`default`則無法要求租VPC用為的 Spot 執行個體`dedicated`。

除了 T 執行個體之外，所有執行個體群都支援專用 競價型執行個體。對於每個支援的執行個體系列而言，只有最大型的執行個體大小或硬體大小可支援專用 競價型執行個體。

Spot 執行個體定價歷史記錄

競價型執行個體價格由 Amazon 設定，EC2並根據競價型執行個體容量的長期供需趨勢逐步調整。

滿足您的 Spot 請求後，您會以目前的 Spot 價格啟動 Spot 執行個體，而不會超過隨需價格。您可以檢視過去 90 天的 Spot 價格歷史，根據執行個體類型、作業系統和可用區域來進行篩選。

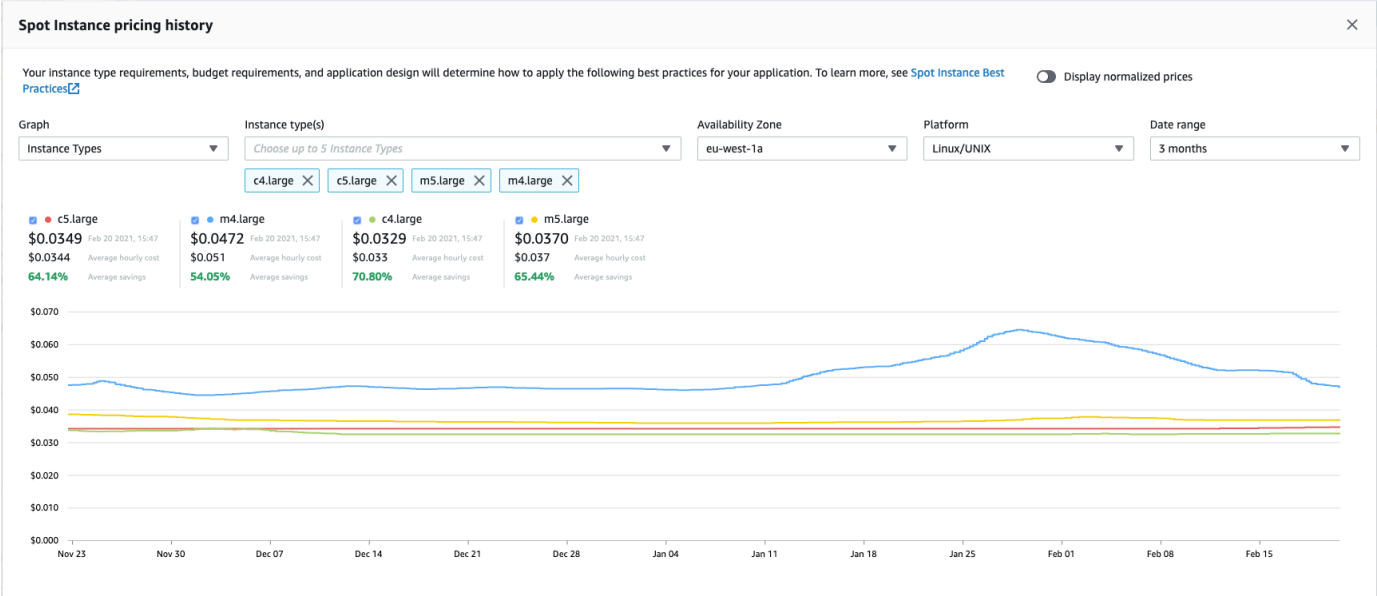
檢視目前的 Spot 價格

如需目前競價型執行個體價格，請參閱 Amazon EC2 Spot 執行個體定價。

使用主控台檢視現貨價格歷史記錄

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選擇定價歷史。
4. 在 Graph (圖表) 中，選擇依 Availability Zones (可用區域) 或 Instance Types (執行個體類型) 來比較價格歷史記錄。
   - 若您選擇 Availability Zones (可用區域)，則選擇 Instance type (執行個體類型)、作業系統 (Platform (平台))，以及要檢視價格歷史記錄的 Date range (日期範圍)。
   - 若您選擇 Instance Types (執行個體類型)，則選擇最多五種 Instance type(s) (執行個體類型)、Availability Zone (可用區域)、作業系統 (Platform (平台))，以及要檢視價格歷史記錄的 Date range (日期範圍)。

   下列螢幕擷取畫面顯示不同執行個體類型的價格比較。



5. 將您的指標懸停至圖表上方，可顯示所選取日期範圍中特定時間點的價格。價格顯示在圖表上方的資訊區塊中。顯示於頂列的價格會顯示指定日期的价格。顯示於第二列的價格會顯示所選日期範圍的平均價格。
6. 若要顯示每 v 的價格CPU，請開啟 [顯示標準化價格]。若要顯示該執行個體類型的價格，請關閉Display normalized prices (顯示標準化價格)。

使用命令列來檢視 Spot 價格歷史

您可以使用下列其中一個命令。如需詳細資訊，請參閱訪問 Amazon EC2。

- describe-spot-price-history (AWS CLI)
- Get-EC2SpotPriceHistory (AWS Tools for Windows PowerShell)

購買 競價型執行個體省下的成本

您可檢視每個叢集之 競價型執行個體或所有執行中 競價型執行個體的用量和節省資訊。每個叢集的用量和節省資訊包括叢集啟動及終止的所有執行個體。您可檢視前一小時或前三天的資訊。

Savings (節省) 區段中的下列螢幕擷取畫面顯示 Spot 用量和 Spot Fleet 的節省資訊。

Spot usage and savings

4	266	700	$9.55	$2.99	69%
Spot Instances	vCPU-hours	Mem(GiB)-hours	On-Demand total	Spot total	Savings
				$0.0112	$0.0043
				Average cost per VCPU-hour	Average cost per mem(GiB)-hour

Details

t3.medium (1)	2 vCPU hours	4 mem(GiB)-hours	$0.01 total	70% savings
m4.large (1)	144 vCPU hours	576 mem(GiB)-hours	$2.52 total	68% savings
t2.micro (2)	120 vCPU hours	120 mem(GiB)-hours	$0.46 total	70% savings

您可以檢視下列用量和節省資訊：

- Spot 執行個體— Spot Fleet 啟動和終止的 Spot 執行個體數量。檢視節省摘要時，數字代表所有正在執行的 競價型執行個體。
- v CPU-hours — 所選時間範圍CPU內所有競價型執行個體使用的 v 小時數。
- Mem(GiB)-hours (記憶體 (GiB) 時數) – 所選時段內所有 競價型執行個體使用的 GiB 時數。
- On-Demand total (隨需總額) – 如果在所選時段內啟動這些執行個體做為隨需執行個體使用，所要支付的總金額。
- Spot total (Spot 總額) – 所選時段要支付的總金額。
- Savings (節省) – 不支付隨需價格的節省百分比。
- 每 v CPU 小時的平均成本 — 所選時間範圍內所有競價型執行個體使用的平均小時成本，計算方式如下：每 v 小時的平均成本 = Spot 總計/v CPU CPU 小時。 vCPUs
- 每 Mem 的平均成本 (GiB)-小時 — 所選時間範圍內所有競價型執行個體使用的平均小時成本，計算方式如下：每 Mem (GiB) 小時的平均每小時成本 = Spot 總計/Mem (GiB)-小時。 GiBs
- Details (詳細資訊) 表 – 構成 Spot Fleet 的不同執行個體類型 (每種執行個體類型的執行個體數量所佔百分比)。檢視節省摘要時，這些項目構成您所有正在執行的 競價型執行個體。

節省的資訊只能使用 Amazon 主EC2控台檢視。

使用主控台檢視 Spot 叢集的節省資訊

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格上，選擇 Spot Requests (Spot 請求)。
3. 選取 Spot Fleet 請求的 ID 並捲動至 Savings (節省) 區段。

   或者，選取 Spot Fleet 請求 ID 旁邊的核取方塊，然後選取 Savings (節省) 索引標籤。
4. 根據預設，頁面會顯示前三天的用量和節省資訊。您可選擇 last hour (前一小時) 或 last three days (前三天)。若為啟動不到一小時的 Spot Fleets，頁面顯示一小時的預估節省。

使用主控台檢視所有執行中 Spot 執行個體的節省資訊

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格上，選擇 Spot Requests (Spot 請求)。
3. 選擇 Savings Summary (節省摘要)。

建立 Spot 執行個體請求

若要使用 Spot 執行個體，您可以建立 Spot 執行個體請求，其中包含所需的執行個體數量、執行個體類型，以及 Availability Zone (可用區域)。如果容量可用，Amazon 會立即EC2滿足您的要求。否則，Amazon 會EC2等待您的請求完成或取消請求為止。

您可以使用 Amazon EC2 主控台中的啟動執行個體精靈或執行執行個體 AWS CLI 命令來請求競價型執行個體，方法與啟動隨需執行個體相同。僅出於下列原因，建議使用此方法：

- 您已經在使用啟動執行個體精靈或 run-instances 命令來啟動隨需執行個體，而且您只想透過變更單一參數來變更為啟動 Spot 執行個體。
- 您不需要具有不同執行個體類型的多個執行個體。

通常不建議在啟動 Spot 執行個體時使用此方法，因為您無法指定多個執行個體類型，並且無法在同一請求中同時啟動 Spot 執行個體和隨需執行個體。如需有關啟動 Spot 執行個體的慣用方法，其中包括啟動含有使用多個執行個體類型的 Spot 執行個體和隨需執行個體的機群，請參閱使用哪種 Spot 請求方法最好？

如果您一次申請多個 Spot 執行個體，Amazon EC2 會建立單獨的競價型執行個體請求，以便您可以分別追蹤每個請求的狀態。如需追蹤 Spot 執行個體請求的詳細資訊，請參閱 取得 Spot 執行個體請求的狀態。

Console

使用啟動執行個體精靈建立 Spot 執行個體請求

步驟 1–9 與您用於啟動隨需執行個體的步驟相同。在步驟 10 中，您可以設定 Spot 執行個體請求。

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在螢幕上方的導覽列上，選取「區域」。
3. 在 Amazon 主EC2控台儀表板中，選擇啟動執行個體。
4. (選用) 在 Name and tags (名稱和標籤) 下，您可以命名執行個體，並標記 Spot 執行個體請求、執行個體、磁碟區和彈性圖形。如需標籤的相關資訊，請參閱標記您的 Amazon EC2 資源。

   a. 對於 Name (名稱)，輸入執行個體的描述性名稱。

      執行個體名稱是一個標籤，其中鍵是 Name (名稱)，而值是您指定的名稱。如果您未指定名稱，則可以透過其 ID 來標識執行個體，該 ID 將在您啟動執行個體時自動產生。

   b. 若要標記 Spot 執行個體請求、執行個體、磁碟區和彈性圖形，請選擇 Add additional tags (新增其他標籤)。選取 Add tag (新增標籤)，然後輸入鍵和值，然後選取要標記的資源類型。為每個要新增的其他標籤重新選擇 Add tag (新增標籤)。
5. 在應用程式和作業系統映像 (Amazon Machine Image) 下，選擇執行個體的作業系統 (OS)，然後選取AMI. 如需詳細資訊，請參閱應用程式和作業系統映像 (Amazon Machine Image)。
6. 在 Instance type (執行個體類型) 下，選取符合您執行個體硬體組態和大小要求的執行個體類型。如需詳細資訊，請參閱執行個體類型。
7. 在 Key pair (login) (金鑰對 (登入))，選擇現有的金鑰對，或選擇 Create new key pair (建立新的金鑰對) 以建立新的金鑰對。如需詳細資訊，請參閱Amazon EC2 密鑰對和 Amazon EC2 實例。

   > **Important**
   > 如果您選擇不使用 key pair 繼續 (不建議) 選項，則除非您選擇設定為允許使用者以其他方式登入AMI的方式，否則您將無法連線至執行個體。

8. 在 Network settings (網路設定) 下，使用預設設定，或視需要選擇 Edit (編輯) 以便設定網路設定。

   安全群組構成網路設定的一部分，並定義執行個體的防火牆規則。這些規則會指定應交付至您執行個體的傳入網路流量。

如需詳細資訊，請參閱Network settings (網路設定)。

9. AMI您選取的包括一或多個儲存磁碟區，包括根裝置磁碟區。在 Configure storage (設定儲存) 下，您可以透過選擇 Add new volume (新增新磁碟區)，指定要連接到執行個體的額外磁碟區。如需詳細資訊，請參閱設定儲存。
10. 在 Advanced details (進階詳細資訊)，請設定 Spot 執行個體請求，如下所示：

    a. 在 Purchasing option (購買選項)，請選取 Request Spot Instances (請求 Spot 執行個體) 核取方塊。

    b. 您可以保留 Spot 執行個體請求的預設組態，也可以選擇右側的 Customize (自訂) 為您的 Spot 執行個體請求指定自訂設定。

    當您選擇 Customize (自訂) 時，會出現下列欄位。

    i. Maximum price (最高價格)：您可以採用 Spot 價格 (以隨需價格為上限) 請求 Spot 執行個體，也可以指定您願意支付的最高金額。

    > **Warning**
    > 如果您指定最高價格，您的執行個體將會比選擇無最高價時更頻繁地遭到中斷。

    - No maximum price (無最高價格)：您的 Spot 執行個體將以目前的 Spot 價格啟動。價格絕不會超過隨需價格。(建議)
    - Set your maximum price (per instance/hour) (設定您的最高價格 (每個執行個體/小時))：您可以指定您願意支付的最高金額。
      - 如果您指定的最高價小於目前的 Spot 價格，則不會啟動 Spot 執行個體。
      - 如果您指定的最高價格大於目前的 Spot 價格，Spot 執行個體將會啟動並按目前的 Spot 價格收費。執行競價型執行個體後，如果競價型價格上升到您的最高價格之上，Amazon 就EC2會中斷您的競價型執行個體。
      - 無論您指定的最高價格如何，都將始終按當前 Spot 價格向您收取費用。

    若要回顧 Spot 價格的趨勢，請參閱 Spot 執行個體定價歷史記錄。

    ii. Request type (請求類型)：您選擇的 Spot 執行個體請求類型會決定 Spot 執行個體中斷時會發生什麼狀況。

- 一次性：Amazon EC2 會為您的競價型執行個體提出一次性請求。如果您的 Spot 執行個體中斷，即不會重新提交請求。
- 持續性請求：Amazon EC2 會為您的競價型執行個體提出持續性請求。如果您的 Spot 執行個體中斷，系統會重新提交請求以補充中斷的 Spot 執行個體。

若未指定數值，則預設值為一次性請求。

iii. Valid to (有效至)：持久性 Spot 執行個體請求的到期日期。

一次性請求不支援此欄位。一次性請求在請求中的所有執行個體均啟動，或是您取消請求之前，都會繼續維持有效運作。

- No request expiry date (無請求過期日期)：請求會維持在有效狀態，直到您將其取消為止。
- Set your request expiry date (設定您的請求過期日期)：持久性請求在您指定的日期之前會維持在有效狀態，或者直到您將其取消為止。

iv. Interruption behavior (中斷行為)：您選擇的行為會決定 Spot 執行個體中斷時會發生什麼狀況。

- 針對持久性請求，有效值為 Stop (停止) 和 Hibernate (休眠)。執行個體停止時，會收取EBS磁碟區儲存的費用。

> **Note**
>
> Spot 執行個體現在使用與隨需執行個體相同的休眠功能。若要啟用休眠功能，您可以在此選擇休眠，也可以從停止 - 休眠行為欄位 (顯示於啟動執行個體精靈下半部) 中選擇啟用。如需休眠必要條件，請參閱Amazon EC2 執行個體休眠的先決條件。

- 針對一次性請求，有效值僅有 Terminate (終止)。

若未指定值，則預設值為 Terminate (終止)，這對於持久性 Spot 執行個體請求無效。如果您保留預設值，並嘗試啟動持久性 Spot 執行個體請求，則會出現錯誤。

如需詳細資訊，請參閱Spot 執行個體中斷的行為。

11. 在 Summary (摘要) 面板上，針對 Number of instances (執行個體的數量)，輸入要啟動的執行個體數量。

> **Note**
>
> Amazon EC2 會為每個競價型執行個體建立個別的請求。

12. 在 Summary (摘要) 面板上，檢閱執行個體的詳細資訊，並進行任何必要的變更。提交 Spot 執行個體請求後，您就無法變更請求的參數。您可以直接導覽至啟動執行個體精靈中的某個區段，方法是在 Summary (摘要) 面板中選擇其連結。如需詳細資訊，請參閱Summary。
13. 當您準備好啟動執行個體時，請選擇 Launch instance (啟動執行個體)。

    如果執行個體無法啟動或狀態立即進入 `terminated` 而非 `running`，請參閱解決 Amazon EC2 執行個體啟動問題。

AWS CLI

若要使用 run-instances 建立 Spot 執行個體請求

使用 run-instances 命令，並在 `--instance-market-options` 參數中指定 Spot 執行個體選項。

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --instance-type t2.micro \
    --count 5 \
    --subnet-id subnet-08fc749671b2d077c \
    --key-name MyKeyPair \
    --security-group-ids sg-0b0384b66d7d692f9 \
    --instance-market-options file://spot-options.json
```

以下是要在JSON檔案中指定的資料結構`--instance-market-options`。您也可以指 `ValidUntil` 和 `InstanceInterruptionBehavior`。如果您沒有在資料結構中指定欄位，則會使用預設值。

以下範例會建立 `persistent` 請求。

```
{
  "MarketType": "spot",
  "SpotOptions": {
    "SpotInstanceType": "persistent"
  }
```

```
}
```

若要使用建立競價型執行個體請求 request-spot-instances

> **Note**
>
> 我們強烈建議使用該request-spot-instances命令請求競價型執行個體，因為它是沒API有計劃投資的舊版。如需詳細資訊，請參閱 使用哪種 Spot 請求方法最好？

使用指request-spot-instances令建立一次性要求。

```
aws ec2 request-spot-instances \
    --instance-count 5 \
    --type "one-time" \
    --launch-specification file://specification.json
```

使用命request-spot-instances令建立持續性要求。

```
aws ec2 request-spot-instances \
    --instance-count 5 \
    --type "persistent" \
    --launch-specification file://specification.json
```

例如，啟動規格檔案以搭配這些命令使用，請參閱Spot 執行個體請求啟動規格範例。如果您從競價型要求主控台下載啟動規格檔案，則必須改用request-spot-fleet命令 (Spot Request 主控台會使用 Spot 叢集指定競價型執行個體請求)。

Spot 執行個體請求啟動規格範例

下列範例顯示可與request-spot-instances命令搭配使用的啟動設定來建立競價型執行個體請求。如需詳細資訊，請參閱管理 Spot 執行個體。

> **Important**
>
> 我們強烈建議使用該request-spot-instances命令請求競價型執行個體，因為它是沒API有計劃投資的舊版。如需詳細資訊，請參閱 使用哪種 Spot 請求方法最好？

範例

- 範例 1：啟動 競價型執行個體
- 範例 2：在指定的可用區域中啟動 競價型執行個體
- 範例 3：在指定的子網中啟動 競價型執行個體
- 範例 4：啟動專用 Spot 執行個體

範例 1：啟動 競價型執行個體

下列範例不包括可用區域或子網。Amazon 為您EC2選擇一個可用區域。Amazon 會在所選可用區域的預設子網路中EC2啟動執行個體。

```
{
  "ImageId": "ami-0abcdef1234567890",
  "KeyName": "my-key-pair",
  "SecurityGroupIds": [ "sg-1a2b3c4d5e6f7g8h9" ],
  "InstanceType": "m5.medium",
  "IamInstanceProfile": {
      "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
  }
}
```

範例 2：在指定的可用區域中啟動 競價型執行個體

下列範例包括可用區域。Amazon 會在指定可用區域的預設子網路中EC2啟動執行個體。

```
{
  "ImageId": "ami-0abcdef1234567890",
  "KeyName": "my-key-pair",
  "SecurityGroupIds": [ "sg-1a2b3c4d5e6f7g8h9" ],
  "InstanceType": "m5.medium",
  "Placement": {
    "AvailabilityZone": "us-west-2a"
  },
  "IamInstanceProfile": {
      "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
  }
}
```

範例 3：在指定的子網中啟動 競價型執行個體

下列範例包含子網。Amazon 會在指定的子網路中EC2啟動執行個體。如果非預設值VPC，則執行個體預設不會收到公開IPv4位址。VPC

```
{
  "ImageId": "ami-0abcdef1234567890",
  "SecurityGroupIds": [ "sg-1a2b3c4d5e6f7g8h9" ],
  "InstanceType": "m5.medium",
  "SubnetId": "subnet-1a2b3c4d",
  "IamInstanceProfile": {
      "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
  }
}
```

若要將公用位IPv4址指派給非預設值中的執行個體VPC，請指定下列範例所示的`AssociatePublicIpAddress`欄位。指定網路介面時，您必須填入網路介面的子網 ID 和安全群組 ID，而非使用前例程式碼區塊所示的 `SubnetId` 和 `SecurityGroupIds` 欄位。

```
{
  "ImageId": "ami-0abcdef1234567890",
  "KeyName": "my-key-pair",
  "InstanceType": "m5.medium",
  "NetworkInterfaces": [
    {
      "DeviceIndex": 0,
      "SubnetId": "subnet-1a2b3c4d5e6f7g8h9",
      "Groups": [ "sg-1a2b3c4d5e6f7g8h9" ],
      "AssociatePublicIpAddress": true
    }
  ],
  "IamInstanceProfile": {
      "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
  }
}
```

範例 4：啟動專用 Spot 執行個體

下列範例請求具有`dedicated`租用的 Spot 執行個體。必須在中啟動專用競價型執行個體VPC。

```
{
  "ImageId": "ami-0abcdef1234567890",
```

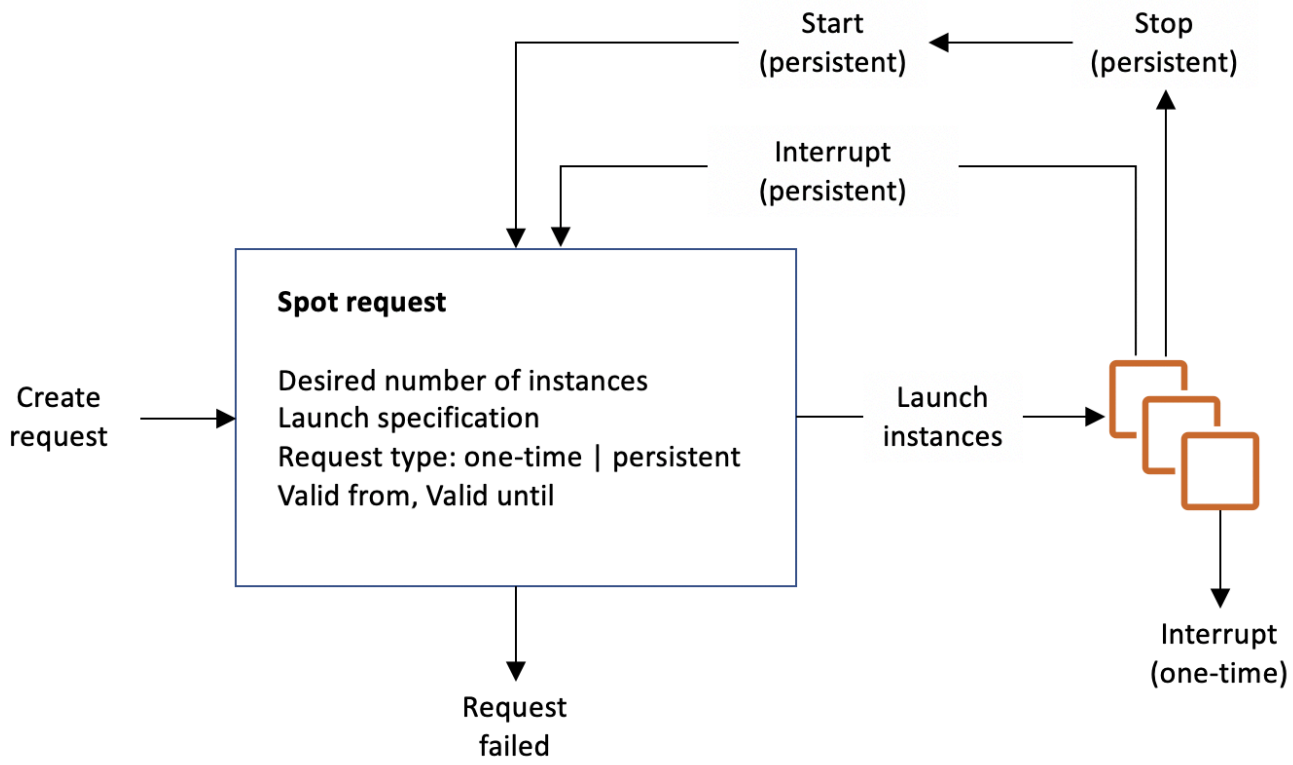
```
  "KeyName": "my-key-pair",
  "SecurityGroupIds": [ "sg-1a2b3c4d5e6f7g8h9" ],
  "InstanceType": "c5.8xlarge",
  "SubnetId": "subnet-1a2b3c4d5e6f7g8h9",
  "Placement": {
    "Tenancy": "dedicated"
  }
}
```

取得 Spot 執行個體請求的狀態

為了協助您追蹤競價型執行個體請求並規劃使用 Spot 執行個體，請使用 Amazon 提供的請求狀態 EC2。例如，請求狀態可提供 Spot 請求尚未完成的理由，或列出阻礙完成 Spot 請求的限制。

在程序的每個步驟中— (此程序也稱為 Spot 請求生命週期)—，特定的事件會決定相繼請求的狀態。

下圖顯示了 Spot 執行個體請求的運作方式。請注意，請求類型 (一次性或永久性) 會決定當 Amazon EC2 中斷 Spot 執行個體或停止 Spot 執行個體時是否再次開啟請求。如果是持久性請求，則會在 Spot 執行個體中斷之後再次開啟。如果是持久性請求，而且您停止 Spot 執行個體，只會在您啟動 Spot 執行個體後開啟請求。



目錄

- 取得請求狀態資訊
- Spot 請求狀態碼
- EC2競價型執行個體請求履行
- Spot 請求的狀態變更

取得請求狀態資訊

您可以使用 AWS Management Console 或命令列工具取得要求狀態資訊。

使用主控台取得要求狀態資訊

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在瀏覽窗格中，選取 Spot 請求 ，然後選取 Spot 請求。
3. 若要檢查狀態，請在描述索引標籤上檢查狀態欄位。

使用命令列來取得請求狀態資訊

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱訪問 Amazon EC2。

- describe-spot-instance-requests (AWS CLI)
- Get-EC2SpotInstanceRequest (AWS Tools for Windows PowerShell)

Spot 請求狀態碼

Spot 請求狀態資訊包含狀態碼、更新時間和狀態訊息。綜合這些資訊，可協助您決定 Spot 請求的處理方式。

下列是 Spot 請求的狀態碼：

`az-group-constraint`

Amazon EC2 無法在同一可用區域啟動您請求的所有執行個體。

`bad-parameters`

Spot 請求的一個或多個參數無效 (例如，AMI您指定的不存在)。狀態訊息會顯示哪個參數無效。

`canceled-before-fulfillment`

Spot 請求在完成之前已遭到使用者取消。

`capacity-not-available`

您所請求的執行個體沒有足夠的可用容量。

`constraint-not-fulfillable`

無法完成 Spot 請求，因為有一個或多個限制條件無效 (例如，可用區域不存在)。狀態訊息會顯示哪個限制條件無效。

`fulfilled`

競價型請求是`active`，Amazon EC2 正在啟動您的競價型執行個體。

`instance-stopped-by-price`

您的執行個體已停止，因為 Spot 價格超過您的最高價格。

`instance-stopped-by-user`

您的執行個體已停止，因為使用者已停止該執行個體或從該執行個體執行關機命令。

`instance-stopped-no-capacity`

您的執行個體因為EC2容量管理需求而停止。

`instance-terminated-by-price`

您的執行個體已終止，因為 Spot 價格超過您的最高價格。如果請求為永久性，程序會重新開始執行，因此請求會處於等待評估的狀態。

`instance-terminated-by-schedule`

您的 Spot 執行個體已在排程持續時間結束時終止。

`instance-terminated-by-service`

執行個體會從停止狀態終止。

`instance-terminated-by-user` 或 `spot-instance-terminated-by-user`

您終止了已完成的 Spot 執行個體，因此請求狀態為 `closed` (除非為持久性請求)，而執行個體狀態為 `terminated`。

`instance-terminated-launch-group-constraint`

啟動群組中的一個或多個執行個體已終止，因此不再滿足啟動群組的限制條件。

`instance-terminated-no-capacity`

由於標準容量管理程序，您的執行個體已終止。

`launch-group-constraint`

Amazon EC2 無法同時啟動您要求的所有執行個體。啟動群組中的所有執行個體會同時啟動和終止。

`limit-exceeded`

超過磁碟區數量或總EBS磁碟區儲存的限制。如需詳細資訊，請參閱 Amazon EBS使用者指南EBS中的 Amazon 配額。

`marked-for-stop`

Spot 執行個體已標記停止。

`marked-for-termination`

Spot 執行個體已標記終止。

`not-scheduled-yet`

在尚未達到排程日期之前，將不會評估 Spot 請求。

`pending-evaluation`

在您發出 Spot 執行個體請求後，在系統評估請求的參數時，該請求會進入 `pending-evaluation` 狀態。

`pending-fulfillment`

Amazon EC2 正在嘗試佈建您的競價型執行個體。

`placement-group-constraint`

尚無法完成 Spot 請求，因為目前無法將 Spot 執行個體加入置放群組。

`price-too-low`

尚無法完成 Spot 請求，因為最高價格低於 Spot 價格。在這種情況中，不會啟動任何執行個體，而請求的狀態仍為 open。

`request-canceled-and-instance-running`

您在 競價型執行個體仍在執行時取消了 Spot 請求。請求已 `cancelled`，但執行個體仍處於 `running` 狀態。

`schedule-expired`

Spot 請求已過期，因為未在指定日期之前完成。

`system-error`

出現未預期的系統錯誤。如果這是一個經常性的問題，請聯繫 AWS Support 尋求幫助。

EC2競價型執行個體請求履行

完成競價型執行個體請求後，Amazon EC2 會將EC2競價型執行個體請求履行事件傳送給 Amazon EventBridge。您可以建立規則以在發生此事件時採取動作，例如叫用 Lambda 函數或通知 Amazon SNS 主題。

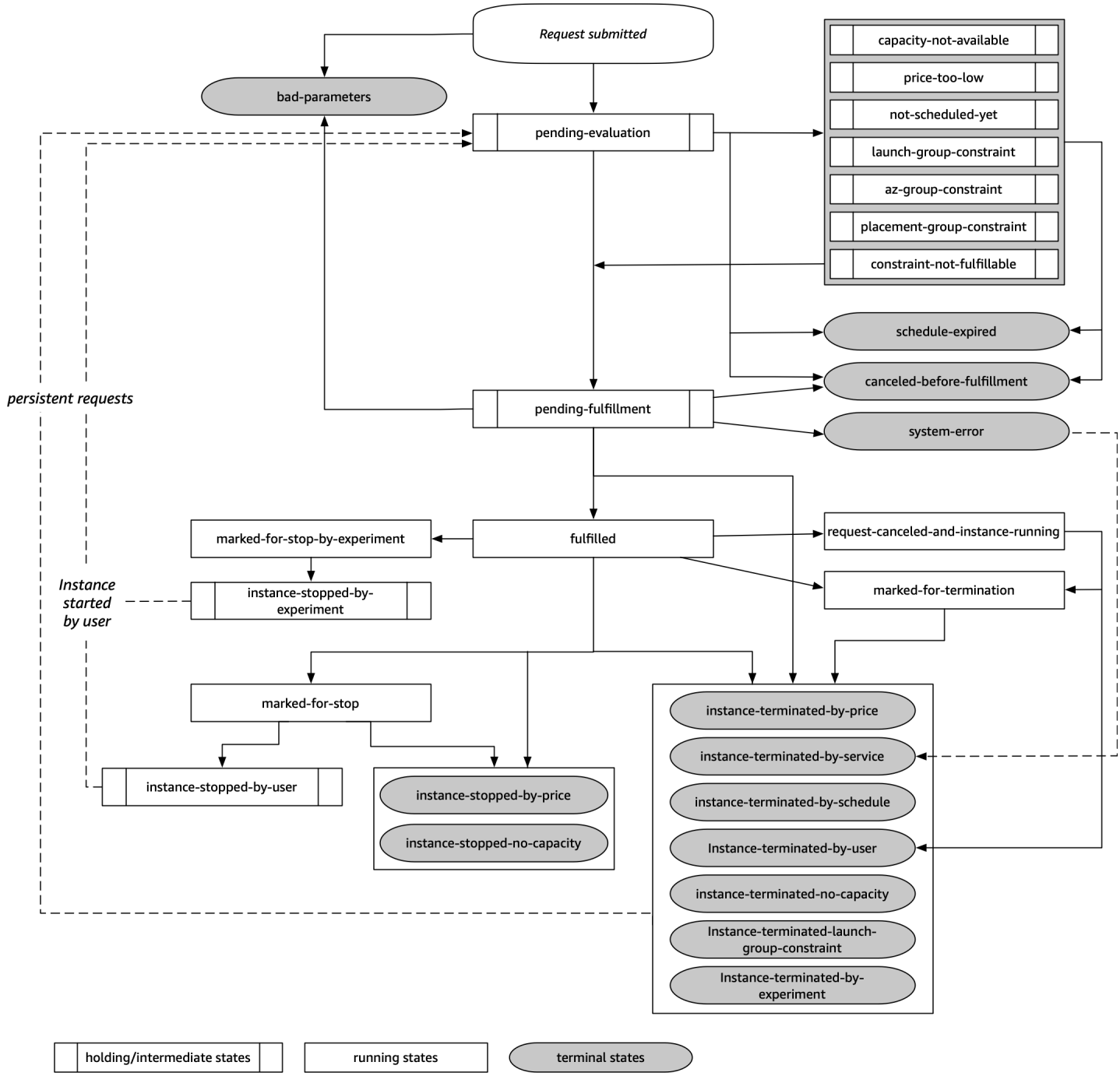
以下是此事件的範例資料。

```
{
    "version": "0",
    "id": "01234567-1234-0123-1234-012345678901",
    "detail-type": "EC2 Spot Instance Request Fulfillment",
    "source": "aws.ec2",
    "account": "123456789012",
    "time": "yyyy-mm-ddThh:mm:ssZ",
    "region": "us-east-2",
    "resources": ["arn:aws:ec2:us-east-2:123456789012:instance/i-1234567890abcdef0"],
    "detail": {
        "spot-instance-request-id": "sir-1a2b3c4d",
        "instance-id": "i-1234567890abcdef0"
    }
}
```

如需詳細資訊，請參閱 Amazon EventBridge 使用者指南。

Spot 請求的狀態變更

下圖顯示 Spot 請求在整個生命週期的期間中可遵循的路徑 (從提交到終止)。每個步驟皆以節點表示，而每個節點的狀態碼則描述了 Spot 請求和 Spot 執行個體的狀態。



等待評估

除非有一個或多個請求參數無效 (`pending-evaluation`)，否則當您建立 Spot 執行個體請求後，它就會立即進入 `bad-parameters` 狀態。

狀態碼	請求狀態	執行個體狀態
pending-evaluation	open	不適用
bad-parameters	closed	不適用

保留

如果有一個或多個請求限制為有效但尚未滿足，或是沒有足夠的容量，則請求會進入保留狀態，以等待限制條件滿足。請求選項會影響請求完成的可能性。例如，如果沒有容量，則請求會持續處於保留狀態，直到有可用的容量為止。如果指定了可用區域群組，則請求會持續處於保留狀態，直到可用區域的限制條件滿足。

如果其中一個可用區域中斷，其他可用區域中的 Spot 執行個體請求可用的備用EC2容量可能會受到影響。

狀態碼	請求狀態	執行個體狀態
capacity-not-available	open	不適用
price-too-low	open	不適用
not-scheduled-yet	open	不適用
launch-group-constraint	open	不適用
az-group-constraint	open	不適用
placement-group-constraint	open	不適用
constraint-not-fulfillable	open	不適用

等待評估/實現-終止

如果所建立的請求只在特定期間內有效，而且在請求進入待完成階段之前，此期間就會到期，則此 Spot 執行個體請求可以進入 `terminal` 狀態。如果您取消請求或系統發生錯誤，也會發生此狀況。

狀態碼	請求狀態	執行個體狀態
`schedule-expired`	`cancelled`	不適用
`canceled-before-fulfillment` [1]	`cancelled`	不適用
`bad-parameters`	`failed`	不適用
`system-error`	`closed`	不適用

[1] 若您取消請求。

等待實現

當您所指定的限制條件 (如果有的話) 已滿足，則 Spot 請求會進入 `pending-fulfillment` 狀態。

此時，Amazon EC2 已準備好佈建您請求的執行個體。如果程序在此時停止，可能是因為使用者已在啟動 Spot 執行個體前取消程序。也可能是因為發生未預期的系統錯誤。

狀態碼	請求狀態	執行個體狀態
`pending-fulfillment`	`open`	不適用

已實現

當滿足 Spot 執行個體的所有規格時，Spot 請求即已完成。Amazon EC2 啟動競價型執行個體，這可能需要幾分鐘的時間。如果 Spot 執行個體已進入休眠或是在中斷時停止，它將會繼續處於此狀態，直到請求可以再次完成或取消。

狀態碼	請求狀態	執行個體狀態
fulfilled	active	pending → running
fulfilled	active	stopped → running

如果您停止 Spot 執行個體，則 Spot 請求會進入 marked-for-stop 或 instance-stopped-by-user 狀態，直到 Spot 執行個體可以再次啟動或取消請求為止。

狀態碼	請求狀態	執行個體狀態
marked-for-stop	active	stopping
instance-stopped-by-user[1]	disabled 或是 cancelled[2]	stopped

[1] 如果您停止執行個體或從執行個體執行 shutdown 指令，Spot 執行個體就會進入 instance-stopped-by-user 狀態。停止執行個體之後，您可以重新啟動它。重新啟動時，競價型執行個體請求會返回pending-evaluation狀態，然後 Amazon 在滿足限制時啟EC2動新的 Spot 執行個體。

[2] 如果您停止 Spot 執行個體但不取消請求，Spot 請求狀態則為 disabled。如果 Spot 執行個體已停止且請求到期，請求狀態則為 cancelled。

已實現-終止

只要有該執行個體類型可用的容量，且您未終止執行個體，您的Spot 執行個體就會持續運作。如果 Amazon EC2 必須終止您的 Spot 執行個體，Spot 請求會進入終端狀態。如果取消 Spot 請求或終止 競價型執行個體，請求也會進入終止狀態。

狀態碼	請求狀態	執行個體狀態
request-canceled-and-instance-running	cancelled	running
marked-for-stop	active	running

狀態碼	請求狀態	執行個體狀態
marked-for-termination	active	running
instance-stopped-by-price	disabled	stopped
instance-stopped-by-user	disabled	stopped
instance-stopped-no-capacity	disabled	stopped
instance-terminated-by-price	closed (一次性)，open (持續性)	terminated
instance-terminated-by-schedule	closed	terminated
instance-terminated-by-service	cancelled	terminated
instance-terminated-by-user	closed 或是 cancelled [1]	terminated
instance-terminated-no-capacity	closed (一次性)，open (持續性)	running †
instance-terminated-no-capacity	closed (一次性)，open (持續性)	terminated
instance-terminated-launch-group-constraint	closed (一次性)，open (持續性)	terminated

[1] 如果您終止執行個體但未取消請求，則請求狀態為 closed。如果終止執行個體而且取消請求，則請求狀態為 cancelled。即使您在取消請求之前終止 Spot 執行個體，Amazon EC2 偵測到您的

Spot 執行個體已終止之前，也可能會有一段延遲。在此種情況中，請求的狀態可以是 `cancelled` 或 `closed`。

† 當 Amazon 在需要恢復容量且將執行個體設定為在中EC2斷時終止時中斷時中斷 Spot 執行個體時，狀態會立即設定為 `instance-terminated-no-capacity` (未設定為`marked-for-termination`)。但是，執行個體會保留在 `running` 狀態 2 分鐘，反映執行個體收到 Spot 執行個體中斷通知的 2 分鐘時段。2 分鐘後，執行個體狀態設定為 `terminated`。

中斷實驗

您可以使用 AWS Fault Injection Service 來啟動 Spot 執行個體中斷，以便測試 Spot 執行個體上的應用程式回應方式。如果 AWS FIS 停止 Spot 執行個體，您的 Spot 請求會進入`marked-for-stop-by-experiment`狀態，然後進入`instance-stopped-by-experiment`狀態。如果 AWS FIS 終止 Spot 執行個體，則您的 Spot 請求會進入該`instance-terminated-by-experiment`狀態。如需詳細資訊，請參閱the section called “啟動中斷”。

狀態碼	請求狀態	執行個體狀態
`marked-for-stop-by-experiment`	`active`	`running`
`instance-stopped-by-experiment`	`disabled`	`stopped`
`instance-terminated-by-experiment`	`closed`	`terminated`

永久性請求

當您的 Spot 執行個體終止 (由您或 AmazonEC2) 時，如果 Spot 請求是永久性請求，它會返回`pending-evaluation`狀態，然後 Amazon EC2 可以在滿足限制時啟動新的競價型執行個體。

標記 Spot 執行個體請求

為了協助分類和管理您的 Spot 執行個體請求，您可以使用自訂中繼資料來標記它們。您可以在建立 Spot 執行個體請求時或之後將標記指派給請求。您可以使用 Amazon EC2 主控台或命令列工具指派標籤。

當您標記 Spot 執行個體請求時，系統不會自動標記 Spot 執行個體請求所啟動的執行個體和磁碟區。您需要明確標記 Spot 執行個體請求啟動的執行個體和磁碟區。您可以在啟動期間或之後將標籤指派給 Spot 執行個體和磁碟區。

如需標籤運作方式的詳細資訊，請參閱標記您的 Amazon EC2 資源。

目錄

- 必要條件
- 標記新的 Spot 執行個體請求
- 標記現有的 Spot 執行個體請求
- 檢視 Spot 執行個體請求標籤

必要條件

授予使用者標記資源的許可。如需有關IAM策略和範例原則的詳細資訊，請參閱範例：標籤資源。

您建立的IAM政策取決於您用於建立 Spot 執行個體請求的方法。

- 如果您使用啟動執行個體精靈或 `run-instances` 來請求 競價型執行個體，請參閱 To grant a user the permission to tag resources when using the launch instance wizard or run-instances。
- 如果您使用 `request-spot-instances` 命令請求 Spot 執行個體，請參閱 To grant a user the permission to tag resources when using request-spot-instances。

若要授予使用者許可，以便在使用啟動執行個體精靈或 run-instances 時標記資源

建立包含下列項目的IAM策略：

- `ec2:RunInstances` 動作。這會授予使用者啟動執行個體的許可。
- 對於 `Resource`，請指定 `spot-instances-request`。這可讓使用者建立 Spot 執行個體請求，它們會請求 Spot 執行個體。
- `ec2:CreateTags` 動作。這會授予使用者建立標籤的許可。
- 對於 `Resource`，請指定 `*`。這可讓使用者標記執行個體啟動期間建立的所有資源。

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

```
            "Sid": "AllowLaunchInstances",
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-1::image/*",
                "arn:aws:ec2:us-east-1:*:subnet/*",
                "arn:aws:ec2:us-east-1:*:network-interface/*",
                "arn:aws:ec2:us-east-1:*:security-group/*",
                "arn:aws:ec2:us-east-1:*:key-pair/*",
                "arn:aws:ec2:us-east-1:*:volume/*",
                "arn:aws:ec2:us-east-1:*:instance/*",
                "arn:aws:ec2:us-east-1:*:spot-instances-request/*"
            ]
        },
        {
            "Sid": "TagSpotInstanceRequests",
            "Effect": "Allow",
            "Action": "ec2:CreateTags",
            "Resource": "*"
        }
    ]
}
```

當您使用該 RunInstances 動作建立競價型執行個體請求並在建立時標記競價型執行個體請求時，您需要瞭解 Amazon 如何在IAM政策中評EC2估`spot-instances-request`資源的 RunInstances陳述式中評估資源，如下所示：

- 如果您在建立時未標記競價型執行個體請求，Amazon EC2 不會評估 RunInstances陳述式中的`spot-instances-request`資源。
- 如果您在建立時標記競價型執行個體請求，Amazon EC2 會評估 RunInstances陳述式中的`spot-instances-request`資源。

因此，對於`spot-instances-request`資源，下列規則會套用至IAM策略：

- 如果您使 RunInstances 用建立競價型執行個體請求，且不打算在建立時標記競價型執行個體請求，則不需要明確允許`spot-instances-request`資源；呼叫將會成功。
- 如果您使 RunInstances 用建立競價型執行個體請求並打算在建立時標記競價型執行個體請求，則必須在 RunInstances allow 陳述式中包含`spot-instances-request`資源，否則呼叫將會失敗。

- 如果您使 RunInstances 用建立競價型執行個體請求並打算在建立時標記競價型執行個體請求，則必須在 CreateTags allow 陳述式中指定`spot-instances-request`資源或包含`*`萬用字元，否則呼叫將失敗。

如需IAM政策範例，包括 Spot 執行個體請求不支援的政策，請參閱使用 競價型執行個體。

授與使用者在使用時標記資源的權限 request-spot-instances

建立包含下列項目的IAM策略：

- `ec2:RequestSpotInstances` 動作。這會授予使用者建立 Spot 執行個體請求的許可。
- `ec2:CreateTags` 動作。這會授予使用者建立標籤的許可。
- 對於 `Resource`，請指定 `spot-instances-request`。這允許使用者只標記 Spot 執行個體請求。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "TagSpotInstanceRequest",
            "Effect": "Allow",
            "Action": [
                "ec2:RequestSpotInstances",
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:us-east-1:111122223333:spot-instances-request/*"
}
```

標記新的 Spot 執行個體請求

Console

若要使用主控台標記新的 Spot 執行個體請求

1. 遵循管理 Spot 執行個體程序。
2. 若要新增標記，請在新增標記頁面上選擇新增標記，然後輸入標記的金鑰和值。為每個額外的標籤選擇新增其他標籤。

對於每個標籤，您可以用相同的標籤來標記 Spot 執行個體請求、Spot 執行個體以及磁碟區。若要標記所有這三個項目，請確保已選取執行個體、磁碟區和 Spot 執行個體請求。若只要標記一個或兩個，請確定已選取要標記的資源，並清除其他資源。

3. 完成建立 Spot 執行個體請求的必要欄位，然後選擇 Launch (啟動)。如需詳細資訊，請參閱管理 Spot 執行個體。

AWS CLI

若要使用標記新的 Spot 執行個體請求 AWS CLI

若要在建立 Spot 執行個體請求時進行標記，請依照下列方式設定 Spot 執行個體請求組態：

- 使用 `--tag-specification` 參數指定 Spot 執行個體請求的標籤。
- 對於 `ResourceType`，請指定 `spot-instances-request`。如果您指定另一個值，Spot 執行個體請求將失敗。
- 對於 `Tags`，指定金鑰值對。您可以指定多個金鑰值對。

在下面的例子中，Spot 執行個體請求標記了兩個標籤：Key=Environment and Value=Production 和 Key=Cost-Center and Value=123。

```
aws ec2 request-spot-instances \
    --instance-count 5 \
    --type "one-time" \
    --launch-specification file://specification.json \
    --tag-specification 'ResourceType=spot-instances-
request,Tags=[{Key=Environment,Value=Production},{Key=Cost-Center,Value=123}]'
```

標記現有的 Spot 執行個體請求

Console

若要使用主控台標記現有的 Spot 執行個體請求

建立 Spot 執行個體請求後，您可以使用主控台為 Spot 執行個體請求新增標籤。

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在瀏覽窗格中，選擇 Spot 請求。

3. 選取您的 Spot 執行個體請求。
4. 選擇標籤索引標籤，然後選擇建立標籤。

若要使用主控台標記現有的 Spot 執行個體

在您的 Spot 執行個體請求啟動 Spot 執行個體之後，您可以使用主控台將標籤新增到執行個體。如需詳細資訊，請參閱使用主控台新增和移除標籤。

AWS CLI

若要使用標記現有的 Spot 執行個體請求或 Spot 執行個體 AWS CLI

使用建立標籤命令來標記現有資源。在下面的例子中，現有的 Spot 執行個體請求和 Spot 執行個體會使用 Key=purpose and Value=test 進行標記。

```
aws ec2 create-tags \
    --resources sir-08b93456 i-1234567890abcdef0 \
    --tags Key=purpose,Value=test
```

檢視 Spot 執行個體請求標籤

Console

若要使用主控台檢視 Spot 執行個體請求

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選取您的 Spot 執行個體請求並選取 Tags (標籤) 標籤。

AWS CLI

若要描述 Spot 執行個體請求標籤

您可以透過描述 Spot 執行個體請求來檢視 Spot 執行個體請求的標籤。使用命describe-spot-instance-requests令檢視指定競價型執行個體請求的組態，其中包括為請求指定的任何標籤。

```
aws ec2 describe-spot-instance-requests \
    --spot-instance-request-ids sir-EXAMPLE1 \
    --query "SpotInstanceRequests[*].Tags"
```

下列為範例輸出。

```
[
    [
        {
            "Key": "Environment",
            "Value": "Production"
        },
        {
            "Key": "Department",
            "Value": "101"
        }
    ]
]
```

若要取消 Spot 執行個體請求

如果您不再需要 Spot 執行個體請求，可以取消它。您只能取消狀態為 `open`、`active` 或 `disabled` 的 Spot 執行個體請求。

- 當請求尚未處理完成，而且尚未啟動任何執行個體時，Spot 執行個體請求的狀態為 `open`。
- 當請求已處理完成，而且因此啟動了 Spot 執行個體時，Spot 執行個體請求的狀態為 `active`。
- 當您停止 Spot 執行個體時，您的 Spot 執行個體請求的狀態為 `disabled`。

如果您的 Spot 執行個體請求的狀態為 `active`，並且擁有相關聯的執行中 Spot 執行個體，則取消請求並不會終止執行個體。如需有關終止 Spot 執行個體的詳細資訊，請參閱終止 Spot 執行個體。

Console

使用主控台取消 Spot 執行個體請求

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選取競價型執行個體請求。
4. 選擇動作、取消請求。
5. (選擇性) 如果您已完成與 競價型執行個體 的關聯，您可以終止它們。在取消 Spot 請求對話方塊中，選取終止執行個體，然後選取確認。

AWS CLI

若要取消 Spot 執行個體請求 AWS CLI

使用命cancel-spot-instance-requests令取消指定的 Spot 執行個體請求。

```
aws ec2 cancel-spot-instance-requests --spot-instance-request-ids sir-08b93456
```

管理 Spot 執行個體

當容量可用時，Amazon 會EC2啟動競價型執行個體。Spot 執行個體會執行，直到它中斷或您自己終止它。

目錄

- 尋找您的 Spot 執行個
- 停止 Spot 執行個體
- 啟動 Spot 執行個體
- 終止 Spot 執行個體

尋找您的 Spot 執行個

Spot 執行個體會與隨需執行個體一起出現在主控台的「執行個體」頁面中。請使用下列程序來尋找您的 Spot 執行個體。

Console

使用主控台尋找您的 Spot 執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 若要尋找所有 Spot 執行個體，請在搜尋窗格中選擇「執行個體生命週期 = 定點」。
4. 若要確認例證是否為 Spot 例證，請選取例證，選擇「詳細資料」頁籤，然後檢查「生命週期」的值。Spot 執行個體的值為`spot`，隨需執行個體的值為`normal`。

AWS CLI

若要使用 AWS CLI

使用描述實例命令搭配選項。`--filters`

```
aws ec2 describe-instances \
    --filters "Name=instance-lifecycle,Values=spot"
```

判斷執行個體是否為競價型執行個體

使用描述執行個體指令，使用`--query`選項來檢查生命週期值。

```
aws ec2 describe-instances \
    --instance-ids i-0123a456700123456 \
    --query "Reservations[*].Instances[*].InstanceLifecycle" \
    --output text
```

如果輸出為`spot`，則執行個體為 Spot 執行個體。如果沒有輸出，則執行個體為隨需執行個體。

使用下列程序尋找從特定競價型執行個體或 Spot 叢集請求啟動的 Spot 執行個體。

Console

使用主控台尋找要求的 Spot 執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在瀏覽窗格中，選擇 Spot 請求。此清單包含 Spot 執行個體請求和 Spot 叢集請求。
3. 如果已滿足競價型執行個體請求，容量就是競價型執行個體的 ID。對於 Spot Fleet ，容量表示已完成所請求容量的程度。若要檢視 Spot 叢集中IDs的執行個體，請選擇展開箭頭，或選取叢集並選擇執行個體。
4. 對於競價型叢集，容量會指出要求的容量已滿足多少。若要檢視 Spot 叢集中IDs的執行個體，請選擇要開啟其詳細資料頁面的叢集 ID，然後找到執行個體窗格。

AWS CLI

若要使用 AWS CLI

使用指describe-spot-instance-requests令搭配選`--query`項。

```
aws ec2 describe-spot-instance-requests \
    --query "SpotInstanceRequests[*].{ID:InstanceId}"
```

下列為範例輸出：

```
[
    {
        "ID": "i-1234567890abcdef0"
    },
    {
        "ID": "i-0598c7d356eba48d7"
    }
]
```

停止 Spot 執行個體

如果您現在不需要 Spot 執行個體，但想要稍後重新啟動它們，而不會遺失 Amazon EBS 磁碟區中保存的資料，則可以停止它們。停止 Spot 執行個體的步驟與停止隨需執行個體的步驟類似。

> **Note**
>
> 停止 Spot 執行個體時，您可以修改其某些執行個體屬性，但無法修改執行個體類型。我們不會針對停止的競價型執行個體或資料傳輸費用收取使用費，但我們會對任何 Amazon EBS 磁碟區的儲存收取費用。

限制

- 如果從 `persistent` Spot 執行個體請求中啟動 Spot 執行個體，則只能停止 Spot 執行個體。
- 如果已取消關聯的 Spot 執行個體請求，則無法停止 Spot 執行個體。當 Spot 執行個體請求被取消時，您只能終止 Spot 執行個體。
- 如果 Spot 執行個體屬於某個機群或啟動群組或可用區域群組，則無法停止它。

Console

使用主控台停止 Spot 執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取定點例證。如果您未儲存競價型執行個體的執行個體 ID，請參閱the section called “尋找您的 Spot 執行個”。

4. 選擇 Instance state (執行個體狀態)、Stop instance (停止執行個體)。
5. 出現確認提示時，請選擇 Stop (停止)。

AWS CLI

若要使用停止定點例證 AWS CLI

使用停止執行個體指令手動停止 Spot 執行個體。

```
aws ec2 stop-instances --instance-ids i-1234567890abcdef0
```

啟動 Spot 執行個體

您可以啟動之前已停止的 Spot 執行個體。

必要條件

只有在下列情況下，才能啟動 Spot 執行個體：

- 您手動停止了 Spot 執行個體。
- Spot 執行個體是一個EBS支援的執行個體。
- Spot 執行個體容量可用。
- Spot 價格低於您的最高價格。

限制

- 如果 Spot 執行個體屬於某個機群或啟動群組或可用區域群組，則無法啟動它。

啟動 Spot 執行個體的步驟與啟動隨需執行個體的步驟類似。

Console

使用主控台啟動 Spot 執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取定點例證。如果您未儲存競價型執行個體的執行個體 ID，請參閱the section called “尋找您的 Spot 執行個”。

4. 選擇 Instance state (執行個體狀態)、Start instance (啟動執行個體)。

AWS CLI

若要啟動定點例證 AWS CLI

使用啟動執行個體指令手動啟動 Spot 執行個體。

```
aws ec2 start-instances --instance-ids i-1234567890abcdef0
```

終止 Spot 執行個體

如果終止持久性 Spot 執行個體請求啟動的執行中或已停止的 Spot 執行個體，Spot 執行個體請求會轉變為 `open` 狀態，以便啟動新的 Spot 執行個體。若要確保沒有啟動新的 Spot 執行個體，您必須先取消 Spot 執行個體請求。

如果您取消具有執行中 Spot 執行個體的 `active` Spot 執行個體請求，則執行中的 Spot 執行個體不會自動終止；您必須手動終止 Spot 執行個體。

如果您取消已停止 `disabled` Spot 執行個體的競價型執行個體請求，Amazon S EC2 pot 服務會自動終止已停止的競價型執行個體。取消 Spot 執行個體請求和 Spot 服務終止 Spot 執行個體之間可能有短暫延遲。

如需詳細資訊，請參閱若要取消 Spot 執行個體請求。

Console

若要使用主控台手動終止 Spot 執行個體

1. 終止執行個體之前，請確認您不會遺失任何資料，方法是確認您的 Amazon EBS 磁碟區在終止時不會被刪除，以及您已將所需的任何資料從執行個體存放磁碟區複製到永久性儲存，例如 Amazon EBS 或 Amazon S3。
2. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
3. 在導覽窗格中，選擇 Instances (執行個體)。
4. 選取定點例證。如果您未儲存競價型執行個體的執行個體 ID，請參閱the section called “尋找您的 Spot 執行個”。
5. 選擇 Instance state (執行個體狀態)、Terminate instance (終止執行個體)。

6. 出現確認提示時，請選擇 Terminate (終止)。

AWS CLI

若要使用 AWS CLI

使用終止執行個體命令手動終止 Spot 執行個體。

```
aws ec2 terminate-instances --instance-ids i-1234567890abcdef0 i-0598c7d356eba48d7
```

Spot 執行個體中斷

您可以在備用EC2容量上啟動競價型執行個體以獲得大幅折扣，以換取在 Amazon EC2 需要返回容量時將其退還。Amazon EC2 回收 Spot 執行個體時，我們稱此事件為 Spot 執行個體中斷。

對競價型執行個體的需求可能會有很大的不同，而且 Spot 執行個體的可用性也會因未使用的EC2執行個體可用數量而有很大差異。Spot 執行個體隨時都有可能中斷。以下是 Amazon 可EC2能會中斷您的 Spot 執行個體的可能原因：

容量

Amazon EC2 可以在需要時中斷您的競價型執行個體。EC2回收執行個體主要是為了重新規劃容量，但也可能因為主機維護或硬體停用等其他原因而發生。

價格

Spot 價格高於您的最高價格。

您可以在 Spot 請求中指定最高價格。然而，如果您指定了最高價格，執行個體將比未指定最高價時更頻繁受到中斷。

限制條件

如果 Spot 請求包含限制條件，例如啟動群組或可用區域群組，則不再滿足限制條件時，Spot 執行個體會以群組的形式終止。

Amazon 中EC2斷 Spot 執行個體時，會根據您在建立 Spot 請求時指定的中斷行為，終止、停止或休眠執行個體。

目錄

- Spot 執行個體中斷的行為
- 為 Spot 執行個體中斷做好準備
- 啟動 Spot 執行個體中斷
- Spot 執行個體中斷通知
- 尋找中斷的 競價型執行個體
- 判斷 Amazon 是否EC2終止競價型執行個體
- 針對已中斷的 競價型執行個體收費

Spot 執行個體中斷的行為

建立 Spot 要求時，您可以指定中斷行為。以下是可能的中斷行為：

- 停止
- 休眠
- 終止

預設行為是 Amazon EC2 會在 Spot 執行個體中斷時終止它們。

停止中斷的 競價型執行個體

您可以指定 Amazon 在 Spot 執行個體中斷時EC2停止它們。Spot 執行個體請求類型必須為 `persistent`。您無法在 Spot 執行個體請求中指定啟動群組。對於「EC2叢集」或「競價型叢集」，請求類型必須是`maintain`。

考量事項

- 只有 Amazon EC2 可以重新啟動中斷的停止 Spot 執行個體。
- 對於由競價型執行個體請求啟動的`persistent`競價型執行個體：當容量在相同的可用區域中可用時，Amazon 會EC2重新啟動已停止的執行個體，以及與停止執行個體相同的執行個體類型 (必須使用相同的啟動規格)。
- 停止 Spot 執行個體時，您可以修改其某些執行個體屬性，但無法修改執行個體類型。如果分離或刪除EBS磁碟區，則在啟動 Spot 例證時不會貼附該磁碟區。如果您分離根磁碟區，而 Amazon EC2 嘗試啟動競價型執行個體，則執行個體將無法啟動，而 Amazon EC2 將終止已停止的執行個體。
- 您可以在 Spot 執行個體停止時將其終止。
- 如果您取消競價型執行個體請求、EC2叢集或 Spot 叢集，Amazon EC2 會終止任何已停止的相關競價型執行個體。

- 當中斷的 Spot 執行個體停止時，您只需支付保留EBS磁碟區的費用。使用EC2叢集和 Spot Fleet 時，如果您有許多停止的執行個體，您可以超過帳戶的EBS磁碟區數量上限。如需在 Spot 執行個體中斷時計費方式的詳細資訊，請參閱 針對已中斷的 競價型執行個體收費。
- 確定您熟悉停止執行個體的含義。如需停止執行個體時會發生狀況的資訊，請參閱 實例狀態之間的差異。

休眠中斷的 競價型執行個體

您可以指定 Amazon 在 Spot 執行個EC2體中斷時將其休眠。如需詳細資訊，請參閱休眠您的 Amazon EC2 實例。

Amazon EC2 現在提供與隨需執行個體目前可用的 Spot 執行個體相同的休眠體驗。支援更廣泛，現在針對 Spot 執行個體休眠支援下列內容：

- 更多支援 AMIs
- 更多支援的執行個體系列
- 使用者啟動的休眠

終止已中斷的 Spot 執行個體

Amazon 中EC2斷 Spot 執行個體時，預設會終止執行個體，除非您指定不同的中斷行為，例如停止或休眠。如需詳細資訊，請參閱終止 Amazon EC2 實例。

為 Spot 執行個體中斷做好準備

對競價型執行個體的需求可能會有很大的不同，而且 Spot 執行個體的可用性也會因未使用的EC2執行個體可用數量而有很大差異。Spot 執行個體隨時都有可能中斷。因此，您必須確定應用程式已經為 Spot 執行個體中斷做好準備。

建議您遵循這些最佳實務，以便為 Spot 執行個體中斷做好準備。

- 使用 Auto Scaling 群組建立 Spot 請求。如果您的 Spot 執行個體中斷，Auto Scaling 群組將自動啟動替換執行個體。如需詳細資訊，請參閱 Amazon Auto Scaling 使用者指南中的具有多個執行個體類型的 EC2 Auto Scaling 群組和購買選項。
- 使用包含所需軟體組態的 Amazon Machine Image (AMI)，確保您的執行個體在完成請求後立即準備就緒。您也可以利用使用者資料，在啟動時執行命令。
- 在停止或終止執行個體時，執行個體儲存體磁碟區上的資料會遺失。將執行個體存放磁碟區上的任何重要資料備份到更持久的儲存，例如 Amazon S3 EBS、亞馬遜或 Amazon DynamoDB。

- 請將重要的資料定期存放於當 Spot 執行個體終止時不會受到影響的位置。例如，您可以使用 Amazon S3 EBS、Amazon 或 DynamoDB。
- 將工作分割為較小的任務 (使用 Grid、Hadoop 或採用佇列的架構) 或使用檢查點，如此您就可以經常儲存工作。
- 當執行個EC2體有中斷風險增加時，Amazon 會向競價型執行個體發出重新平衡建議信號。您可以依賴重新平衡建議，主動管理 Spot 執行個體中斷，而不必等待兩分鐘的 Spot 執行個體中斷通知。如需詳細資訊，請參閱 EC2例項重新平衡建議。
- 使用兩分鐘的 Spot 執行個體中斷通知來監控 Spot 執行個體的狀態。如需詳細資訊，請參閱 Spot 執行個體中斷通知。
- 雖然我們努力地盡快提供這些警示，但您的 Spot 執行個體還是有可能在警示發布之前中斷。測試您的應用程式，確保它能夠從容地處理非預期的執行個體中斷，即使您正在監控重新平衡建議訊號和中斷通知。您可以執行使用隨需執行個體的應用程式，然後自行終止隨需執行個體，從而完成此任務。
- 執行受控故障注入實驗， AWS Fault Injection Service 以測試 Spot 執行個體中斷時應用程式的回應方式。如需詳細資訊，請參閱 AWS Fault Injection Service 使用者指南中教學課程：使用 AWS FIS 測試 Spot 執行個體中斷。

啟動 Spot 執行個體中斷

您可以在 Amazon EC2 主控台中選取 Spot 執行個體請求或 Spot 叢集請求，並啟動競價型執行個體中斷，以便測試 Spot 執行個體上的應用程式如何處理被中斷。當您啟動競價型執行個體中斷時，Amazon 會EC2通知您 Spot 執行個體將在兩分鐘內中斷，然後兩分鐘後執行個體中斷。

執行 Spot 執行個體中斷的基礎服務為 AWS Fault Injection Service (AWS FIS)。如需有關的資訊 AWS FIS，請參閱AWS Fault Injection Service。

> **Note**
>
> 中斷行為是 `terminate`、`stop` 以及 `hibernate`。如果將中斷行為設定為 `hibernate`，當您啟動 Spot 執行個體中斷時，休眠程序將立即開始。

AWS 區域 除亞太區域 (雅加達)、亞太區域 (大阪)、中國 (北京)、中國 (寧夏) 和中東 (UAE) 以外的所有項目都支援啟動 Spot 執行個體中斷。

目錄

- 啟動 Spot 執行個體中斷

- 確認 Spot 執行個體中斷
- 配額

啟動 Spot 執行個體中斷

您可以使用主EC2控台快速起始 Spot 執行個體中斷。當您選取 Spot 執行個體請求時，您可以啟動一個 Spot 執行個體的中斷。當您選取 Spot 機群請求時，您可以一次啟動多個 Spot 執行個體的中斷。

如需測試 Spot 執行個體中斷的更進階實驗，您可以使用 AWS FIS 主控台建立自己的實驗。

使用EC2主控台啟動 Spot 執行個體請求中的一個 Spot 執行個體中斷

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在瀏覽窗格中，選擇 Spot Requests (Spot 請求)。
3. 選取 Spot 執行個體請求，然後選取 Actions (動作)、Initiate interruption (啟動中斷)。您無法選取多個 Spot 執行個體請求來啟動中斷。
4. 在 Initiate Spot Instance interruption (啟動 Spot 執行個體中斷) 對話方塊，於 Service access (服務存取) 下，使用預設角色或選擇現有角色。若要選擇現有角色，請選擇 [使用現有的服務角色]，然後針對IAM角色選取要使用的角色。
5. 當您準備好要啟動 Spot 執行個體中斷時，請選擇 Initiate interruption (啟動中斷)。

使用EC2主控台啟動 Spot 叢集請求中的一或多個 Spot 執行個體中斷

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在瀏覽窗格中，選擇 Spot Requests (Spot 請求)。
3. 選取 Spot 機群請求，然後選取動作、啟動中斷。您無法選取多個 Spot 機群請求來啟動中斷。
4. 在指定 Spot 執行個體的數目對話方塊中，對於要中斷之執行個體的數目，輸入要中斷之 Spot 執行個體的數目，然後選擇確認。

   > **Note**
   >
   > 該數量不得超過叢集中的 Spot 執行個體數量，或者您每個實驗 AWS FIS 可能中斷的 Spot 執行個體數量配額。

5. 在 Initiate Spot Instance interruption (啟動 Spot 執行個體中斷) 對話方塊，於 Service access (服務存取) 下，使用預設角色或選擇現有角色。若要選擇現有角色，請選擇 [使用現有的服務角色]，然後針對IAM角色選取要使用的角色。

6. 當您準備好要啟動 Spot 執行個體中斷時，請選擇 Initiate interruption (啟動中斷)。

使用 AWS FIS 主控台建立更進階的實驗以測試 Spot 執行個體中斷

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在瀏覽窗格中，選擇 Spot Requests (Spot 請求)。
3. 選擇 Actions (動作)，Create advanced experiments (建立進階實驗)。

   AWS FIS 主控台隨即開啟。如需詳細資訊，請參閱《AWS Fault Injection Service 使用者指南》中的教學課程：使用 AWS FIS測試 Spot 執行個體中斷。

確認 Spot 執行個體中斷

在您啟動中斷後，會發生下列情況：

- Spot 執行個體會收到執行個體重新平衡建議。
- Spot 執行個體中斷通知會在執行個體中 AWS FIS 斷前兩分鐘發出。
- 兩分鐘後，系統會中斷 Spot 執行個體。
- 由停止的 Spot 執行個體會 AWS FIS 保持停止狀態，直到您重新啟動為止。

確認執行個體在您啟動中斷後是否已中斷

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 從導覽窗格中，在單獨的瀏覽器索引標籤或視窗中開啟 Spot Requests (Spot 請求) 和 Instances (執行個體)。
3. 對於 Spot 請求，選取 Spot 執行個體請求或 Spot 機群請求。起始狀態為 `fulfilled`。執行個體中斷後，根據中斷行為，狀態會有如下變更：
   - `terminate` – 狀態會變成 `instance-terminated-by-experiment`。
   - `stop` – 狀態會變成 `marked-for-stop-by-experiment`，然後再變成 `instance-stopped-by-experiment`。
4. 對於 Instances (執行個體)，選取 Spot 執行個體。起始狀態為 `Running`。收到 Spot 執行個體中斷通知後兩分鐘，根據中斷行為，狀態會有如下變更：
   - `stop` – 狀態會變成 `Stopping`，然後再變成 `Stopped`。
   - `terminate` – 狀態會變成 `Shutting-down`，然後再變成 `Terminated`。

配額

對於每 AWS 帳戶 個實驗 AWS FIS 可中斷的 Spot 執行個體數量，您的預設配額如下。

名稱	預設	可調整	描述
aws SpotInstances 的目標：ec2：send-spot-instance-interruptions	每個受支援的區域：5	是	當您使用每個實驗使用標籤識別目標時，aw: ec2: send-spot-instance-interruptions 可以鎖定的 Spot 執行個體數目上限。

您可以要求增加配額。如需詳細資訊，請參閱《Service Quotas 使用者指南》中的請求提高配額。

若要檢視的所有配額 AWS FIS，請開啟「Service Quotas」主控台。在導覽窗格中，選擇 AWS 服務，然後選取 AWS Fault Injection Service。您也可以在 AWS Fault Injection Service 使用者指南中檢視 AWS Fault Injection Service 的所有配額。

Spot 執行個體中斷通知

競價型執行個體中斷通知是在 Amazon EC2 停止或終止 Spot 執行個體前兩分鐘發出警告。如果將休眠指定為中斷時的行為，您會收到中斷通知，但不會提前兩分鐘收到警告，因為休眠程序會立刻開始執行。

從容處理 Spot 執行個體中斷的最佳方法，就是建立容錯的應用程式架構。若要達成此目的，您可以利用 Spot 執行個體中斷通知。建議您每隔 5 秒檢查這些中斷通知。

中斷通知可作為 EventBridge 事件和 Spot 執行個體上執行個體中繼資料中的項目提供。中斷通知的原則是盡可能發出。

EC2 Spot Instance Interruption Warning 事件

當 Amazon EC2 打算中斷您的 Spot 執行個體時，它會在實際中斷前兩分鐘發出事件 (休眠除外，這會取得中斷通知，但不會提前兩分鐘發出，因為休眠會立即開始)。Amazon 可以檢測到此事件 EventBridge。如需有關 EventBridge 事件的詳細資訊，請參閱 Amazon EventBridge 使用者指南。如需逐步引導您如何建立和使用事件規則的詳細範例，請參閱利用 Amazon EC2 Spot 執行個體中斷通知。

下列是 Spot 執行個體中斷事件的範例。`instance-action` 可能的值為 `hibernate`、`stop` 或 `terminate`。

```
{
    "version": "0",
    "id": "12345678-1234-1234-1234-123456789012",
    "detail-type": "EC2 Spot Instance Interruption Warning",
    "source": "aws.ec2",
    "account": "123456789012",
    "time": "yyyy-mm-ddThh:mm:ssZ",
    "region": "us-east-2",
    "resources": ["arn:aws:ec2:us-east-2a:instance/i-1234567890abcdef0"],
    "detail": {
        "instance-id": "i-1234567890abcdef0",
        "instance-action": "action"
    }
}
```

> **Note**
>
> Spot 執行個體中斷事件的ARN格式為`arn:aws:ec2:availability-zone:instance/instance-id`。此格式與資EC2源格ARN式不同。

instance-action

該`instance-action`項目指定動作和大概的時間, 在UTC, 當動作將發生.

如果您的競價型執行個體被 Amazon 標記為停止或終止EC2，則該`instance-action`項目會出現在執行個體中繼資料中。否則不會存在。您可以`instance-action`使用執行個體中繼資料服務版本 2 (IMDSv2) 擷取，如下所示。

cURL

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/spot/instance-action
```

PowerShell

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/meta-data/spot/instance-action
```

下列範例輸出指出此執行個體將會停止的時間點。

```
{"action": "stop", "time": "2017-09-18T08:22:00Z"}
```

下列範例輸出指出此執行個體將會終止的時間點。

```
{"action": "terminate", "time": "2017-09-18T08:22:00Z"}
```

如果 Amazon EC2 不準備停止或終止執行個體，或者您自行終止執行個體，則執行`instance-action`個體中繼資料中不存在，且您在嘗試擷取執行個體時會收到 HTTP 404 錯誤訊息。

termination-time

此`termination-time`項目會指定執行個體UTC接收關機訊號的大約時間。

> **Note**
>
> 保留此項目是為了回溯相容性；您應改用 `instance-action`。

如果您的競價型執行個體被 Amazon 標記為終止 EC2 (可能是因為中斷行為設定為的競價型執行個體中斷`terminate`，或是因為永久 Spot 執行個體請求取消)，則該`termination-time`項目會出現在執行個體中繼資料中。否則不會存在。您可以`termination-time`使用如下方IMDSv2式檢索。

cURL

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"`
[ec2-user ~]$ if curl -H "X-aws-ec2-metadata-token: $TOKEN" -s
 http://169.254.169.254/latest/meta-data/spot/termination-time | grep -q .*T.*Z;
 then echo termination_scheduled; fi
```

PowerShell

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/meta-data/spot/termination-time
```

下列為範例輸出。

```
2015-01-05T18:02:00Z
```

如果 Amazon EC2 沒有準備終止執行個體 (可能是因為沒有競價型執行個體中斷或是因為您的中斷行為設定為`stop`或`hibernate`)，或者如果您自行終止 Spot 執行個體，則該`termination-time`項目可能不存在於執行個體中繼資料中 (因此您會收到 HTTP 404 錯誤)，或者包含非時間值的值。

如果 Amazon EC2 無法終止執行個體，請求狀態會設為`fulfilled`。`termination-time` 值仍會存在於執行個體中繼資料中，包含原來的約略時間 (現在這個時間已經成為過去)。

尋找中斷的 競價型執行個體

在主控台中，執行個體窗格會顯示所有執行個體，包括 競價型執行個體。Spot 執行個體的執行個體生命週期為 `spot`。Spot 執行個體的執行個體狀態為 `stopped` 或 `terminated`，視您設定的中斷行為而定。若為已休眠的 Spot 執行個體，執行個體狀態為 `stopped`。

使用主控台尋找中斷的 Spot 執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 套用下列篩選條件：執行個體生命週期=spot。
4. 根據您設定的中斷行為，套用執行個體狀態=已停止或執行個體狀態=已終止篩選條件。
5. 對於每個 Spot 執行個體，在詳細資訊索引標籤的執行個體詳細資訊下，找到狀態轉換訊息 下列程式碼指出 Spot 執行個體已中斷。
   - `Server.SpotInstanceShutdown`
   - `Server.SpotInstanceTermination`
6. 如需中斷原因的其他詳細資訊，請檢查 Spot 要求狀態碼。如需詳細資訊，請參閱the section called “取得 Spot 執行個體請求的狀態”。

若要尋找中斷的 Spot 執行個體，請 AWS CLI

您可以使用 describe-instances 命令搭配 `--filters` 參數來列出中斷的 競價型執行個體。若只要列出輸出IDs中的例證，請包括`--query`參數。

如果執行個體中斷行為是終止 Spot 執行個體，請使用下列命令：

```
aws ec2 describe-instances \
    --filters Name=instance-lifecycle,Values=spot Name=instance-state-
name,Values=terminated Name=state-reason-code,Values=Server.SpotInstanceTermination \
    --query "Reservations[*].Instances[*].InstanceId"
```

如果執行個體中斷行為是停止 Spot 執行個體，請使用下列命令：

```
aws ec2 describe-instances \
    --filters Name=instance-lifecycle,Values=spot Name=instance-state-
name,Values=stopped Name=state-reason-code,Values=Server.SpotInstanceShutdown \
    --query "Reservations[*].Instances[*].InstanceId"
```

判斷 Amazon 是否EC2終止競價型執行個體

Spot 執行個體EC2會一直執行，直到 Amazon 因應競價型執行個體中斷而終止執行，或直到您自行終止為止。如需詳細資訊，請參閱the section called “中斷行為”。

Spot 執行個體終止後，您可以使用 AWS CloudTrail 來查看 Amazon 是否EC2已終止該執行個體。如果 CloudTrail 日誌中包含一個`BidEvictedEvent`，這表明 Amazon EC2 終止了競價型執行個體。如果您看到`TerminateInstances`事件，則表示使用者終止了 Spot 執行個體。

或者，如果您想收到 Amazon EC2 將中斷 Spot 執行個體的通知，請使用 Amazon EventBridge 回應EC2競價型執行個體中斷警告事件。

若要檢視中的BidEvictedEvent事件 CloudTrail

1. 在開啟 CloudTrail 主控台https://console.aws.amazon.com/cloudtrail/。
2. 在導覽窗格中，選擇 Event history (事件歷史記錄)。
3. 從篩選器清單中，選擇 [事件名稱]，然後在右側的篩選欄位中輸入**`BidEvictedEvent`**。
4. (選擇性) 選取時間範圍。
5. 如果清單不是空的，請BidEvictedEvent從結果項目中選擇以開啟其詳細資訊頁面。您可以在事件記錄窗格中找到有關競價型執行個體的資訊，包括競價型執行個體的 ID。以下是事件記錄的範例。

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "accountId": "123456789012",
        "invokedBy": "ec2.amazonaws.com"
    },
    "eventTime": "2016-08-16T22:30:00Z",
    "eventSource": "ec2.amazonaws.com",
    "userAgent": "ec2.amazonaws.com",
    "sourceIPAddress": "ec2.amazonaws.com",
    "eventName": "BidEvictedEvent",
    "awsRegion": "us-east-2",
    "eventID": "d27a6096-807b-4bd0-8c20-a33a83375054",
    "eventType": "AwsServiceEvent",
    "recipientAccountId": "123456789012",
    "RequestParameters": null,
    "ResponseElements": null,
    "serviceEventDetails": {
        "instanceIdSet": [
          "i-1eb2ac8eEXAMPLE"
        ]
    }
}
```

6. 如果找不到`BidEvictedEvent`事件的項目，請輸入**`TerminateInstances`**事件名稱。若要取得有關的事件記錄的更多資訊`TerminateInstances`，請參閱the section called “Amazon EC2 API 活動示例”。

針對已中斷的 競價型執行個體收費

Spot 執行個體中斷時，系統會依執行個體和EBS磁碟區使用量向您收費，並且可能會產生其他費用，如下所示。

執行個體用量

誰中斷 Spot 執行個體	作業系統	在第一個小時中斷	在第一個小時之後的任何小時中斷
如果您停止或終止 Spot 執行個體	視窗和 Linux 系統 (不包括SUSE)	按使用秒數收費	按使用秒數收費

誰中斷 Spot 執行個體	作業系統	在第一個小時中斷	在第一個小時之後的任何小時中斷
如果 Amazon 中EC2斷競價型執行個體	SUSE	按整整一小時收費，即使只使用半小時亦同	按使用的完整時數收費，中斷半小時以整整一小時收費
	視窗和 Linux 系統 (不包括SUSE)	免費	按使用秒數收費
	SUSE	免費	按使用的完整時數收費，但中斷半小時免費

EBS磁碟區使用

當中斷的 Spot 執行個體停止時，您只需支付保留EBS磁碟區的費用。

使用EC2叢集和 Spot Fleet 時，如果您有許多停止的執行個體，您可以超過帳戶的EBS磁碟區數量上限。

EC2例項重新平衡建議

EC2執行個體重新平衡建議是一種訊號，會在 Spot 執行個體處於中斷風險增加時通知您。該訊號能夠比兩分鐘 Spot 執行個體中斷通知更快到達，讓您有機會主動管理 Spot 執行個體。您可以決定將工作負載重新平衡至未處於提高之中斷風險的新的或現有 競價型執行個體。

Amazon 並非總是可EC2以在兩分鐘 Spot 執行個體中斷通知之前傳送重新平衡建議訊號。因此，重新平衡建議訊號可以與兩分鐘的中斷通知一起到達。

重新平衡建議可作為 EventBridge 事件和 Spot 執行個體上執行個體中繼資料中的項目提供。盡可能發出事件。

> **Note**
>
> 只有在 2020 年 11 月 5 日 00:00 UTC 之後啟動的 Spot 執行個體才支援重新平衡建議。

目錄

- 您可以採取的重新平衡動作
- 監控重新平衡建議訊號
- 使用重新平衡建議訊號的服務

您可以採取的重新平衡動作

以下是您可以採取的一些可能的重新平衡動作：

順利關機

當您收到 Spot 執行個體的重新平衡建議訊號時，您可以啟動執行個體關機程序，其中可能包括確保在程序停止之前完成程序。例如，您可以將系統或應用程式日誌上傳到 Amazon 簡單儲存服務 (Amazon S3)、關閉 Amazon SQS 工作者，或從網域名稱系統 (DNS) 完成取消註冊。您也可以將工作儲存在外部儲存體中，稍後再繼續執行。

防止排程新工作

當您收到 Spot 執行個體的重新平衡建議訊號時，您可以防止在執行個體上排程新工作，同時繼續使用執行個體，直到排定的工作完成為止。

主動啟動新的替代執行個體

您可以將 Auto Scaling 群組、EC2叢集或 Spot 叢集設定為在發出重新平衡建議訊號時自動啟動替換 Spot 執行個體。如需詳細資訊，請參閱 Amazon EC2Auto Scaling 使用者指南和使用叢集和 S EC2 pot 叢集中的容量重新平衡來取代有風險的 Spot 執行個體本使用者指南中的使用容量重新平衡處理 Amazon EC2 Spot 中斷。

監控重新平衡建議訊號

您可以監視重新平衡建議訊號，以便在發出重新平衡建議訊號時，您可以採取在上一部分中指定的動作。重新平衡建議信號可作為傳送至 Amazon EventBridge (以前稱為 Amazon E CloudWatch vents) 的事件提供，以及作為競價型執行個體上的執行個體中繼資料。

監控重新平衡建議訊號：

- 使用 Amazon EventBridge
- 使用執行個體中繼資料

使用 Amazon EventBridge

當針對競價型執行個體發出重新平衡建議訊號時，訊號的事件會傳送至 Amazon。 EventBridge如果 EventBridge 偵測到符合規則中定義的模式的事件模式，則會 EventBridge 叫用規則中指定的一個或多個目標。

以下是重新平衡建議訊號的範例事件。

```
{
    "version": "0",
    "id": "12345678-1234-1234-1234-123456789012",
    "detail-type": "EC2 Instance Rebalance Recommendation",
    "source": "aws.ec2",
    "account": "123456789012",
    "time": "yyyy-mm-ddThh:mm:ssZ",
    "region": "us-east-2",
    "resources": ["arn:aws:ec2:us-east-2:123456789012:instance/i-1234567890abcdef0"],
    "detail": {
        "instance-id": "i-1234567890abcdef0"
    }
}
```

下列欄位會形成規則中定義的事件模式：

`"detail-type": "EC2 Instance Rebalance Recommendation"`

識別事件是重新平衡建議事件

`"source": "aws.ec2"`

標識該事件來自 Amazon EC2

建立 EventBridge 規則

您可以撰寫 EventBridge 規則，並在事件模式符合規則時自動執行哪些動作。

下列範例會建立 EventBridge 規則，以便在 Amazon EC2 每次發出重新平衡建議訊號時傳送電子郵件、文字訊息或行動推播通知。將該信號作為 `EC2 Instance Rebalance Recommendation` 事件發出 ，這觸發了規則定義的動作。

在建立 EventBridge 規則之前，您必須為電子郵件、文字訊息或行動推播通知建立 Amazon SNS 主題。

若要建立重新平衡建議事件的 EventBridge 規則

1. 在打開 Amazon EventBridge 控制台https://console.aws.amazon.com/events/。
2. 選擇建立規則。
3. 針對 Define rule detail (定義規則詳細資訊) 執行下列動作：
   a. 輸入規則的Name (名稱)，或者輸入描述。

      在同一個區域和同一個事件匯流排上，規則不能與另一個規則同名。
   b. 針對 Event bus (事件匯流排) 選擇 default (預設值)。當您帳戶中的 AWS 服務產生事件時，一律會前往您帳戶的預設事件匯流排。
   c. 針對 Rule type (規則類型) 選擇 Rule with an event pattern (具有事件模式的規則)。
   d. 選擇 Next (下一步)。
4. 針對 Build event pattern (建置事件模式) 執行下列動作：
   a. 對於事件來源，請選擇AWS 事件或 EventBridge合作夥伴事件。
   b. 針對此範例的 Event pattern (事件模式)，您需指定下列事件模式，以便與 `EC2 Instance Rebalance Recommendation` 事件相符，然後選擇 Save (儲存)。

      ```
      {
       "source": ["aws.ec2"],
       "detail-type": ["EC2 Instance Rebalance Recommendation"]
      }
      ```

      若要新增事件模式，您可以選擇事件模式表單來使用範本，或選擇自訂模式 (JSON編輯器) 來指定您自己的模式，如下所示：

      i. 若要使用範本建立事件模式，請執行下列動作：
         A. 選擇 Event pattern form (事件模式表單)。
         B. 在 Event source (事件來源) 欄位中，選擇 AWS services (服務)。
         C. 對於「AWS 服務」，請選擇「EC2現貨艦隊
         D. 對於「事件」類型，請選擇「EC2執行處理重新平衡
         E. 若要自訂範本，請選擇 Edit pattern (編輯模式) 並進行變更以與範例事件模式相符。
      ii. (替代) 若要指定自訂事件模式，請執行下列動作：
         A. 選擇自定義模式（JSON編輯器）。

B. 在 Event pattern (事件模式) 方塊中，為此範例新增事件模式。

c. 選擇 Next (下一步)。

5. 針對 Select target(s) (選取目標) 執行下列動作：

   a. 在 Target types (目標類型) 欄位中，選擇 AWS service (服務)。

   b. 針對 [選取目標]，選擇要在事件發生時傳送電子郵件、簡訊或行動推播通知的SNS主題。

   c. 針對 Topic (主題)，請選擇現有的主題。您首先需要使用 Amazon 控制台創建一個 Amazon SNS SNS 主題。如需詳細資訊，請參閱 Amazon 簡單通知服務開發人員指南中的使用 Amazon SNS 進行 application-to-person (A2P) 簡訊。

   d. (選用) 在 Additional settings (其他設定) 下，您可以選擇性地設定其他設定。如需詳細資訊，請參閱 Amazon EventBridge 使用者指南中的建立對事件做出反應的 Amazon EventBridge 規則 (步驟 16)。

   e. 選擇 Next (下一步)。

6. (選用) 針對 Tags (標籤)，您可以選擇性地將一或多個標籤指派給您的規則，然後選擇 Next (下一步)。

7. 針對 Review and create (檢閱和建立) 執行下列動作：

   a. 檢閱規則的詳細資訊，然後視需求進行修改。

   b. 選擇建立規則。

有關更多信息，請參閱 Amazon EventBridge 用戶指南中的 Amazon EventBridge 規則和 Amazon EventBridge 事件模式

使用執行個體中繼資料

執行個體中UTC繼資料類別`events/recommendations/rebalance`提供針對 Spot 執行個體發出重新平衡建議訊號時的大約時間 (英寸)。

我們建議您每 5 秒檢查重新平衡建議訊號，這樣您就不會錯過對重新平衡建議採取行動的機會。

如果 Spot 執行個體收到重新平衡建議，則發出訊號的時間會出現在執行個體中繼資料中。您可以擷取如下所示的發出訊號時間。

cURL

IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-
aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-
data/events/recommendations/rebalance
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/events/recommendations/
rebalance
```

PowerShell

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/events/
recommendations/rebalance
```

以下是範例輸出，表示針對 Spot 執行UTC個體發出重新平衡建議訊號的時間。

```
{"noticeTime": "2020-10-27T08:22:00Z"}
```

如果尚未發出實例的信號，`events/recommendations/rebalance`則不存在，並且在嘗試檢索它時收到 HTTP 404 錯誤。

使用重新平衡建議訊號的服務

Amazon EC2 Auto Scaling、EC2叢集和 Spot Fleet 使用重新平衡建議訊號，在執行中的執行個體收到兩分鐘的 Spot 執行個體中斷通知之前，主動使用新的 Spot 執行個體擴充叢集，讓您輕鬆維持工作負載可用性。您可以讓這些服務主動地監控並回應影響 競價型執行個體 的可用性的變更 如需詳細資訊，請參閱下列內容：

- 在 Amazon EC2Auto Scaling 使用者指南中，使用容量重新平衡來處理 Amazon EC2 Spot 中斷情況
- 使用叢集和 S EC2 pot 叢集中的容量重新平衡來取代有風險的 Spot 執行個體本使用者指南中的 EC2「艦隊與 Spot 叢集」主題中

Spot 配置分數

Spot 放置分數功能可以根據您的競價型容量需求推薦區 AWS 域或可用區域。Spot 容量會波動，而且您無法確定您始終都能取得所需的容量。Spot 配置分數表示 Spot 請求在區域或可用區域中成功的可能性。

> **Note**
>
> Spot 配置分數不會就可用容量或中斷風險提供任何保證。Spot 配置分數僅作為建議提供。

使用案例

您可以針對下列情況使用 Spot 配置分數功能：

- 視需要在不同區域中重新放置並擴展 Spot 運算容量，以回應目前區域中增加的容量需求或降低的可用容量。
- 識別執行單一可用區域工作負載的最佳可用區域。
- 模擬未來的 Spot 容量需求，以便您可以挑選最佳區域來擴充 Spot 型工作負載。
- 尋找執行個體類型的最佳組合以滿足您的 Spot 容量需求。

目錄

- 限制
- 成本
- Spot 配置分數的運作方式
- Spot 放置分數所需的權限
- 計算現貨放置分數

限制

- 目標容量限制 – 您的 Spot 配置分數目標容量限制是基於最近的 Spot 用量，同時考量潛在的用量成長。如果沒有最近的 Spot 用量，我們會為您提供符合 Spot 請求限制的低預設限制。
- 請求組態限制 – 如果我們偵測到未與 Spot 配置分數功能之預期用途相關聯的模式，可以在 24 小時期間內限制新請求組態的數目。如果達到限制，您可以重試已使用的請求組態，但無法指定新的請求組態，直到下一個 24 小時期間。
- 執行個體類型的最少數量 — 如果您指定執行個體類型，則必須至少指定三種不同的執行個體類型，否則 Amazon EC2 將傳回較低的 Spot 放置分數。同樣地，如果您指定執行個體屬性，它們必須解析為至少三種不同的執行個體類型。如果執行個體類型具有不同的名稱，則會將其視為不同。例如，m5.8xlarge、m5a.8xlarge 和 m5.12xlarge 都會視為不同。

成本

使用 Spot 配置分數功能無需額外費用。

Spot 配置分數的運作方式

使用競價型放置分數功能時，首先要為 Spot 執行傳回前 10 個區域。每個區域或可用區域的分數範圍從 1 到 10，其中 10 表示您的 前 Spot 請求很可能成功，1 表示您的 Spot 請求不可能成功。

若要使用 Spot 配置分數功能，請遵循下列步驟：

- 步驟 1：指定您的 Spot 需求
- 步驟 2：篩選 Spot 配置分數回應
- 步驟 3：檢閱建議
- 步驟 4：使用建議

步驟 1：指定您的 Spot 需求

首先，您可以指定所需的目標 Spot 容量和運算需求，如下所示：

1. 指定目標 Spot 容量，並選擇性指定目標容量單位。

   您可以根據執行個體數量或 vCPUs MiB 中的記憶體容量來指定所需的目標 Spot 容量。若要以記憶體數目 vCPUs 或容量指定目標容量，您必須將目標容量單位指定為`vcpu`或`memory-mib`。否則，其會預設為執行個體的數量。

   透過根據記憶體數量 vCPUs 或記憶體容量來指定目標容量，您可以在計算總容量時使用這些單位。例如，如果您想要混合使用不同大小的執行個體，可以將目標容量指定為的總數vCPUs。然後，Spot 放置分數功能會依據要求中的每個執行個體類型進行考量vCPUs，並在總計目標容量時計算的執行個體總數， vCPUs 而非執行個體總數。

   例如，假設您指定的總目標容量為 30vCPUs，而您的執行個體類型清單包含 c5.xlarge (4vCPUs)、m5.2xlarge (8) 和 r5.large (2vCPUs)。vCPUs要實現總共 30 個vCPUs，您可以獲得 2 厘米超大（2* 4vCPUs），2 立方米 5.2 倍大（2* 8）和 3 R5 大（3* 2）的混合。vCPUs vCPUs
2. 指定執行個體類型或執行個體屬性。

   您可以指定要使用的執行個體類型，也可以指定運算需求所需的執行個體屬性，然後讓 Amazon EC2 識別具有這些屬性的執行個體類型。這就是所謂的屬性型執行個體類型選項。

   您無法在相同的 Spot 配置分數請求中同時指定執行個體類型和執行個體屬性。

如果指定執行個體類型，則必須至少指定三種不同的執行個體類型，否則 Amazon EC2 將傳回較低的 Spot 放置分數。同樣地，如果您指定執行個體屬性，它們必須解析為至少三種不同的執行個體類型。

如需指定 Spot 需求的不同方式範例，請參閱 範例組態。

步驟 2：篩選 Spot 配置分數回應

Amazon EC2 會計算每個區域或可用區域的 Spot 放置分數，並傳回 Spot 請求可能成功的前 10 個區域或前 10 個可用區域。預設值是傳回得分區域的清單。如果您計劃將所有 Spot 容量啟動至單一可用區域，請求得分可用區域的清單很有用。

您可以指定區域篩選條件來縮小將在回應中傳回的區域。

您可以結合區域篩選條件和得分可用區域的請求。如此一來，得分可用區域僅限於您已篩選的區域。若要找出區域中得分最高的可用區域，請僅指定該區域，然後回應將傳回該區域中所有可用區域的得分清單。

步驟 3：檢閱建議

每個區域或可用區域的 Spot 配置分數是根據目標容量、執行個體類型的組成、歷史和目前 Spot 使用趨勢，以及請求的時間來計算。由於 Spot 容量持續波動，因此在不同時間計算時，相同的 Spot 配置分數請求可能會產生不同的分數。

區域和可用區域的分數範圍是從 1 到 10。分數 10 表示您的 Spot 請求很有可能 (但不保證) 成功。分數 1 表示您的 Spot 請求完全不可能成功。對於不同的區域或可用區域，可能會傳回相同的分數。

如果傳回低分數，您可以編輯運算需求並重新計算分數。您也可以在一天中的不同時間，針對相同的運算需求，請求 Spot 配置分數建議。

步驟 4：使用建議

僅在 Spot 請求的組態與 Spot 配置分數組態 (目標容量、目標容量單位，以及執行個體類型或執行個體屬性) 完全相同，且該請求設定為使用 `capacity-optimized` 配置策略時，Spot 配置分數才相關。否則，取得可用 Spot 容量的可能性將不符合分數。

雖然 Spot 配置分數充當指導方針，而且沒有任何分數保證您的 Spot 請求會完全或部分實現，但您可以使用下列資訊來取得最佳結果：

- 使用相同的組態 — 只有當 Auto Scaling 群組、EC2叢集或 Spot 叢集中的 Spot 請求組態 (目標容量、目標容量單位以及執行個體類型或執行個體屬性) 與您為取得 Spot 放置分數所輸入的相同時，Spot 放置分數才有關聯。

  如果您在 Spot 放置分數請求中使用了以屬性為基礎的執行個體類型選擇，則可以使用以屬性為基礎的執行個體類型選擇來設定 Auto Scaling 群組、EC2叢集或 Spot 叢集。如需詳細資訊，請參閱建立 Auto Scaling 群組，其中包含所使用執行個體類型和的一組需求指定EC2叢集或 Spot 叢集的執行個體類型選取屬性。

  > **Note**
  >
  > 如果您根據記憶體數量 vCPUs 或數量指定目標容量，並且在競價型放置分數設定中指定了執行個體類型，請注意，您目前無法在 Auto Scaling 群組、EC2叢集或 Spot 叢集中建立此組態。反之，您必須使用 `WeightedCapacity` 參數，手動設定執行個體加權。

- 使用 **`capacity-optimized`** 分配策略 – 任何分數都假設您的機群請求將設定為使用所有的可用區域 (用於跨區域請求容量) 或單一可用區域 (如果在某個可用區域請求容量) 和 `capacity-optimized` Spot 分配策略，以便您可以成功請求 Spot 容量。如果您使用其他配置策略 (例如 `lowest-price`)，取得可用 Spot 容量的可能性將不符合分數。
- 立即對分數採取行動 – Spot 配置分數建議會反映請求時的可用 Spot 容量，並且由於 Spot 容量波動，因此在不同時間計算時，相同的組態可能會產生不同的分數。雖然分數 10 表示您的 Spot 容量請求很可能 (但不保證) 成功，但為了取得最佳結果，我們建議您立即對分數採取行動。我們也建議您在每次嘗試容量請求時取得全新的分數。

Spot 放置分數所需的權限

根據預設，IAM身分識別 (使用者、角色或群組) 沒有使用權限the section called “Spot 配置分數”。若要允許IAM身分使用 Spot 放置分數，您必須建立授與使用`ec2:GetSpotPlacementScores`EC2API動作之權限的IAM原則。然後，您可以將原則附加到需要此權限的IAM身分識別。

以下是授與使用動作之權限的範例IAM原`ec2:GetSpotPlacementScores`EC2API則。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ec2:GetSpotPlacementScores",
            "Resource": "*"
```

```
        }
    ]
}
```

若要取得有關編輯IAM策略的資訊，請參閱《IAM使用指南》中的編輯IAM策略。

若要提供存取權，請新增權限至您的使用者、群組或角色：

- 使用者和群組位於 AWS IAM Identity Center：

  建立權限合集。請按照 AWS IAM Identity Center 使用者指南 中的 建立權限合集 說明進行操作。
- IAM透過身分識別提供者管理的使用者：

  建立聯合身分的角色。請遵循《使用指南》中的〈為第三方身分識別提供IAM者 (同盟) 建立角色〉中的指示進行。
- IAM使用者：
  - 建立您的使用者可擔任的角色。請按照《用戶指南》中的「為IAM用戶創建角色」中的IAM說明進行操作。
  - (不建議) 將政策直接附加至使用者，或將使用者新增至使用者群組。遵循《使用者指南》中的「向使用者 (主控台) 新增權限」IAM 中的指示進行。

計算現貨放置分數

您可以根據目標容量和運算需求計算 Spot 放置分數。如需詳細資訊，請參閱the section called “Spot 配置分數的運作方式”。

所需的 許可

請確定您擁有必要的權限。如需詳細資訊，請參閱the section called “所需的許可”。

選項

- 使用實例屬性計算
- 使用實例類型計算
- 使用計算 AWS CLI

使用實例屬性計算

執行個體屬性來計算 Spot 配置分數

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選擇 Spot placement score (Spot 配置分數)。
4. 選擇 Enter requirements (輸入需求)。
5. 針對目標容量，根據執行個體數量或vCPUs記憶體容量 (MiB) 輸入所需的容量。
6. 對於執行個體類型需求，若要指定運算需求並讓 Amazon EC2 識別符合這些需求的最佳執行個體類型，請選擇指定符合您運算需求的執行個體屬性。
7. 對於 vCPUs，輸入所需的最小和最大數目vCPUs。若要指定無限制，請選取 No minimum (無下限)、No maximum (無上限)，或兩者。
8. 對於 Memory (GiB) (記憶體 (GiB))，輸入所需記憶體數量的下限和上限。若要指定無限制，請選取 No minimum (無下限)、No maximum (無上限)，或兩者。
9. 針對CPU架構，請選取所需的執行個體架構。
10. (選用) 對於 Additional instance attributes (其他執行個體屬性)，您可以選擇性地指定一或多個屬性，以更詳細地表達您的運算需求。每個額外屬性都會將進一步的限制新增至您的請求。您可以省略額外屬性；省略時，就會使用預設值。如需每個屬性及其預設值的說明，請參閱 Amazon 命EC2令列參考get-spot-placement-scores中的。
11. (選用) 若要檢視具有所指定屬性的執行個體類型，請展開 Preview matching instance types (預覽相符的執行個體類型)。若要排除執行個體類型，使其不在配置評估中使用，請選取執行個體，然後選取 Exclude selected instance types (排除選取的執行個體類型)。
12. 選擇 Load placement scores (載入配置分數)，然後檢閱結果。
13. (選用) 若要顯示特定區域的 Spot 配置分數，請針對 Regions to evaluate (要評估的區域)，選取要評估的區域，然後選取 Calculate placement scores (計算配置分數)。
14. (選用) 若要顯示所顯示區域中可用區域的 Spot 配置分數，請選取 Provide placement scores per Availability Zone (提供每個可用區域的配置分數) 核取方塊。如果您想要將所有 Spot 容量啟動至單一可用區域，得分可用區域的清單很有用。
15. (選用) 若要編輯您的運算需求並取得新的配置分數，請選擇 Edit (編輯)、進行必要的調整，然後選擇 Calculate placement scores (計算配置分數)。

使用實例類型計算

指定執行個體類型來計算 Spot 配置分數

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選擇 Spot placement score (Spot 配置分數)。
4. 選擇 Enter requirements (輸入需求)。
5. 針對目標容量，根據執行個體數量或vCPUs記憶體容量 (MiB) 輸入所需的容量。
6. 對於 Instance type requirements (執行個體類型需求)，若要指定要使用的執行個體類型，請選取 Manually select instance types (手動選取執行個體類型)。
7. 選取 Select instance types (選取執行個體類型)、選取要使用的執行個體類型，然後選取 Select (選取)。若要快速尋找執行個體類型，您可以使用篩選條件列，依不同屬性篩選執行個體類型。
8. 選擇 Load placement scores (載入配置分數)，然後檢閱結果。
9. (選用) 若要顯示特定區域的 Spot 配置分數，請針對 Regions to evaluate (要評估的區域)，選取要評估的區域，然後選取 Calculate placement scores (計算配置分數)。
10. (選用) 若要顯示所顯示區域中可用區域的 Spot 配置分數，請選取 Provide placement scores per Availability Zone (提供每個可用區域的配置分數) 核取方塊。如果您想要將所有 Spot 容量啟動至單一可用區域，得分可用區域的清單很有用。
11. (選用) 若要編輯執行個體類型的清單並取得新的配置分數，請選擇 Edit (編輯)、進行必要的調整，然後選擇 Calculate placement scores (計算配置分數)。

使用計算 AWS CLI

計算 Spot 配置分數

1. (選擇性) 若要產生可為 Spot 放置分數組態指定的所有可能參數，請使用指get-spot-placement-scores令和`--generate-cli-skeleton`參數。

```
aws ec2 get-spot-placement-scores \
    --region us-east-1 \
    --generate-cli-skeleton
```

下列為範例輸出。

```
{
```

```
    "InstanceTypes": [
        ""
    ],
    "TargetCapacity": 0,
    "TargetCapacityUnitType": "vcpu",
    "SingleAvailabilityZone": true,
    "RegionNames": [
        ""
    ],
    "InstanceRequirementsWithMetadata": {
        "ArchitectureTypes": [
            "x86_64_mac"
        ],
        "VirtualizationTypes": [
            "hvm"
        ],
        "InstanceRequirements": {
            "VCpuCount": {
                "Min": 0,
                "Max": 0
            },
            "MemoryMiB": {
                "Min": 0,
                "Max": 0
            },
            "CpuManufacturers": [
                "amd"
            ],
            "MemoryGiBPerVCpu": {
                "Min": 0.0,
                "Max": 0.0
            },
            "ExcludedInstanceTypes": [
                ""
            ],
            "InstanceGenerations": [
                "previous"
            ],
            "SpotMaxPricePercentageOverLowestPrice": 0,
            "OnDemandMaxPricePercentageOverLowestPrice": 0,
            "BareMetal": "excluded",
            "BurstablePerformance": "excluded",
            "RequireHibernateSupport": true,
            "NetworkInterfaceCount": {
```

```
                "Min": 0,
                "Max": 0
            },
            "LocalStorage": "included",
            "LocalStorageTypes": [
                "hdd"
            ],
            "TotalLocalStorageGB": {
                "Min": 0.0,
                "Max": 0.0
            },
            "BaselineEbsBandwidthMbps": {
                "Min": 0,
                "Max": 0
            },
            "AcceleratorTypes": [
                "fpga"
            ],
            "AcceleratorCount": {
                "Min": 0,
                "Max": 0
            },
            "AcceleratorManufacturers": [
                "amd"
            ],
            "AcceleratorNames": [
                "vu9p"
            ],
            "AcceleratorTotalMemoryMiB": {
                "Min": 0,
                "Max": 0
            }
        }
    },
    "DryRun": true,
    "MaxResults": 0,
    "NextToken": ""
}
```

2. 使用上一個步驟的輸出建立JSON組態檔案，並依照下列方式進行配置：

   a. 對於`TargetCapacity`，根據執行個體數量或vCPUs記憶體容量 (MiB) 輸入所需的 Spot 容量。

b. 對於 `TargetCapacityUnitType`，輸入目標容量的單位。如果省略此參數，則其會預設為 `units`。

   有效值：`units` (其會轉換為執行個體的數量) | `vcpu` | `memory-mib`

c. 對於 `SingleAvailabilityZone`，指定 `true`，以取得傳回得分可用區域清單的回應。如果您想要將所有 Spot 容量啟動至單一可用區域，得分可用區域的清單很有用。如果省略此參數，則其會預設為 `false`，並且回應會傳回得分區域的清單。

d. (選用) 對於 `RegionNames`，請指定要用作篩選條件的區域。您必須指定區域代碼，例如 `us-east-1`。

   搭配區域篩選條件，回應只會傳回您指定的區域。如果您針對 `SingleAvailabilityZone` 指定了 `true`，則回應只會傳回所指定區域中的可用區域。

e. 您可以包含 `InstanceTypes` 或 `InstanceRequirements`，但兩者不能同時位於相同的組態中。

   在您的組態中指定下列其中一JSON項：

   - 若要指定執行個體類型的清單，請在 `InstanceTypes` 參數中指定執行個體類型。指定至少三種不同的執行個體類型。如果您只指定一個或兩個執行個體類型，Spot 配置分數會傳回低分數。如需執行個體類型的清單，請參閱 Amazon EC2 執行個體類型。
   - 若要指定執行個體屬性，以EC2便 Amazon 識別符合這些屬性的執行個體類型，請指定位於`InstanceRequirements`結構中的屬性。

     您必須提供 `VCpuCount`、`MemoryMiB` 和 `CpuManufacturers` 的值。您可以省略其他屬性；省略時，就會使用預設值。如需每個屬性及其預設值的說明，請參閱 Amazon 命EC2令列參考get-spot-placement-scores中的。

   如需組態範例，請參閱 範例組態。

3. 若要取得您在JSON檔案中指定需求的 Spot 放置分數，請使用指get-spot-placement-scores令，並使用`--cli-input-json`參數指定JSON檔案的名稱和路徑。

```
aws ec2 get-spot-placement-scores \
    --region us-east-1 \
    --cli-input-json file://file_name.json
```

如果設定`SingleAvailabilityZone`為`false`或省略，則輸出範例 (如果省略，則預設為`false`) — 傳回計清單。

```
"SpotPlacementScores": [
    {
        "Region": "us-east-1",
        "Score": 7
    },
    {
        "Region": "us-west-1",
        "Score": 5
    },
   ...
```

如果設定`SingleAvailabilityZone`為，則輸出範例 `true` — 傳回可用區域的計分清單。

```
"SpotPlacementScores": [
    {
        "Region": "us-east-1",
        "AvailabilityZoneId": "use1-az1",
        "Score": 8
    },
    {
        "Region": "us-east-1",
        "AvailabilityZoneId": "usw2-az3",
        "Score": 6
    },
   ...
```

範例組態

使用時 AWS CLI，您可以使用下列範例組態。

範例組態

- 範例：指定執行個體類型和目標容量
- 範例：根據記憶體指定執行個體類型和目標容量
- 範例：為屬性型執行個體類型選取指定屬性
- 範例：為屬性型執行個體類型選取指定屬性，並傳回可用區域的得分清單

範例：指定執行個體類型和目標容量

下列範例組態指定三種不同的執行個體類型，以及有 500 個 Spot 執行個體的目標 Spot 容量。

```
{
    "InstanceTypes": [
        "m5.4xlarge",
        "r5.2xlarge",
        "m4.4xlarge"
    ],
    "TargetCapacity": 500
}
```

範例：根據記憶體指定執行個體類型和目標容量

下列範例組態指定了三種不同的執行個體類型，以及有 500,000 MiB 記憶體的目標 Spot 容量，其中要啟動的 Spot 執行個體數目必須提供總計 500,000 MiB 的記憶體。

```
{
    "InstanceTypes": [
        "m5.4xlarge",
        "r5.2xlarge",
        "m4.4xlarge"
    ],
    "TargetCapacity": 500000,
    "TargetCapacityUnitType": "memory-mib"
}
```

範例：為屬性型執行個體類型選取指定屬性

下列範例組態是針對屬性型執行個體類型選取設定的，其後是範例組態的文字說明。

```
{
    "TargetCapacity": 5000,
    "TargetCapacityUnitType": "vcpu",
    "InstanceRequirementsWithMetadata": {
        "ArchitectureTypes": ["arm64"],
        "VirtualizationTypes": ["hvm"],
        "InstanceRequirements": {
            "VCpuCount": {
                "Min": 1,
                "Max": 12
```

```
            },
            "MemoryMiB": {
                "Min": 512
            }
        }
    }
}
```

InstanceRequirementsWithMetadata

若要使用屬性型執行個體類型選取，您必須在組態中包含 `InstanceRequirementsWithMetadata` 結構，並為 Spot 執行個體指定所需的屬性。

在上述範例中，指定了下列必要的執行個體屬性：

- `ArchitectureTypes` – 執行個體類型的架構類型必須是 `arm64`。
- `VirtualizationTypes` – 執行個體類型的虛擬化類型必須是 `hvm`。
- `VCpuCount`— 執行個體類型必須至少為 1 且最多 12 個vCPUs。
- `MemoryMiB` – 執行個體類型必須具有最少 512 MiB 的記憶體。省略 `Max` 參數，表示沒有最大限制。

請注意，有幾個您可以指定的其他選用屬性。若要取得屬性清單，請參閱《AWS CLI 指令參考》get-spot-placement-scores中的〈〉。

TargetCapacityUnitType

`TargetCapacityUnitType` 參數會指定目標容量的單位。在此範例中，目標容量為`5000`且目標容量單位類型為`vcpu`，它們共同指定所需的目標容量 5000vCPUs，其中要啟動的 Spot 執行個體數目必須提供總計 5000 個vCPUs。

範例：為屬性型執行個體類型選取指定屬性，並傳回可用區域的得分清單

下列範例組態是針對屬性型執行個體類型選取而設定的。透過指定 `"SingleAvailabilityZone": true`，回應將傳回得分可用區域的清單。

```
{
    "TargetCapacity": 1000,
    "TargetCapacityUnitType": "vcpu",
    "SingleAvailabilityZone": true,
    "InstanceRequirementsWithMetadata": {
```

```
        "ArchitectureTypes": ["arm64"],
        "VirtualizationTypes": ["hvm"],
        "InstanceRequirements": {
            "VCpuCount": {
                "Min": 1,
                "Max": 12
            },
            "MemoryMiB": {
                "Min": 512
            }
        }
    }
}
```

使用競價型執行個體資料饋送追蹤 Spot 執行個體成本

為了協助您瞭解 Spot 執行個體的費用，Amazon EC2 提供資料饋送說明您的競價型執行個體用量和定價。當您訂閱資料摘要時，此資料摘要會傳送到您所指定的 Amazon S3 儲存貯體。

資料摘要檔案通常每小時送達一次您的儲存貯體。如果在某個小時內沒有執行 Spot 執行個體，您不會收到那個小時的資料摘要檔案。

Spot 執行個體的每小時使用量通常涵蓋在單一資料檔案中。這些檔案在傳送到您的儲存貯體之前，會先經過壓縮 (gzip)。Amazon EC2 可以在特定小時內寫入多個檔案，其中檔案很大 (例如，壓縮前一小時的檔案內容超過 50 MB 時)。

> **Note**
>
> 每個執行個體只能建立一個 Spot 執行個體資料饋送 AWS 帳戶。

除中國 (北京)、中國 (寧夏)、 AWS GovCloud (美國) 和預設停用的 AWS 區域以外的所有區域均支援 Spot 執行個體資料饋送。

目錄

- 資料摘要檔案名稱與格式
- Amazon S3 儲存貯體需求
- 訂閱 Spot 執行個體資料摘要
- 檢視資料饋送中的資料
- 刪除 Spot 執行個體資料摘要

資料摘要檔案名稱與格式

競價型執行個體資料饋送檔案名稱使用下列格式 (日期和小時為單位UTC)：

```
bucket-name.s3.amazonaws.com/optional-prefix/aws-account-id.YYYY-MM-DD-HH.n.unique-id.gz
```

例如，如果儲存貯體的名稱為 **amzn-s3-demo-bucket**、前綴字首為 **my-prefix**，則檔案名稱會類似於下列名稱：

```
amzn-s3-demo-bucket.s3.amazonaws.com/my-prefix/111122223333.2023-12-09-07.001.b959dbc6.gz
```

如需儲存貯體名稱的詳細資訊，請參閱《Amazon S3 使用者指南》中的儲存貯體命名規則。

Spot 執行個體資料摘要檔案以製表符 (tab) 分隔。資料檔案中的每一行皆對應至一個執行個體小時，而且包含下表中所列出的欄位。

欄位	描述
Timestamp	時間戳記，用來判定使用此執行個體的收費價格。
UsageType	計費的使用類型和執行個體類型。對於 m1.small 競價型執行個體，此欄位設定為 SpotUsage 。如果是其他所有的執行個體類型，此欄位會設定為 SpotUsage: {instance-type}。例如：SpotUsage:c1.medium 。
Operation	收費的產品。若是 Linux 競價型執行個體，此欄位會設定為 RunInstances 。若是 Windows 競價型執行個體，此欄位會設定為 RunInstances:0002 。Spot 的使用會根據可用區域分組。
InstanceID	產生此執行個體用量的 Spot 執行個體的 ID。
MyBidID	產生此執行個體用量的 Spot 執行個體請求的 ID。
MyMaxPrice	針對此 Spot 請求所指定的最高價格。

欄位	描述
MarketPrice	Timestamp 欄位中所指定時間點當時的 Spot 價格。
Charge	針對此執行個體用量的計費價格。
Version	資料饋送版本。可用的版本為 1.0。

Amazon S3 儲存貯體需求

訂閱資料摘要時，您必須指定用來存放資料摘要檔案的 Amazon S3 儲存貯體。

在選擇資料摘要的 Amazon S3 儲存貯體之前，請考量下列事項：

- 您必須具有儲存貯體的 FULL_CONTROL 許可。如果您是儲存貯體擁有者，預設會具有此項權限。否則，值區擁有者必須授予您 AWS 帳戶 此權限。
- 當您訂閱資料饋送時，系統會使用這些權限來更新值區，ACL以授與資 AWS 料饋送帳戶FULL_CONTROL權限。資 AWS 料饋送帳戶會將資料饋送檔案寫入值區。如果您的帳戶沒有所需的權限，則資料摘要檔案無法寫入儲存貯體。如需詳細資訊，請參閱 Amazon 日誌使用者指南中的傳送至 Amazon S3 的 CloudWatch 日誌。

  如果您更新ACL並移除資 AWS 料饋送帳戶的權限，則無法將資料饋送檔案寫入值區。您必須重新訂閱資料摘要，以接收資料摘要檔案。
- 每個資料饋送檔案都有自己的 ACL (獨立ACL於值區的)。儲存貯體擁有者具有對資料檔案的 FULL_CONTROL 權限。資 AWS 料饋送帳戶具有讀取和寫入權限。
- 如果您刪除資料饋送訂閱，Amazon EC2 不會移除值區或資料檔案上 AWS 資料饋送帳戶的讀取和寫入許可。您必須自行移除這些權限。
- 如果您使用存放在 AWS Key Management Service (SSE-KMS) 中的 AWS KMS 金鑰使用伺服器端加密 Amazon S3 儲存貯體，則必須使用客戶受管金鑰。如需詳細資訊，請參閱 Amazon S3 儲存貯體伺服器端加密的 Amazon CloudWatch 日誌使用者指南。

訂閱 Spot 執行個體資料摘要

若要訂閱資料饋送，請使用create-spot-datafeed-subscription AWS CLI 指令。

```
aws ec2 create-spot-datafeed-subscription \
```

```
    --bucket amzn-s3-demo-bucket \
    [--prefix my-prefix]
```

以下是示例輸出。

```
{
    "SpotDatafeedSubscription": {
        "OwnerId": "111122223333",
        "Bucket": "amzn-s3-demo-bucket",
        "Prefix": "my-prefix",
        "State": "Active"
    }
}
```

如果您收到值區沒有足夠權限的錯誤訊息，請參閱下列文章以取得疑難排解資訊：疑難排解 Spot 執行個體的資料饋送。

檢視資料饋送中的資料

在中 AWS Management Console，開啟 AWS CloudShell。使用下列 s3 sync 命令，從 S3 儲存貯體取得資料饋送的 .gz 檔案，並將它們存放在您指定的資料夾中。

```
aws s3 sync s3://amzn-s3-demo-bucket ./data-feed
```

若要顯示 .gz 檔案的內容，請變更至您存放 S3 儲存貯體內容的資料夾。

```
cd data-feed
```

使用 ls 命令檢視檔案的名稱。使用具有檔案名稱的 zcat 命令，顯示壓縮檔案的內容。下列是範例命令。

```
zcat  111122223333.2023-12-09-07.001.b959dbc6.gz
```

下列為範例輸出。

```
#Version: 1.0
#Fields: Timestamp UsageType Operation InstanceID MyBidID MyMaxPrice MarketPrice Charge
 Version
2023-12-09 07:13:47 UTC USE2-SpotUsage:c7a.medium       RunInstances:SV050
 i-0c3e0c0b046e050df     sir-pwq6nmfp    0.0510000000 USD        0.0142000000 USD
  0.0142000000 USD        1
```

刪除 Spot 執行個體資料摘要

若要刪除資料饋送，請使用delete-spot-datafeed-subscription AWS CLI 指令。

```
aws ec2 delete-spot-datafeed-subscription
```

Spot 執行個體請求的服務連結角色

Amazon EC2 使用服務連結角色來取得代表您呼叫其他 AWS 服務所需的許可。服務連結角色是一種獨特的IAM角色類型，直接連結至. AWS 服務服務連結角色提供委派權限的安全方式， AWS 服務 因為只有連結的服務可以擔任服務連結的角色。如需詳細資訊，請參閱IAM使用指南中的服務連結角色。

Amazon EC2 使用名為AWSServiceRoleForEC2Spot的服務連結角色代表您啟動和管理 Spot 執行個體。

AWSServiceRoleForEC2Spot 授予的許可

Amazon EC2 使AWSServiceRoleForEC2Spot用完成以下操作：

- `ec2:DescribeInstances` - 描述 Spot 執行個體
- `ec2:StopInstances` - 停止 Spot 執行個體
- `ec2:StartInstances` - 啟動 Spot 執行個體

建立服務連結角色

在大部分情況下，您不需要手動建立服務連結角色。當您第一次使用主控台請求 Spot 執行個體時，Amazon EC2 會建立AWSServiceRoleForEC2Spot服務連結角色。

如果您在 2017 年 10 月之前有一個有效的競價型執行個體請求，當 Amazon EC2 開始支援此服務連結角色時，Amazon 就會在您的 AWS 帳戶中EC2建立該AWSServiceRoleForEC2Spot角色。如需詳細資訊，請參閱《IAM使用者指南》中的「我的帳戶」中的「新角色」。

如果您使用 AWS CLI 或API要求競價型執行個體，則必須首先確保此角色存在。

使用主控台建立 AWSServiceRoleForEC2Spot

1. 在開啟IAM主控台https://console.aws.amazon.com/iam/。
2. 在導覽窗格中，選擇 Roles (角色)。

3. 選擇 Create Role (建立角色)。
4. 在 [選取信任實體的類型] 頁面上 EC2，選擇 EC2--競價型執行個體，下一個：權限。
5. 在下一頁選擇 Next:Review (下一步：檢閱)。
6. 在檢閱頁面上選擇建立角色。

若要AWSServiceRoleForEC2Spot使用建立 AWS CLI

使用 create-service-linked-role 命令，如下所示。

```
aws iam create-service-linked-role --aws-service-name spot.amazonaws.com
```

如果您不再需要使用 Spot 執行個體，建議您刪除該AWSServiceRoleForEC2Spot角色。從您的帳戶中刪除此角色後，如果您請求 Spot 執行個體，Amazon EC2 將再次建立角色。

授予客戶受管金鑰的存取權，以便搭配加密AMIs和EBS快照使用

如果您為競價型執行個體指定加密AMI或加密的 Amazon EBS 快照，並使用客戶受管金鑰進行加密，則必須授與該AWSServiceRoleForEC2Spot角色使用客戶受管金鑰的權限，以便 Amazon EC2 可以代表您啟動 Spot 執行個體。若要執行此動作，您必須將授予新增至客戶受管金鑰，如下列程序所示。

提供許可時，授與為金鑰政策的替代方案。如需詳細資訊，請參閱 開發人員指南中的使用授予 AWS KMS和在 AWS Key Management Service 中使用金鑰政策。

若要授予 AWSServiceRoleForEC2Spot 角色許可以使用客戶受管金鑰

- 使用 create-grant 命令可將授權新增至客戶管理的金鑰，並指定授與執行授權之作業的主體 (AWSServiceRoleForEC2Spot服務連結角色)。客戶管理金鑰由`key-id`參數和客戶管理金鑰指定。ARN主體是由`grantee-principal`參數和AWSServiceRoleForEC2Spot服務連結角色所指定。ARN

  ```
  aws kms create-grant \
      --region us-east-1 \
      --key-id arn:aws:kms:us-
  east-1:444455556666:key/1234abcd-12ab-34cd-56ef-1234567890ab \
      --grantee-principal arn:aws:iam::111122223333:role/aws-service-role/
  spot.amazonaws.com/AWSServiceRoleForEC2Spot \
      --operations "Decrypt" "Encrypt" "GenerateDataKey"
   "GenerateDataKeyWithoutPlaintext" "CreateGrant" "DescribeKey" "ReEncryptFrom"
   "ReEncryptTo"
  ```

Spot 執行個體配額

每個區域每個 AWS 帳戶 的執行中 Spot 執行個體請求和待定 Spot 執行個體請求有數量配額。待處理的 Spot 執行個體請求完成後，請求將不再計入配額中，因為執行中的執行個體會計入配額中。

Spot 執行個體配額是根據執行中的 Spot 執行個體正在使用的虛擬中央處理器單元數量 (vCPUs) 來管理，或將在未完成的 Spot 執行個體請求履行時使用的虛擬中央處理單元數量。如果您終止 Spot 執行個體但未取消競價型執行個體請求，則請求會計入您的 Spot 執行個體 v CPU 配額，直到 Amazon EC2 偵測到 Spot 執行個體終止並關閉請求為止。

我們為 Spot 執行個體提供下列配額類型。

名稱	預設	可調整
所有 DL Spot 執行個體請求	0	是
所有 F Spot 執行個體請求	0	是
所有 G 和 VT Spot 執行個體請求	0	是
所有 Inf Spot 執行個體請求	0	是
所有 P4、P3 和 P2 競價型執行個體請求	0	是
所有 P5 競價型執行個體請求	0	是
所有標準 (A, C, D, H, I, M, R, T, Z) Spot 執行個體請求	5	是
所有 Trn Spot 執行個體請求	0	是
所有 X Spot 執行個體請求	0	是

即使 Amazon EC2 會根據您的用量自動增加 Spot 執行個體配額，但您也可以視需要申請增加配額。例如，如果您打算啟動的 Spot 執行個體超過目前配額允許的範圍，您可以請求提高配額。如果您送出 Spot 執行個體請求並收到錯誤訊息 `Max spot instance count exceeded`，您可以要求增加請求的配額。若要請求增加配額，可以使用 Amazon EC2 服務配額 中所述的 Service Quotas 主控台。

您可以啟動任意執行個體類型的組合，這些類型組合可滿足不斷變化的應用程式需求。例如，如果所有標準競價型執行個體請求配額為 256 個vCPUs，您可以請求 32 個`m5.2xlarge`競價型執行個體 (32 x 8vCPUs) 或 16 個`c5.4xlarge`競價型執行個體 (16 x 16vCPUs)。

透過 Amazon CloudWatch 指標整合，您可以根據配額監控EC2使用情況。您也可以設定警示，在接近配額時提出警告。如需詳細資訊，請參閱 Ser vice Quotas 使用指南中的 Service Quotas 和 Amazon CloudWatch 警示，以。

Amazon EC2 專用主機

Amazon EC2 專用主機是完全專供您使用的實體伺服器。 您可以選擇與其他 AWS 帳戶共用執行個體容量。如需詳細資訊，請參閱跨帳戶 Amazon EC2 專用主機共享。

專用主機提供執行個體放置的可見度和控制權，並支援主機相似性。這表示您可以在特定主機上啟動和執行執行個體，而且可以確保執行個體僅在特定主機上執行。如需詳細資訊，請參閱Amazon EC2 專用主機自動放置和主機親和性。

專用主機提供全面的自攜授權 (BYOL) 支援。根據您的授權條款，它們可讓您使用現有的每個插槽、每個核心或虛擬機器的軟體授權，SQL包括 Windows 伺服器、伺服器、Linux 企業伺服器、Red Hat Enterprise Linux 或其他綁定至VMs通訊端或實體核心的軟體授權。SUSE

如果您要求執行個體在專用硬體上執行，但不需要掌握或控制執行個體放置，而且不需要使用每個插槽或每核心的軟體授權，則可以考慮改用專用執行個體。專用執行個體和專用主機都可以在專用實體伺服器上啟動 Amazon EC2 執行個體。專用執行個體和專用執行個體上的執行個體之間沒有效能、安全或實體差異。但是，它們之間存在一些關鍵差異。下表特別列出了專用執行個體與專用執行個體之間的主要差異：

	Dedicated Host	Dedicated Instance
專屬實體伺服器	實體伺服器具有完全專供您使用的執行個體容量。	專用於單一客戶帳戶的實體伺服器。
實例容量共享	可與其他帳戶共用執行個體容量。	不支援
計費	按每台主機計費	按每個執行個體計費
通訊端、核心和主機 ID 的可見度	提供通訊端與實體核心數目的可見度	沒有可見度

	Dedicated Host	Dedicated Instance
主機和執行個體親和性	可讓您隨時間一致地將執行個體部署到相同的實體伺服器	不支援
訂定執行個體配置目標	對於執行個體在實體伺服器上的配置方式，提供額外可見性和控制權	不支援
自動執行個體復原功能	支援。如需詳細資訊，請參閱 Amazon EC2 專用主機恢復。	支援
攜帶您自己的授權 (BYOL)	支援	部分支援 *
Capacity Reservations	不支援	支援

* 透過軟體保證提供授權行動性的 Microsoft SQL 伺服器，以及 Windows 虛擬桌面存取 (VDA) 授權可搭配專用執行個體使用。

如需專用預留執行個體的詳細資訊，請參閱 Amazon EC2 專用實例。

專用執行個體 限制

在您配置專用執行個體之前，請記下以下的法規與限制：

- 若要在專用主機上執行 RHEL SUSE Linux 和SQL伺服器，您必須自行攜帶AMIs。RHEL提供 AWS 或可用AMIs的、SUSE Linux 和SQL伺服器 AWS Marketplace 無法與專用主機搭配使用。如需如何建立自己的詳細資訊AMI，請參閱將您自己的軟體授權帶到 Amazon EC2 專用主機。

  此限制不適用於配置給記憶體密集型執行個體 (`u-6tb1.metal`、`u-9tb1.metal`、`u-12tb1.metal`、`u-18tb1.metal` 和 `u-24tb1.metal`) 的主機。RHELAMIs以及由提供 AWS 或可用的 SUSE Linux AWS Marketplace 可用於這些主機。
- 每個區域每個 AWS 帳戶每個執行個體系列的執行中專用執行個體有數量限制。配額僅適用於執行中的執行個體。如果執行個體處於待定、停止中或已停止狀態，則不會計入您的配額。若要檢視帳戶配額，或請求提升配額，可使用 Service Quotas 主控台。
- 在專用主機上執行的執行個體只能在VPC.

- 僅在使用指定主機資源群組的啟動範本時，才支援 Auto Scaling 群組。如需詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南中的使用進階設定建立啟動範本。
- 不支援 Amazon RDS 執行個體。
- AWS 免費用量層不適用於專用主機。
- 執行個體的配置控制是指針對專用執行個體中所啟動的執行個體進行管理。您無法在置放群組中啟動專用執行個體。
- 如果您為虛擬執行個體類型分配主機，則在分配主機之後，無法將執行個體類型修改為 `.metal` 執行個體類型。例如，如果您為 `m5.large` 執行個體類型分配主機，則無法將執行個體類型修改為 `m5.metal`。

  同樣，如果您為 `.metal` 執行個體類型分配主機，則在分配主機之後，無法將執行個體類型修改為虛擬執行個體類型。例如，如果您為 `m5.metal` 執行個體類型分配主機，則無法將執行個體類型修改為 `m5.large`。

目錄

- Amazon EC2 專用主機定價和計費
- Amazon EC2 專用主機執行個體容量組態
- Amazon EC2 專用主機上的高載 T3 執行個體
- 將您自己的軟體授權帶到 Amazon EC2 專用主機
- Amazon EC2 專用主機自動放置和主機親和性
- 分配 Amazon EC2 專用主機以在您的帳戶中使用
- 在 EC2 Amazon EC2 專用主機上啟動 Amazon 實例
- 將 Amazon EC2 執行個體啟動到主機資源群組
- 修改現有 Amazon EC2 專用主機的自動放置設定
- 修改現有 Amazon EC2 專用主機支援的執行個體類型
- 修改 Amazon EC2 執行個體的 Amazon EC2 專用主機租用和親和性
- 發布 Amazon EC2 專用主機
- 購買專用主機保留以享有專用主機帳單折扣
- 跨帳戶 Amazon EC2 專用主機共享
- Amazon EC2 專用主機 AWS Outposts
- Amazon EC2 專用主機恢復

- Amazon EC2 專用主機的主機維護
- 監控 Amazon EC2 專用主機的狀態
- 使用追蹤 Amazon EC2 專用主機組態變更 AWS Config

Amazon EC2 專用主機定價和計費

專用主機 定價會根據付款選項有所差異。

付款選項

- 隨需專用執行個體
- Dedicated Host Reservations
- Savings Plans
- 在專用執行個體的 Windows Server 定價

隨需專用執行個體

當您將專用執行個體配置到帳戶時，會自動啟用隨需計費。

專用執行個體的隨需價格依執行個體系列和區域而異。不論您選擇在專用執行個體中啟動的執行個體數量或大小為何，都會以每秒 (最低 60 秒) 費率計費。如需隨需定價的詳細資訊，請參閱 Amazon EC2 專用主機隨需定價。

您可以隨時釋出隨需專用執行個體以停止累積計費。如需如何釋出專用執行個體的資訊，請參閱發布 Amazon EC2 專用主機。

Dedicated Host Reservations

專用執行個體預留提供相當於執行隨需專用執行個體的計費折扣。保留有三種付款選項可用：

- 無預付— 無預付保留可提供您一段期間的專用執行個體用量折扣，且不需要預付款。提供一年期和三年期。只有部分執行個體系列支援三年期無預付保留。
- 部分預付— 必須預先支付其中一部分的保留，期間內剩餘的小時數則會以折扣後的費率計費。提供一年期和三年期。
- 全額預付— 提供最低的有效價位。提供一年期和三年期，並預付整期的成本，日後不需另外付費。

您的帳戶中必須具備作用中專用執行個體，才能購買保留。每個保留可以涵蓋單一可用區域中支援相同執行個體系列的一或多部主機。保留會套用至主機上的執行個體系列，而不是執行個體大小。若您有三個具有不同大小執行個體的專用執行個體 (`m4.xlarge`、`m4.medium` 和 `m4.large`)，您可以將單一 `m4` 保留與所有那些專用執行個體建立關聯。保留的執行個體系列和可用區域必須符合您想與其建立關聯之專用執行個體的執行個體系列和可用區域。

當保留與專用執行個體建立關聯時，直到保留的期間結束前都不能釋出專用執行個體。

如需有關保留定價的詳細資訊，請參閱 Amazon EC2 專用主機定價。

Savings Plans

Savings Plans 是個彈性定價模式，提供 隨需執行個體 大幅節省成本。透過 Savings Plans，您承諾以一或三年為單位，以USD每小時為單位的一致使用量。這可提供您彈性使用 專用執行個體 以達到更佳的需求，且持續的節省金費，而不是與特定的 專用執行個體 綁約。如需詳細資訊，請參閱 AWS Savings Plans 使用者指南。

> **Note**
>
> Savings Plans 不支援 `u-6tb1.metal`、`u-9tb1.metal`、`u-12tb1.metal`、`u-18tb1.metal` 及 `u-24tb1.metal` 專用執行個體。

在專用執行個體的 Windows Server 定價

根據 Microsoft 授權條款，您可以將現有的 Windows 伺服器和SQL伺服器授權帶到專用主機。如果您選取自帶授權，則軟體使用不收取額外費用。

此外，您還可以使用 Amazon AMIs 提供的 Windows 服務器在專用主機上運行最新版本的 Windows 服務器。如果您有資格在專用主機上執行的現有SQL伺服器授權，但需要 Windows Server 來執行伺服器工作負載，則這種情況很常見。SQLAmazon AMIs 提供的 Windows 伺服器僅支援目前一代的執行個體類型。 如需詳細資訊，請參閱 Amazon EC2 專用主機定價。

Amazon EC2 專用主機執行個體容量組態

專用主機支援不同的組態 (實體核心、通訊端和VCPUs)，可讓您執行不同系列和大小的執行個體。

在帳戶中配置專用執行個體時，您可以選擇能支援單一執行個體類型的組態，或相同執行個體系列中多種執行個體類型的組態。您可在主機上執行的執行個體數量將依您選擇的組態而定。

目錄

- 單一執行個體類型支援
- 多種執行個體類型支援

單一執行個體類型支援

您可以配置僅支援一種執行個體類型的專用執行個體。若使用此組態，您在專用執行個體上啟動的每個執行個體，其執行個體類型皆必須與配置主機時指定的執行個體類型相同。

例如，您可以配置僅支援 `m5.4xlarge` 執行個體類型的主機。在此情況下，您僅能執行該主機上的 `m5.4xlarge` 執行個體。

您可在主機上啟動的執行個體數量，將依該主機提供的實體核心數量以及指定執行個體類型使用的核心數目而定。例如，如果您為 `m5.4xlarge` 執行個體配置主機，且該主機提供 48 個實體核心，則每個 `m5.4xlarge` 執行個體會消耗 8 個實體核心。這表示您最多可在該主機上啟動 6 個執行個體 (48 個實體核心 / 每個執行個體 8 個核心 = 6 個執行個體)。

多種執行個體類型支援

您可以配置能支援相同執行個體系列中多種執行個體類型的專用執行個體。此舉可讓您在相同主機上執行不同類型的執行個體，只要這些執行個體位於相同執行個體系列中，且主機具有足夠的執行個體容量即可。

例如，您可以配置能支援 R5 執行個體系列中不同執行個體類型的主機。在此情況下，您可以在該主機上啟動任意 R5 執行個體類型的組合，例如 `r5.large`、`r5.xlarge`、`r5.2xlarge` 以及 `r5.4xlarge`，實際情況依主機的實體核心容量而定。

下列執行個體系列可支援具有多種執行個體類型支援的專用執行個體：

- 一般用途：A1、M5、M5n、M6i 和 T3
- 運算優化：C5、C5n 和 C6i
- 記憶體優化：R5、R5n 和 R6i

您可在主機上執行的執行個體數量，將依該主機提供的實體核心數量以及在該主機上執行的每種執行個體類型所消耗的核心數目而定。例如，如果配置一個會提供 48 個實體核心的 R5 主機，在執行兩個 `r5.2xlarge` 執行個體 (4 個核心 x 2 個執行個體) 以及三個 `r5.4xlarge` 執行個體 (8 個核心 x 3 個執行個體) 時，這些執行個體總共會消耗 32 個核心，而且只要不超過剩餘的 16 個核心，您就可執行任意組合的 R5 執行個體。

但是，對於每個執行個體系列，每種執行個體大小都有限制可執行的執行個體數目。例如，R5 專用執行個體最多可支援 2 個 `r5.8xlarge` 執行個體，使用 32 個實體核心。在此情況下，可使用其他較小的 R5 執行個體來將主機填滿至核心容量。如需了解每個執行個體系列支援的執行個體大小數量，請參閱專用執行個體組態表格。

下表所示為執行個體類型組合範例：

執行個體系列	範例執行個體大小組合
R5	• 範例 1：4 x `r5.4xlarge` + 4 x `r5.2xlarge` • 範例 2：1 x `r5.12xlarge` + 1 x `r5.4xlarge` + 1 x `r5.2xlarge` + 5 x `r5.xlarge` + 2 x `r5.large`
C5	• 範例 1：1 x `c5.9xlarge` + 2 x `c5.4xlarge` + 1 x `c5.xlarge` • 範例 2：4 x `c5.4xlarge` + 1 x `c5.xlarge` + 2 x `c5.large`
M5	• 範例 1：4 x `m5.4xlarge` + 4 x `m5.2xlarge` • 範例 2：1 x `m5.12xlarge` + 1 x `m5.4xlarge` + 1 x `m5.2xlarge` + 5 x `m5.xlarge` + 2 x `m5.large`

考量事項

在使用支援多種執行個體類型的專用執行個體時，請注意下列各項：

- 若為 N 類型專用執行個體 (例如 C5n、M5n 和 R5n)，您無法以組合方式使用更小型的執行個體 (`2xlarge` 和更小型) 及更大型的執行個體 (`4xlarge` 和更大型，包括 `metal`)。若您要在 N 類型專用執行個體上同時用更小型和更大型執行個體，則必須為更小型和更大型的執行個體配置不同主機。
- 建議您先啟動較大類型的執行個體，然後視需要以較小類型的執行個體填滿剩餘的執行個體容量。

Amazon EC2 專用主機上的高載 T3 執行個體

專用執行個體支援爆量效能 T3 執行個體。T3 執行個體可在專用硬體上使用合格BYOL授權軟體，提供符合成本效益的方式。T3 執行個體CPU佔用空間較小，可讓您將工作負載整合到較少的主機上，並將每核心授權使用率最大化。

T3 專用主機最適合執行CPU使用率低至中等的BYOL軟體。這包括符合資格的每個插槽、每個核心或個別虛擬機器的軟體授權，例如 Windows 伺服器、Windows 桌面、SQL伺服器、SUSE企業 Linux 伺服器、RHEL 和 Oracle 資料庫。例如，適用於 T3 專用執行個體的工作負載包括中小型資料庫、虛擬桌面、開發及測試環境、程式碼儲存庫和產品原型等。T3 專用主機不建議用於持續高CPU使用率的工作負載，或同時遇到相關CPU突發的工作負載。

專用執行個體上的 T3 執行個體使用的額度模型是與共用租用硬體上的 T3 執行個體相同。但是，其僅支援 `standard` 額度模型；並不支援 `unlimited` 額度模型。在 `standard` 模式中，專用執行個體上的 T3 執行個體使用與共用租用硬體上爆量效能執行個體相同的方式獲得、花費以及累積額度。它們提供了基準CPU性能，並且能夠超出基準級別。為了超出基準線，執行個體會花費在其CPU信用餘額中累積的積分。當累積的信用額度耗盡時，CPU使用率會降低至基準層級。如需 `standard` 模式的詳細資訊，請參閱 標準爆量效能執行個體如何運作。

T3 專用主機支援 Amazon EC2 專用主機提供的所有功能，包括單一主機上的多個執行個體大小、主機資源群組和BYOL。

支援的 T3 執行個體大小和組態

T3 專用主機會執行一般用途的高載 T3 執行個體，這些執行個體會共用主機的CPU資源，提供基準CPU效能，並在需要時提升到更高層級的能力。如此一來，擁有 48 個核心的 T3 專用執行個體，每台主機最多可支援 192 個執行個體。為了有效利用主機的資源並提供最佳的執行個體效能，Amazon EC2 執行個體放置演算法會自動計算可在主機上啟動的執行個體和執行個體大小組合的支援數量。

T3 專用執行個體支援同一主機上的多個執行個體類型。所有 T3 執行個體大小都受專用執行個體的支援。您可以在主機CPU限制內執行不同的 T3 執行個體組合。

下表列出支援的執行個體類型、摘要說明每個執行個體類型的效能，並指明可啟動之每種大小的執行個體數目上限。

執行個體類型	vCPUs	記憶體 (GiB)	每 v 的基準 CPU使用率 CPU	網路高載頻寬 (Gbps)	Amazon EBS 突發帶寬 (Mbps)	每個專用執行個體的執行個體數目上限
t3.na	2	0.5	5%	5	最高 2,085	192
t3.m	2	1	10%	5	最高 2,085	192
t3.sn	2	2	20%	5	最高 2,085	192
t3.m	2	4	20%	5	最高 2,085	192
t3.la	2	8	30%	5	2,780	96
t3.xl	4	16	40%	5	2,780	48
t3.2x e	8	32	40%	5	2,780	24

監控 T3 專CPU用主機的使用率

您可以使用 `DedicatedHostCPUUtilization` Amazon CloudWatch 指標監控專用主機的 v 使用CPU率。`EC2` 命名空間和 `Per-Host-Metrics` 維度均可使用此指標。如需詳細資訊，請參閱專用執行個體指標。

將您自己的軟體授權帶到 Amazon EC2 專用主機

專用執行個體可讓您使用現有的每個通訊端、每個核心或每個 VM 軟體授權。當您使用自有授權時，需負責管理自己的授權。不過，Amazon EC2 的功能可協助您維持授權合規性，例如執行個體親和性和目標放置。

這些是將您自己的大量授權機器映像帶入 Amazon 時應遵循的一般步驟EC2。

1. 確定授權條款 (控管機器映像的使用辦法) 允許您在虛擬化雲端環境中使用機器映像。如需 Microsoft 授權的詳細資訊，請參閱 Amazon Web Services 和 Microsoft 授權。

2. 驗證您的機器映像可以在 Amazon 中使用後EC2，請使用 VM 匯入/匯出來匯入。如需如何匯入機器映像的資訊，請參閱 VM Import/Export 使用者指南。
3. 匯入機器映像之後，您就可以透過該映像在帳戶中的作用中專用執行個體上啟動執行個體。
4. 當您執行這些執行個體時，視作業系統而定，您可能需要針對自己的KMS伺服器 (例如 Windows 伺服器或 Windows SQL 伺服器) 啟動這些執行個體。您無法AMI對 Amazon 視窗KMS服務器激活導入的窗口。

> **Note**
>
> 若要追蹤影像的使用方式 AWS，請在中啟用主持人錄製 AWS Config。您可以使用 AWS Config 將組態變更記錄到專用主機，並使用輸出作為授權報告的資料來源。如需詳細資訊，請參閱使用追蹤 Amazon EC2 專用主機組態變更 AWS Config。

Amazon EC2 專用主機自動放置和主機親和性

您可以在執行個體層級和主機層級上進行 專用執行個體 配置控制。

自動配置

您會在主機層級設定自動配置。這可讓您管理執行個體要在特定主機上啟動，或在任何具有相符組態的可用主機上啟動。

當專用執行個體的自動配置為停用時，它僅會接受主機租用執行個體的啟動 (指定其唯一主機 ID)。這是新專用執行個體的預設設定。

當專用執行個體的自動配置為啟用時，它會接受符合其執行個體類型組態之任何未訂定目標的執行個體啟動。

在啟動執行個體時，您必須設定其租用。若在專用執行個體上啟動執行個體而未提供特定 `HostId`，該執行個體就會在啟用自動配置且符合其執行個體類型的任何專用執行個體上啟動。

主機親和性

您必須在執行個體層級設定主機親和性。它會建立執行個體和專用執行個體之間的啟動關係。

當親和性設為 `Host` 時，先前在特定主機上啟動的執行個體若曾停止，執行個體一律會在相同的主機上重新啟動。這適用於有訂定與未訂定目標的啟動。

當親和性設為 `Default` 且您停止並重新啟動執行個體時，它會在任何可用的主機上重新啟動。不過，它會嘗試在先前執行的專用執行個體上啟動 (以最佳做法為主)。

分配 Amazon EC2 專用主機以在您的帳戶中使用

若要開始使用專用主機，您必須先將其配置到您的帳戶中。配置 專用執行個體 之後，會立即將 專用執行個體 容量提供給您的帳戶，而且您可以開始在 專用執行個體 上啟動執行個體。

在帳戶中配置專用執行個體時，您可以選擇能支援單一執行個體類型的組態，或相同執行個體系列中多種執行個體類型的組態。您可在主機上執行的執行個體數量將依您選擇的組態而定。如需詳細資訊，請參閱 Amazon EC2 專用主機執行個體容量組態。

Console

配置專用執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Dedicated Hosts (專用執行個體)，然後選擇 Allocate 專用執行個體 (配置專用執行個體)。
3. 對於 Instance family (執行個體系列)，選擇適用於 專用執行個體 的執行個體系列。
4. 指定 專用執行個體 是否支援所選執行個體系列內的多種執行個體容量，或是僅支援特定執行個體類型。執行下列操作。
   - 若要將 專用執行個體 設定為支援所選執行個體系列內的多個執行類型，請針對 Support multiple instance types (支援多個執行個體類型) 選取 Enable (啟用)。啟用此功能，您就能在 專用執行個體 上啟動來自相同執行個體系列的不同執行個體容量。例如，如果您選擇 `m5` 執行個體系列且選擇了此選項，您就能在 專用執行個體 上啟動 `m5.xlarge` 及 `m5.4xlarge`。
   - 若要將 專用執行個體 設定為支援所選執行個體系列內的單一執行特定類型，請清除 Support multiple instance types (支援多個執行個體類型)，然後針對 Instance type (執行個體類型) 選取要支援的執行個體類型。這可讓您在 專用執行個體 上啟動單一執行個體類型。例如，如果您選擇此選項並指定 `m5.4xlarge` 作為支援的執行個體類型，則您僅能在 專用執行個體 上啟動 `m5.4xlarge`。
5. 對於 Availability Zone (可用區域)，選擇要配置 專用執行個體 的可用區域。
6. 若要允許 專用執行個體 接受符合其執行個體類型的未鎖定執行個體啟動，對於 Instance auto-placement (執行個體自動放置)，請選擇 Enable (啟用)。如需自動配置的詳細資訊，請參閱Amazon EC2 專用主機自動放置和主機親和性。

7. 若要啟用 專用執行個體 的主機復原，對於 Host recovery (主機復原)，請選擇 Enable (啟用)。如需詳細資訊，請參閱Amazon EC2 專用主機恢復。
8. 對於 Quantity (數量)，輸入要配置的 專用執行個體 數量。
9. (選用) 選擇新增標籤，然後輸入標籤金鑰和標籤值。
10. 選擇 Allocate (配置)。

AWS CLI

配置專用執行個體

使用分配主機命令 AWS CLI 。下列命令會配置一個專用執行個體，其可支援 `us-east-1a` 可用區域中來自 `m5` 執行個體系列的多個執行個體類型。主機也已啟用主機復原功能，而且已停用自動配置功能。

```
aws ec2 allocate-hosts --instance-family "m5" --availability-zone "us-east-1a" --
auto-placement "off" --host-recovery "on" --quantity 1
```

以下命令會配置一個專用執行個體，其可支援在 `eu-west-1a` 可用區域中啟動之未訂定目標的 `m4.large` 執行個體、啟用主機復原，並套用含 `purpose` 金鑰和 `production` 值的標籤。

```
aws ec2 allocate-hosts --instance-type "m4.large" --availability-zone "eu-west-1a"
 --auto-placement "on" --host-recovery "on" --quantity 1 --tag-specifications
 'ResourceType=dedicated-host,Tags=[{Key=purpose,Value=production}]'
```

PowerShell

配置專用執行個體

使用指New-EC2Host AWS Tools for Windows PowerShell 令。下列命令會配置一個專用執行個體，其可支援 `us-east-1a` 可用區域中來自 `m5` 執行個體系列的多個執行個體類型。主機也已啟用主機復原功能，而且已停用自動配置功能。

```
PS C:\> New-EC2Host -InstanceFamily m5 -AvailabilityZone us-east-1a -
AutoPlacement Off -HostRecovery On -Quantity 1
```

以下命令會配置一個專用執行個體，其可支援在 `eu-west-1a` 可用區域中啟動之未訂定目標的 `m4.large` 執行個體、啟用主機復原，並套用含 `purpose` 金鑰和 `production` 值的標籤。

TagSpecification 參數 (在建立專用執行個體時用來對其套用標籤) 需要使用物件來指定要套用標籤的資源類型、標籤鍵和標籤值。下列命令會建立所需的物件。

```
PS C:\> $tag = @{ Key="purpose"; Value="production" }
PS C:\> $tagspec = new-object Amazon.EC2.Model.TagSpecification
PS C:\> $tagspec.ResourceType = "dedicated-host"
PS C:\> $tagspec.Tags.Add($tag)
```

以下命令會配置專用執行個體，並套用 $tagspec 物件指定的標籤。

```
PS C:\> New-EC2Host -InstanceType m4.large -AvailabilityZone eu-west-1a -
AutoPlacement On -HostRecovery On -Quantity 1 -TagSpecification $tagspec
```

在 EC2 Amazon EC2 專用主機上啟動 Amazon 實例

配置好專用執行個體之後，您就可以在其中啟動執行個體。如果您不具備作用中的專用執行個體，且其可用容量不足以啟動您的執行個體類型，您就無法啟動含 host 租用的執行個體。

> **Tip**
> 對於支援多種執行個體規模的專用執行個體，建議您先啟動較大的執行個體，然後視需要以較小的執行個體填滿剩餘的執行個體容量。

在您啟動執行個體之前，請記下以下的限制。如需詳細資訊，請參閱專用執行個體 限制。

您可以使用下列方法，將執行個體啟動至專用執行個體。

Console

在專用執行個體頁面於特定專用執行個體啟動執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 專用執行個體。
3. 在 Dedicated Hosts (專用執行個體) 頁面上，選取一個主機，然後選取 Actions (動作)、Launch Instance(s) onto Host (在主機啟動執行個體)。
4. 在「應用程式和作業系統映像」區段中，AMI從清單中選取。

> **Note**
>
> SQLAmazon RHEL AMIs 提供的服務器和服務器不EC2能與專用主機一起使用。SUSE

5. 在 Instance type (執行個體類型) 區段中，選取要啟動的執行個體類型。

> **Note**
>
> 如果 專用執行個體 僅支援單一執行個體類型，則預設會選取支援的執行個體類型，而且不可變更。
> 如果專用執行個體支援多個執行個體類型，您必須根據專用執行個體的可用執行個體容量，選取支援執行個體系列內的執行個體類型。建議您先啟動較大的執行個體，然後視需要以較小的執行個體填滿剩餘的執行個體容量。

6. 在 Key pair (金鑰對) 區段中，選取要與執行個體建立關聯的金鑰對。
7. 在「進階詳細資料」區段中，針對「租賃相似性」，選擇下列其中一項：
   - 關閉 — 已停用主機相似性。執行個體會啟動到指定的主機，但如果停止，則不保證會在相同的專用主機上重新啟動。
   - 專用主機 ID — 已啟用主機相似性。如果停止，執行個體一律會在此指定主機上重新啟動 (如果有容量)。如果主機沒有容量，則無法重新啟動執行個體；您必須與其他主機建立相似性。

   如需親和性的詳細資訊，請參閱Amazon EC2 專用主機自動放置和主機親和性。

> **Note**
>
> Tenancy (租用) 和 Host (主機) 選項會依據您選取的主機預先設定。

8. 視需要設定其餘的執行個體選項。如需詳細資訊，請參閱Amazon EC2 執行個體組態參數的參考。
9. 選擇啟動執行個體。

使用啟動執行個體精靈，在專用執行個體中啟動執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。

2. 在導覽窗格中，選擇 Instances (執行個體)、Launch Instance (啟動執行個體)。
3. 在「應用程式和作業系統映像」區段中，AMI從清單中選取。

> **Note**
>
> SQLAmazon RHEL AMIs 提供的服務器和服務器不EC2能與專用主機一起使用。SUSE

4. 在 Instance type (執行個體類型) 區段中，選取要啟動的執行個體類型。
5. 在 Key pair (金鑰對) 區段中，選取要與執行個體建立關聯的金鑰對。
6. 在 Advanced details (進階詳細資料) 區段中，執行下列操作：
   a. 對於 Tenancy (租用)，選取 Dedicated Host (專用執行個體)。
   b. 對於 Target host by (目標主機依據) 中，選取 Host ID (主機 ID)。
   c. 對於 Target host ID (目標主機 ID)，選取要在其上啟動執行個體的主機。
   d. 對於租賃親和性，請選擇下列其中一項：
      - 關閉 — 已停用主機相似性。執行個體會啟動到指定的主機，但如果停止，則不保證會在相同的專用主機上重新啟動。
      - 專用主機 ID — 已啟用主機相似性。如果停止，執行個體一律會在此指定主機上重新啟動 (如果有容量)。如果主機沒有容量，則無法重新啟動執行個體；您必須與其他主機建立相似性。

      如需親和性的詳細資訊，請參閱Amazon EC2 專用主機自動放置和主機親和性。
7. 視需要設定其餘的執行個體選項。如需詳細資訊，請參閱Amazon EC2 執行個體組態參數的參考。
8. 選擇啟動執行個體。

AWS CLI

將執行個體啟動至專用執行個體

使用執行個體 AWS CLI 命令，並在`Placement`要求參數中指定執行個體相似性、租用和主機。

PowerShell

將執行個體啟動至專用執行個體

使用命New-EC2Instance AWS Tools for Windows PowerShell 令並在 `Placement` request 參數中指定執行個體相似性、租用和主機。

將 Amazon EC2 執行個體啟動到主機資源群組

專用主機也與 AWS License Manager. 您可以使用 License Manager 建立主機資源群組，這是做為單一實體管理的專用執行個體集合。建立主機資源群組時，您可以指定專用執行個體的主機管理偏好設定，例如自動配置和自動發佈。這讓您在無需手動配置和管理這些主機的情況下，在專用執行個體中啟動執行個體。如需詳細資訊，請參閱《AWS License Manager 使用者指南》中的 主機資源群組。

當您將執行個體啟動到具有可用執行個體容量的專用主機的主機資源群組時，Amazon 會將執行個體EC2啟動到該主機。如果主機資源群組沒有具有可用執行個體容量的主機，Amazon 會EC2自動在主機資源群組中配置新主機，然後將執行個體啟動到該主機。如需詳細資訊，請參閱《AWS License Manager 使用者指南》中的 主機資源群組。

需求與限制

- 您必須將以核心或插槽為基礎的授權組態與. AMI
- 您不能使用SQL服務器SUSE，或由 Amazon RHEL AMIs 提供的專EC2用主機。
- 您無法透過選擇主機 ID 鎖定特定主機為目標，而且在將執行個體啟動至主機資源群組時，也無法啟用執行個體親和性。

您可以使用下列方法，將執行個體啟動至主機資源群組。

Console

將執行個體啟動至主機資源群組

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)、Launch Instance (啟動執行個體)。
3. 在「應用程式和作業系統映像」區段中，AMI從清單中選取。

> **Note**
>
> SQLAmazon RHEL AMIs 提供的服務器和服務器不EC2能與專用主機一起使用。SUSE

4. 在 Instance type (執行個體類型) 區段中，選取要啟動的執行個體類型。

5. 在 Key pair (金鑰對) 區段中，選取要與執行個體建立關聯的金鑰對。
6. 在 Advanced details (進階詳細資料) 區段中，執行下列操作：
   a. 對於 Tenancy (租用)，選取 Dedicated Host (專用執行個體)。
   b. 對於 Target host by (目標主機依據)，選取 Host resource group (主機資源群組)。
   c. 對於 Tenancy host resource group (租用主機資源群組)，選取要在其中啟動執行個體的主機資源群組。
   d. 對於 Tenancy affinity (租用親和性)，執行下列其中一個操作：
      - 選取 Off (關閉) – 在指定主機上啟動的執行個體，不保證停止後會在相同的專用執行個體上重新啟動。
      - 選取專用執行個體 ID – 執行個體在停止後始終會在此特定主機上重新啟動。

      如需親和性的詳細資訊，請參閱Amazon EC2 專用主機自動放置和主機親和性。
7. 視需要設定其餘的執行個體選項。如需詳細資訊，請參閱Amazon EC2 執行個體組態參數的參考。
8. 選擇啟動執行個體。

AWS CLI

將執行個體啟動至主機資源群組

使用執行個體 AWS CLI 命令，並在`Placement`要求參數中省略租用選項，並指定主機資源群組。ARN

PowerShell

將執行個體啟動至主機資源群組

使用命New-EC2Instance AWS Tools for Windows PowerShell 令，並在 `Placement` request 參數中省略「租用」選項，並指定主機資源群組ARN。

修改現有 Amazon EC2 專用主機的自動放置設定

將專用主機配置到 AWS 帳戶後，您可以使用下列其中一種方法修改專用主機的自動放置設定。

Console

修改專用執行個體的自動配置

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 專用主機。
3. 選取主機並選取 Actions (動作)、Modify host (修改主機)。
4. 對於 Instance auto-placement (執行個體自動配置)，選擇 Enable (啟用) 以啟用自動配置，或清除 Enable (啟用) 以停用自動配置。如需詳細資訊，請參閱Amazon EC2 專用主機自動放置和主機親和性。
5. 選擇 Save (儲存)。

AWS CLI

修改專用執行個體的自動配置

使用「修改主 AWS CLI 機」指令。下列範例會針對指定的專用執行個體啟用自動配置。

```
aws ec2 modify-hosts --auto-placement on --host-ids h-012a3456b7890cdef
```

PowerShell

修改專用執行個體的自動配置

使用指Edit-EC2Host AWS Tools for Windows PowerShell 令。下列範例會針對指定的專用執行個體啟用自動配置。

```
PS C:\> Edit-EC2Host --AutoPlacement 1 --HostId h-012a3456b7890cdef
```

修改現有 Amazon EC2 專用主機支援的執行個體類型

您可以修改 專用執行個體 來變更其支援的執行個體類型。如果其目前支援單一執行個體類型，您可以進行修改，以支援該執行個體系統內的多個執行個體類型。同樣地，如果其目前支援多個執行個體類型，您可以進行修改，僅支援特定執行個體類型。

若要修改 專用執行個體 以支援多個執行個體類型，您必須先停止主機上所有執行的執行個體。此修改需要約 10 分鐘的時間完成。進行修改時，專用執行個體 會轉移為 pending 狀態。當執行個體處於 pending 狀態時，您無法開始已停止的執行個體，或是啟動 專用執行個體 上的新執行個體。

若要將支援多個執行個體類型的 專用執行個體 修改為僅支援單一執行個體類型，主機必須沒有執行中的執行個體，或是執行中的執行個體必須是您希望主機支援的執行個體類型。例如，若要將支援 m5 執行個體系列內多個執行個體類型的主機修改為僅支援 m5.large 執行個體，專用執行個體 必須沒有執行中的執行個體，或是其上必須只有 m5.large 個執行個體在執行中。

如果您為虛擬執行個體類型分配主機，則在分配主機之後，無法將執行個體類型修改為 .metal 執行個體類型。例如，如果您為 m5.large 執行個體類型分配主機，則無法將執行個體類型修改為 m5.metal。同樣，如果您為 .metal 執行個體類型分配主機，則在分配主機之後，無法將執行個體類型修改為虛擬執行個體類型。例如，如果您為 m5.metal 執行個體類型分配主機，則無法將執行個體類型修改為 m5.large。

您可以使用下列其中一種方法，來修改支援的執行個體類型。

Console

修改專用執行個體支援的執行個體類型

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Dedicated Host (專用執行個體)。
3. 選取要修改的專用執行個體，然後選取 Actions (動作)、Modify host (修改主機)。
4. 根據專用執行個體的目前組態，執行下列其中一個動作：

   - 如果專用執行個體目前支援特定執行個體類型，則不會啟用 Support multiple instance types (支援多個執行個體類型)，且 Instance type (執行個體類型) 會列出支援的執行個體類型。若要將主機修改為支援目前執行個體系列中的多個類型，請針對 Support multiple instance types (支援多個執行個體類型)，選取 Enable (啟用)。

     您必須先停止主機上所有執行的執行個體，然後再將之修改為支援多個執行個體類型。

   - 如果專用執行個體目前支援執行個體系列內的多個執行個體類型，請針對 Support multiple instance types (支援多個執行個體類型) 選取 Enabled (已啟用)。若要將主機修改為支援特定執行個體類型，請針對 Support multiple instance types (支援多個執行個體類型) 清除 Enable (啟用)，然後針對 Instance type (執行個體類型) 選取要支援的特定執行個體類型。

     您無法變更 專用執行個體 支援的執行個體系列。

5. 選擇 Save (儲存)。

AWS CLI

修改專用執行個體支援的執行個體類型

使用「修改主 AWS CLI 機」指令。

下列命令會修改 專用執行個體 以支援 m5 執行個體系列內的多個執行個體類型。

```
aws ec2 modify-hosts --instance-family m5 --host-ids h-012a3456b7890cdef
```

下列命令會修改 專用執行個體，以便僅支援 m5.xlarge 執行個體。

```
aws ec2 modify-hosts --instance-type m5.xlarge --instance-family --host-
ids h-012a3456b7890cdef
```

PowerShell

修改專用執行個體支援的執行個體類型

使用指Edit-EC2Host AWS Tools for Windows PowerShell 令。

下列命令會修改 專用執行個體 以支援 m5 執行個體系列內的多個執行個體類型。

```
PS C:\> Edit-EC2Host --InstanceFamily m5 --HostId h-012a3456b7890cdef
```

下列命令會修改 專用執行個體，以便僅支援 m5.xlarge 執行個體。

```
PS C:\> Edit-EC2Host --InstanceType m5.xlarge --HostId h-012a3456b7890cdef
```

修改 Amazon EC2 執行個體的 Amazon EC2 專用主機租用和親和性

啟動執行個體之後，就可以變更執行個體的租用。您也可以修改執行個體的親和性以鎖定特定主機，或允許在帳戶中具有相符屬性的任何可用專用執行個體上啟動。若要修改執行個體租用或親和性，執行個體必須處於 stopped 狀態。

執行個體的作業系統詳細資料以及是否已安裝 SQL Server，都會影響支援的轉換。如需執行個體可用租用轉換路徑的詳細資訊，請參閱《License Manager 使用指南》中的租用轉換。

> **Note**
>
> 若使用 T3 執行個體，您必須在專用執行個體上啟動執行個體，才能使用 `host` 的租用。使用 T3 執行個體時，您無法將租用從 `host` 變更為 `dedicated` 或 `default`。嘗試進行上述任一不受支援的租用變更時，會顯示 `InvalidRequest` 錯誤代碼。

您可以使用下列方法來修改執行個體的租用和親和性。

Console

修改執行個體租用或親和性

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 選取 Instances (執行個體)，再選取要修改的執行個體。
3. 選擇 Instance state (執行個體狀態)、Stop (停止)。
4. 在已選取執行個體的情況下，選擇動作、執行個體設定、修改執行個體的配置。
5. 在 [修改執行個體放置] 頁面上，設定下列項目：
   - Tenancy (租用)— 選擇下列其中一項：
     - 執行專用硬體執行個體 – 以專用預留執行個體形式啟動執行個體。如需詳細資訊，請參閱Amazon EC2 專用實例。
     - 在專用執行個體上啟動執行個體 – 在可設定親和性的專用執行個體上啟動執行個體。
   - Affinity (親和性)— 選擇下列其中一項：
     - 此執行個體可在我的任一台主機上執行 – 執行個體會在您帳戶中任何支援其執行個體類型的可用專用執行個體上啟動。
     - 此執行個體僅能在所選的主機上執行 – 執行個體僅能在為 Target Host (目標主機) 選取的專用執行個體上執行。
   - Target Host (目標主機)— 選取執行個體必須在其上執行的專用執行個體。如果沒有列出任何目標主機，您的帳戶中可能沒有可用且相容的專用執行個體。

   如需詳細資訊，請參閱Amazon EC2 專用主機自動放置和主機親和性。
6. 選擇 Save (儲存)。

AWS CLI

修改執行個體租用或親和性

使用指modify-instance-placement AWS CLI 令。以下範例會將指定執行個體的親和性從 `default` 變更為 `host`，並指定執行個體要與哪個專用執行個體保有親和性。

```
aws ec2 modify-instance-placement --instance-id i-1234567890abcdef0 --affinity host
 --tenancy host --host-id h-012a3456b7890cdef
```

PowerShell

修改執行個體租用或親和性

使用指Edit-EC2InstancePlacement AWS Tools for Windows PowerShell 令。以下範例會將指定執行個體的親和性從 `default` 變更為 `host`，並指定執行個體要與哪個專用執行個體保有親和性。

```
PS C:\> Edit-EC2InstancePlacement -InstanceId i-1234567890abcdef0 -Affinity host -
Tenancy host -HostId h-012a3456b7890cdef
```

發布 Amazon EC2 專用主機

如果您不再需要專用主機，可以停止在主機上執行的執行個體，指示它們在不同的主機上啟動，然後釋放主機。

您必須先停止在專用執行個體上執行的任何執行個體，才能釋出主機。您可以將這些執行個體遷移到帳戶中的其他專用執行個體，以便繼續使用。這些步驟僅適用於隨需專用執行個體。

您可以使用下列方法來釋出專用執行個體。

Console

釋出專用執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 專用主機。
3. 在 專用執行個體 (專用執行個體) 頁面中，選取要釋出的專用執行個體。
4. 選擇 Actions (動作)、Release host (釋出主機)。
5. 若要確認，請選擇 Release (釋出)。

AWS CLI

釋出專用執行個體

使用「發布 AWS CLI 主機」命令。

```
aws ec2 release-hosts --host-ids h-012a3456b7890cdef
```

PowerShell

釋出專用執行個體

使用指Remove-EC2Hosts AWS Tools for Windows PowerShell 令。

```
PS C:\> Remove-EC2Hosts -HostId h-012a3456b7890cdef
```

在您釋出專用執行個體之後，即無法再次使用相同的主機或主機 ID，也不會再以隨需計費費率向您收費。專用執行個體的狀態會變更為 `released`，且您將無法在該主機上啟動任何執行個體。

> **Note**
>
> 如果您最近曾釋出專用執行個體，其可能需要一些時間才能停止計入限制。在這段期間，您可能會在嘗試配置新的專用執行個體時遇到 `LimitExceeded` 錯誤。若是這種情況，請嘗試在幾分鐘後再次配置新的主機。

停止的執行個體仍可供使用，並會列在 Instances (執行個體) 頁面中。這些執行個體仍保有其 `host` 租用設定。

購買專用主機保留以享有專用主機帳單折扣

與隨選專用主機定價相比，專用主機保留可為您提供高達 70% 的 discount。您必須在帳戶中配置作用中的專用主機，才能購買專用主機保留區。如需詳細資訊，請參閱Dedicated Host Reservations。

您可以使用下列方法購買專用主機保留區：

Console

購買保留

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。

2. 選擇 專用執行個體、專用執行個體預留、Purchase 專用執行個體預留 (購買 專用執行個體預留)。
3. 在「尋找提供項目」畫面上，執行下列動作：
   a. 針對執行個體系列，選取要為其購買專用主機保留項目之專用主機的執行個體系列。
   b. 對於付款選項，選擇並配置您偏好的付款選項。
4. 選擇 Next (下一步)。
5. 選取要與專用主機保留區產生關聯的專用主機，然後選擇下一步。
6. (選擇性) 將標記指派給專用主機保留區。
7. 檢查您的訂單並選擇「購買」。

AWS CLI

購買保留

1. 使用指describe-host-reservation-offerings AWS CLI 令列出符合您需求的可用供應項目。下列範例列出的方案可支援 m4 執行個體系列中的執行個體，並具有一年期限。

   > **Note**
   > 此期限以秒為單位指定。一年期限包括 31,536,000 秒，三年期限則為 94,608,000 秒。

   ```
   aws ec2 describe-host-reservation-offerings --filter Name=instance-family,Values=m4 --max-duration 31536000
   ```

   此命令會傳回符合您條件的方案清單。請記下要購買方案的 `offeringId`。

2. 使用指purchase-host-reservation AWS CLI 令購買訂購項目，並提供上一個步驟中`offeringId`所述的項目。下列範例會購買指定的保留區，並將其與 AWS 帳戶中已配置的特定專用主機產生關聯，並套用具有金鑰`purpose`且值為的標記`production`。

   ```
   aws ec2 purchase-host-reservation --offering-id hro-03f707bf363b6b324 --host-id-set h-013abcd2a00cbd123 --tag-specifications 'ResourceType=host-reservation,Tags={Key=purpose,Value=production}'
   ```

PowerShell

購買保留

1. 使用指Get-EC2HostReservationOffering AWS Tools for Windows PowerShell 令列出符合您需求的可用供應項目。下列範例列出的方案可支援 m4 執行個體系列中的執行個體，並具有一年期限。

   > **Note**
   >
   > 此期限以秒為單位指定。一年期限包括 31,536,000 秒，三年期限則為 94,608,000 秒。

   ```
   PS C:\> $filter = @{Name="instance-family"; Value="m4"}
   ```

   ```
   PS C:\> Get-EC2HostReservationOffering -filter $filter -MaxDuration 31536000
   ```

   此命令會傳回符合您條件的方案清單。請記下要購買方案的 `offeringId`。

2. 使用指New-EC2HostReservation AWS Tools for Windows PowerShell 令購買訂購項目，並提供上一個步驟中`offeringId`所述的項目。下列範例會購買指定的保留區，並將其與 AWS 帳戶中已配置的特定專用主機產生關聯。

   ```
   PS C:\> New-EC2HostReservation -OfferingId hro-03f707bf363b6b324 -
   HostIdSet h-013abcd2a00cbd123
   ```

跨帳戶 Amazon EC2 專用主機共享

專用主機共用可讓專用主機擁有者與其他 AWS 帳戶或 AWS 組織內共用其專用主機。這可讓您集中建立和管理專用主機，以及跨多個 AWS 帳戶或 AWS 組織內共用專用主機。

在此模型中，擁有專用主機 (擁有者) 的 AWS 帳戶會與其他 AWS 帳戶 (消費者) 共用該帳戶。消費者可以在與他們共用的專用執行個體中啟動執行個體，就好像他們在自己的帳戶中配置的專用執行個體中啟動執行個體一樣。擁有者負責管理專用執行個體和他們在其中啟動的執行個體。擁有者無法修改消費者在共用的專用執行個體上啟動的執行個體。消費者負責管理在分享給他們的專用執行個體中啟動的執行個體。消費者無法檢視或修改其他消費者或專用執行個體擁有者所擁有的執行個體，也無法修改與他們共用的專用執行個體。

專用執行個體擁有者可以將專用執行個體分享給下列對象：

- AWS 組織內部或外部的特定 AWS 帳戶
- 組織內的組 AWS 織單位
- 它的整個 AWS 組織

目錄

- 共用 專用執行個體 的先決條件
- 共用 專用執行個體 的限制
- 相關服務
- 跨可用區域共用
- 共用的 專用執行個體 許可
- 計費和計量
- 專用執行個體 限制
- 主機復原和 專用執行個體 共用
- 跨 AWS 帳戶共享 Amazon EC2 專用主機
- 取消共用與其 AWS 他帳戶共用的專用主機
- 在您的 AWS 帳戶中查看共享的 Amazon EC2 專用主機

共用 專用執行個體 的先決條件

- 若要共用專用主機，您必須在 AWS 帳戶中擁有該主機。您無法將已分享給您的專用執行個體再分享出去。
- 若要與您的組織或 AWS 組織中的組織單位共用專用主機，您 AWS 必須啟用與共用 AWS Organizations。如需詳細資訊，請參閱 AWS RAM 使用者指南中的透過 AWS Organizations啟用共用。

共用 專用執行個體 的限制

您無法共用已針對下列執行個體類型配置的 專用執行個體：`u-6tb1.metal`、`u-9tb1.metal`、`u-12tb1.metal`、`u-18tb1.metal` 和 `u-24tb1.metal`。

相關服務

AWS Resource Access Manager

專用主機共用與 AWS Resource Access Manager (AWS RAM) 整合。 AWS RAM 是一項服務，可讓您與任何 AWS 帳戶或透過其他帳戶共用 AWS 資源 AWS Organizations。使用 AWS RAM，您可以透過建立資源共用來共用您擁有的資源。資源共享指定要共用的資源，以及共用它們的消費者。消費者可以是個人 AWS 帳戶，也可以是組織單位或整個組織 AWS Organizations。

若要取得有關的更多資訊 AWS RAM，請參閱AWS RAM 使用者指南。

跨可用區域共用

為確保資源分配至區域中的所有可用區域，可用區域會獨立對應至各個帳戶的名稱。這可能導致帳戶之間的可用區域命名出現差異。例如，您 AWS 帳戶的可`us-east-1a`用區域可能與其他 AWS 帳戶`us-east-1a`的位置不同。

若要基於您的帳戶來識別專用執行個體的相對位置，您必須使用可用區域 ID (AZ ID)。可用區域 ID 是所有 AWS 帳戶中可用區域的唯一且一致的識別碼。例如，`use1-az1` 是 `us-east-1` 區域的可用區域 ID，它在每一個 AWS 帳戶的位置都相同。

若要檢視您帳戶中可IDs用區域的可用區域

1. 請在 https://console.aws.amazon.com/ram 開啟 AWS RAM 主控台。
2. 目前區域IDs的可用區域會顯示在畫面右側的「您的 AZ ID」面板中。

共用的 專用執行個體 許可

擁有者的許可

擁有者負責管理他們共用的專用執行個體以及在其上啟動的執行個體。擁有者可以檢視在共用的專用執行個體上執行的所有執行個體，包括消費者啟動的執行個體。不過，擁有者無法對消費者啟動的執行個體採取任何動作。

消費者的許可

消費者負責管理他們在共用的專用執行個體上啟動的執行個體。消費者完全無法修改共用的專用執行個體，也無法檢視或修改其他消費者或專用執行個體擁有者所擁有的執行個體。

計費和計量

共用專用執行個體無須額外收費。

擁有者會按他們共用的專用執行個體計費。對於消費者在共用的專用執行個體上啟動的執行個體則不計費。

專用執行個體預留會繼續為共用的專用執行個體提供帳單折扣。只有專用執行個體擁有者可以為他們擁有的共用專用執行個體購買專用執行個體預留。

專用執行個體 限制

共用的專用執行個體只會計入擁有者的專用執行個體限制。消費者的專用執行個體限制不受已與他們共用的專用執行個體所影響。同樣地，消費者在共用的專用執行個體上啟動的執行個體不會計入他們的執行個體限制。

主機復原和 專用執行個體 共用

主機復原可復原專用執行個體擁有者和其所共用之消費者啟動的執行個體。替換專用執行個體會配置到擁有者的帳戶。它會新增至與原始專用執行個體相同的資源共享，並與相同的消費者共用。

如需詳細資訊，請參閱Amazon EC2 專用主機恢復。

跨 AWS 帳戶共享 Amazon EC2 專用主機

當擁有者分享專用執行個體，它可讓消費者在主機上啟動執行個體。在共用主機上可用容量所允許下，消費者可以在當中啟動任意數量的執行個體。

> **Important**
>
> 請注意，您有責任確保您擁有適當的授權權限，以便在專用主機上共用任何BYOL授權。

如果您在啟用自動配置時共用專用執行個體，請記住下列事項，因為這可能會導致意外的專用執行個體使用量：

- 如果消費者使用專用執行個體租用啟動執行個體，但他們的帳戶中擁有的專用執行個體上沒有容量，則該執行個體會自動啟動到共用的專用執行個體。

若要分享專用執行個體，您必須將它新增至資源共享。資源共用是一 AWS RAM 種可讓您跨 AWS 帳號共用資源的資源。資源共享指定要分享的資源，以及共用它們的消費者。您可以將專用執行個體新增至現有的資源，或將其新增至新的資源共享。

如果您是組織的一員， AWS Organizations 且已啟用組織內的共用功能，則組織中的取用者會自動獲得共用專用主機的存取權。否則，消費者會收到加入資源共享的邀請，並且在接受邀請後便能存取共用的專用執行個體。

> **Note**
>
> 分享專用執行個體後，消費者可能需要幾分鐘的時間才能存取它。

您可以使用下列其中一種方法，來共用您擁有的專用執行個體。

Amazon EC2 console

使用 Amazon 主EC2控台共用您擁有的專用主機

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 專用主機。
3. 選擇專用執行個體來共用，然後選擇 Actions (動作) > Share host (共用主機)。
4. 選取資源共享以新增專用執行個體，然後選取 Share host (共用主機)。

   可能需要經過幾分鐘，消費者才能存取共用的主機。

AWS RAM console

共用您使用主控台擁有的專用 AWS RAM 主機

請參閱《AWS RAM 使用者指南》中的建立資源共享。

AWS CLI

共用您使用的專用主機 AWS CLI

使用指create-resource-share令。

取消共用與其 AWS 他帳戶共用的專用主機

專用執行個體擁有者可以隨時取消共享已共用的專用執行個體。當您取消共享已共用的專用執行個體時，請注意以下規則：

- 與其共用專用執行個體的消費者無法再於其上啟動新的執行個體。

- 取消共用時在專用執行個體上執行中的消費者所擁有的執行個體會繼續執行，但排定進行淘汰。消費者會收到執行個體的淘汰通知，而且他們有兩週的時間可對通知採取行動。但是，如果在淘汰通知期間內與消費者重新共用專用執行個體，則會取消執行個體淘汰。

若要取消共享您擁有的已共用專用執行個體，您必須從資源共享中移除它。您可以使用下列其中一種方法來執行這項作業。

Amazon EC2 console

使用 Amazon EC2 主控台取消共用您擁有的共用專用主機

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 專用主機。
3. 選擇要取消共享的專用執行個體，然後選擇 Sharing (共用) 索引標籤。
4. Sharing (共用) 索引標籤列出已新增專用執行個體的資源共享。選取要從中移除專用執行個體的資源共享，然後選取 Remove host from resource share (從資源共享中移除主機)。

AWS RAM console

使用 AWS RAM 主控台來取消共用您擁有的已共用專用執行個體

請參閱 AWS RAM 使用者指南中的更新資源共享。

Command line

若要取消共用您所擁有的共用專用主機 AWS CLI

使用指disassociate-resource-share令。

在您的 AWS 帳戶中查看共享的 Amazon EC2 專用主機

您可以檢視與其他帳戶共用的專用主機，以及與您共用的專用主機。如果您擁有專用主機，則可以查看主機上執行的所有執行個體，包括取用者啟動的執行個體。如果專用主機與您共用，您只能看到您在共用主機上啟動的執行個體，而不能看到其他用戶啟動的執行個體。

擁有者和消費者可以使用下列其中一種方法，來識別共用的專用執行個體。

Amazon EC2 console

使用 Amazon 主EC2控台識別共用專用主機

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 專用主機。畫面會列出您擁有的專用執行個體和已分享給您的專用執行個體。Owner (擁有者) 資料欄會顯示專用執行個體擁有者的 AWS 帳戶 ID。若要檢視主機上執行的執行處理，請選取執行處理頁籤。

Command line

使用識別共用專用主機 AWS CLI

使用 describe-hosts 命令。此命令會傳回您擁有的專用執行個體且和已分享給您的專用執行個體。

Amazon EC2 專用主機 AWS Outposts

AWS Outposts 是一項完全受控的服務，可將 AWS 基礎架構APIs、服務和工具延伸至您的場所。透過提供 AWS 受管理基礎結構的本機存取權， AWS Outposts 可讓您使用與 AWS 區域相同的程式設計介面在內部部署建置和執行應用程式，同時使用本機運算和儲存資源來降低延遲和本機資料處理需求。

Outpost 是部署在客戶站點的 AWS 計算和儲存容量集區。 AWS 作為 AWS 區域的一部分來操作、監控和管理此容量。

您可以在帳戶中擁有的 Outpost 上配置專用執行個體。這讓您可以更輕鬆地將需要專用實體伺服器的現有軟體授權和工作負載帶至 AWS Outposts。您也可以鎖定 Outpost 上的特定硬體資產，協助將工作負載之間的延遲降到最低。

專用主機可讓您在 Amazon 上使用合格的軟體授權EC2，從而獲得使用自有授權的靈活性和成本效益。綁定至虛擬機器、通訊端或實體核心的其他軟體授權也可以在專用執行個體上使用，但需遵守其授權條款。雖然 Outposts 一直都是符合BYOL工作負載資格的單一租用戶環境，但專用主機可讓您將需要的授權限制在單一主機，而不是整個 Outpost 部署。

此外，在 Outpost 上使用專用執行個體可為您在執行個體類型部署方面提供更大的靈活性，並對執行個體放置進行更精細的控制。您可以針對特定主機進行執行個體啟動，並使用主機親和性來確保執行個體始終在該主機上執行，您也可以使用自動放置，在具有相符組態和可用容量的任何可用主機上啟動執行個體。

內容

- 必要條件
- 支援的功能
- 考量事項
- 在上分配一個 Amazon EC2 專用主機 AWS Outposts

必要條件

您的站點必須安裝 Outpost。如需詳細資訊，請參閱 AWS Outposts 使用者指南中的建立 Outpost 並訂購 Outpost 容量。

支援的功能

- 支援以下執行個體系列：C5、M5、R5、C5d、M5d、R5d、G4dn 和 i3en。
- Outpost 上的專用執行個體可設定為支援多種執行個體規模。下列執行個體系列可支援多個執行個體類型：C5、M5、R5、C5d、M5d 和 R5d。如需詳細資訊，請參閱Amazon EC2 專用主機執行個體容量組態。
- Outpost 上的專用執行個體支援自動放置和鎖定執行個體啟動。如需詳細資訊，請參閱Amazon EC2 專用主機自動放置和主機親和性。
- Outpost 上的專用執行個體支援主機親和性。如需詳細資訊，請參閱Amazon EC2 專用主機自動放置和主機親和性。
- Outposts 上的專用主機支持與 AWS RAM共享。如需詳細資訊，請參閱跨帳戶 Amazon EC2 專用主機共享。

考量事項

- Outpost 上不支援專用執行個體保留功能。
- Outposts 不支援主機資源群組和 AWS License Manager 主機資源群組。
- Outpost 上的專用執行個體不支援爆量 T3 執行個體。
- Outpost 上的專用執行個體不支援主機復原。
- 在 Outposts 上使用專用主機租用的執行個體不支援簡化的自動復原。

在上分配一個 Amazon EC2 專用主機 AWS Outposts

您在 Outpost 上配置和使用專用執行個體的方式與在 AWS 區域中使用專用執行個體的方式相同。

必要條件

在 Outpost 上建立子網。如需詳細資訊，請參閱 AWS Outposts 使用者指南中的建立子網。

若要在 Outpost 中配置專用執行個體，請使用下列其中一種方法：

AWS Outposts console

1. 在開啟 AWS Outposts 主控台https://console.aws.amazon.com/outposts/。
2. 在導覽窗格中，選擇 Outposts。選取 Outpost，然後選取 Actions (動作)、Allocate Dedicated Host (配置專用執行個體)。
3. 視需設定專用執行個體。如需詳細資訊，請參閱分配 Amazon EC2 專用主機以在您的帳戶中使用。

   > **Note**
   > 可用區域和前哨ARN應預先填入可用區域和ARN所選前哨。

4. 選擇 Allocate (配置)。

Amazon EC2 console

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Dedicated Hosts (專用執行個體)，然後選擇 Allocate Dedicated Host (配置專用執行個體)。
3. 對於 Availability Zone (可用區域)，選取與 Outpost 關聯的可用區域。
4. 對於「前哨」ARN，輸入前哨ARN的。
5. 若要鎖定 Outpost 上的特定硬體資產，請選擇 啟用 鎖定 Outpost 上的特定硬體資產。針對要鎖定的每項硬體資產，請選擇 新增資產 ID，然後輸入硬體資產的 ID。

   > **Note**
   > 您為「數量」指定的值必須等於您指定IDs的資產數目。例如，如果您指定 3 個資產 IDs，則「數量」也必須為 3。

6. 視需進行其餘的專用執行個體設定。如需詳細資訊，請參閱分配 Amazon EC2 專用主機以在您的帳戶中使用。
7. 選擇 Allocate (配置)。

AWS CLI

使用分配主機命令 AWS CLI 。對於 `--availability-zone`，指定與 Outpost 關聯的可用區域。對於`--outpost-arn`，指定「前哨」ARN 的。(可選) 對`--asset-ids`，指定要鎖定目標IDs的 Outpost 硬體資產。

```
aws ec2 allocate-hosts --availability-zone "us-east-1a" --outpost-arn
 "arn:aws:outposts:us-east-1a:111122223333:outpost/op-4fe3dc21baEXAMPLE" --asset-
ids asset_id --instance-family "m5" --auto-placement "off" --quantity 1
```

將執行個體啟動至 Outpost 上的專用執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 專用主機。選取您在上一步驟中配置的專用執行個體，然後選取 Actions (動作)、Launch instance onto host (在主機上啟動執行個體)。
3. 根據需要設定執行個體，然後啟動執行個體。如需詳細資訊，請參閱在 EC2 Amazon EC2 專用主機上啟動 Amazon 實例。

Amazon EC2 專用主機恢復

如果在您的專用執行個體上偵測到特定問題狀況，專用執行個體自動復原會自動將您的執行個體重新啟動到新的替代主機。若有關於系統電源或網路連線事件的非預期專用執行個體失敗，主機復原可減少手動介入的需要，並降低操作負擔。其他專用執行個體問題將需要手動介入才能從中復原。

目錄

- Amazon EC2 專用主機如何工作
- 支援的執行個體類型
- 定價
- 管理 Amazon EC2 專用主機恢復
- 檢視 Amazon EC2 專用主機的主機復原設定
- 手動復原 Amazon EC2 專用主機復原不支援的執行個體

Amazon EC2 專用主機如何工作

專用執行個體和主機資源群組復原程序會使用主機層級運作狀態檢查，來存取專用執行個體可用性，以及偵測基礎系統失敗。專用執行個體失敗的類型決定專用執行個體自動復原是否可行。導致主機層級運作狀態檢查失敗的問題範例包括：

- 網路連線中斷
- 系統電力中斷
- 實體主機上的硬體或軟體問題

> **Important**
>
> 主機排定為淘汰時，不會發生專用執行個體自動復原。

專用執行個體自動復原

當您的專用主機上偵測到系統電源或網路連線失敗時，會啟 auto 專用主機EC2自動復原，Amazon 會在與原始專用主機相同的可用區域中自動配置替換專用主機。替代的 專用執行個體 會收到新的主機 ID，但保留與原始 專用執行個體 相同的屬性，包括：

- 可用區域
- 執行個體類型
- 標籤
- 自動配置設定
- 保留

當配置替代的專用執行個體時，會將執行個體復原至替代的專用執行個體。復原的執行個體會保留與原始執行個體相同的屬性，包括：

- 執行個體 ID
- 私有 IP 地址
- 彈性 IP 地址
- EBS磁碟區附件
- 所有執行個體中繼資料

此外，與 L AWS icense Manager 的內置集成可以自動跟踪和管理許可證。

> **Note**
>
> AWS 僅在可使用 License Manager 的地區支援 L AWS icense Manager 整合。

如果執行個體與受損的 專用執行個體 具有主機親和性關係，復原的執行個體會與替代的 專用執行個體 建立主機親和性。

當所有執行個體都已復原至替代的 專用執行個體 上時，受損的 專用執行個體 會被釋出，並且替代的 專用執行個體 會變得可供使用。

啟動主機復原時，會透過電子郵件和 AWS Health Dashboard 事件通知 AWS 帳戶擁有者。在主機復原已成功完成時，會傳送第二個通知。

如果您使用 L AWS icense Manager 來追蹤授權，則 Lic AWS ense Manager 會根據授權組態限制為替換的專用主機配置新的授權。如果授權組態具有因主機復原而違反的硬性限制，則不允許復原程序，且會透過 Amazon SNS 通知 (如果已為 L AWS icense Manager 設定通知設定) 通知您主機復原失敗。如果授權組態具有軟限制，而這些限制會因為主機復原而違反，則允許復原繼續，並透過 Amazon SNS 通知通知通知您有關違反限制的通知。如需詳細資訊，請參閱 AWS License Manager 使用者指南中的使用授權組態和 License Manager 中的設定。

主機復原狀態

偵測到專用執行個體失敗時，受損的專用執行個體會進入 `under-assessment` 狀態，並且所有執行個體會進入 `impaired` 狀態。當受損的 專用執行個體 處於 `under-assessment` 狀態時，您無法對其啟動執行個體。

配置替代的 專用執行個體 之後，它會進入 `pending` 狀態。在主機復原程序完成之前，它會保持在此狀態。當替代的 專用執行個體 處於 `pending` 狀態時，您無法對其啟動執行個體。在復原程序期間，替代的 專用執行個體 上復原的執行個體會維持為 `impaired` 狀態。

在主機復原完成之後，替代的 專用執行個體 會進入 `available` 狀態，而復原的執行個體會回到 `running` 狀態。當替代的 專用執行個體 進入 `available` 狀態時，您可以對其啟動執行個體。原始受損的 專用執行個體 會永久釋出並且進入 `released-permanent-failure` 狀態。

如果受損的 專用執行個體 具有不支援主機復原的執行個體，例如具備執行個體存放區磁碟區的執行個體，則不會釋出 專用執行個體。而是會將它標記為供淘汰，並進入 `permanent-failure` 狀態。

沒有專用執行個體自動復原的案例

主機排定為淘汰時，不會發生專用執行個體自動復原。您將在 Amazon CloudWatch 事件中收到退休通知 AWS Health Dashboard， AWS 帳戶擁有者電子郵件地址會收到有關專用主機故障的訊息。請在指定的時間段內遵循淘汰通知中所述的補救步驟，以手動復原淘汰主機中的執行個體。

已停止的執行個體不會復原到替代的 專用執行個體 上。如果您嘗試啟動以受損的 專用執行個體 為目標的停止的執行個體，該執行個體啟動會失敗。建議您將已停止的執行個體修改為以不同的 專用執行個體 為目標，或是在具有相符組態並且已啟用自動配置的任何可用的 專用執行個體 上啟動。

具有執行個體儲存體的執行個體不會復原到替代的專用執行個體上。作為補救措施，會將受損的 專用執行個體 標示為供淘汰，並且您會在主機復原完成之後收到淘汰通知。請在指定的時間期間內遵循淘汰通知中所述的補救步驟，以手動復原受損的 專用執行個體 上其餘的執行個體。

支援的執行個體類型

下列執行個體系列支援主機復原：A1、C3、C4、C5、C5n、C6a、C6g、C6i、資訊 1、G3、G5G、M4、M5、M5、M5n、M5zn、M6a、M6g、M6i、P2、M4、R5B、R54、R5B N, R 6 克, R 6 I, T3, X1, X1, X1e, X2iezn, u-6 TB1, U-九 TB1, U-十二 TB1, U-十八泰比 1, 和 U-24 泰比 1.

若要復原不受支援的執行個體，請參閱手動復原 Amazon EC2 專用主機復原不支援的執行個體。

> **Note**
>
> 支援的金屬執行個體類型的專用執行個體自動復原將需要比非金屬執行個體類型更長的時間來偵測和復原。

定價

使用主機復原沒有額外的費用，但適用一般的 專用執行個體 費用。如需詳細資訊，請參閱 Amazon EC2 專用主機定價。

一旦初始化主機復原，將不再就受損的 專用執行個體 向您收費。替代專用執行個體的計費方式只會在進入 `available` 狀態之後開始。

如果受損的 專用執行個體 是使用隨需費率計費，則替代的 專用執行個體 也會使用隨需費率計費。如果受損的 專用執行個體 有作用中 專用執行個體預留，則會傳輸到替代的 專用執行個體。

管理 Amazon EC2 專用主機恢復

如果在您的專用執行個體上偵測到特定問題狀況，專用執行個體自動復原會自動將您的執行個體重新啟動到新的替代主機。您可以在配置專用主機時或配置後啟用主機復原。

配置主機時，請使用下列程序啟用主機復原。

Console

若要在配置時啟用主機復原

使用 Amazon 主EC2控台配置專用主機時，對於主機復原，請選擇啟用。如需詳細資訊，請參閱分配 Amazon EC2 專用主機以在您的帳戶中使用。

AWS CLI

若要在配置時啟用主機復原

使用分配主機命令並指定參數。`host-recovery`

```
$ aws ec2 allocate-hosts \
    --instance-type m5.large \
    --availability-zone eu-west-1a \
    --auto-placement on \
    --host-recovery on \
    --quantity 1
```

使用下列程序來管理專用主機的主機復原。

Console

管理配置後的主機復原

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 專用主機。
3. 選取專用主機。
4. 選擇動作、修改主機。
5. 對於主機復原，請選取或清除啟用。
6. 選擇 Save (儲存)。

AWS CLI

若要在配置後啟用主機復原

使用「修改主機」指令，並指定值為的`host-recovery`參數。`on`

```
$ aws ec2 modify-hosts --host-recovery on --host-ids h-012a3456b7890cdef
```

若要在配置後停用主機復原

使用「修改主機」指令，並指定值為的`host-recovery`參數。`off`

```
$ aws ec2 modify-hosts --host-recovery off --host-ids h-012a3456b7890cdef
```

檢視 Amazon EC2 專用主機的主機復原設定

您可以隨時檢視 專用執行個體 的主機復原組態。

使用主控台檢視 專用執行個體 的主機復原組態

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 專用主機。
3. 選取 專用執行個體，並在 Description (描述) 索引標籤中，檢閱 Host Recovery (主機復原) 欄位。

使用 AWS CLI檢視專用執行個體的主機復原組態

使用 describe-hosts 命令。

```
$ aws ec2 describe-hosts --host-ids h-012a3456b7890cdef
```

`HostRecovery` 回應元素指出主機復原是否已啟用或已停用。

手動復原 Amazon EC2 專用主機復原不支援的執行個體

主機復原不支援復原使用執行個體存放區磁碟區的執行個體。遵循以下的指示來手動復原您無法自動復原的任何執行個體。

> **Warning**
>
> 位在執行個體存放區磁碟區的資料都會在執行個體停止、休眠或終止時遺失。這包括連接到具有磁碟區做為根裝置的執行個體的執行個體儲存磁EBS碟區。若要保護執行個體存放區磁碟區的資料，請在執行個體停止或終止之前，將資料備份至持久性儲存。

手動復原EBS支援的執行個體

對於無法自動復原的已EBS支援執行個體，建議您手動停止並啟動執行個體，以將其復原到新的專用主機上。如需停止您執行個體的詳細資訊，以及在停止時您執行個體組態中發生的變更，請參閱停止並啟動 Amazon EC2 執行個體。

手動復原執行個體存放區後端執行個體

對於無法自動復原的執行個體存放區後端執行個體，建議您執行下列：

1. 在最新的專用主機上啟動替代執行個體AMI。
2. 將所有必要資料遷移至替代的執行個體。
3. 終止受損的 專用執行個體 上的原始執行個體。

Amazon EC2 專用主機的主機維護

透過主機維護，降級專用主機上的 Amazon EC2 執行個體會在排定的維護事件期間，在替換的專用主機上自動重新啟動。這有助於減少應用程式停機時間，並將無差異化的繁重維護工作卸載到 AWS。主機維護也會針對規劃和例行 Amazon EC2 維護執行。

透過 Amazon 主EC2控台進行的所有新專用主機分配均支援主機維護。對於您 AWS 帳戶 或透過配置的任何新專用主機中的任何專用主機 AllocateHostsAPI，您可以為支援的專用主機設定主機維護。如需詳細資訊，請參閱the section called “設定主機維護”。

目錄

- 主機維護與主機復原
- 支援的執行個體類型
- 限制
- 相關服務
- 定價
- Amazon EC2 專用主機的主機維護如何運作

- 設定 Amazon EC2 專用主機的主機維護設定

主機維護與主機復原

下表顯示主機復原與主機維護之間的主要差異。

	主機復原	主機維護
可存取性	無法連接	可連接
State	`under-assessment`	`permanent-failure`
動作	立即復原	已排定維護
排程彈性	無法重新排程	可重新排程
主機資源群組	支援	不支援

如需有關主機復原的詳細資訊，請參閱主機復原。

支援的執行個體類型

下列執行個體系列支援主機維護：

- 一般用途：`A1` | `M4` | `M5` | `M5a` | `M5n` | `M5zn` | `M6a` | `M6g` | `M6i` | `M6in` | `M7a` | `M7g` | `M7i` | `T3`
- 運算最佳化：`C4` | `C5` | `C5a` | `C5n` | `C6a` | `C6g` | `C6gn` | `C6i` | `C6in` | `C7g` | `C7gn` | `C7i`
- 記憶體最佳化：`R4` | `R5` | `R5a` | `R5b` | `R5n` | `R6a` | `R6g` | `R6i` | `R6in` | `R7a` | `R7g` | `R7iz` | `u-12tb1` | `u-18tb1` | `u-24tb1` | `u-3tb1` | `u-6tb1` | `u-9tb1` | `X2iezn`
- 加速運算：`G3` | `G5g` | `Inf1` | `P2` | `P3`

限制

- Local Zones 和 AWS Wavelength AWS 區不支援主機維護。 AWS Outposts
- 對於已在主機資源群組中的主機，無法開啟或關閉主機維護。新增至主機資源群組的主機會保留其主機維護設定。如需詳細資訊，請參閱主機資源群組。
- 只有特定執行個體類型才支援主機維護。如需詳細資訊，請參閱the section called “支援的執行個體類型”。

相關服務

專用主機與 Lic AWS ense Manager 整合 — 跨 Amazon EC2 專用主機追蹤授權 (僅在提供 L AWS icense Manager 的區域支援)。如需詳細資訊，請參閱 AWS License Manager 使用者指南。

您必須在您的新專用主 AWS 帳戶 機中擁有足夠的授權。當主機在排定的維護事件完成後釋放時，會釋放與降級主機相關聯的授權。

定價

使用主機維護沒有額外的費用，但通常收取專用執行個體費用。如需詳細資訊，請參閱 Amazon EC2 專用主機定價。

啟動主機維護之後，將不再為降級專用執行個體付費。替代專用執行個體的計費方式只會在進入 `available` 狀態之後開始。

如果降級的專用主機是使用隨需費率計費，則更換的專用主機也會以隨需費率計費。如果降級的專用執行個體有作用中專用執行個體預留，則會傳輸到新專用執行個體。

Amazon EC2 專用主機的主機維護如何運作

在專用執行個體上偵測到降級時，會配置新的專用執行個體。降級可能是由於基礎硬體降級或偵測到某些有問題的情況而引起。降級的專用主機上的執行個體會排程在替換的專用主機上自動重新開機。

替代的專用執行個體會收到新的主機 ID，但保留與原始專用執行個體相同的屬性。這些屬性包含下列項目。

- 自動配置設定
- 可用區域
- 保留
- 主機親和性
- 主機維護設定
- 主機復原設定
- 執行個體類型
- 標籤

所有受支援的專用主 AWS 區域 機均可進行主機維護。如需不支援主機維護之專用執行個體的詳細資訊，請參閱 the section called “限制”。

在所有執行個體重新啟動到新的專用執行個體或停止後，會釋放降級的專用執行個體。您可以在排定的維護事件發生之前存取降級之專用執行個體上的執行個體，但不支援在降級的專用執行個體上啟動執行個體。

在排定的維護事件發生之前，您可以使用取代的專用主機在主機上啟動新執行個體。不過，替換主機上的某些執行個體容量會保留給必須從降級主機移轉的執行個體使用。您無法在此保留容量中啟動新執行個體。如需詳細資訊，請參閱the section called “專用執行個體上的執行個體”。

專用執行個體上的執行個體

Amazon 會為將從降級主機EC2自動遷移的執行個體自動保留替換主機上的容量。Amazon EC2 不會為無法自動遷移的執行個體 (例如具有執行個體存放根磁碟區的執行個體) 保留替代主機上的容量。保留容量無法用於啟動新執行個體。

> **Note**
>
> Amazon 主EC2控台會將保留容量顯示為已使用容量。可能看起來執行個體正在降級的主機和取代主機上執行。不過，執行個體只會在效能降級的主機上繼續執行，直到它們停止或移轉至取代主機上的保留容量為止。

如果您在降級的主機上手動停止可自動移轉的執行個體，則替換主機上為該執行個體保留的容量將會釋出並可供使用。

在排定的維護事件期間，降級主機上的執行個體會重新開機，並移轉至替換專用主機上的保留容量。遷移的執行個體會保留與降級主機上的相同屬性，包括下列項目。

- Amazon EBS 卷附件
- 彈性 IP 地址
- 執行個體 ID
- 執行個體中繼資料
- 私有 IP 地址

您可以在排程維護事件啟動之前的任何時間點，在降級的主機上停止和啟動執行個體。這樣做會將您的執行個體重新啟動到另一台主機，而您的執行個體將不會進行排程維護。您必須將執行個體的主機相似性更新至將要重新啟動執行個體的新主機。如果您在維護事件起始之前停止降級主機上的所有執行個體，降級的主機會被釋放，並取消維護事件。如需詳細資訊，請參閱停止並啟動 Amazon EC2 執行個體。

> **Note**
>
> 在停止和啟動執行個體時，不會保留任何本機儲存體磁碟區上的資料。

執行個體儲存體磁碟區為根裝置的執行個體會在指定的終止日期後終止。執行個體終止時，會刪除執行個體儲存體磁碟區上的任何資料。終止的執行個體會永久刪除，且無法再次啟動。對於執行個體儲存體磁碟區為根裝置的執行個體，建議使用最新的 Amazon Machine Image 在不同的專用執行個體上啟動替換執行個體，並在指定的終止日期之前將所有可用資料遷移到替換執行個體。如需詳細資訊，請參閱針對執行個體淘汰採取的動作。

無法重新啟動的執行個體會在指定日期之後自動停止。您可以在不同的主機上重新啟動這些執行個體。在新主機上啟動後，使用 Amazon EBS 磁碟區做為根裝置的執行個體會繼續使用相同的 Amazon EBS 磁碟區。

您可以重新排程執行個體重新啟動的開始時間，以設定執行個體重新啟動的https://console.aws.amazon.com/ec2/順序。

維護事件

偵測到降級時，會在 14 天後排程維護事件，以便在新的專用執行個體上重新啟動執行個體。您會收到電子郵件通知，其中提供有關降級的主機、排程的維護事件和維護時段的詳細資訊。如需詳細資訊，請參閱檢視排程事件。

可以將維護事件重新排程為排程事件日期後七天內的任何一天。如需有關重新排程的詳細資訊，請參閱對已排定事件進行重新排程。

維護事件通常需要幾分鐘來完成。在少數發生失敗事件的情況下，您會收到電子郵件通知，以便在指定時間範圍內移出降級主機上的執行個體。

主機維護狀態

偵測到降級時，專用執行個體會設定為 `permanent-failure` 狀態。您無法在狀態為 `permanent-failure` 的專用執行個體上啟動執行個體。維護事件完成後，會釋放降級主機並進入 `released, permanent-failure` 狀態。

在偵測到專用主機上的降級情況並排定維護事件之前，主機維護會自動在您的帳戶中配置替換的專用主機。此取代主機會保持pending狀態，直到排定維護事件為止。排定維護事件之後，取代的專用主機會移至該available狀態。

在排定的維護事件發生之前，您可以使用取代的專用主機在主機上啟動新執行個體。不過，替換主機上的某些執行個體容量會保留給必須從降級主機移轉的執行個體使用。您無法在此保留容量中啟動新執行個體。如需詳細資訊，請參閱the section called “專用執行個體上的執行個體”。

設定 Amazon EC2 專用主機的主機維護設定

您可以透過 AWS Management Console 或，為所有受支援的專用主機設定主機維護 AWS CLI。如需詳細資訊，請參閱下表。

AWS Management Console

若要使用啟用專用主機的主機維護 AWS Management Console。

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 專用主機。
3. 詳情「專用執行個體」 > 「動作」 > 「修改主機」。
4. 在主機維護欄位中選取開啟。

使用 AWS Management Console停用專用執行個體的主機維護。

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 專用主機。
3. 詳情「專用執行個體」 > 「動作」 > 「修改主機」。
4. 在主機維護欄位中選取關閉。

使用 AWS Management Console檢視專用執行個體的主機維護組態。

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 專用主機。
3. 選取「專用執行個體」，並在描述索引標籤中，檢閱主機維護欄位。

AWS CLI

使用 AWS CLI，在配置期間啟用或停用專用執行個體的主機維護。

使用 allocate-hosts 命令。

啟用

```
aws ec2 allocate-hosts --region us-east-1 --quantity 1 --instance-type m3.large --
availability-zone us-east-1b --host-maintenance on
```

停用

```
aws ec2 allocate-hosts --region us-east-1 --quantity 1 --instance-type m3.large --
availability-zone us-east-1b --host-maintenance off
```

使用 AWS CLI啟用或停用現有專用執行個體的主機維護。

使用 modify-hosts 命令。

啟用

```
aws ec2 modify-hosts --region us-east-1 --host-maintenance on --host-
ids h-0d123456bbf78910d
```

停用

```
aws ec2 modify-hosts --region us-east-1 --host-maintenance off --host-
ids h-0d123456bbf78910d
```

使用 AWS CLI檢視專用執行個體的主機維護組態。

使用 describe-hosts 命令。

```
aws ec2 describe-hosts --region us-east-1 --host-ids h-0d123456bbf78910d
```

> **Note**
>
> 如果停用主機維護，則會收到電子郵件通知，以移出降級主機，並在 28 天內將執行個體手動遷移到其他主機。如果您有專用執行個體預留，則會配置替代主機。28 天後，在降級主機上執行的執行個體將終止，並自動釋放主機。

監控 Amazon EC2 專用主機的狀態

Amazon 會EC2持續監控專用主機的狀態。更新是在 Amazon EC2 控制台上進行通訊。您可以使用下列方法，檢視專用執行個體的相關資訊。

Console

檢視專用執行個體的狀態

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 專用主機。
3. 在清單中找出專用執行個體，並檢閱 State (狀態) 欄的值。

AWS CLI

檢視專用執行個體的狀態

使用描述主機 AWS CLI 指令，然後檢閱回應元素中的`statehostSet`性質。

```
aws ec2 describe-hosts --host-id h-012a3456b7890cdef
```

PowerShell

檢視專用執行個體的狀態

使用指Get-EC2Host AWS Tools for Windows PowerShell 令，然後檢閱`hostSet`回應元素中的`state`性質。

```
PS C:\> Get-EC2Host -HostId h-012a3456b7890cdef
```

下表說明可能的專用執行個體狀態。

州	Description
`available`	AWS 沒有偵測到專用主機的問題。沒有排定的維護或修復。執行個體可在此專用執行個體上啟動。
`released`	專用執行個體已釋出。主機 ID 已不再使用。已釋出的主機即無法重複使用。
`under-assessment`	AWS 正在探索專用主機的可能問題。如果必須採取行動，系統會透過 AWS Management Console 或電子郵件通知您。執行個體無法在此狀態的專用執行個體上啟動。

州	Description
pending	專用執行個體 無法用於新的執行個體啟動。 已經修改為支援多個執行個體類型，或是主機復原正在進行中。
permanent-failure	已偵測到無法復原的失敗。您會透過執行個體和電子郵件收到移出通知。執行個體可繼續運作。如果您停止或終止具有此狀態的專用主機上的所有執行個體，則會 AWS 淘汰主機。 AWS 不會重新啟動處於此狀態的執行個體。執行個體無法在此狀態下於專用執行個體啟動。
released-permanent-failure	AWS 永久釋出失敗且不再具有執行中執行個體的專用主機。專用執行個體 ID 將無法再使用。

使用追蹤 Amazon EC2 專用主機組態變更 AWS Config

您可以用 AWS Config 來記錄專用主機的組態變更，以及在其上啟動、停止或終止的執行處理。接著，您就可以將 AWS Config 擷取的資訊做為授權回報的資料來源。

AWS Config 個別記錄專用主機和執行處理的組態資訊，並透過關係配對此資訊。有下列三種回報條件：

- AWS Config 錄製狀態 — 開啟時， AWS Config 正在記錄一或多個 AWS 資源類型，其中可包括專用主機和專用執行個體。為了擷取授權回報所需的資訊，請確認主機和執行個體的下列欄位都有記錄到。
- 主機記錄狀態— 為 Enabled (啟用) 時，會記錄專用執行個體的組態資訊。
- 執行個體記錄狀態 – 為 Enabled (啟用) 時，會記錄專用預留執行個體的組態資訊。

如果上述三個條件中有任一個停用，Edit Config Recording (編輯組態記錄) 按鈕的圖示就會是紅色。若要獲得這項工具的完整效益，請確認所有三個記錄方法均為啟用。所有三個項目均啟用時，圖示即為綠色。若要編輯設定，請選擇 Edit Config Recording (編輯組態記錄)。系統會將您導向至主 AWS Config 控台中的 [設定 AWS Config] 頁面，您可以在此頁面為主機、執行個體 AWS Config 和其他支援的資源類型設定並開始錄製。如需詳細資訊，請參閱AWS Config 開發人員指南中的 AWS Config 使用主控台進行設定。

> **Note**
>
> AWS Config 在發現資源後記錄您的資源，這可能需要幾分鐘的時間。

AWS Config 開始記錄主機和執行個體的組態變更後，您可以取得已配置或釋放之任何主機的組態歷史記錄，以及您已啟動、停止或終止的任何執行個體。例如，您可以在任何時間點於專用執行個體的組態歷史記錄中，查詢該主機上啟動了多少執行個體，以及該主機上的通訊端與核心數目。對於這些執行個體中的任何一個，您也可以查詢其 Amazon 機器映像 (AMI) 的 ID。您可以使用這些資訊來回報您自有伺服器固定的軟體授權 (依據每個通訊端或每個核心)。

您可以透過以下任何方式來檢視組態歷史記錄：

- 通過使用控 AWS Config 制台。對於每個記錄的資源，您可以查看時間軸頁面，其中提供組態詳細資訊的歷史記錄。若要查看此頁面，請選擇專用執行個體頁面中的 Config Timeline (組態時間軸) 欄內的灰色圖示。如需詳細資訊，請參閱AWS Config 開發人員指南中的檢視 AWS Config 主控台中的組態詳細資料。
- 通過運行 AWS CLI 命令。首先，您可以使用命list-discovered-resources令來取得所有主機和執行個體的清單。然後，您可以使用命get-resource-config-history令取得特定時間間隔內主機或執行個體的組態詳細資料。如需詳細資訊，請參閱AWS Config 開發人員指南CLI中的使用檢視組態詳細資料。
- 在您的應用程式 AWS Config API中使用。首先，您可以使用此ListDiscoveredResources動作來取得所有主機和執行個體的清單。然後，您可以使用此GetResourceConfigHistory動作取得特定時間間隔內主機或執行處理的組態詳細資訊。

例如，若要從中取得所有專用主機的清單 AWS Config，請執行如下CLI命令。

```
aws configservice list-discovered-resources --resource-type AWS::EC2::Host
```

若要從中取得專用主機的組態歷程記錄 AWS Config，請執行如下CLI命令。

```
aws configservice get-resource-config-history --resource-type AWS::EC2::Instance --
resource-id i-1234567890abcdef0
```

使用主控台管理 AWS Config 設定

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在 專用執行個體 頁面上，選擇 Edit Config Recording (編輯組態記錄)。

3. 在 AWS Config 主控台中，依照提供的步驟開啟錄製功能。如需詳細資訊，請參閱AWS Config 使用主控台進行設定。

如需詳細資訊，請參閱在AWS Config 主控台中檢視組態詳細資料。

若要 AWS Config 使用指令行啟動，或 API

- AWS CLI：檢視AWS Config 開發人員指南中的組態詳細資料 (AWS CLI)。
- Amazon EC2API：GetResourceConfigHistory。

Amazon EC2 專用實例

依預設，EC2執行個體會在共用租用硬體上執行。這表示多個 AWS 帳戶可能會共用相同的實體硬體。

專用執行個體是指在單一 AWS 帳戶專用的硬體上EC2執行的執行個體。這表示專用執行個體在主機硬體層級與屬於其他執行個體的執行個體實際隔離 AWS 帳戶，即使這些帳戶連結到單一付款人帳戶也一樣。不過，專用執行個體可能會與非專用執行個體的其他執行 AWS 帳戶 個體共用硬體。

專用執行個體不提供執行個體放置的可見性或控制權，也不支援主機相似性。如果您停止並啟動專用執行個體，則該執行個體可能無法在相同主機上執行。同樣地，您無法鎖定要啟動或執行執行個體的特定主機。此外，專用執行個體對自攜授權 (BYOL) 提供有限的支援。

如果您需要對執行個體放置的能見度和控制權，以及更全面的BYOL支援，請考慮改用專用主機。專用執行個體和專用主機都可以在專用實體伺服器上啟動 Amazon EC2 執行個體。專用執行個體和專用執行個體上的執行個體之間沒有效能、安全或實體差異。但是，它們之間存在一些關鍵差異。下表特別列出了專用執行個體與專用執行個體之間的主要差異：

	Dedicated Host	Dedicated Instance
專屬實體伺服器	實體伺服器具有完全專供您使用的執行個體容量。	專用於單一客戶帳戶的實體伺服器。
實例容量共享	可與其他帳戶共用執行個體容量。	不支援
計費	按每台主機計費	按每個執行個體計費
通訊端、核心和主機 ID 的可見度	提供通訊端與實體核心數目的可見度	沒有可見度

	Dedicated Host	Dedicated Instance
主機和執行個體親和性	可讓您隨時間一致地將執行個體部署到相同的實體伺服器	不支援
訂定執行個體配置目標	對於執行個體在實體伺服器上的配置方式，提供額外可見性和控制權	不支援
自動執行個體復原功能	支援。如需詳細資訊，請參閱 Amazon EC2 專用主機恢復。	支援
攜帶您自己的授權 (BYOL)	支援	部分支援 *
Capacity Reservations	不支援	支援

* 透過軟體保證提供授權行動性的 Microsoft SQL 伺服器，以及 Windows 虛擬桌面存取 (VDA) 授權可搭配專用執行個體使用。

如需專用預留執行個體的詳細資訊，請參閱 Amazon EC2 專用主機。

主題

- 專用執行個體 基本概念
- 支援的功能
- 專用執行個體 限制
- 專用執行個體定價
- 使用預設租用將專用執行VPC個體啟動到
- 變更 Amazon EC2 執行個體的租用
- 變更執行個體租用 VPC

專用執行個體 基本概念

A VPC 可以擁有`default`或的租賃`dedicated`。根據預設，VPCs您的`default`租賃和啟動到租賃中的執行個`default`體VPC擁有`default`租賃。可以執行下列作業來啟動專用執行個體：

- VPC使用的租用建立dedicated，讓所有執VPC行個體都以專用執行個體的形式執行。如需詳細資訊，請參閱使用預設租用將專用執行VPC個體啟動到。
- VPC使用的租用建立，default並手動指定租用，讓執行個體dedicated做為專用執行個體執行個體執行。如需詳細資訊，請參閱使用預設租用將專用執行VPC個體啟動到。

支援的功能

專用執行個體支援下列功能和 AWS 服務整合：

主題

- 預留執行個體
- 自動調整規模
- 自動復原
- 專用的 競價型執行個體
- 爆量效能執行個體

預留執行個體

若要保留專用執行個體的容量，您可以購買專用預留執行個體或容量保留。如需詳細資訊，請參閱 Amazon 預留執行個體EC2概觀 和 透過隨需容量保留來保留運算容量。

購買專用預留執行個體時，您購買的是以較低的使用費用將專用執行個體啟動到專用執行個VPC體的容量；只有當您啟動具有專用租用的執行個體時，才會收取使用費中的多量折扣價格。當您購買搭配預設租用的預留執行個體時，它僅適用於搭配 default 租用的運作中執行個體，不適用於搭配 dedicated 租用的運作中執行個體。

購買後，您就無法使用修改程序變更預留執行個體的租用。但您可以交換可轉換預留執行個體和具有不同租用的新可轉換預留執行個體。

自動調整規模

您可以使用 Amazon EC2 Auto Scaling 來啟動專用執行個體。如需詳細資訊，請參閱 Amazon Auto Scaling 使用者指南VPC中的啟動 EC2 Auto Scaling 執行個體。

自動復原

如果專用執行個體因基礎硬體故障或需要 AWS 參與修復的問題而受損，您可以為專用執行個體設定自動復原。如需詳細資訊，請參閱執行個體恢復。

專用的 競價型執行個體

您可以在建立 Spot 執行個體請求時，藉由指定 `dedicated` 的租用來執行專用 Spot 執行個體。如需詳細資訊，請參閱 在單租戶硬體上啟動。

爆量效能執行個體

您可以將在專用租用硬體上執行的好處運用於 the section called “爆量效能執行個體”。T3 專用執行個體預設會以無限制模式啟動，並提供基準CPU效能等級，並能在工作負載需要時提升至更高CPU層級。T3 基準性能和爆發能力由CPU積分控制。由於 T3 執行個體類型的高載本質，我們建議您監控 T3 執行個體如何使用專用硬體的CPU資源，以獲得最佳效能。T3 專用執行個體適用於具有顯示隨機CPU行為的各種工作負載，但理想情況下，平均CPU使用量達到或低於基準使用量的客戶。如需詳細資訊，請參閱the section called “重要概念”。

Amazon EC2 已設有系統來識別和修正效能的變化。但是，如果您啟動多個具有相CPU關使用模式的 T3 專用執行個體，仍然可能會遇到短期變化。對於這些更高要求或相關的工作負載，我們建議使用 M5 或 M5a 專用執行個體，而不是 T3 專用執行個體。

專用執行個體 限制

使用專用執行個體時，請謹記以下事項：

- 執行個VPC體租用設定為時，不支援某些 AWS 服務或其功能`dedicated`。請參閱各自服務的文件以確認是否有任何限制。
- 在執行個體租VPC用設定為的情況下，無法將某些執行個體類型啟動到`dedicated`。如需支援執行個體類型的詳細資訊，請參閱 Amazon EC2 專用執行個體。
- 當您啟動 Amazon 支援的專用執行個體時EBS，該EBS磁碟區不會在單租用戶硬體上執行。

專用執行個體定價

專用執行個體定價與隨需執行個體定價不同。如需詳細資訊，請參閱 Amazon EC2 專用執行個體。

使用預設租用將專用執行VPC個體啟動到

建立時VPC，您可以選擇指定其執行個體租用。如果您將執行個體啟動到具有執行個體租用為的執行個體`dedicated`，則執行個體一律會在專供您使用的硬體上以專用執行個體的形式執行。VPC

如需有關建立VPC和選擇租用選項的詳細資訊，請參閱 Amazon VPC 使用者指南VPC中的建立。

您可以使用 Amazon 啟動執行個體精靈EC2啟動專用執行個體。

Console

使用主控台將專用執行個體啟動至預設租VPC用

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)、Launch Instance (啟動執行個體)。
3. 在「應用程式和作業系統映像」區段中，AMI從清單中選取。
4. 在 Instance type (執行個體類型) 區段中，選取要啟動的執行個體類型。

> **Note**
> 請務必選擇支援為專用執行個體的執行個體類型。如需詳細資訊，請參閱 Amazon EC2 專用執行個體。

5. 在 Key pair (金鑰對) 區段中，選取要與執行個體建立關聯的金鑰對。
6. 在 Advanced details (進階詳細資訊) 區段，對於 Tenancy (租用)，選取 Dedicated (專用)。
7. 視需要設定其餘的執行個體選項。如需詳細資訊，請參閱Amazon EC2 執行個體組態參數的參考。
8. 選擇啟動執行個體。

AWS CLI

若要在啟動期間設定執行個體的租用選項，請使用 AWS CLI

使用執行例證指令並包含`Tenancy`在選項中`--placement`。

PowerShell

若要在啟動期間使用工具設定執行個體的租用選項 PowerShell

將New-EC2Instance指令程式與`-Placement_Tenancy`參數搭配使用。

如需啟動具有 `host` 租用之執行個體的詳細資訊，請參閱在 EC2 Amazon EC2 專用主機上啟動 Amazon 實例。

變更 Amazon EC2 執行個體的租用

可在啟動後變更已停止的執行個體的租用。您所做的變更會在執行個體下次開始時生效。

執行個體的作業系統詳細資料以及是否已安裝 SQL Server，都會影響支援的轉換。如需執行個體可用租用轉換路徑的詳細資訊，請參閱《License Manager 使用指南》中的租用轉換。

或者，您也可以變更虛擬私有雲 (VPC) 的租用。如需詳細資訊，請參閱the section called “更改租約的租賃 VPC”。

限制

- 若使用 T3 執行個體，您必須在專用執行個體上啟動執行個體，才能使用 `host` 的租用。您無法將租用從 `host` 變更為 `dedicated` 或 `default`。嘗試進行上述任一不受支援的租用變更時，會顯示 `InvalidRequest` 錯誤代碼。

Console

使用主控台變更執行個體的租用

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選取 Instances (執行個體)，然後選取您的執行個體。
3. 選擇 Instance state (執行個體狀態)，Stop instance (停止執行個體)，Stop (停止)。
4. 選擇 Actions (動作)、Instance settings (執行個體設定)、Modify instance placement (修改執行個體的配置)。
5. 如果是 Tenancy (租用)，請選擇是否要在專用硬體還是 專用執行個體 上執行您的執行個體。選擇 Save (儲存)。

AWS CLI

若要使用修改執行個體的租用值 AWS CLI

使用指 modify-instance-placement令。

```
aws ec2 modify-instance-placement --instance-id i-1234567890abcdef0 --
tenancy dedicated
```

PowerShell

若要使用修改執行個體的租用值 AWS CLI

使用指Edit-EC2InstancePlacement令程式。

```
Edit-EC2InstancePlacement -InstanceId i-1234567890abcdef0 -Tenancy Dedicated
```

變更執行個體租用 VPC

您可以在建立虛擬私有雲端 (VPC) `default` 之後，`dedicated`將其執行個體租用從變更為。修改的執行個體租用VPC不會影響中任何現有執行個體的租用VPC。下次在中啟動執行個體時VPC，它的租用期為`default`，除非您在執行個體啟動期間另有指定。

或者，您也可以變更特定執行個體的租用。如需詳細資訊，請參閱the section called “變更執行個體租用”。

限制

- 建立`dedicated`之後，您無法將執行個體租`default`用VPC從變更為。
- 您無法使用修改的執行個體租VPC用 AWS Management Console 您可以使用 AWS CLI AWS SDK、或 Amazon 來修改它EC2API。

AWS CLI

若要修改使用的執行個體租VPC用屬性 AWS CLI

使用命modify-vpc-tenancy令並指定VPC和執行個體租用值的 ID。唯一支援的值為 `default`。

```
aws ec2 modify-vpc-tenancy --vpc-id vpc-1a2b3c4d --instance-tenancy default
```

PowerShell

若要修改使用工具的執行個體租VPC用屬性 PowerShell

使用指Edit-EC2VpcTenancy令程式並指定和執行個體租用VPC值的識別碼。唯一支援的值為 `Default`。

```
Edit-EC2VpcTenancy -VpcId vpc-1a2b3c4d -InstanceTenancy Default
```

ML 的隨需容量保留和容量區塊

容量保留可讓您為特定可用區域中的 Amazon EC2 執行個體保留運算容量。容量保留有兩種類型，用於不同的使用案例。

容量保留類型

- 隨需容量保留
- ML 的容量區塊

以下是隨需容量保留的一些常見使用案例：

- 擴展事件 – 在關鍵業務事件之前建立隨需容量保留，以確保您可以在需要時進行擴展。
- 法規要求和災難復原 – 使用隨需容量保留來滿足法規要求，以達到高可用性，並在不同可用區域或區域中保留容量以進行災難復原。

以下是 ML 容量區塊的一些常見使用案例：

- 機器學習 (ML) 模型訓練和微調 — 取得您保留的GPU執行個體不間斷存取，以完成 ML 模型訓練和微調。
- ML 實驗和原型 — 執行實驗並建立需要短時間GPU執行個體的原型。

使用隨需容量保留的時機

如果您有嚴格的容量需求，且正在執行需要容量保證的關鍵業務工作負載，請使用隨需容量保留。使用隨需容量保留，您可以確保在需要的時間內始終可以存取已保留的 Amazon EC2 容量。

使用 ML 容量區塊的時機

當您需要確保在 future 日期開始的定義時段內不間斷地存取GPU執行個體時，請使用 ML 的容量區塊。容量區塊適合訓練和微調 ML 模型、短期實驗執行，以及處理未來暫時提高的推論需求。使用容量區塊，您可以確保在特定日期存取GPU資源以執行機器學習工作負載。

透過隨需容量保留來保留運算容量

隨需容量保留可讓您在任何持續時間內為特定可用區域內的 Amazon EC2 執行個體保留運算容量。如果您有嚴格的容量需求，並且正在執行需要特定等級的長期或短期容量保證的關鍵業務工作負載，我們建議您建立EC2容量保留，以確保在需要時隨時都能存取 Amazon 容量。

您可以在任何時間建立容量保留，無須綁約一年或三年，且容量可立即生效。一旦帳戶佈建了容量保留，容量就可立即使用並開始計費。當您不再需要容量保證時，請取消容量保留以釋出容量並停止產生費用。您也可以使用 Savings Plans 和區域預留執行個體提供的帳單折扣來降低容量保留的成本。

建立 容量預留 時，您可以指定：

- 要預留容量的可用區域
- 要預留容量的執行個體數
- 執行個體屬性，包括執行個體類型、平台、可用區域和租用

容量預留 只能用於符合屬性的執行個體。根據預設，容量保留會自動比對具有相符屬性 (執行個體類型、平台、可用區域和租用) 的新執行個體和執行中執行個體。這代表任何屬性相符的執行個體會在 容量預留 中自動執行。不過，您也可以針對特定工作負載的容量預留。這可讓您明確控制允許哪些執行個體以該保留容量執行。

您可以指定保留的結束方式。您可以選擇取消容量預留或在指定的時間自動結束。如果指定結束時間，則容量預留會在指定時間的一小時內取消。例如，如果您指定 2019/5/31 13:30:55，則容量預留保證會在 2019/5/31 的 13:30:55 和 14:30:55 之間結束。

在保留結束後，您就不能再以針對容量預留的執行個體。使用預留容量執行的執行個體會持續不間斷執行。如果針對容量預留的執行個體停止，在移除其容量預留針對的偏好設定或設定它們針對不同的容量預留之前，都無法重新啟動它們。如需詳細資訊，請參閱修改執行個體的容量保留設定。

內容

- 容量預留、預留執行個體和 Savings Plans 之間的差異
- 支援的平台
- 配額
- 限制
- 容量預留定價和帳單
- 建立 容量預留
- 檢視容量保留的狀態
- 使用現有的 容量預留 啟動執行個體
- 修改作用中的容量保留
- 修改執行個體的容量保留設定
- 在容量保留之間移動容量
- 從現有的容量保留中分割可用產能
- 取消 容量預留
- 容量保留群組
- 在叢集置放群組中建立容量保留區

- Local Zones 中的容量保留
- 在 Wavelength 區域中的容量保留
- 容量保留 AWS Outposts
- 共用容量保留
- 容量預留機群
- 使用指標監視容量保留使用 CloudWatch 量
- 監視容量保留使用率 EventBridge
- 產能保留使用率通知來源 AWS Health

容量預留、預留執行個體和 Savings Plans 之間的差異

下表反白容量預留、預留執行個體和 Savings Plans 之間的主要差異：

	Capacity Reservations	區域預留執行個體	地區預留執行個體	Savings Plans
期間	不綁約。可視需要建立和取消。	需要綁約一年或三年		
容量優勢	在特定的可用區域中預留的容量。		無預留容量。	
帳單折扣	無帳單折扣。†	提供計費折扣。		
執行個體限制	適用每個區域的 隨需執行個體 限制。	每個可用區域預設為 20。您可以請求提高限制。	每個區域預設為 20。您可以請求提高限制。	沒有限制。

† 您可以將容量預留與 Savings Plans 或區域預留執行個體結合，以獲得折扣。

如需詳細資訊，請參閱下列內容：

- Amazon 預留執行個體EC2概觀
- Savings Plans 使用者指南

支援的平台

您必須使用正確的平台來建立容量預留，以確保它與您的執行個體正確相符。容量預留支援下列平台：

- Linux/ UNIX
- Linux 與SQL伺服器標準
- Linux 與網頁SQL伺服器
- Linux 與企業SQL伺服器
- SUSELinux
- Red Hat Enterprise Linux
- RHEL使用SQL伺服器標準
- RHEL使用企業SQL伺服器
- RHEL使用網頁SQL伺服器
- RHEL與醫管局
- RHEL具備 HA 和SQL伺服器標準
- RHEL與 HA 和SQL伺服器企業
- Ubuntu Pro
- Windows
- 帶SQL伺服器的視窗
- 帶有 Web SQL 服務器的窗口
- SQL伺服器標準視窗
- 視窗與企業SQL伺服器

當您購買 容量預留 時，您必須指定代表您執行個體作業系統的平台。

- 對於 SUSE RHEL Linux 和發行版，不包括在內BYOL，您必須選擇特定的平台。例如，SUSELinux 或紅帽企業版 Linux 平台。
- 對於所有其他 Linux 發行版（包括 Ubuntu），請選擇 Linux UNIX 平台。
- 如果您攜帶現有的RHEL訂閱 (BYOL)，則必須選擇 Linux/ UNIX 平台。
- 對於使用SQL標準的視窗、含SQL伺服器企業版的 Windows，以及含SQL伺服器 Web 的視窗，您必須選擇特定的平台。
- 對於所有其他 Windows 版本 (不包BYOL括不支援的版本)，請選擇 Windows 平台。

配額

您可以預留容量的執行個體數目是以您帳戶的隨需執行個體配額為基礎。您可以預留容量的執行個體數目，為該配額所允許的數目減去已在執行的執行個體數目。

配額僅適用於執行中的執行個體。如果執行個體處於待定中、停止中、已停止或已休眠狀態，則不會計入配額中。

限制

在您建立容量預留之前，請記下以下的限制與法規。

- 作用中和未使用的 容量預留 都計入 隨需執行個體 限制。
- 容量保留不能從一個 AWS 帳戶轉移到另一個帳戶。不過，您可以與其他 AWS 帳戶共用容量保留。如需詳細資訊，請參閱共用容量保留。
- 區域 預留執行個體 帳單折扣不適用於 容量預留。
- 集群放置群組中無法建立容量保留。不支援分散和分區置放群組。
- 容量預留 不能與 專用執行個體 搭配使用。容量預留不能與專用執行個體搭配使用。
- [Windows 執行個體] 容量保留不能與自攜授權 (BYOL) 搭配使用。
- 容量預留 不確保休眠執行個體可以在嘗試啟動後恢復。

容量預留定價和帳單

本節中的主題提供容量保留的定價和計費概觀。

主題

- 定價
- 帳單
- 帳單折扣
- 檢視您的帳單

定價

無論您是否以預留容量執行執行個體，都需要支付按隨需費率計算的容量保留費用。如果您不使用預訂，則在您的 Amazon EC2 帳單上顯示為未使用的預訂。當您執行符合保留屬性的執行個體時，您只需要支付執行個體的費用，不必支付保留費用。沒有前期或額外費用。

例如，如果您建立了 20 個 m4.large Linux 執行個體的 容量預留，在同一可用區域中執行了 15 個 m4.large Linux 執行個體，則您需要支付 15 個作用中執行個體和保留之 5 個未使用執行個體的費用。

Savings Plans 和區域預留執行個體的帳單折扣適用於容量預留。如需詳細資訊，請參閱帳單折扣。

如需詳細資訊，請參閱 Amazon EC2 定價。

帳單

一旦帳戶中佈建了容量，容量保留就會立即開始計費，並在帳戶容量保留保持佈建狀態時持續計費。

容量預留按秒計費。這表示不按小時計費。例如，如果您帳戶的容量保留作用了 24 小時又 15 分鐘，您需要支付 24.25 小時的保留費用。

以下範例示範容量預留的計費方式。您建立了一個 m4.large Linux 執行個體的 容量預留，其隨需費率為小時 0.10 USD。在此範例中，該帳戶的容量保留佈建了 5 個小時。第一個小時未使用 容量預留，所以未使用的這個小時按 m4.large 執行個體類型的標準隨需費率計費。在第二至第五個小時，m4.large 執行個體佔用了 容量預留。在這段期間，容量預留 不產生任何費用，但帳戶需要支付 m4.large 執行個體佔用它的費用。第六個小時取消了 容量預留，m4.large 執行個體在預留容量外正常執行。這個小時按 m4.large 執行個體類型的隨需費率收費。

Hour	1	2	3	4	5	6	Total cost
Unused Capacity Reservation	$0.10	$0.00	$0.00	$0.00	$0.00	$0.00	**$0.10**
On-demand Instance Usage	$0.00	$0.10	$0.10	$0.10	$0.10	$0.10	**$0.50**
Hourly cost	**$0.10**	**$0.10**	**$0.10**	**$0.10**	**$0.10**	**$0.10**	**$0.60**

帳單折扣

S avings Plans 和區域預留執行個體的帳單折扣適用於容量保留。 AWS 會自動將這些折扣套用至具有相符屬性的產能保留。當執行個體使用 容量預留，折扣會套用至該執行個體。在涵蓋未使用的 容量預留 之前，折扣會優先套用至執行個體用量。

用於區域 預留執行個體 的帳單折扣不適用於 容量預留。

如需詳細資訊，請參閱下列內容：

- Amazon 預留執行個體EC2概觀
- Savings Plans 使用者指南
- 帳單和購買選項

檢視您的帳單

您可以在 AWS Billing and Cost Management 主機上查看帳戶的費用和費用。

- Dashboard (儀表板) 會顯示您帳戶的支出摘要。
- 在 Bills (帳單) 頁面上，於 Details (詳細資訊) 下方，展開 Elastic Compute Cloud 區段及區域，以取得容量預留的帳單資訊。

您可以在線查看費用，也可以下載CSV文件。如需詳細資訊，請參閱 AWS Billing and Cost Management 使用者指南中的容量預留明細項目。

建立 容量預留

您可以建立容量保留，以確保在特定可用區域中具有可用的運算容量。如果您建立容量保留的請求成功，則可立即使用容量。只要容量預留在作用中，就有預留容量供您使用，而且您可以隨時使用它啟動執行個體。如果容量預留為開啟中，則具有相符屬性的新執行個體和現有執行個體都會使用容量預留的容量自動執行。如果容量預留為 `targeted`，則執行個體必須特別針對它使用預留容量執行。

如果符合以下條件之一，則您建立容量預留的請求可能會失敗：

- Amazon EC2 沒有足夠的容量來滿足請求。請稍後再試、嘗試其他可用區域，或是嘗試較小的請求。如果您的應用程式能因應各種執行個體類型和大小，請嘗試不同的執行個體屬性。
- 請求的數量超過所選執行個體系列的隨需執行個體限制。請提高執行個體系列的隨需執行個體限制，然後再試一次。如需詳細資訊，請參閱隨需執行個體配額。

使用主控台建立容量預留

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 選擇 容量預留，接著選擇 Create 容量預留 (建立 容量預留)。
3. 在 Create a 容量預留 (建立容量預留) 頁面的 Instance details (執行個體詳細資訊) 區段中設定以下設定。您啟動的執行個體類型、平台、可用區域和租用必須符合您在此處指定的執行個體類型、平台、可用區域和租用，否則不會套用容量保留。例如，如果開啟的 容量預留 不相符，則執行個體啟動 容量預留 明確目標將會失敗。

    a. Instance Type (執行個體類型)—使用預留容量啟動的執行個體類型。

    b. 啟動EBS最佳化執行個體 — 指定是否為最佳化執行個體保留容量。EBS根據預設，某些執行個體類型會選取此選項。如需詳細資訊，請參閱the section called “EBS最佳化”。

c. Platform (平台)—執行個體的作業系統。如需詳細資訊，請參閱支援的平台。

d. Availability Zone (可用區域)—要預留容量的可用區域。

e. Tenancy (租用)—指定是要執行共用硬體執行個體 (預設) 或是專用執行個體。

f. (選擇性) 放置群組 ARN — 要在其中建立容量保留ARN的叢集置放群組。

   如需詳細資訊，請參閱在叢集置放群組中建立容量保留區。

g. Quantity (數量) – 要預留容量的執行個體數目。如果您指定的數量超過所選執行個體類型的剩餘隨需執行個體限制，請求會被拒絕。

4. 在 Reservation details (保留詳細資訊) 區段中設定以下設定：

   a. Reservation Ends (保留結束) – 選擇下列其中一個選項：

      - Manually (手動) – 預留容量直到您明確取消為止。
      - Specific time (特定時間) – 在指定的日期和時間自動取消容量保留。

   b. Instance eligibility (執行個體資格) – 選擇下列其中一個選項：

      - open — (預設) 容量保留符合任何具有相符屬性 (執行個體類型、平台、可用區域和租用) 的執行個體。如果啟動具有相符屬性的執行個體，它會自動置放到預留容量。
      - target — 容量保留僅接受具有相符屬性 (執行個體類型、平台、可用區域和租用) 且明確鎖定保留項目的執行個體。

5. 選擇 Request reservation (請求保留)。

若要使用建立容量保留 AWS CLI

使用指create-capacity-reservation令。如需詳細資訊，請參閱支援的平台。

下列命令會建立一個容量保留，為AMIs在`us-east-1a`可用區域中`m5.2xlarge`執行 Red Hat Enterprise Linux 的三個執行個體保留容量。

```
aws ec2 create-capacity-reservation --instance-type m5.2xlarge --instance-platform Red
 Hat Enterprise Linux --availability-zone us-east-1a --instance-count 3
```

下列命令會建立容量保留，以保留三個`m5.2xlarge`執行 Windows 並在`us-east-1a`可用區域AMIs中執行 Windows 的SQL執行個體的容量保留。

```
aws ec2 create-capacity-reservation --instance-type m5.2xlarge --instance-
platform Windows with SQL Server --availability-zone us-east-1a --instance-count 3
```

檢視容量保留的狀態

Amazon 會EC2持續監控您的容量保留狀態。更新是在 Amazon EC2 控制台上進行通訊。您可以使用下列其中一種方法來檢視容量保留的相關資訊。

使用主控台檢視您的容量預留

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 選取 容量預留 (容量預留)，然後選取要檢視的容量預留。

若要使用檢視您的「容量保留」 AWS CLI

使用describe-capacity-reservations命令：

例如，下面的命令描述所有 容量預留。

```
aws ec2 describe-capacity-reservations
```

範例輸出。

```
{
    "CapacityReservations": [
        {
            "CapacityReservationId": "cr-1234abcd56EXAMPLE ",
            "EndDateType": "unlimited",
            "AvailabilityZone": "eu-west-1a",
            "InstanceMatchCriteria": "open",
            "Tags": [],
            "EphemeralStorage": false,
            "CreateDate": "2019-08-16T09:03:18.000Z",
            "AvailableInstanceCount": 1,
            "InstancePlatform": "Linux/UNIX",
            "TotalInstanceCount": 1,
            "State": "active",
            "Tenancy": "default",
            "EbsOptimized": true,
            "InstanceType": "a1.medium",
            "PlacementGroupArn": "arn:aws:ec2:us-east-1:123456789012:placement-group/
MyPG"
        },
        {
```

```
            "CapacityReservationId": "cr-abcdEXAMPLE9876ef ",
            "EndDateType": "unlimited",
            "AvailabilityZone": "eu-west-1a",
            "InstanceMatchCriteria": "open",
            "Tags": [],
            "EphemeralStorage": false,
            "CreateDate": "2019-08-07T11:34:19.000Z",
            "AvailableInstanceCount": 3,
            "InstancePlatform": "Linux/UNIX",
            "TotalInstanceCount": 3,
            "State": "cancelled",
            "Tenancy": "default",
            "EbsOptimized": true,
            "InstanceType": "m5.large"
        }
    ]
}
```

容量預留 擁有以下可能的狀態：

- `active`—容量可供使用。
- `expired` – 容量保留在保留請求指定的日期和時間自動過期。預留容量將不再提供使用。
- `cancelled` – 容量保留已取消。預留容量將不再提供使用。
- `pending` – 容量保留請求成功，但容量佈建仍待定。
- `failed` – 容量保留請求失敗。請求會因為請求參數無效、容量限制或執行個體限制而失敗。失敗的請求可供您檢視 60 分鐘。

> **Note**
>
> 由於 Amazon 之後採用最終一致性模型 EC2APIs，因此在您建立容量保留之後，主控台最多可能需要 5 分鐘的時間，而且describe-capacity-reservations回應表示容量保留處於`active`狀態。在這段期間，主控台和 `describe-capacity-reservations` 回應可能表示容量保留處於 `pending` 狀態。但是，容量保留可能已經可用，您可以嘗試在其中啟動執行個體。

使用現有的 容量預留 啟動執行個體

當您啟動執行個體時，您可以指定是否在任何 open 容量預留、特定 容量預留、或容量預留群組中啟動執行個體。您只能在具有相符屬性 (執行個體類型、平台、可用區域和租用) 和足夠容量的容量保留

中啟動執行個體。或者，您可以設定執行個體以避免在容量預留中執行，即使您的 open 容量預留 有相符屬性和可用容量。

使用容量預留啟動執行個體可依啟動的執行個體數量降低其可用容量。例如，如果啟動三個執行個體，則容量預留的可用容量會減少三個。

使用主控台以現有的容量預留啟動執行個體

1. 請遵循程序啟動執行個體，但在完成下列步驟以指定置放群組和容量保留的設定之前，請勿啟動執行個體。
2. 展開進階詳細資料並執行下列操作：
   a. 在「放置」群組中，選取要在其中啟動執行個體的叢集置放群組。
   b. 對於 Capacity Reservation (容量保留)，根據容量保留的組態選取下列其中一個選項：
      - 無 — 防止執行個體啟動到容量保留。執行個體以隨需容量執行。
      - 開啟 — 將執行個體啟動至任何具有相符屬性和足夠容量供您選取的執行個體數量使用的容量保留。如果沒有容量足夠的相符容量預留，執行個體將使用隨需容量。
      - 按 ID 鎖定目標 — 將執行個體啟動至選取的容量保留。如果選取的容量預留在您所選之執行個體數量沒有足夠的容量，執行個體啟動會失敗。
      - 按群組鎖定 — 使用所選容量保留群組中相符屬性和可用容量，將執行個體啟動至任何容量保留。如果所選群組沒有 容量預留 具有相符屬性和可用容量的，則執行個體會啟動至隨需容量。
3. 在 Summary (摘要) 面板中，檢閱您的執行個體組態，然後選擇 Launch instance (啟動執行個體)。如需詳細資訊，請參閱使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體。

使用將執行個體啟動至現有容量保留 AWS CLI

使用 run-instances 命令，並指定 `--capacity-reservation-specification` 參數。

以下範例會使用具有相符屬性和可用容量的任何開啟中容量預留，來啟動 `t2.micro` 執行個體：

```
aws ec2 run-instances --image-id ami-abc12345 --count 1 --instance-type t2.micro
 --key-name MyKeyPair --subnet-id subnet-1234567890abcdef1 --capacity-reservation-
specification CapacityReservationPreference=open
```

以下範例會使用 `targeted` 容量預留 啟動 `t2.micro` 執行個體：

```
aws ec2 run-instances --image-id ami-abc12345 --count 1 --instance-type t2.micro
 --key-name MyKeyPair --subnet-id subnet-1234567890abcdef1 --capacity-reservation-
specification CapacityReservationTarget={CapacityReservationId=cr-a1234567}
```

以下範例會在 容量預留 群組啟動 `t2.micro` 執行個體：

```
aws ec2 run-instances --image-id ami-abc12345 --count 1
 --instance-type t2.micro --key-name MyKeyPair --subnet-
id subnet-1234567890abcdef1 --capacity-reservation-specification
 CapacityReservationTarget={CapacityReservationResourceGroupArn=arn:aws:resource-
groups:us-west-1:123456789012:group/my-cr-group}
```

修改作用中的容量保留

如果您的現有容量保留不適合需要容量的工作負載，您可以修改執行個體數量、執行個體資格 (`open`或`targeted`) 和結束時間 (`At specific time`或`Manually`)。您無法在容量保留過期後或明確取消後修改。如果您指定的新執行個體數量超過所選執行個體類型的剩餘隨需執行個體限制，則更新會失敗。

您無法變更現有容量保留的執行個體類型、EBS最佳化、平台、可用區域或租用。如果您需要修改這些屬性的任何一項，建議您取消保留，然後建立具有所需屬性的新保留。

如果您透過將執行處理資格從變更為`targeted`來修改現有的容量保留`open`，則任何符合容量保留屬性、將`CapacityReservationPreference`參數設為且尚未在容量保留中執行的執行中執行處理，都會自動使用已修改的容量保留。`open`

> **Note**
>
> 若要變更執行個體資格，容量保留必須完全閒置 (零用量)，因為當執行個體在保留範圍內執行時，Amazon EC2 無法修改執行個體資格。

使用主控台修改容量預留

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 選取 容量預留 (容量預留)，選取要修改的容量預留，然後選取 Edit (編輯)。
3. 視需要修改總容量、容量保留結束或執行個體資格選項，然後選擇儲存。

若要使用修改容量保留 AWS CLI

使用指modify-capacity-reservation令。例如，下列命令會修改 容量預留 以保留八個執行個體的容量。

```
aws ec2 modify-capacity-reservation --capacity-reservation-id cr-1234567890abcdef0 --
instance-count 8
```

修改執行個體的容量保留設定

您可以隨時修改已停止執行個體的 容量預留 設定：

- 從任何具有相符屬性 (執行個體類型、平台、可用區域和租用) 和可用容量的容量保留開始。
- 在特定容量預留中啟動執行個體。
- 在 Capacity Reservation (容量預留) 群組內具有相符屬性和可用容量的任何 Capacity Reservation (容量預留) 中啟動
- 防止在容量預留中開始執行個體。

使用主控台修改執行個體的容量預留設定

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 選取 Instances (執行個體)，並選取要修改的執行個體。停止執行個體 (若還未停止的話)。
3. 選擇動作、執行個體設定、修改容量保留設定。
4. 針對 容量預留，請選擇下列其中一種選項：
   - Open (開啟) — 使用具有相符屬性以及針對所選執行個體數量具有足夠容量的容量預留來啟動執行個體。如果沒有容量足夠的相符容量預留，執行個體將使用隨需容量。
   - None (無) — 防止使用容量預留啟動執行個體。執行個體以隨需容量執行。
   - Specify Capacity Reservation (指定容量保留) – 在選取的容量保留中啟動執行個體。如果選取的容量預留在您所選之執行個體數量沒有足夠的容量，執行個體啟動會失敗。
   - Specify Capacity Reservation group (指定容量保留群組) – 在選取的容量保留群組內具有相符屬性和可用容量的任何容量保留中啟動執行個體。如果所選群組沒有 容量預留 具有相符屬性和可用容量的，則執行個體會啟動至隨需容量。

使用修改執行個體的容量保留設定 AWS CLI

使用 modify-instance-capacity-reservation-屬性命令。

例如，下列命令會將執行個體的 容量預留 設定變更為 open 或 none。

```
aws ec2 modify-instance-capacity-reservation-attributes --instance-
id i-1234567890abcdef0 --capacity-reservation-specification
 CapacityReservationPreference=none|open
```

例如，下列命令會修改執行個體，以將特定 容量預留 鎖定為目標。

```
aws ec2 modify-instance-capacity-reservation-attributes --instance-
id i-1234567890abcdef0 --capacity-reservation-specification
 CapacityReservationTarget={CapacityReservationId=cr-1234567890abcdef0}
```

例如，下列命令會修改執行個體，以將特定 容量預留 群組鎖定為目標。

```
aws ec2 modify-instance-capacity-reservation-attributes --instance-
id i-1234567890abcdef0 --capacity-reservation-specification
 CapacityReservationTarget={CapacityReservationResourceGroupArn=arn:aws:resource-
groups:us-west-1:123456789012:group/my-cr-group}
```

在容量保留之間移動容量

如果您有多個容量保留，您可以將可用容量從一個保留區移至另一個保留區。例如，如果您需要使用量不斷增加的容量保留中的額外容量，並且有另一個具有可用容量的容量保留，則您可以在兩個保留區之間重新分配容量。

目的地容量保留必須符合來源容量保留的下列特性：

- 執行個體類型
- 平台
- 可用區域
- 租用
- 置放群組
- 結束時間

目的地容量保留執行個體資格 (`open`或`targeted`) 和標記不一定要符合來源容量保留。除了來源保留區中的可用容量減少，以及目的地保留區中增加的可用容量外，來源和目的地容量保留的組態仍保持不變。

如果所有可用容量都從來源容量保留移出，且沒有使用任何容量，則容量保留將會自動取消。如果您嘗試移動超過來源容量保留中的可用容量，則會收到錯誤訊息。

> **Note**
> 不支援移動容量區塊的可用容量。

若要將可用容量從來源容量保留移至目的地容量保留，您可以使用 Amazon EC2 主控台或 AWS CLI.

Console

使用主控台移動可用容量

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側導覽窗格中，選擇「容量保留」。
3. 選取具有可用容量可移動的隨需容量保留 ID。
4. 在動作下的管理容量，選擇移動。
5. 在 [移動容量] 頁面的目的地容量保留下，從清單中選取保留區。
6. 在要移轉的數量下，使用滑桿或輸入要從來源容量保留移至目的地容量保留的執行環境數目。
7. 檢閱摘要，當您準備好時，請選擇 [移動]。
8. 如果移動成功，請從成功通知中選擇檢視明細，以查看 [容量保留詳細資訊] 頁面。

AWS CLI

若要使用移動可用容量 AWS CLI

使用 `move-capacity-reservation-instances` 命令。下列範例會將 10 個執行個體從識別碼為的來源容量保留移`cr-1234567890abcdef0`至識別碼為的目的地容量保留`cr-021345abcdef56789`。

```
aws ec2 move-capacity-reservation-instances --source-capacity-reservation-
id cr-1234567890abcdef0 --destination-capacity-reservation-id cr-021345abcdef56789
 --instance-count 10
```

從現有的容量保留中分割可用產能

如果您在現有的容量保留中有要指派給特定工作負載或用於執行特定動作的可用容量，則可以將可用容量分割為新的容量保留。例如，若要與另一個帳戶部分共用容量保留，您可以分割部分可用容量，以建立較小大小的容量保留。然後，您可以使用 AWS Resource Access Manager (AWS RAM) 與其他帳戶共用較小的容量保留。

當您從現有的「容量保留」分割可用產能時，會自動建立新的「容量保留」。除了從分割執行個體數目減少的總容量外，現有的容量保留將維持不變。在現有容量保留中執行的執行個體不會受到影響。您只能將現有的「容量保留」分割為一個新的「容量保留」。若要分割可用容量，現有的容量保留必須為作用中且由您的 AWS 帳戶擁有。

新的容量保留將具有與現有容量保留相同的組態，但標籤除外。依預設，新的容量保留沒有任何標籤。您可以在分割作業期間指定新標籤。如有必要，也可以在建立新的容量保留之後修改。

要從現有容量保留分割的執行個體數量上限為保留量減去 1 的大小。例如，如果容量保留的保留容量為 10 個插槽，如果全部九個插槽都可用，您最多可以分割 9 個插槽。

考量事項

- 資源群組 — 如果現有的產能保留屬於資源群組，則不會自動將新的產能保留新增至資源群組。如有必要，您可以在建立資源群組後，將其新增至資源群組。
- 共用 — 如果現有的容量保留與消費者帳戶共用，新的容量保留將不會自動與消費者帳戶共用。如有必要，您可以在建立新的容量保留後共用新的容量保留。
- 叢集置放群組 — 如果現有的容量保留是叢集置放群組的一部分，則會在相同的叢集置放群組中建立新的容量保留。

> **Note**
>
> 不支援從容量區塊分割容量。

使用標籤控制分割容量保留的存取權

您可以使用標籤來控制對 Amazon EC2 資源的存取，包括從現有容量保留中拆分可用容量以建立新的容量保留。如需詳細資訊，請參閱《使用指南》中的〈使用標籤控制 AWS 資源存取〉。IAM

若要控制使用標記分割容量保留的存取權，請確定您在原則陳述式中同時指定資源與請求標記，因為IAM原則會同時針對來源容量保留與新建立的容量保留進行評估。下列範例原則包

括ec2:ResourceTag條件索引鍵與來源容量保留的標籤Owner=ExampleDepartment1，以及帶有新建立的容量保留標籤stack=production的ec2:RequestTag條件索引鍵。

```
{
  "Statement": [
    {
      "Sid": "AllowSourceCapacityReservation",
      "Effect": "Allow",
      "Action": "ec2:CreateCapacityReservationBySplitting",
      "Resource": "arn:aws:ec2:region:account:capacity-reservation/
cr-1234567890abcdef0",
      "Condition": {
        "StringEquals": {
          "ec2:ResourceTag/Owner": "ExampleDepartment1"
        }
      }
    },
    {
      "Sid": "AllowNewlyCreatedCapacityReservation",
      "Effect": "Allow",
      "Action": ["ec2:CreateCapacityReservationBySplitting", "ec2:CreateTags"],
      "Resource": "arn:aws:ec2:region:account:capacity-reservation/*",
      "Condition": {
        "StringEquals": {
          "ec2:RequestTag/stack": "production"
        }
      }
    }
  ]
}
```

使用 Amazon 主EC2控台或 AWS CLI

若要從現有容量保留中分割可用容量並建立新的容量保留，您可以使用 Amazon EC2 主控台或 AWS CLI.

Console

使用主控台分割可用容量

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。

2. 在左側導覽窗格中，選擇「容量保留」。
3. 選取具有可用容量進行分割的隨需容量保留 ID。
4. 在動作下的管理容量，選擇分割。
5. 在「分割容量保留」頁面的「要分割的容量」下，使用滑桿或輸入要從目前保留區分割的可用執行個體數目。
6. (選擇性) 為新的容量保留新增標記。
7. 檢閱摘要，當您準備好時，選擇 [分割]。
8. 如果分割成功，請從成功通知中選擇「檢視明細」，以查看新「容量保留」的明細。

AWS CLI

若要使用分割可用容量 AWS CLI

使用 `create-capacity-reservation-by-splitting` 命令。下列範例會從容量保留中拆除 10 個執行個體，識別碼為，以建立新的容量保留`cr-1234567890abcdef0`。

```
aws ec2 create-capacity-reservation-by-splitting --source-capacity-reservation-
id cr-1234567890abcdef0 --instance-count 10
```

取消 容量預留

如果不再需要預留容量，您可以隨時取消容量預留。取消容量預留時，會立即釋出容量，而且不會再預留給您使用。

您可以取消空的容量預留和有執行中之執行個體的容量預留。如果您取消有執行中之執行個體的容量預留，執行個體會繼續在容量保留以外正常執行，依標準 隨需執行個體 費率計價，如有作用中相符的 Savings Plan 或區域預留執行個體，則依折扣費率計價。

在您取消容量預留後，即不再啟動以其為目標的執行個體。修改這些執行個體，讓它們針對不同的容量預留、使用具相符屬性及足夠容量的任何開啟中容量預留啟動，或避免使用容量預留啟動。如需詳細資訊，請參閱修改執行個體的容量保留設定。

使用主控台取消容量預留

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 選取容量預留 (容量預留)，然後選取要取消的容量預留。

3. 選擇 Cancel reservation (取消保留) 和 Cancel reservation (取消保留)。

若要使用取消容量保留 AWS CLI

使用cancel-capacity-reservation命令：

例如，下列命令會取消有 `cr-1234567890abcdef0` ID 的 容量預留。

```
aws ec2 cancel-capacity-reservation --capacity-reservation-id cr-1234567890abcdef0
```

容量保留群組

您可以使用 AWS Resource Groups 建立容量保留的邏輯集合，稱為資源群組。資源群組是全部位於相同 AWS 區域中的 AWS 資源的邏輯群組。如需資源群組的詳細資訊，請參閱 AWS Resource Groups 使用者指南中的什麼是資源群組？。

您可以包含您在帳戶中擁有的「產能保留」，以及單一資源群組中其他 AWS 帳號與您共用的「產能保留」。您也可以在單一資源群組中包含具有不同屬性 (執行個體類型、平台、可用區域和租用) 的容量保留。

為容量保留建立資源群組時，可以選定一個容量保留群組做為容量保留的目標，不要選擇個別的容量保留。以容量保留群組為目標的執行個體符合群組中具有相符屬性 (執行個體類型、平台、可用區域和租用) 和可用容量的任何容量保留。如果群組沒有具有相符屬性和可用容量的，則執行個體 容量預留 會使用隨需容量執行。如果稍後階段將相符的 容量預留 新增至目標群組，則執行個體會自動比對並移入其預留容量。

若要避免在群組中意外使用 容量預留，請將群組中的 容量預留 設定為僅接受明確以容量保留為目標的執行個體。若要這樣做，請在使用 Amazon 主控台建立容量保留時，將執行個體資格設定為目標 (舊主控台) 或僅限指定此保留的執行個體 (新EC2主控台)。使用時 AWS CLI，請在建立容量保留`--instance-match-criteria targeted`時指定。這樣做可確保只有明確鎖定目標群組或群組中的容量預留 執行個體才能在群組中執行。

如果群組中的 容量預留 在其有執行中的執行個體時被取消或過期，則這些執行個體會自動移至群組中具有相符屬性和可用容量的另一個 容量預留。如果群組中剩餘的 容量預留 沒有相符的屬性和可用容量，則執行個體會以隨需容量執行。如果稍後階段將相符的 容量預留 新增至目標群組，則執行個體會自動移至其預留容量中。

主題

- 建立容量保留群組
- 在群組新增容量保留

- 從群組移除容量保留
- 刪除容量保留群組

建立容量保留群組

您可以使用下列資訊來建立「產能保留」的資源群組。

為容量保留建立群組

使用「建立群組」指令 AWS CLI 。在 name 提供群組的描述性名稱，並在 configuration 指定兩個 Type 請求參數：

- AWS::EC2::CapacityReservationPool 可確保將資源群組鎖定為執行個體啟動的目標
- AWS::ResourceGroups::Generic 將 allowed-resource-types 設為 AWS::EC2::CapacityReservation 確保資源群組僅接受「容量預留」

例如，以下命令會建立名為 MyCRGroup 的群組。

```
aws resource-groups create-group --name MyCRGroup --configuration
 '{"Type":"AWS::EC2::CapacityReservationPool"}'
 '{"Type":"AWS::ResourceGroups::Generic", "Parameters": [{"Name": "allowed-resource-
types", "Values": ["AWS::EC2::CapacityReservation"]}]}'
```

下面顯示了範例輸出。

```
{
    "GroupConfiguration": {
        "Status": "UPDATE_COMPLETE",
        "Configuration": [
            {
                "Type": "AWS::EC2::CapacityReservationPool"
            },
            {
                "Type": "AWS::ResourceGroups::Generic",
                "Parameters": [
                    {
                        "Values": [
                            "AWS::EC2::CapacityReservation"
                        ],
                        "Name": "allowed-resource-types"
```

```
                    }
                ]
            }
        ]
    },
    "Group": {
        "GroupArn": "arn:aws:resource-groups:sa-east-1:123456789012:group/MyCRGroup",
        "Name": "MyCRGroup"
    }
}
```

在群組新增容量保留

若將與您共用的容量保留新增到群組，當該容量保留取消共用，該容量保留會自動從群組中移除。

在群組新增容量保留

使用 group-resources AWS CLI 命令。對於`group`，指定要新增容量保留的群組名稱，並指定要新增ARNs的容量保留。`resources`若要新增多個容量保留，請使ARNs用空格分隔。若要取得要新增ARNs的容量保留，請使用describe-capacity-reservations AWS CLI 命令並指定容量保留IDs的。

例如，下列命令會在名為 `MyCRGroup` 的群組中新增兩個 容量預留。

```
aws resource-groups group-resources --group MyCRGroup --resource-arns arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-1234567890abcdef1 arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-54321abcdef567890
```

下面顯示了範例輸出。

```
{
    "Failed": [],
    "Succeeded": [
        "arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-1234567890abcdef1",
        "arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-54321abcdef567890"
    ]
}
```

從群組移除容量保留

從群組中移除 容量預留

使用取消分組 AWS CLI 資源命令。對於`group`，指定ARN要從中移除容量保留的群組，並`resources`指定要移除ARNs的容量保留。若要移除多個容量保留，請使ARNs用空格分隔。

下列範例會從名為 MyCRGroup 的群組中移除兩個 容量預留。

```
aws resource-groups ungroup-resources --group MyCRGroup --resource-arns arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-0e154d26a16094dd arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-54321abcdef567890
```

下面顯示了範例輸出。

```
{
    "Failed": [],
    "Succeeded": [
        "arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-0e154d26a16094dd",
        "arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-54321abcdef567890"
    ]
}
```

刪除容量保留群組

您可以使用下列資訊刪除容量保留群組。

刪除群組

使用刪除群組命令 AWS CLI 。在 group 提供要刪除的群組名稱。

例如，以下命令會刪除名為 MyCRGroup 的檔案。

```
aws resource-groups delete-group --group MyCRGroup
```

下面顯示了範例輸出。

```
{
    "Group": {
        "GroupArn": "arn:aws:resource-groups:sa-east-1:123456789012:group/MyCRGroup",
        "Name": "MyCRGroup"
    }
}
```

在叢集置放群組中建立容量保留區

您可以在叢集置放群組中建立容量保留，以為工作負載保留 Amazon EC2 運算容量。集群放置群組具有低網路延遲和高網路輸送量的優點。

在集群放置群組中建立容量保留，可確保在需要時可隨時存取集群放置群組中的運算容量。這非常適合為需要運算擴展的高效能 (HPC) 工作負載保留容量。它可讓您縮小叢集的規模，同時確保容量仍然可供您使用，以便在需要時進行擴展。

主題

- 限制
- 使用集群放置群組中的容量保留

限制

在集群放置群組中建立容量保留時，請注意下列事項：

- 如果現有的容量保留不在放置群組中，您無法修改容量保留以保留放置群組中的容量。若要預留置放群組中的容量，您必須在置放群組中建立容量保留。
- 在置放群組中建立容量保留後，您不能將其修改為置放群組之外的預留容量。
- 您可以透過修改置放群組中的現有容量保留或在置放群組中建立附加容量保留來增加置放群組中的預留容量。但是，這會增加出現容量不足錯誤的可能性。
- 您不能分享已在集群放置群組中建立的容量保留。
- 您無法刪除具有 `active` 容量預留的集群放置群組。您必須取消您在集群放置群組中建立的所有容量保留之後，才可將其刪除。

使用集群放置群組中的容量保留

要開始對集群放置群組使用容量保留，請執行以下步驟。

> **Note**
>
> 如果要在現有集群放置群組中建立容量保留，請跳過步驟 1。然後，對於步驟 2 和 3，指定現有叢集置放群組ARN的。

主題

- 步驟 1：(有條件) 建立與容量保留搭配使用的集群放置群組
- 步驟 2：在集群放置群組中建立容量保留
- 步驟 3：在集群放置群組中啟動執行個體

步驟 1：(有條件) 建立與容量保留搭配使用的集群放置群組

僅當您需要建立新的集群放置群組時，才執行此步驟。若要使用現有的叢集置放群組，請略過此步驟，然後在步驟 2 和 3 中使用該叢集置放群組ARN的。

您可以使用下列其中一種方法來建立集群放置群組。

Console

使用主控台來建立集群放置群組

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Placement Groups (置放群組)，然後選擇 Create placement group (建立置放群組)。
3. 對於 Name (名稱)，指定置放群組的描述性名稱。
4. 對於 Placement strategy (置放策略)，選擇 Cluster (叢集)。
5. 選擇 Create group (建立群組)。
6. 在「放置群組」表格的「群組ARN」資料欄中，記下您所建立ARN的叢集置放群組。您將需要它來進行下一步。

AWS CLI

使用建立叢集置放群組 AWS CLI

使用指create-placement-group令。對於 `--group-name`，指定置放群組的描述性名稱，然後為 `--strategy` 指定 `cluster`。

下列範例會建立一個名為 `MyPG` 且使用 `cluster` 置放策略的置放群組。

```
aws ec2 create-placement-group \
    --group-name MyPG \
    --strategy cluster
```

記下命令輸出中ARN傳回的放置群組，因為下一個步驟會需要它。

步驟 2：在集群放置群組中建立容量保留

在集群放置群組中建立容量保留的方式與建立任何容量保留的方式相同。不過，您也必須指定要在其中建立容量保留ARN的叢集置放群組。如需詳細資訊，請參閱建立 容量預留。

考量事項

- 指定的集群放置群組必須處於 `available` 狀態。如果集群放置群組處於 `pending`、`deleting` 或 `deleted` 狀態，請求將失敗。
- 容量保留和集群放置群組必須位於相同的可用區域中。如果建立容量保留請求指定的可用區域不同於集群放置群組的可用區域，則請求將失敗。
- 您只能為集群放置群組支援的執行個體類型建立容量保留。若您指定不支援的執行個體類型，請求就會失敗。
- 如果您在叢集置放群組中建立open容量保留，且現有執行中執行個體具有相符屬性 (置放群組 ARN、執行個體類型、可用區域、平台和租用)，則這些執行個體會在容量保留中自動執行。
- 如果符合以下條件之一，則您建立容量預留的請求可能會失敗：
  - Amazon EC2 沒有足夠的容量來滿足請求。請稍後再試、嘗試其他可用區域，或是嘗試較小的容量。如果您的工作負載因應各種執行個體類型和大小，請嘗試不同的執行個體屬性。
  - 請求的數量超過所選執行個體系列的隨需執行個體限制。請提高執行個體系列的隨需執行個體限制，然後再試一次。如需詳細資訊，請參閱 隨需執行個體配額。

您可以使用下列其中一種方法來在集群放置群組中建立容量保留。

Console

使用主控台建立容量預留

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 選擇 容量預留，接著選擇 Create 容量預留 (建立 容量預留)。
3. 在 [建立容量保留] 頁面上，視需要指定執行個體類型、平台、可用區域、租用、數量和結束日期。
4. 在放置群組中，選取要在其中建立容量保留ARN的叢集置放群組。
5. 選擇 Create (建立)。

如需詳細資訊，請參閱建立 容量預留。

AWS CLI

若要使用建立容量保留 AWS CLI

使用指create-capacity-reservation令。在中`--placement-group-arn`，指定要在其中建立容量保留ARN的叢集置放群組。

```
$ aws ec2 create-capacity-reservation \
    --instance-type instance_type \
    --instance-platform platform \
    --availability-zone az \
    --instance-count quantity \
    --placement-group-arn placement_group_ARN
```

如需詳細資訊，請參閱建立 容量預留。

步驟 3：在集群放置群組中啟動執行個體

在集群放置群組中的容量保留內啟動執行個體，其方式與在任何容量保留中啟動執行個體相同。不過，您也必須指定要在其中啟動執行個體ARN的叢集置放群組。如需詳細資訊，請參閱建立 容量預留。

考量事項

- 如果容量保留為 open，您無需在執行個體啟動請求中指定容量保留。如果執行個體具有與指定放置群組中的容量保留相符的屬性 (置放群組ARN、執行個體類型、可用區域、平台和租用)，則執行個體會自動在容量保留中執行。
- 如果容量保留僅接受目標執行個體啟動，則除了請求中的集群放置群組之外，還必須指定目標容量保留。
- 如果容量保留位於容量保留群組中，則除了請求中的集群放置群組之外，還必須指定目標容量保留群組。如需詳細資訊，請參閱 容量保留群組。

您可以使用下列其中一種方法，將執行個體啟動至集群放置群組中的容量保留。

Console

使用主控台以現有的容量預留啟動執行個體

1. 請遵循程序啟動執行個體，但在完成下列步驟以指定置放群組和容量保留的設定之前，請勿啟動執行個體。
2. 展開進階詳細資料並執行下列操作：
   a. 在「放置」群組中，選取要在其中啟動執行個體的叢集置放群組。
   b. 對於 Capacity Reservation (容量保留)，根據容量保留的組態選取下列其中一個選項：

- 開啟 — 將執行個體啟動至叢集置放群組中具有相符屬性和足夠容量的任何open容量保留。
- 按 ID 鎖定 — 將執行個體啟動至僅接受目標執行個體啟動的容量保留。
- 按群組鎖定 — 在選取的容量保留群組中，將執行個體啟動至具有相符屬性和可用容量的任何容量保留。

3. 在 Summary (摘要) 面板中，檢閱您的執行個體組態，然後選擇 Launch instance (啟動執行個體)。如需詳細資訊，請參閱使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體。

如需詳細資訊，請參閱使用現有的 容量預留 啟動執行個體。

AWS CLI

使用將執行個體啟動至現有的容量保留 AWS CLI

使用 run-instances 命令。如果您需要針對特定容量保留或容量保留群組，請指定 `--capacity-reservation-specification` 參數。對於 `--placement`，指定 `GroupName` 參數，然後指定您在上述步驟建立的置放群組的名稱。

以下命令將執行個體啟動到集群放置群組中的 `targeted` 容量保留內。

```
$ aws ec2 run-instances \
    --image-id ami_id \
    --count quantity \
    --instance-type instance_type \
    --key-name key_pair_name \
    --subnet-id subnetid \
    --capacity-reservation-specification
 CapacityReservationTarget={CapacityReservationId=capacity_reservation_id} \
    --placement "GroupName=cluster_placement_group_name"
```

如需詳細資訊，請參閱 使用現有的 容量預留 啟動執行個體。

Local Zones 中的容量保留

本地區域是地理位置靠近您的用戶的 AWS 區域的擴展。在 Local Zone 中建立的資源可以為本機使用者提供非常低延遲的通訊服務。如需詳細資訊，請參閱 AWS Local Zones。

您可以透過在該本機 AWS 區域中建立新的子網路，將其父區域延伸至本機區域。VPC當您在本機區域中建立子網路時，會將您的VPC子網路延伸至該本機區域。本機區域中的子網路運作方式與您VPC中的其他子網路相同。

透過使用 Local Zones，您可以將容量預留放置在更接近使用者的多個位置。您在 Local Zones 中建立和使用容量預留的方式，與在一般可用區域中建立和使用容量預留的方式相同。適用相同的功能和執行個體比對行為。如需本 Local Zones 域支援之定價模式的詳細資訊，請參閱 L AWS ocal Zones FAQs。

考量事項

您無法在 Local Zone 中使用容量預留群組。

若要在 Local Zone 中使用容量預留

1. 啟用本地區域以在您的 AWS 帳戶中使用。如需詳細資訊，請參閱 L AWS ocal Zones 使用指南中的使用 L AWS ocal Zones 入門。
2. 在 Local Zone 中建立容量預留。對於 Availability Zone (可用區域)，請選擇 Local Zone。例如，「本地區域」由「地 AWS 區」代碼表示，後面接著指示位置的識別碼`us-west-2-lax-1a`。如需詳細資訊，請參閱建立 容量預留。
3. 在 Local Zone 中建立子網。對於 Availability Zone (可用區域)，請選擇 Local Zone。如需詳細資訊，請參閱 Amazon VPC 使用者指南VPC中的「建立子網路」。
4. 啟動執行個體。對於子網，選擇 Local Zone 中的子網 (例如 `subnet-123abc | us-west-2-lax-1a`)，對於容量預留，請選擇您在 Local Zone 中建立容量預留所必需的規格 (open 或透過 ID 將其鎖定為目標)。如需詳細資訊，請參閱使用現有的 容量預留 啟動執行個體。

在 Wavelength 區域中的容量保留

AWS Wavelength 可讓開發人員建立提供極低延遲的應用程式給行動裝置與最終使用者。Wavelength 將標準的 AWS 運算與儲存服務部署至電信業者 5G 網路的邊緣。您可以將 Amazon Virtual Private Cloud (VPC) 擴展到一個或多個 Wavelength 區域。然後，您可以使用 Amazon EC2 執行個體等 AWS 資源來執行需要超低延遲和區域 AWS 服務連線的應用程式。如需詳細資訊，請參閱AWS Wavelength 區域。

當您建立隨需 容量預留 時，可以選擇 Wavelength 區域，並透過指定與 Wavelength 區域相關聯的子網，在 Wavelength 區域中對 容量預留 啟動執行個體。Wavelength 區域是以一個 AWS 區域代碼加上一個識別符來表示，用以代表位置，例如 `us-east-1-wl1-bos-wlz-1`。

並非每個區域都有 Wavelength 區域。如需支援 Wavelength 區域的區域相關資訊，請參閱 AWS Wavelength 開發人員指南中的可用 Wavelength 區域。

考量事項

您無法使用在 Wavelength 區域的 容量預留 群組。

在 Wavelength 區域中使用 容量預留

1. 啟用 Wavelength 區域以便在您的 AWS 帳戶中使用。如需詳細資訊，請參閱 AWS Wavelength 開發人員指南中的 AWS Wavelength入門。
2. 在 Wavelength 區域中建立 容量預留。對於可用區域，請選擇 Wavelength。例`us-east-1-wl1-bos-wlz-1`如，「Wavelength」由「 AWS 區域」代碼表示，後面接著指示位置的識別碼。如需詳細資訊，請參閱建立 容量預留。
3. 在 Wavelength 區域中建立子網。對於可用區域，請選擇 Wavelength 區域。如需詳細資訊，請參閱 Amazon VPC 使用者指南VPC中的「建立子網路」。
4. 啟動執行個體。在子網中，選擇在 Wavelength 區域中的子網 (例如 `subnet-123abc | us-east-1-wl1-bos-wlz-1`)，對於 容量預留，請選擇規格 (open 或透過 ID 將其鎖定為目標)，這對於您在 Wavelength 中建立的 容量預留 為必要條件。如需詳細資訊，請參閱使用現有的 容量預留 啟動執行個體。

容量保留 AWS Outposts

AWS Outposts 是一項全受管服務，可將 AWS 基礎架構APIs、服務和工具延伸至客戶場所。透過提供 AWS 受管理基礎架構的本機存取權， AWS Outposts 讓客戶能夠使用與 Region 相同的程式設計介面在 AWS 內部部署建置和執行應用程式，同時使用本機運算和儲存資源來降低延遲和本機資料處理需求。

Outpost 是部署在客戶站點的 AWS 計算和儲存容量集區。 AWS 作為 AWS 區域的一部分來操作、監控和管理此容量。

您可以在帳戶中建立的 Outposts 上建立容量保留。這可讓您在站點的 Outpost 上預留運算容量。您在 Outposts 上建立和使用容量預留的方式，與在一般可用區域中建立和使用容量預留的方式相同。適用相同的功能和執行個體比對行為。

您也可以使用與組織內的其他 AWS 帳戶共用 AWS Resource Access Manager Outposts 上的容量保留。如需共用容量保留的資訊，請參閱 共用容量保留。

先決條件

您的站點必須安裝 Outpost。如需詳細資訊，請參閱 AWS Outposts 使用者指南中的建立 Outpost 並訂購 Outpost 容量。

考量

- 您無法在 Outpost 上使用容量預留群組。

若要在 Outpost 上使用容量預留

1. 在 Outpost 上建立子網。如需詳細資訊，請參閱 AWS Outposts 使用者指南中的建立子網。
2. 在 Outpost 上建立容量保留。
   a. 在開啟 AWS Outposts 主控台https://console.aws.amazon.com/outposts/。
   b. 在導覽窗格中，選擇 Outposts，然後選擇 Actions (動作)、Create Capacity Reservation (建立容量預留)。
   c. 視需要設定容量預留，然後選擇 Create (建立)。如需詳細資訊，請參閱建立 容量預留。

   > **Note**
   > Instance Type (執行個體類型) 下拉式清單只會列出選取的 Outpost 支援的執行個體類型，而 Availability Zone (可用區域) 下拉式清單只會列出與選取的 Outpost 相關聯的可用區域。

3. 在容量預留中啟動執行個體。對於子網，選取您在步驟 1 中建立的子網，對於容量預留，選取您在步驟 2 中建立的容量預留。如需詳細資訊，請參閱 AWS Outposts 使用者指南中的在 Outpost 上啟動執行個體。

共用容量保留

容量保留共用可讓容量保留擁有者與其他 AWS 帳戶或 AWS 組織內共用其保留容量。這可讓您集中建立和管理容量保留，並在多個 AWS 帳戶或 AWS 組織內共用保留容量。

在此模型中，擁有容量保留 (擁有者) 的 AWS 帳戶會與其他 AWS 帳戶 (取用者) 共用該帳戶。消費者可以在與他們共用的 容量預留 中啟動執行個體，就好像他們在自己的帳戶中擁有的 容量預留 中啟動執行個體一樣。容量預留擁有者負責管理 容量預留和他們在其中啟動的執行個體。擁有者在他們共用的 容量預留中無法修改消費者啟動的執行個體。消費者在分享給他們的 容量預留中負責管理他們啟動的執行個體。消費者無法檢視或修改其他消費者或 容量預留擁有者所擁有的執行個體。

容量預留擁有者可以將 容量預留分享給下列對象：

- AWS 組織內部或外部的特定 AWS 帳戶
- 組織內的組 AWS 織單位
- 其整個 AWS 組織

共用容量預留的先決條件

- 若要共用容量保留，您必須在 AWS 帳戶中擁有該容量保留。您無法將已分享給您的 容量預留 再分享出去。
- 針對租用執行個體您只能共用 容量預留。針對專用租用執行個體您無法共用 容量預留。
- 容量保留共用不適用於 AWS 帳單歷史記錄有限的新 AWS 帳戶或帳戶。
- 若要與您的組織或 AWS 組織中的組織單位共用容量保留，您 AWS 必須啟用與共用 AWS Organizations。如需詳細資訊，請參閱《AWS RAM 使用者指南》中的透過 AWS Organizations啟用共用。

相關服務

容量保留共用與 AWS Resource Access Manager (AWS RAM) 整合。 AWS RAM 是一項服務，可讓您與任何 AWS 帳戶或透過共用 AWS 資源 AWS Organizations。您可以透過建立資源共用 AWS RAM 來共用您所擁有的資源。資源共享指定要共用的資源，以及共用它們的消費者。消費者可以是個人 AWS 帳戶，或組織單位或整個組織 AWS Organizations。

若要取得有關的更多資訊 AWS RAM，請參閱AWS RAM 使用者指南。

跨可用區域共用

為確保資源分配至區域中的所有可用區域，可用區域會獨立對應至各個帳戶的名稱。這可能導致帳戶之間的可用區域命名出現差異。例如，您 AWS 帳戶的可`us-east-1a`用區域可能與其他 AWS 帳戶`us-east-1a`的位置不同。

若要基於您的帳戶來識別 容量預留的相對位置，您必須使用可用區域 ID (AZ ID)。AZ ID 是所有 AWS 帳戶中可用區域的唯一且一致的識別碼。例如，`use1-az1`是`us-east-1`區域的 AZ ID，每個 AWS 帳戶中的位置都相同。

檢視您帳IDs戶中可用區域的 AZ

1. 請在 https://console.aws.amazon.com/ram 開啟 AWS RAM 主控台。
2. 目前區域IDs的 AZ 會顯示在畫面右側的「您的 AZ ID」面板中。

共用 容量預留

當您與其他人共用您擁有的容量保留時 AWS 帳戶，您可以讓他們將執行個體啟動到您的預留容量中。如果您共用開啟的容量預留，請謹記下列事項，因為有可能導致容量預留用量超出預料：

- 如果消費者的執行中執行個體符合 容量預留的屬性、將 `CapacityReservationPreference` 參數設為 `open`，以及尚未執行預留容量，他們會自動使用共用的 容量預留。
- 如果取用者啟動具有相符屬性 (執行個體類型、平台、可用區域和租用) 的執行個體，並將`CapacityReservationPreference`參數設定為`open`，則會自動啟動至共用容量保留。

若要分享 容量預留，您必須將它新增至資源共享。資源共用是一 AWS RAM 種可讓您跨 AWS 帳號共用資源的資源。資源共享指定要共用的資源，以及共用它們的消費者。使用 Amazon EC2 主控台共用容量保留時，請將其新增至現有的資源共用。若要將容量預留新增至新的資源共享，您必須使用 AWS RAM 主控台建立資源共享。

如果您是組織中的一員， AWS Organizations 且已啟用組織內的共用功能，則您組織中的用戶會被授與存取共用「容量保留」(如果符合共用的前提網站)。如果容量保留與外部帳戶共享，他們會收到加入資源共享的邀請，並且在接受邀請後便能存取共享的容量保留。

> **Important**
>
> 將執行個體啟動到與您共用的容量保留之前，請在主控台中檢視或使用describe-capacity-reservations AWS CLI 命令說明共用容量保留，以確認您是否具有存取共用容量保留的權限。如果您可以在主控台中檢視共用的容量保留或使用說明共用的容量保留 AWS CLI，則可供您使用，而且您可以在其中啟動執行個體。如果您嘗試將執行個體啟動至容量保留中，但由於共享失敗而無法存取，則執行個體將啟動至隨需容量中。

您可以使用 Amazon EC2 主控台、 AWS RAM 主控台或共用您擁有的容量保留 AWS CLI。

使用 Amazon EC2 主控台共用您擁有的容量保留

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 容量預留。
3. 選擇要分享的 容量預留，然後選擇 Actions (動作)、Share reservation (共用保留)。
4. 選取資源共享以新增 容量預留，然後選取 Share 容量預留 (共用 容量預留)。

   可能需要經過幾分鐘，消費者才能存取共用的 容量預留。

共用您使用 AWS RAM 主控台擁有的容量保留

請參閱《AWS RAM 使用者指南》中的建立資源共享。

若要共用您所擁有的容量保留 AWS CLI

使用指create-resource-share令。

停止共用容量預留

容量預留擁有者可以隨時停止共用容量預留。適用的規定如下：

- 在共用停止時以共用容量執行的取用者擁有的執行個體會在預留容量之外繼續正常執行，且容量會還原至容量保留 (視 Amazon EC2 容量可用性而定)。
- 共用 容量預留的消費者不能再於預留容量中啟動新的執行個體。

若要停止共用您擁有的容量預留，您必須從資源共用中移除它。您可以使用 Amazon 主EC2控台、AWS RAM 主控台或 AWS CLI.

停止共用您使用 Amazon EC2 主控台擁有的容量保留

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 容量預留。
3. 選取容量預留，然後選取共用索引標籤。
4. Sharing (共用) 索引標籤列出已新增 容量預留的資源共享。選取要從中移除 容量預留的資源共享，然後選取 Remove from resource share (從資源共享中移除)。

停止共用您使用 AWS RAM 主控台擁有的容量保留

請參閱《AWS RAM 使用者指南》中的更新資源共享。

若要停止共用您所擁有的容量保留，請使用 AWS CLI

使用指disassociate-resource-share令。

共用的容量預留許可

擁有者的許可

擁有者負責管理和取消他們共用的 容量預留。擁有者無法修改共用的 容量預留中由其他帳戶擁有的執行中執行個體。擁有者仍然負責管理他們在共用的 容量預留中啟動的執行個體。

消費者的許可

消費者負責管理他們在共用的 容量預留中執行的執行個體。消費者完全無法修改共用的 容量預留，也無法檢視或修改其他消費者或 容量預留擁有者所擁有的執行個體。

計費和計量

共用 容量預留無須額外收費。

容量預留擁有者對於他們在 容量預留內執行的執行個體及未用的預留容量，需要付費。消費者對於他們在共用的 容量預留內執行的執行個體，需要付費。

如果容量保留擁有者屬於不同的付款人帳戶，且區域預留執行個體或 Savings Plan 涵蓋容量保留，則會繼續針對區域預留執行個體或 Savings Plan 向容量保留擁有者收取費用。在這些情況下，容量保留擁有者會支付區域預留執行個體或 Savings Plan 的費用，並且會針對在共用容量保留中執行的執行個體向消費者收費。

執行個體限制

所有 容量預留用量都計入 容量預留擁有者的 隨需執行個體 限制。其中包括：

- 未用的預留容量
- 容量預留擁有者所擁有者的執行個體的用量
- 消費者擁有的執行個體的用量

消費者在共享容量中啟動的執行個體計入 容量預留擁有者的 隨需執行個體 限制。消費者的執行個體限制等於他們自己的 隨需執行個體 限制和他們在共用的 容量預留中可存取的可用容量的總和。

容量預留機群

隨需容量預留機群是一組容量預留。

容量預留機群要求包含啟動容量預留機群所需的所有組態資訊。使用單一請求，您可以為跨多個執行個體類型的工作負載預留大EC2量 Amazon 容量，最多可達指定的目標容量。

建立容量預留機群之後，您可以修改或取消容量預留機群，共同管理機群中的容量預留。

主題

- 容量預留機群的運作方式

- 考量事項
- 定價
- 容量預留叢集概念與規劃
- 建立容量預留機群
- 檢視容量預留機群
- 修改容量預留機群
- 取消容量預留機群
- 容量預留機群組態範例
- 使用容量保留機群的服務連結角色

容量預留機群的運作方式

建立容量預留機群時，機群會嘗試建立個別容量預留，符合您在機群要求中指定的總目標容量。

機群預留容量的執行個體數量，取決於您指定的總目標容量與執行個體類型權重。預留容量的執行個體類型，取決於您使用的分配策略和執行個體類型優先順序。

如果建立機群時容量不足，且無法立即滿足總目標容量，則機群會非同步嘗試建立容量預留，直到已預留要求的容量數量為止。

機群達到總目標容量時，會嘗試維持該容量。如果取消機群中的容量預留，機群會根據您的機群組態，自動建立一或多個容量預留，取代失去的容量並維持總目標容量。

無法個別管理機群中的容量預留。必須修改機群來集體管理。修改機群時，機群中的容量預留會自動更新以反映變更。

容量預留機群目前支援 open 執行個體符合條件，而機群啟動的所有容量預留會自動使用此執行個體符合條件。根據此準則，具有相符屬性 (執行個體類型、平台、可用區域和租用) 的新執行個體和現有執行個體會自動在叢集建立的容量保留中執行。容量預留機群不支援 target 執行個體符合條件。

考量事項

使用容量預留機群時請記住下列事項：

- 您可以使用和建立、修改、檢視及取消容量保留叢集 AWS API。 AWS CLI
- 無法個別管理機群中的容量預留。必須修改或取消機群來集體管理。
- 容量預留機群無法跨區域。

- 容量預留機群無法跨可用區域。
- 由容量保留叢集建立的容量保留會自動標記為下列 AWS 產生的標記：
  - 索引鍵 – `aws:ec2-capacity-reservation-fleet`
  - 值 – *`fleet_id`*

  您可以使用此標籤，識別容量預留機群建立的容量預留。

定價

使用容量預留機群無須額外收費。系統會針對您容量預留機群建立的個別容量預留計費。如需容量預留計費方式的詳細資訊，請參閱容量預留定價和帳單。

容量預留叢集概念與規劃

下列資訊說明如何規劃容量保留叢集，並說明容量保留叢集概念，包括目標容量總計、配置策略、執行個體類型權重以及執行個體類型優先順序。

主題

- 規劃容量保留叢集
- 總目標容量
- 分配策略
- 執行個體類型權重
- 執行個體類型優先順序

規劃容量保留叢集

規劃容量保留叢集時，建議您執行下列動作：

1. 決定工作負載所需的運算容量。
2. 決定要使用的執行個體類型和可用區域。
3. 根據您的需求和偏好，為每個執行個體類型指派優先順序。如需詳細資訊，請參閱 執行個體類型優先順序。
4. 建立適合您工作負載的容量加權系統。為每個執行個體類型指派權重，並決定總目標容量。如需詳細資訊，請參閱執行個體類型權重及總目標容量。
5. 決定您是否無限期需要容量保留，或是只需要一段特定時間。

總目標容量

總目標容量定義容量預留機群預留的運算容量總量。建立容量預留機群時，您可以指定總目標容量。建立叢集之後，Amazon EC2 會自動建立容量保留，以保留最多總目標容量的容量。

容量預留機群為其預留容量的執行個體數量，取決於總目標容量，以及您為容量預留機群中每個執行個體類型指定的執行個體類型權重 (`total target capacity/instance type weight=number of instances`)。

您可以根據對您工作負載有意義的單位數量，指派總目標容量。例如，如果您的工作負載需要特定數目 vCPUs，您可以根據 vCPUs 所需的數目指定總目標容量。如果您的工作負載需要 2048vCPUs，請指定的目標容量總計，2048然後根據叢集中執行個體類型 vCPUs 提供的數目指派執行個體類型權重。如需範例，請參閱執行個體類型權重。

分配策略

容量預留機群中的分配策略，決定如何從容量預留機群組態中的執行個體類型規格，履行您的預留容量要求。

目前只支援 `prioritized` 分配策略。使用此策略，容量預留機群會使用您指派給容量預留機群組態中每個執行個體類型規格的優先順序，建立容量預留。優先順序值越低，表示使用的優先順序越高。例如，假設您建立使用下列執行個體類型和優先順序的容量預留機群：

- `m4.16xlarge` – 優先順序 = 1
- `m5.16xlarge` – 優先順序 = 3
- `m5.24xlarge` – 優先順序 = 2

機群會先嘗試為 `m4.16xlarge` 建立容量保留。如果 Amazon EC2 的`m4.16xlarge`容量不足，叢集會嘗試為`m5.24xlarge`. 如果 Amazon EC2 的`m5.24xlarge`容量不足，則叢集會為`m5.16xlarge`.

執行個體類型權重

執行個體類型權重是您指派給容量預留機群中每個執行個體類型的權重。權重決定計入機群總目標容量的該特定執行個體類型每個執行個體容量單位量。

您可以根據對您工作負載有意義的單位數量來指派權重。例如，如果您的工作負載需要特定數量的 vCPUs，您可以根據容量保留叢集中每個執行個體類型 vCPUs 提供的數目來指派權重。在此情況下，如果您使用`m4.16xlarge`和`m5.24xlarge`執行個體建立容量保留叢集，您可以指定與每個執行個體數目相對應的 vCPUs 加權，如下所示：

- m4.16xlarge— 64vCPUs，重量 = 64 單位
- m5.24xlarge— 96vCPUs，重量 = 96 單位

執行個體類型權重決定容量預留機群為其預留容量的執行個體數量。例如，如果總目標容量為 384 單位的容量預留機群，使用先前範例中的執行個體類型和權重，則機群可以為 6 個 m4.16xlarge 執行個體 (384 總目標容量/64 執行個體類型權重 = 6 個執行個體)，或 4 個 m5.24xlarge 執行個體 (384/96 = 4) 預留容量。

如果您未指派執行個體類型權重，或將執行個體類型權重指派為 1，則總目標容量會只根據執行個體數量。舉例來說，如果總目標容量為 384 個單位的容量預留機群使用先前範例中的執行個體類型，但省略權重或對這兩個執行個體類型指定 1，機群可為 384 個 m4.16xlarge 執行個體或 384 個 m5.24xlarge 執行個體預留容量。

執行個體類型優先順序

執行個體類型優先順序是您指派給機群中執行個體類型的值。優先順序是用來決定應優先使用哪些為機群指定的執行個體類型。

優先順序值越低，表示使用的優先順序越高。

建立容量預留機群

建立容量預留機群時，它會自動為機群請求中指定的執行個體類型建立容量預留，最多達指定的總目標容量。容量預留機群預留容量的執行個體數量，取決於您在請求中指定的總目標容量和執行個體類型權重。如需詳細資訊，請參閱執行個體類型權重及總目標容量。

建立機群時，您必須指定要使用的執行個體類型，以及每個執行個體類型的優先順序。如需詳細資訊，請參閱 分配策略 和 執行個體類型優先順序。

> **Note**
>
> 第一次建立容量保留叢集時，會在您的帳戶中自動建立AWSServiceRoleForEC2CapacityReservationFleet服務連結角色。如需詳細資訊，請參閱使用容量保留機群的服務連結角色。

容量預留機群目前支援 open 執行個體符合條件。

若要建立容量預留機群

使用指create-capacity-reservation-fleet AWS CLI 令。

```
aws ec2 create-capacity-reservation-fleet \
--total-target-capacity capacity_units \
--allocation-strategy prioritized \
--instance-match-criteria open \
--tenancy dedicated|default \
--end-date yyyy-mm-ddThh:mm:ss.000Z \
--instance-type-specifications file://instanceTypeSpecification.json
```

`instanceTypeSpecification.json` 內容如下。

```
[
    {
        "InstanceType": "instance_type",
        "InstancePlatform":"platform",
        "Weight": instance_type_weight,
        "AvailabilityZone":"availability_zone",
        "AvailabilityZoneId" : "az_id",
        "EbsOptimized": true|false,
        "Priority" : instance_type_priority
    }
]
```

預期的輸出結果。

```
{
    "Status": "status",
    "TotalFulfilledCapacity": fulfilled_capacity,
    "CapacityReservationFleetId": "cr_fleet_id",
    "TotalTargetCapacity": capacity_units
}
```

範例

```
aws ec2 create-capacity-reservation-fleet \
--total-target-capacity 24 \
--allocation-strategy prioritized \
--instance-match-criteria open \
--tenancy default \
--end-date 2021-12-31T23:59:59.000Z \
--instance-type-specifications file://instanceTypeSpecification.json
```

instanceTypeSpecification.json

```
[
  {
    "InstanceType": "m5.xlarge",
    "InstancePlatform": "Linux/UNIX",
    "Weight": 3.0,
    "AvailabilityZone":"us-east-1a",
    "EbsOptimized": true,
    "Priority" : 1
  }
]
```

範例輸出。

```
{
    "Status": "submitted",
    "TotalFulfilledCapacity": 0.0,
    "CapacityReservationFleetId": "crf-abcdef01234567890",
    "TotalTargetCapacity": 24
}
```

檢視容量預留機群

您可以隨時檢視容量預留機群的組態和容量資訊。檢視機群也會提供有關機群內個別容量預留的詳細資訊。

若要檢視容量預留機群

使用指describe-capacity-reservation-fleets AWS CLI 令。

```
aws ec2 describe-capacity-reservation-fleets \
--capacity-reservation-fleet-ids cr_fleet_ids
```

下列為範例輸出。

```
{
    "CapacityReservationFleets": [
        {
            "Status": "status",
            "EndDate": "yyyy-mm-ddThh:mm:ss.000Z",
            "InstanceMatchCriteria": "open",
```

```
            "Tags": [],
            "CapacityReservationFleetId": "cr_fleet_id",
            "Tenancy": "dedicated|default",
            "InstanceTypeSpecifications": [
                {
                    "CapacityReservationId": "cr1_id",
                    "AvailabilityZone": "cr1_availability_zone",
                    "FulfilledCapacity": cr1_used_capacity,
                    "Weight": cr1_instance_type_weight,
                    "CreateDate": "yyyy-mm-ddThh:mm:ss.000Z",
                    "InstancePlatform": "cr1_platform",
                    "TotalInstanceCount": cr1_number of instances,
                    "Priority": cr1_instance_type_priority,
                    "EbsOptimized": true|false,
                    "InstanceType": "cr1_instance_type"
                },
    {
                    "CapacityReservationId": "cr2_id",
                    "AvailabilityZone": "cr2_availability_zone",
                    "FulfilledCapacity": cr2_used_capacity,
                    "Weight": cr2_instance_type_weight,
                    "CreateDate": "yyyy-mm-ddThh:mm:ss.000Z",
                    "InstancePlatform": "cr2_platform",
                    "TotalInstanceCount": cr2_number of instances,
                    "Priority": cr2_instance_type_priority,
                    "EbsOptimized": true|false,
                    "InstanceType": "cr2_instance_type"
                },
            ],
            "TotalTargetCapacity": total_target_capacity,
            "TotalFulfilledCapacity": total_target_capacity,
            "CreateTime": "yyyy-mm-ddThh:mm:ss.000Z",
            "AllocationStrategy": "prioritized"
        }
    ]
}
```

範例

```
aws ec2 describe-capacity-reservation-fleets \
--capacity-reservation-fleet-ids crf-abcdef01234567890
```

下列為範例輸出。

```
{
    "CapacityReservationFleets": [
        {
            "Status": "active",
            "EndDate": "2021-12-31T23:59:59.000Z",
            "InstanceMatchCriteria": "open",
            "Tags": [],
            "CapacityReservationFleetId": "crf-abcdef01234567890",
            "Tenancy": "default",
            "InstanceTypeSpecifications": [
                {
                    "CapacityReservationId": "cr-1234567890abcdef0",
                    "AvailabilityZone": "us-east-1a",
                    "FulfilledCapacity": 5.0,
                    "Weight": 1.0,
                    "CreateDate": "2021-07-02T08:34:33.398Z",
                    "InstancePlatform": "Linux/UNIX",
                    "TotalInstanceCount": 5,
                    "Priority": 1,
                    "EbsOptimized": true,
                    "InstanceType": "m5.xlarge"
                }
            ],
            "TotalTargetCapacity": 5,
            "TotalFulfilledCapacity": 5.0,
            "CreateTime": "2021-07-02T08:34:33.397Z",
            "AllocationStrategy": "prioritized"
        }
    ]
}
```

容量預留機群狀態

容量預留機群可以是下列任一狀態：

- `submitted`— 已提交容量保留叢集請求，Amazon EC2 正準備建立容量保留。
- `modifying` – 正在修改容量保留機群。修改完前，機群會一直維持在此狀態。
- `active` – 容量保留機群已達到其總目標容量，並嘗試維持此容量。機群會一直維持在該狀態，直到遭修改或刪除。
- `partially_fulfilled` – 容量保留機群已部分達到總目標容量。Amazon EC2 容量不足以滿足總目標容量。機群正嘗試以非同步方式達到總目標容量。

- `expiring` – 容量保留機群已到結束日期，且即將過期。其一或多個容量預留可能仍作用中。
- `expired` – 容量保留機群已到結束日期。機群及其容量預留已過期。機群無法建立新的容量預留。
- `cancelling` – 容量保留機群正遭取消。其一或多個容量預留可能仍作用中。
- `cancelled` – 容量保留機群已手動取消。機群及其容量預留已取消，且機群無法建立新的容量預留。
- `failed` – 容量保留機群無法為指定的執行個體類型預留容量。

修改容量預留機群

您可以隨時修改容量預留機群的總目標容量和日期。修改容量預留叢集的總目標容量時，機群會自動建立新的容量預留，或者修改或取消機群中的現有容量預留，以滿足新的總目標容量。修改機群的結束日期時，會相應地更新所有個別容量預留的結束日期。

修改機群後，其狀態會轉換為 `modifying`。您無法在機群處於 `modifying` 狀態時，嘗試對機群進行額外修改。

您無法修改容量預留機群使用的租用、可用區域、執行個體類型、執行個體平台、優先順序或權重。如果您必須變更任一參數，可能需要取消現有機群，並以所需參數建立新機群。

若要修改容量預留機群

使用指modify-capacity-reservation-fleet AWS CLI 令。

> **Note**
>
> 您無法在相同命令中指定 `--end-date` 和 `--remove-end-date`。

```
aws ec2 modify-capacity-reservation-fleet \
--capacity-reservation-fleet-id cr_fleet_ids \
--total-target-capacity capacity_units \
--end-date yyyy-mm-ddThh:mm:ss.000Z \
--remove-end-date
```

下列為範例輸出。

```
{
```

```
    "Return": true
}
```

範例：修改總目標容量

```
aws ec2 modify-capacity-reservation-fleet \
--capacity-reservation-fleet-id crf-01234567890abcedf \
--total-target-capacity 160
```

範例：修改結束日期

```
aws ec2 modify-capacity-reservation-fleet \
--capacity-reservation-fleet-id crf-01234567890abcedf \
--end-date 2021-07-04T23:59:59.000Z
```

範例：移除結束日期

```
aws ec2 modify-capacity-reservation-fleet \
--capacity-reservation-fleet-id crf-01234567890abcedf \
--remove-end-date
```

下列為範例輸出。

```
{
    "Return": true
}
```

取消容量預留機群

不再需要容量預留機群及其預留的容量時，您可以取消該機群。取消機群時，其狀態會變更為 `cancelled`，且無法再建立新的容量預留。此外，叢集中的所有個別容量保留都會被取消。先前以預留容量執行的執行個體會在共用容量中繼續正常執行。

若要取消容量預留機群

使用指cancel-capacity-reservation-fleets AWS CLI 令。

```
aws ec2 cancel-capacity-reservation-fleets \
--capacity-reservation-fleet-ids cr_fleet_ids
```

下列為範例輸出。

```
{
    "SuccessfulFleetCancellations": [
        {
            "CurrentFleetState": "state",
            "PreviousFleetState": "state",
            "CapacityReservationFleetId": "cr_fleet_id_1"
        },
        {
            "CurrentFleetState": "state",
            "PreviousFleetState": "state",
            "CapacityReservationFleetId": "cr_fleet_id_2"
        }
    ],
    "FailedFleetCancellations": [
        {
         "CapacityReservationFleetId": "cr_fleet_id_3",
         "CancelCapacityReservationFleetError": [
             {
              "Code": "code",
                    "Message": "message"
          }
         ]
     }
    ]
}
```

範例：成功取消

```
aws ec2 cancel-capacity-reservation-fleets \
--capacity-reservation-fleet-ids crf-abcdef01234567890
```

下列為範例輸出。

```
{
    "SuccessfulFleetCancellations": [
        {
            "CurrentFleetState": "cancelling",
            "PreviousFleetState": "active",
            "CapacityReservationFleetId": "crf-abcdef01234567890"
        }
```

```
    ],
    "FailedFleetCancellations": []
}
```

容量預留機群組態範例

以下範例會建立容量預留機群，此機群使用兩種執行個體類型：m5.4xlarge 和 m5.12xlarge。

它使用基於指定例證類型 vCPUs 提供的數量的加權系統。總目標容量為480vCPUs。提m5.4xlarge供 16 vCPUs 並獲得的重量16，而提m5.12xlarge供 48 vCPUs 並獲得的重量48。此加權系統會設定容量預留機群，為 30 個 m5.4xlarge 執行個體 (480/16=30)，或 10 個 m5.12xlarge 執行個體 (480/48= 10) 預留容量。

機群設定為優先預留 m5.12xlarge 容量，取得優先順序 1，而 m5.4xlarge 取得較低的優先順序 2。這表示叢集會先嘗試保留m5.12xlarge容量，並且只會在 Amazon m5.4xlarge EC2 容量不足時嘗試保留m5.12xlarge容量。

叢集會保留Windows執行個體的容量，且保留區會October 31, 2021在自動到期23:59:59UTC。

```
aws ec2 create-capacity-reservation-fleet \
--total-target-capacity 480 \
--allocation-strategy prioritized \
--instance-match-criteria open \
--tenancy default \
--end-date 2021-10-31T23:59:59.000Z \
--instance-type-specifications file://instanceTypeSpecification.json
```

instanceTypeSpecification.json 內容如下。

```
[
    {
        "InstanceType": "m5.4xlarge",
        "InstancePlatform":"Windows",
        "Weight": 16,
        "AvailabilityZone":"us-east-1a",
        "EbsOptimized": true,
        "Priority" : 2
    },
    {
        "InstanceType": "m5.12xlarge",
        "InstancePlatform":"Windows",
        "Weight": 48,
```

```
        "AvailabilityZone":"us-east-1a",
        "EbsOptimized": true,
        "Priority" : 1
 }
]
```

使用容量保留機群的服務連結角色

隨選容量保留叢集使用 AWS Identity and Access Management (IAM) 服務連結角色。服務連結角色是直接連結至容量保留叢集的唯一IAM角色類型。服務連結角色由容量保留叢集預先定義，並包含服務代表您呼叫其他服 AWS 務所需的所有權限。

服務連結角色可讓設定容量預留機群更為簡單，因為您不必手動新增必要的許可。容量預留機群定義其服務連結角色的許可，除非另有定義，否則僅有容量預留機群可以擔任其角色。定義的權限包括信任原則和權限原則，而且該權限原則無法附加至任何其他IAM實體。

您必須先刪除服務連結角色的相關資源，才能將其刪除。如此可保護您的容量預留機群資源，避免您不小心移除資源的存取許可。

容量保留機群的服務連結角色許可

容量保留叢集使用名為的服務連結角色AWSServiceRoleForEC2CapacityReservationFleet來代表您建立、描述、修改及取消先前由容量保留叢集建立的容量保留。

AWSServiceRoleForEC2CapacityReservationFleet 服務連結角色會信任下列實體擔任該角色：`capacity-reservation-fleet.amazonaws.com`

角色使用包含下列權限的AWSEC2CapacityReservationFleetRolePolicy原則：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeCapacityReservations",
                "ec2:DescribeInstances"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
```

```
            "Action": [
                "ec2:CreateCapacityReservation",
                "ec2:CancelCapacityReservation",
                "ec2:ModifyCapacityReservation"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:capacity-reservation/*"
            ],
            "Condition": {
                "StringLike": {
                    "ec2:CapacityReservationFleet": "arn:aws:ec2:*:*:capacity-
reservation-fleet/crf-*"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:capacity-reservation/*"
            ],
            "Condition": {
                "StringEquals": {
                    "ec2:CreateAction": "CreateCapacityReservation"
                }
            }
        }
    ]
}
```

您必須設定權限，才能允許IAM實體 (例如使用者、群組或角色) 建立、編輯或刪除服務連結角色。如需詳細資訊，請參閱IAM使用指南中的服務連結角色權限。

建立容量保留機群的服務連結角色

您不需要手動建立一個服務連結角色。當您使用`create-capacity-reservation-fleet` AWS CLI 命令或建立容量保留叢集時 `CreateCapacityReservationFleet`API，會自動為您建立服務連結角色。

若您刪除此服務連結角色，之後需要再次建立，您可以在帳戶中使用相同程序重新建立角色。建立容量預留機群時，容量預留機群會再次為您建立服務連結角色。

編輯容量保留機群的服務連結角色

容量保留叢集不允許您編輯 AWSServiceRoleForEC2CapacityReservationFleet 服務連結角色。因為有各種實體可能會參考服務連結角色，所以您無法在建立角色之後變更角色名稱。但是，您可以使用編輯角色的描述IAM。如需詳細資訊，請參閱IAM使用指南中的編輯服務連結角色。

刪除容量保留機群的服務連結角色

若您不再使用需要服務連結角色的功能或服務，我們建議您刪除該角色。如此一來，您就沒有未主動監控或維護的未使用實體。然而，在手動刪除服務連結角色之前，您必須先刪除資源。

> **Note**
>
> 若容量預留機群服務在您試圖刪除資源時正在使用該角色，刪除可能會失敗。若此情況發生，請等待數分鐘後並再次嘗試操作。

若要刪除 AWSServiceRoleForEC2CapacityReservationFleet 服務連結角色

1. 使用`delete-capacity-reservation-fleet` AWS CLI 命令或刪`DeleteCapacityReservationFleet`API除帳戶中的容量保留叢集。
2. 使用IAM主控台 AWS CLI、或刪除 AWSServiceRoleForEC2CapacityReservationFleet 服務連結角色。 AWS API如需詳細資訊，請參閱IAM使用指南中的刪除服務連結角色。

容量保留機群服務連結角色的支援區域

容量預留機群支援在所有提供服務的區域中，使用服務連結角色。如需詳細資訊，請參閱 AWS 區域與端點。

使用指標監視容量保留使用 CloudWatch 量

透過 CloudWatch 指標，您可以透過設定CloudWatch 警示來在滿足使用量閾值時通知您，有效率地監控容量保留並識別未使用的容量。這可協助您維護經常使用的 容量預留 磁碟區，並達到更高層級的使用率。

隨需容量保留會 CloudWatch 每五分鐘傳送一次指標資料。作用中時間少於五分鐘的 容量預留 不支援指標。

如需有關在 CloudWatch 主控台中檢視指標的詳細資訊，請參閱使用 Amazon CloudWatch 指標。如需建立警示的詳細資訊，請參閱建立 Amazon CloudWatch 警示。

目錄

- 容量預留 用量指標
- 容量預留 指標維度
- 檢視容 CloudWatch 量保留的指標

容量預留 用量指標

AWS/EC2CapacityReservations 命名空間包含下列用量指標，可讓您用來監控和維護您針對預留指定之閾值內的隨需容量。

指標	描述
UsedInstanceCount	目前正在使用的執行個體數量。 單位：計數
AvailableInstanceCount	可供使用的執行個體數量。 單位：計數
TotalInstanceCount	您已預留的執行個體總數。 單位：計數
InstanceUtilization	目前正在使用的預留容量執行個體百分比。 單位：百分比

容量預留 指標維度

您可以使用下列維度，在選取的「區域」和帳戶中調整上一個表格中列出的量度。

維度	描述
(無尺寸)	此維度會篩選所有「容量保留」的指定量度。

維度	描述
CapacityReservationId	此維度會篩選已識別「容量保留」的指定測量結果。
InstanceType	此維度會篩選已識別之執行處理類型的指定量度。
AvailabilityZone	此維度會篩選已識別之可用區域的指定測量結果。
InstanceMatchCriteria	此維度會篩選已識別執行處理符合條件 (open或targeted) 的指定量度。
Platform	此維度會篩選已識別平台的指定測量結果資料。
Tenancy	此維度會篩選已識別租用的指定量度。

檢視容 CloudWatch 量保留的指標

指標會先依服務命名空間分組，再依支援的維度分組。您可以使用下列程序檢視 容量預留 的指標。

使用 CloudWatch 主控台檢視容量保留測量結果

1. 在開啟 CloudWatch 主控台https://console.aws.amazon.com/cloudwatch/。
2. 如有必要請變更區域。從導覽列，選取您 容量預留 所在的「區域」。如需詳細資訊，請參閱區域與端點。
3. 在導覽窗格中，選擇 指標。
4. 對於所有量度，請選擇EC2容量保留。
5. 從「所有容量保留」、「依容量保留」、「依執行個體類型」、「依可用區域」、「依平台」、「依執行個體比對準則」或「依租用」的前述量度維度中進行選擇，量度將分別依「無」維度CapacityReservationIdInstanceType、AvailabilityZone、、PlatformInstanceMatch和Tenancy分別進行分組。
6. 若要排序指標，請使用直欄標題。若要將指標圖形化，請選取指標旁的核取方塊。

檢視容量保留指標 (AWS CLI)

使用下列 list-metrics 命令：

```
aws cloudwatch list-metrics --namespace "AWS/EC2CapacityReservations"
```

監視容量保留使用率 EventBridge

AWS Health EventBridge 當您帳戶中的容量保留在特定期間的使用量低於 20% 時，會將事件傳送至 Amazon。使用 EventBridge，您可以建立規則來觸發程式設計動作以回應此類事件。例如您可以建立一個「容量保留在過去 7 天的使用率低於 20% 時，自動取消該容量保留」的規則。

中的事 EventBridge 件表示為JSON物件。事件唯一的欄位包含在JSON物件的「詳細資訊」區段中。"event" 欄位則包含事件名稱。"result" 欄位包含觸發事件之動作的完成狀態。如需詳細資訊，請參閱 Amazon EventBridge 使用者指南中的 Amazon EventBridge 事件模式。

如需詳細資訊，請參閱 Amazon EventBridge 使用者指南。

中不支援此功能 AWS GovCloud (US)。

目錄

- 事件
- 建立 EventBridge 規則

事件

AWS Health 當容量保留的容量使用量低於 20% 時，會傳送下列事件。

事件

- AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION
- AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION_SUMMARY

AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION

以下例子顯示新建立的容量保留在過去 24 小時的使用量低於 20% 時產生的事件。

```
{
    "version": "0",
    "id": "b3e00086-f271-12a1-a36c-55e8ddaa130a",
    "detail-type": "AWS Health Event",
```

```
    "source": "aws.health",
    "account": "123456789012",
    "time": "2023-03-10T12:03:38Z",
    "region": "ap-south-1",
    "resources": [
        "cr-01234567890abcdef"
    ],
    "detail": {
        "eventArn": "arn:aws:health:ap-south-1::event/EC2/
AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION/
AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION_cr-01234567890abcdef-6211-4d50-9286-0c9fbc243f04",
        "service": "EC2",
        "eventTypeCode": "AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION",
        "eventTypeCategory": "accountNotification",
        "startTime": "Fri, 10 Mar 2023 12:03:38 GMT",
        "endTime": "Fri, 10 Mar 2023 12:03:38 GMT",
        "eventDescription": [
            {
                "language": "en_US",
                "latestDescription": "A description of the event will be provided here"
            }
        ],
        "affectedEntities": [
            {
                "entityValue": "cr-01234567890abcdef"
            }
        ]
    }
    }
```

AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION_SUMMARY

以下例子顯示一個或多個容量保留在過去 7 天的使用量低於 20% 時產生的事件。

```
{
    "version": "0", "id":"7439d42b-3c7f-ad50-6a88-25e2a70977e2",
    "detail-type": "AWS Health Event",
    "source": "aws.health",
    "account": "123456789012",
    "time": "2023-03-07T06:06:01Z",
    "region": "us-east-1",
    "resources": [
        "cr-01234567890abcdef | us-east-1b | t3.medium | Linux/UNIX | 0.0%",
        "cr-09876543210fedcba | us-east-1a | t3.medium | Linux/UNIX | 0.0%"
```

```
    ],
    "detail": {
        "eventArn": "arn:aws:health:us-east-1::event/EC2/
AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION_SUMMARY/
AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION_SUMMARY_726c1732-d6f6-4037-b9b8-
bec3c2d3ba65",
        "service": "EC2",
        "eventTypeCode": "AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION_SUMMARY",
        "eventTypeCategory": "accountNotification",
        "startTime": "Tue, 7 Mar 2023 06:06:01 GMT",
        "endTime": "Tue, 7 Mar 2023 06:06:01 GMT",
        "eventDescription": [
            {
                "language": "en_US",
                "latestDescription": "A description of the event will be provided
 here"
            }
        ],
        "affectedEntities": [
            {
                "entityValue": "cr-01234567890abcdef | us-east-1b | t3.medium | Linux/
UNIX | 0.0%"
            },
            {
                "entityValue": "cr-09876543210fedcba | us-east-1a | t3.medium | Linux/
UNIX | 0.0%"
            }
        ]
    }
}
```

建立 EventBridge 規則

若要在容量保留使用率降至 20% 以下時接收電子郵件通知，請建立 Amazon SNS 主題，然後為`AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION`事件建立 EventBridge 規則。

要創建 Amazon SNS 主題

1. 在 https://console.aws.amazon.com/sns/v3/ home 上打開 Amazon SNS 控制台。
2. 在導覽窗格中選擇 Topics (主題)，然後選擇 Create topic (建立主題)。
3. 針對類型，選擇標準。
4. 在名稱中輸入新主題的名稱。

5. 請選擇建立主題。
6. 選擇 Create subscription (建立訂閱)。
7. 在通訊協定中選擇電子郵件，然後在端點中輸入接收通知的電子郵件地址。
8. 選擇 Create subscription (建立訂閱)。
9. 您會在上方輸入的電子郵件地址收到主旨列如下的電子郵件訊息：`AWS Notification - Subscription Confirmation`。請依照指示來確認訂閱。

若要建立 EventBridge 規則

1. 在打開 Amazon EventBridge 控制台https://console.aws.amazon.com/events/。
2. 在導覽窗格中，選擇 Rules (規則)，然後選擇 Create rule (建立規則)。
3. 在名稱中輸入新規則的名稱。
4. 針對規則類型，選擇具有事件模式的規則。
5. 選擇 Next (下一步)。
6. 在事件模式中執行下列動作：
   a. 在事件來源欄位中，選擇 AWS 服務。
   b. 在AWS 服務中選擇 AWS Health。
   c. 針對「事件」型態，選擇EC2ODCR使用率不足通知。
7. 選擇 Next (下一步)。
8. 對目標 1執行下列動作：
   a. 在目標類型欄位中，選擇 AWS 服務。
   b. 針對 [選取目標]，選擇 [SNS主題]。
   c. 為主題選擇您剛剛建立的主題。
9. 選擇下一步，然後再選擇一次下一步。
10. 選擇建立規則。

產能保留使用率通知來源 AWS Health

AWS Health 當您帳戶中「容量保留」的容量使用率降至 20% 以下時，會傳送下列電子郵件與 AWS Health Dashboard 通知。

- 每個新建立的容量保留在過去 24 小時使用率低於 20% 的個別通知。

- 所有容量保留在過去 7 天使用率低於 20% 的摘要通知。

電子郵件通知和 AWS Health Dashboard 通知會傳送至與擁有容量保留的 AWS 帳戶相關聯的電子郵件地址。通知包含以下資訊：

- 容量保留 ID。
- 容量保留的可用區域。
- 容量保留的平均使用率。
- 容量保留的執行個體類型和平台 (作業系統)。

此外，當您帳戶中「容量保留」的產能使用率在 24 小時與 7 天期間內降至 20% 以下時， AWS Health 會將事件傳送至 EventBridge。使用 EventBridge，您可以建立規則來啟動自動動作，例如傳送電子郵件通知或觸發 AWS Lambda 函數，以回應此類事件。如需詳細資訊，請參閱監視容量保留使用率 EventBridge。

ML 的容量區塊

ML 的容量區塊可讓您在 future 日期保留備受歡迎的GPU執行個體，以支援短期機器學習 (ML) 工作負載。在容量區塊內執行的執行個體會自動放置在 Amazon 內部 EC2 UltraClusters，以實現低延遲、PB 級、無阻塞的聯網。

使用容量區塊，您可以查看 future 日期何時可用GPU執行個體容量，並且可以排定容量區塊在最適合您的時間啟動。當您保留容量區塊時，您可以獲得可預測的GPU執行個體容量保證，同時只需支付所需時間的費用。如果您需GPUs要一次支援數天或數週的 ML 工作負載，而且不想在GPU執行個體不使用時支付保留費用，我們建議您使用容量區塊。

以下是容量區塊的一些常見使用案例：

- ML 模型訓練和微調 — 不間斷地存取您保留的GPU執行個體，以完成 ML 模型訓練和微調。
- ML 實驗和原型 — 執行實驗並建立需要短時間GPU執行個體的原型。

容量區塊目前可用於`p5.48xlarge`和`p4d.24xlarge`執行個體。這些`p5.48xlarge`執行個體可在美國東部 (俄亥俄) 和美國東部 (維吉尼亞北部) 區域使用。這些`p4d.24xlarge`執行個體可在美國東部 (俄亥俄) 和美國西部 (奧勒岡) 區域使用。您可以保留最多未來八週內的容量區塊。

您可以使用容量區塊來預留具有下列保留持續時間`p5`和`p4d`執行個體數量選項的執行個體。

- 1 天增量的預訂持續時間，最多 14 天和 7 天增量，最多共 28 天

- 1、2、4、8、16、32 或 64 個執行個體的保留執行個體數量選項

若要預留容量區塊，請先指定容量需求，包括執行個體類型、執行個體數量、時間量、最早開始日期，以及所需的最晚結束日期。您就可以看到符合您規格的可用容量區塊方案。容量區塊方案包含詳細資訊，例如開始時間、可用區域和保留價格。容量區塊方案的價格取決於交付方案時的可用供給與需求。保留容量區塊後，價格不會變更。如需詳細資訊，請參閱容量區塊定價和計費。

購買容量區塊方案時，系統會根據您選取的日期和執行個體數量保留。容量區塊保留開始時，您可以在啟動請求中指定保留 ID，鎖定執行個體啟動。

您可以使用所有保留的執行個體，直到容量區塊結束時間前 30 分鐘為止。容量區塊保留剩 30 分鐘時，我們就會開始終止容量區塊中執行的所有執行個體。我們會利用這段時間清理您的執行個體，再將容量區塊交付給下一位客戶。保留的最後 30 分鐘不會以容量區塊的價格收費。我們會在終止程序開始前 EventBridge 10 分鐘發出事件。如需詳細資訊，請參閱使用監視容量區塊 EventBridge。

主題

- 支援平台
- 考量事項
- 相關資源
- 容量區塊定價和計費
- 使用容量區塊
- 使用監視容量區塊 EventBridge
- 記錄容量區塊API呼叫 AWS CloudTrail

支援平台

ML 的容量區塊目前支援，以`p5.48xlarge`及具有預設租用的`p4d.24xlarge`執行個體。當您使用購買 AWS Management Console 容量區塊時，預設平台選項為 LinuxUNIX/。當您使用 AWS Command Line Interface (AWS CLI) 或購買 AWS SDK容量區塊時，可以使用下列平台選項：

- Linux/UNIX
- Red Hat Enterprise Linux
- RHEL與醫管局
- SUSELinux
- Ubuntu Pro

考量事項

使用容量區塊之前，請考慮下列詳細資訊和限制。

- 容量區塊在協調世界時間上午 11:30 開始和結束 () UTC。
- 在容量區塊中執行的執行個體的終止程序會在保留的最後一天上午 11:00 開始協調世界時間 (UTC)。
- 可以保留最多未來 8 週內的容量區塊。
- 不可修改和取消容量區塊。
- 容量區塊無法跨 AWS 帳戶或 AWS 組織內共用。
- 容量區塊無法在容量保留群組中使用。
- 在特定日期， AWS 組織中所有帳戶的容量區塊中可保留的執行個體總數不得超過 64 個執行個體。
- 若要使用容量區塊，執行個體必須專門鎖定保留 ID。
- 容量區塊中的執行個體不會計入您的隨需執行個體限制。
- 對於使用自訂的 P5 執行個體AMI，請確定您具備所需的軟體和組態。EFA
- 對於 Amazon EKS 受管節點群組，請參閱使用適用於 ML 的 Amazon EC2 容量區塊建立受管節點群組。對於 Amazon EKS 自我管理節點群組，請參閱將容量區塊與自我管理節點搭配使用 ML。

相關資源

建立容量區塊之後，您可以使用容量區塊執行下列作業：

- 將執行個體啟動至容量區塊。如需詳細資訊，請參閱在容量區塊中啟動執行個體。
- 創建一個 Amazon EC2 Auto Scaling 組。如需詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南中的機器學習工作負載使用容量區塊。

> **Note**
>
> 如果您使用 Amazon EC2 Auto Scaling 或 AmazonEKS，您可以排程擴展以在容量區塊保留開始時執行。透過排定的擴充功能， AWS 會自動為您處理重試，因此您不必擔心實作重試邏輯來處理暫時性失敗。

- 使用 AWS ParallelCluster. 如需詳細資訊，請參閱使用 AWS ParallelCluster 和適用於 ML 的 Amazon EC2 容量區塊來增強 ML 工作流程。

如需有關的詳細資訊 AWS ParallelCluster，請參閱什麼是 AWS ParallelCluster。

容量區塊定價和計費

使用適用於 ML 的 Amazon EC2 容量區塊，您只需為保留的金額付費。容量區塊的價格取決於購買時容量區塊的可用供給與需求。您可以在保留前檢視容量區塊方案的價格。我們會在保留時預先收取容量區塊的費用。當您在某個日期範圍內搜尋容量區塊時，我們會傳回可用的最低價格容量區塊方案。保留容量區塊之後，價格不會變更。

使用容量區塊時，您會為執行個體執行時使用的作業系統付費。如需有關作業系統價格的詳細資訊，請參閱 ML 定價的 Amazon EC2 容量區塊。

帳單

我們會預先收取容量區塊方案的費用。您購買容量區塊後 12 小時內，我們會向您的 AWS 帳戶收取費用。處理您的付款時，您的容量區塊保留資源會維持 `payment-pending` 狀態。如果無法在 12 小時內處理您的付款，您的容量區塊就會釋出，且保留狀態會變更為 `payment-failed`。

成功處理您的付款後，容量區塊資源狀態會從 `payment-pending` 變更為 `scheduled`。您會收到反映一次性預付款的發票。在發票中，您可以將已付金額與容量區塊保留 ID 相關聯。

容量區塊保留開始時，系統只會根據保留中執行個體執行時，您使用的作業系統向您收費。您可以在 AWS Cost and Usage Report的年度帳單中查看使用月份的使用量和相關費用。

> **Note**
>
> Savings Plans 和預留執行個體折扣不適用於容量區塊。

檢視您的帳單

您可以在 AWS Billing and Cost Management 主控台中檢視帳單。容量區塊的預付款會顯示在您購買保留的月份。

保留開始後，您的帳單會分別顯示區塊保留已使用和未使用時間的明細。您可以透過這些明細項目，查看保留中使用了多少時間。如果您使用付費作業系統，只會在明細中看到已使用時間的使用費用。如需詳細資訊，請參閱容量區塊定價和計費。未使用時間不會收費。

如需詳細資訊，請參閱 AWS Billing and Cost Management 使用者指南中的檢視您的帳單。

如果您的容量區塊在與購買保留時不同的月份開始，預付費用和保留用量會顯示在不同的帳單月份下。在您的中 AWS Cost and Usage Report，產能區塊預留識別碼會列在預付費用的預留/預留ARN明細行項目，以及週年紀念帳單中的明細行項目/ResourceID 中，以便您可以將用量與對應的預付價格產生關聯。

使用容量區塊

若要開始使用容量區塊，您要先尋找並購買符合您保留大小、時間長度和時間需求的可用容量區塊。接著，保留開始時，您可以透過啟動鎖定保留 ID 的執行個體，使用容量區塊。在保留到期前三十分鐘，我們會開始終止任何仍在容量區塊中執行的執行個體。

容量區塊會在單一可用區域中以 `targeted` 容量保留的形式提供。若要在容量區塊中執行執行個體，您必須在啟動執行個體時指定保留 ID。如果您自行停止執行個體且容量區塊到期，則必須先鎖定狀態為 `active` 的另一個容量區塊，才能重新啟動執行個體。

容量區塊預設會在容量區塊內的執行個體之間提供低延遲、高輸送量的網路連線，因此不需要搭配容量區塊使用叢集置放群組。

主題

- 必要條件
- 尋找和購買容量區塊
- 在容量區塊中啟動執行個體
- 檢視容量區塊

必要條件

您必須 AWS 區域 針對要使用的執行個體類型使用對應的。如需詳細資訊，請參閱區域。

下列提供具有`p5.48xlarge`執行個體的容量區塊 AWS 區域。

區域名稱	區域代碼
美國東部 (俄亥俄)	us-east-2
美國東部 (維吉尼亞北部)	us-east-1

下列提供具有`p4d.24xlarge`執行個體的容量區塊 AWS 區域。

區域名稱	區域代碼
美國東部 (俄亥俄)	us-east-2
美國西部 (奧勒岡)	us-west-2

> **Note**
>
> 並非所有執行個體類型都支援 64 個執行個體的容量區塊大小 AWS 區域。

尋找和購買容量區塊

若要保留容量區塊，您必須先找到符合您需求的可用容量時間區塊。若要尋找可供預留的容量區塊，請指定下列項目。

- 您需要的執行個體數量
- 您需要執行個體的時間長度
- 您需要保留的日期範圍

若要搜尋可用的容量區塊方案，請指定保留時間長度和執行個體數量。您必須選取下列任一選項。

- 對於預訂時間-以 1 天為單位最多 14 天或以 7 天為單位最多 28 天
- 例如計數 — 1、2、4、8、16、32 或 64 個執行個體

如果可用的容量區塊符合您的規格，我們會傳回單一容量區塊方案的詳細資訊。方案詳細資訊包括保留的開始時間、保留的可用區域，以及保留的價格。如需詳細資訊，請參閱容量區塊定價和計費。

您可以購買顯示的容量區塊方案，也可以修改搜尋條件以查看其他可用選項。方案沒有預先設定的到期時間，但僅採先到先用方式。

購買容量區塊方案時，您會立即獲得回應，確認已保留您的容量區塊。確認後，您會在帳戶中看到新的容量保留，保留類型為 `capacity-block`，且 `start-date` 設定為您所購買方案的開始時間。保留您的容量區塊時，其狀態為 `payment-pending`。成功處理預付款後，保留狀態會變更為 `scheduled`。如需詳細資訊，請參閱帳單。

您可以使用下列任一方法來尋找和購買容量區塊。

Console

使用主控台尋找和購買容量區塊

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在畫面頂端的導覽列中，選取一個 AWS 區域。此選項很重要，因為並非所有區域中的所有執行個體類型都支援 64 個執行個體的容量區塊大小。

3. 在導覽窗格中，依序選擇容量保留和購買容量區塊。
4. 在容量屬性中，您可以設定容量區塊搜尋參數。默認情況下，該平台是 Linux。如果您要選取不同的作業系統，請使用 AWS CLI。如需詳細資訊，請參閱支援平台。
5. 在總容量中，選取您要保留的執行個體數量。
6. 在持續時間中，輸入您需要保留的天數。
7. 在「日期範圍」下，以搜尋「產能區塊」，輸入您要預留開始的最早日期。
8. 選擇尋找容量區塊。
9. 如果可用的容量區塊符合您的規格，您會在建議的容量區塊下看到方案。如果有多個符合您規格的供應項目，則會顯示最早可用的容量區塊供應項目。若要檢視其他容量區塊方案，請調整您的搜尋輸入內容，並再次選擇尋找容量區塊。
10. 找到要購買的容量區塊方案時，請選擇下一步。
11. (選用) 在新增標籤頁面上，選擇新增標籤。
12. 檢閱和購買頁面會列出開始和結束日期、時間長度、執行個體總數和價格。

> **Note**
> 容量區塊在您保留之後就無法修改或取消。

13. 在購買容量區塊快顯視窗中，輸入「confirm」，然後選擇購買。

AWS CLI

若要使用尋找容量區塊 AWS CLI

使用 `describe-capacity-block-offerings` 命令。

下列範例會搜尋具有 16 個 `p5.48xlarge` 執行個體的容量區塊，日期範圍從 `2023-08-14` 到 `2023-10-22`，時間長度為 48 小時。執行個體數量必須為預先設定選項 1、2、4、8、16、32、64 之中的整數。容量時間長度必須是介於 24 到 336 間，24 的倍數整數，以小時表示天數。

```
aws ec2 describe-capacity-block-offerings --instance-type p5.48xlarge \
  --instance-count 16 --start-date-range 2023-08-14T00:00:00Z \
  --end-date-range 2023-10-22-T00:00:00Z --capacity-duration 48
```

若要使用購買容量區塊 AWS CLI

使用 `purchase-capacity-block` 命令，然後指定您要購買之容量區塊的方案 ID，以及執行個體平台。

```
aws ec2 purchase-capacity-block \
  --capacity-block-offering-id cbr-0123456789abcdefg \
  --instance-platform Linux/UNIX
```

在容量區塊中啟動執行個體

若要使用容量區塊，您必須在啟動執行個體時指定容量區塊保留 ID。在容量區塊中啟動執行個體，會依啟動的執行個體數量減少可用容量。舉例來說，如果您購買的執行個體容量為八個執行個體，而您啟動四個執行個體，則可用容量會減少四個。

如果您在保留結束之前終止在容量區塊中執行的執行個體，可以在其位置啟動新執行個體。當您停止或終止容量區塊中的執行個體時，需要幾分鐘時間清理執行個體，才能啟動另一個執行個體加以取代。在此期間，您的執行個體將處於 stopping 或 `shutting-down` 狀態。完成此程序後，您的執行個體狀態會變更為 `stopped` 或 `terminated`。接著，容量區塊中的可用容量將會更新，顯示另一個可用的執行個體。

如需如何使用容量區塊設定EKS受管節點群組的詳細資訊，請參閱 Amazon EKS 使用者指南中的使用 ML 容量區塊建立受管節點群組。

如需如何 AWS ParallelCluster 使用容量區塊設定的詳細資訊，請參閱上的 ML AWS ParallelCluster。

如需如何使用EC2叢集將執行個體啟動至容量區塊的相關資訊，請參閱教學課程：設定EC2叢集以將執行個體啟動至容量區塊。

如需如何建立鎖定容量區塊的啟動範本資訊，請參閱：在 Amazon 啟動範本中存放執行個體EC2啟動參數。

下列步驟說明如何使用 AWS Management Console 或將執行個體啟動到`active`狀態中的容量區塊 AWS CLI。

Console

使用主控台在容量區塊中啟動執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在螢幕上方的導覽列上，為容量區塊保留選取區域。

3. 在 Amazon 主EC2控台儀表板中，選擇啟動執行個體。
4. (選用) 在名稱和標籤中，您可以為執行個體命名並標記執行個體。如需標籤的相關資訊，請參閱標記您的 Amazon EC2 資源
5. 在應用程式和作業系統映像下，選取 Amazon 機器映像 (AMI)。
6. 在執行個體類型中，選取適合您容量區塊保留的執行個體類型。
7. 在金鑰對 (登入) 中，選擇現有的金鑰對，或選擇建立新的金鑰對以建立新的金鑰對。如需詳細資訊，請參閱Amazon EC2 密鑰對和 Amazon EC2 實例。
8. 在 Network settings (網路設定) 下，使用預設設定，或視需要選擇 Edit (編輯) 以便設定網路設定。

   > **Important**
   >
   > 您的執行個體無法在與容量區塊所在可用區域不同的可用區域子網路中啟動。

9. 在進階詳細資訊中，設定執行個體，如下所示。
   a. 在購買選項 (市場類型) 中，選取容量區塊。
   b. 在容量保留中，選取依據 ID 的目標。
   c. 選取容量區塊保留的容量保留 ID。
10. 在 Summary (摘要) 面板上，針對 Number of instances (執行個體的數量)，輸入要啟動的執行個體數量。
11. 選擇啟動執行個體。

AWS CLI

使用將執行個體啟動到容量區塊 AWS CLI

- 使用 `run-instances` 命令，然後在 `instance-market-options` 結構中將 `MarketType` 指定為 `capacity-block`。您也必須指定 `capacity-reservation-specification` 參數。

  以下範例會在具有相符屬性和可用容量的作用中容量區塊中，啟動單一 `p5.48xlarge` 執行個體。

```
aws ec2 run-instances --image-id ami-abc12345 --count 1 \
  --instance-type p5.48xlarge --key-name MyKeyPair \
  --subnet-id subnet-1234567890abcdef1 \
  --instance-market-options MarketType='capacity-block'
```

```
 --capacity-reservation-specification
CapacityReservationTarget={CapacityReservationId=cr-a1234567}
```

檢視容量區塊

保留容量區塊後，您可以在 AWS 帳戶中檢視容量區塊保留。您可以檢視 `start-date` 和 `end-date`，查看保留開始和結束時間。在容量區塊保留開始之前，可用容量會顯示為零。您可以依標籤索引鍵 `aws:ec2capacityreservation:incrementalRequestedQuantity` 的標籤值，查看容量區塊中可用的執行個體數量。

容量區塊保留開始時，保留狀態會從 `scheduled` 變更為 `active`。我們透過 Amazon 發出事件 EventBridge，通知您可以使用容量區塊。如需詳細資訊，請參閱使用監視容量區塊 EventBridge。

容量區塊具有下列狀態：

- `payment-pending` – 尚未處理預付款。
- `payment-failed` – 無法在 12 小時內處理付款。您的容量區塊已釋出。
- `scheduled` – 已處理付款，容量區塊保留尚未開始。
- `active` – 預留容量可供使用。
- `expired` – 容量區塊保留在保留請求中指定的日期和時間自動過期。預留容量將不再提供使用。

您可以使用下列任一種方法來檢視容量區塊預留。

Console

使用主控台檢視容量區塊

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 容量預留。
3. 在容量保留概觀頁面上，您會看到一個資源表格，其中包含所有容量保留資源的詳細資訊。若要尋找您的容量區塊保留，請從容量保留 ID上方的下拉式清單中選取容量區塊。在表格中，您可以查看容量區塊的相關資訊，例如開始和結束日期、時間長度和狀態。
4. 如需容量區塊的詳細資訊，請選取要檢視之容量區塊的保留 ID。容量保留詳細資訊頁面會顯示保留項目的所有屬性，以及容量區塊中使用中和可用的執行個體數量。

> **Note**
>
> 在容量區塊保留開始之前，可用容量會顯示為零。您可以為標籤鍵使用下列標籤值，以查看容量區塊保留開始時可用的執行個體數量：`aws:ec2capacityreservation:incrementalRequestedQuantity`。

AWS CLI

使用檢視容量區塊的步驟 AWS CLI

依預設，當您使用describe-capacity-reservations命令時，會同時列出隨需容量保留和容量區塊保留區。若只要檢視容量區塊保留，請為 `capacity-reservation-type` 參數使用 `capacity-block` 來篩選。

例如，下列命令說明目前的一或多個容量區塊保留項目 AWS 區域。

```
aws ec2 describe-capacity-reservations -reservation-type capacity-block
```

範例輸出。

```
{
    "CapacityReservations": [
        {
            "CapacityReservationId": "cr-12345678",
            "EndDateType": "limited",
            "ReservationType": "capacity-block"
            "AvailabilityZone": "eu-east-2a",
            "InstanceMatchCriteria": "targeted",
            "EphemeralStorage": false,
            "CreateDate": "2023-11-29T14:22:45Z",
            "StartDate": "2023-12-15T12:00:00Z",
            "EndDate": "2023-08-19T12:00:00Z",
            "AvailableInstanceCount": 0,
            "InstancePlatform": "Linux/UNIX",
            "TotalInstanceCount": 16,
            "State": "payment-pending",
            "Tenancy": "default",
            "EbsOptimized": true,
            "InstanceType": "p5.48xlarge"
        },
```

```
        ...
```

使用監視容量區塊 EventBridge

當您的容量區塊保留開始時，Amazon EC2 將發出一個事件，表示您的 EventBridge 容量已準備就緒可供使用。在容量區塊保留結束前四十分鐘，您會收到另一個 EventBridge 事件，告訴您在保留區中執行的任何執行個體將在 10 分鐘內開始終止。如需有關 EventBridge 事件的詳細資訊，請參閱 Amazon EventBridge 活動。

以下事件為針對容量區塊所發出事件的結構：

已提供容量區塊

以下範例說明已提供容量區塊的事件。

```
{
  "customer_event_id": "[Capacity Reservation Id]-delivered",
  "detail_type": "Capacity Block Reservation Delivered",
  "source": "aws.ec2",
  "account": "[Customer Account ID]",
  "time": "[Current time]",
  "resources": [
    "[ODCR ARN]"
  ],
  "detail": {
    "capacity-reservation-id": "[ODCR ID]",
    "end-date": "[ODCR End Date]"
  }
}
```

容量區塊到期警告

以下範例說明容量區塊到期警告的事件。

```
{
  "customer_event_id": "[Capacity Reservation Id]-approaching-expiry",
  "detail_type": "Capacity Block Reservation Expiration Warning",
  "source": "aws.ec2",
  "account": "[Customer Account ID]",
  "time": "[Current time]",
  "resources": [
    "[ODCR ARN]"
```

```
  ],
  "detail": {
    "capacity-reservation-id": "[ODCR ID]",
    "end-date": "[ODCR End Date]"
  }
}
```

記錄容量區塊API呼叫 AWS CloudTrail

容量區塊與一項服務整合 AWS CloudTrail，可提供使用者、角色或服務在容量區塊中所採取的動作記錄的 AWS 服務。 CloudTrail 將容量區塊的API呼叫擷取為事件。擷取的呼叫包括來自容量區塊主控台的呼叫，以及對容量區塊API作業的程式碼呼叫。如果您建立追蹤，您可以啟用持續交付 CloudTrail 事件到 Amazon S3 儲存貯體，包括容量區塊的事件。如果您未設定追蹤，您仍然可以在 [事件歷程記錄] 中檢視 CloudTrail 主控台中最近的事件。使用收集的資訊 CloudTrail，您可以判斷對容量區塊提出的要求、提出要求的來源 IP 位址、提出要求的人員、提出要求的時間，以及其他詳細資訊。

若要進一步了解 CloudTrail，請參閱使AWS CloudTrail 用者指南。

容量區塊資訊 CloudTrail

CloudTrail 在您創建帳戶 AWS 帳戶 時啟用。當活動發生在容量區塊中時，該活動會與 CloudTrail 事件歷史記錄中的其他 AWS 服務事件一起記錄在事件中。您可以查看，搜索和下載最近的事件 AWS 帳戶。如需詳細資訊，請參閱使用 CloudTrail 事件歷程記錄檢視事件。

如需您 AWS 帳戶的事件的持續記錄 (包括容量區塊的事件)，請建立追蹤。追蹤可 CloudTrail 將日誌檔交付到 Amazon S3 儲存貯體。依預設，當您在主控台中建立追蹤時，該追蹤會套用至所有的 AWS 區域。追蹤記錄來自 AWS 分區中所有區域的事件，並將日誌檔傳送到您指定的 Amazon S3 儲存貯體。此外，您還可以設定其他 AWS 服務，以進一步分析 CloudTrail 記錄中收集的事件資料並採取行動。如需詳細資訊，請參閱下列內容：

- 建立追蹤的概觀
- CloudTrail 支援的服務與整合
- 設定 Amazon SNS 通知 CloudTrail
- 從多個區域接收 CloudTrail 日誌文件並從多個帳戶接收 CloudTrail 日誌文件

所有容量區塊動作均由 Amazon 參考記錄， CloudTrail 並記錄在 Amazon EC2 API 參考中。例如，呼叫和`CapacityBlockActive`動作會`CapacityBlockScheduled`在 CloudTrail 記錄檔中產生項目。

每一筆事件或日誌專案都會包含產生請求者的資訊。身分資訊可協助您判斷下列事項：

- 要求是使用 root 或 AWS Identity and Access Management (IAM) 使用者認證提出的。
- 提出該請求時，是否使用了特定角色或聯合身分使用者的暫時安全憑證。
- 請求是否由其他 AWS 服務提出。

如需詳細資訊，請參閱CloudTrail userIdentity 元素。

了解容量區塊日誌檔案項目

追蹤是一種組態，可讓事件以日誌檔的形式傳遞到您指定的 Amazon S3 儲存貯體。 CloudTrail 記錄檔包含一或多個記錄項目。事件代表來自任何來源的單一請求，包括有關請求的操作，動作的日期和時間，請求參數等信息。 CloudTrail 日誌文件不是公共API調用的有序堆棧跟踪，因此它們不會以任何特定順序顯示。

下列範例顯示下列項目的 CloudTrail 記錄項目：

- TerminateCapacityBlocksInstances
- CapacityBlockPaymentFailed
- CapacityBlockScheduled
- CapacityBlockActive
- CapacityBlockFailed
- CapacityBlockExpired

> **Note**
> 已修改資料隱私權範例的某些欄位。

TerminateCapacityBlocksInstances

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "accountId": "123456789012",
    "invokedBy": "AWS Internal;"
  },
  "eventTime": "2023-10-02T00:06:08Z",
  "eventSource": "ec2.amazonaws.com",
```

```
  "eventName": "TerminateCapacityBlockInstances",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "203.0.113.25",
  "userAgent": "aws-cli/1.15.61 Python/2.7.10 Darwin/16.7.0 botocore/1.10.60",
  "requestParameters": null,
  "responseElements": null,
  "eventID": "a1b2c3d4-EXAMPLE",
  "readOnly": false,
  "resources": [
    {
      "accountId": "123456789012",
      "type": "AWS::EC2::Instance",
      "ARN": "arn:aws:ec2:US East (N. Virginia):123456789012:instance/
i-1234567890abcdef0"
    }
    {
      "accountId": "123456789012",
      "type": "AWS::EC2::Instance",
      "ARN": "arn:aws::ec2:US East (N. Virginia):123456789012:instance/
i-0598c7d356eba48d7"
    }
  ],
  "eventType": "AwsServiceEvent",
  "recipientAccountId": "123456789012",
  "serviceEventDetails": {
      "capacityReservationId": "cr-12345678",
      }
}
```

CapacityBlockPaymentFailed

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "accountId": "123456789012",
    "invokedBy": "AWS Internal;"
  },
  "eventTime": "2023-10-02T00:06:08Z",
  "eventSource": "ec2.amazonaws.com",
  "eventName": "CapacityBlockPaymentFailed",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "203.0.113.25",
  "userAgent": "aws-cli/1.15.61 Python/2.7.10 Darwin/16.7.0 botocore/1.10.60",
```

```
  "requestParameters": null,
  "responseElements": null,
  "eventID": "a1b2c3d4-EXAMPLE",
  "readOnly": false,
  "resources": [
    {
      "ARN": "arn:aws:ec2:US East (N. Virginia):123456789012:capacity-reservation/
cr-12345678",
      "accountId": "123456789012",
      "type": "AWS::EC2::CapacityReservation"
    }
  ],
  "eventType": "AwsServiceEvent",
  "recipientAccountId": "123456789012",
  "serviceEventDetails": {
      "capacityReservationId": "cr-12345678",
      "capacityReservationState": "payment-failed"
      }
}
```

CapacityBlockScheduled

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "accountId": "123456789012",
    "invokedBy": "AWS Internal;"
  },
  "eventTime": "2023-10-02T00:06:08Z",
  "eventSource": "ec2.amazonaws.com",
  "eventName": "CapacityBlockScheduled",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "203.0.113.25",
  "userAgent": "aws-cli/1.15.61 Python/2.7.10 Darwin/16.7.0 botocore/1.10.60",
  "requestParameters": null,
  "responseElements": null,
  "eventID": "a1b2c3d4-EXAMPLE",
  "readOnly": false,
  "resources": [
    {
      "ARN": "arn:aws:ec2:US East (N. Virginia):123456789012:capacity-reservation/
cr-12345678",
      "accountId": "123456789012",
```

```
      "type": "AWS::EC2::CapacityReservation"
    }
  ],
  "eventType": "AwsServiceEvent",
  "recipientAccountId": "123456789012",
  "serviceEventDetails": {
      "capacityReservationId": "cr-12345678",
      "capacityReservationState": "scheduled"
      }
}
```

CapacityBlockActive

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "accountId": "123456789012",
    "invokedBy": "AWS Internal;"
  },
  "eventTime": "2023-10-02T00:06:08Z",
  "eventSource": "ec2.amazonaws.com",
  "eventName": "CapacityBlockActive",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "203.0.113.25",
  "userAgent": "aws-cli/1.15.61 Python/2.7.10 Darwin/16.7.0 botocore/1.10.60",
  "requestParameters": null,
  "responseElements": null,
  "eventID": "a1b2c3d4-EXAMPLE",
  "readOnly": false,
  "resources": [
    {
      "ARN": "arn:aws:ec2:US East (N. Virginia):123456789012:capacity-reservation/
cr-12345678",
      "accountId": "123456789012",
      "type": "AWS::EC2::CapacityReservation"
    }
  ],
  "eventType": "AwsServiceEvent",
  "recipientAccountId": "123456789012",
  "serviceEventDetails": {
      "capacityReservationId": "cr-12345678",
      "capacityReservationState": "active"
      }
```

```
 }
```

CapacityBlockFailed

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "accountId": "123456789012",
    "invokedBy": "AWS Internal;"
  },
  "eventTime": "2023-10-02T00:06:08Z",
  "eventSource": "ec2.amazonaws.com",
  "eventName": "CapacityBlockFailed",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "203.0.113.25",
  "userAgent": "aws-cli/1.15.61 Python/2.7.10 Darwin/16.7.0 botocore/1.10.60",
  "requestParameters": null,
  "responseElements": null,
  "eventID": "a1b2c3d4-EXAMPLE",
  "readOnly": false,
  "resources": [
    {
      "ARN": "arn:aws:ec2:US East (N. Virginia):123456789012:capacity-reservation/
cr-12345678",
      "accountId": "123456789012",
      "type": "AWS::EC2::CapacityReservation"
    }
  ],
  "eventType": "AwsServiceEvent",
  "recipientAccountId": "123456789012",
  "serviceEventDetails": {
      "capacityReservationId": "cr-12345678",
      "capacityReservationState": "failed"
      }
 }
```

CapacityBlockExpired

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "accountId": "123456789012",
    "invokedBy": "AWS Internal;"
```

```
  },
  "eventTime": "2023-10-02T00:06:08Z",
  "eventSource": "ec2.amazonaws.com",
  "eventName": "CapacityBlockExpired",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "203.0.113.25",
  "userAgent": "aws-cli/1.15.61 Python/2.7.10 Darwin/16.7.0 botocore/1.10.60",
  "requestParameters": null,
  "responseElements": null,
  "eventID": "a1b2c3d4-EXAMPLE",
  "readOnly": false,
  "resources": [
    {
      "ARN": "arn:aws:ec2:US East (N. Virginia):123456789012:capacity-reservation/
cr-12345678",
      "accountId": "123456789012",
      "type": "AWS::EC2::CapacityReservation"
    }
  ],
  "eventType": "AwsServiceEvent",
  "recipientAccountId": "123456789012",
  "serviceEventDetails": {
      "capacityReservationId": "cr-12345678",
      "capacityReservationState": "expired"
      }
 }
```

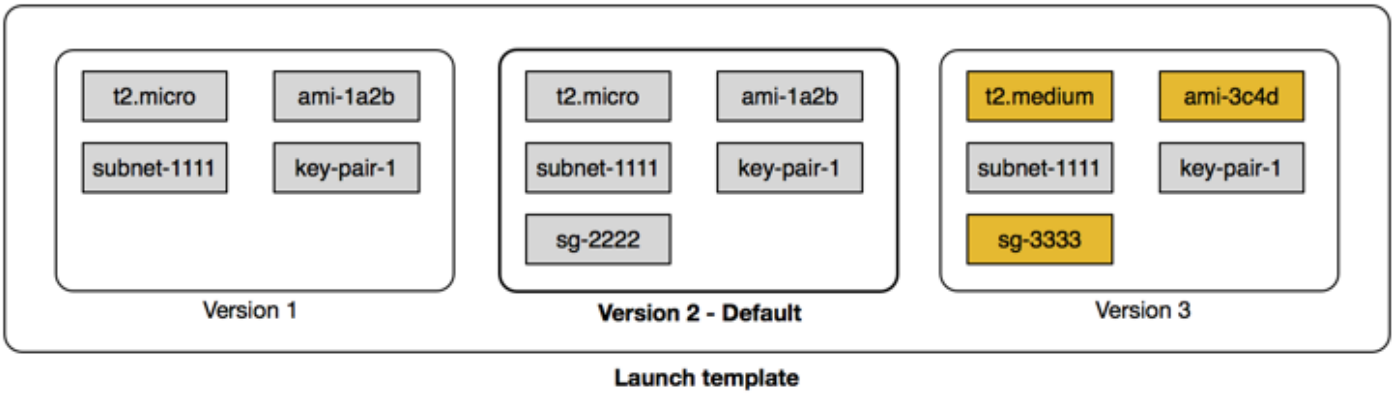

在 Amazon 啟動範本中存放執行個體EC2啟動參數

您可以使用 Amazon EC2 啟動範本來存放執行個體啟動參數，這樣就不必在每次啟動 Amazon EC2 執行個體時指定它們。例如，您可以建立一個啟動範本，以儲存您通常用於啟動執行個體的 AMI ID、執行個體類型和網路設定。使用 Amazon EC2 主控台、或命令列工具啟動執行個 AWS SDK體時，您可以指定啟動範本，而不必再次輸入參數。

針對每個啟動範本，您可以建立一或多個已編號的 啟動範本版本。每個版本都可以有不同的啟動參數。當您從啟動範本啟動執行個體時，可以使用任意版本的啟動範本。如果您未指定版本，則會使用預設版本。您可以設定任意版本的啟動範本做為預設—版本，而該版本即預設為啟動範本的第一個版本。

下圖顯示具有三個版本的啟動範本。第一個版本會指定用來啟動執行個體的執行個體類型、AMI ID、子網路和 key pair。第二個版本以第一個版本為基礎，同時指定執行個體的安全群組。第三個版本針對

部分參數使用不同的值。第 2 版設定為預設版本。如果您已從此啟動範本啟動執行個體，則會在未指定其他版本時使用第 2 版的啟動參數。



目錄

- Amazon EC2 啟動模板的限制
- IAMAmazon EC2 啟動模板所需的許可
- 使用 Amazon EC2 啟動範本控制啟動 Amazon EC2 執行個體
- 創建一個 Amazon EC2 啟動模板
- 修改啟動範本 (管理啟動範本版本)
- 刪除啟動範本或啟動範本版本

Amazon EC2 啟動模板的限制

下列限制適用於啟動範本與啟動範本版本：

- 配額 — 若要檢視啟動範本的配額和啟動範本版本，請開啟 Service Quotas 主控台或使用 list-service-quotas AWS CLI 指令。每個 AWS 帳戶每個區域最多可以有 5,000 個啟動範本，每個啟動範本最多可有 10,000 個版本。您帳戶的配額可能不太一樣，視年齡和使用記錄而定。
- 參數是選用值 – 啟動範本參數是選用的。不過，您必須確保執行個體啟動要求包含所有必要參數。例如，如果您的啟動範本不包含 AMI ID，您必須在使用此啟動範本啟動執行個體時指定 AMI ID。
- 參數尚未驗證 – 在您建立啟動範本時，啟動範本參數尚未完整驗證。如果您指定不正確的值或使用不支援的參數組合，則無法使用此啟動範本啟動實例。若要避免發生問題，請務必指定正確的值並使用支援的參數組合。例如，若要在置放群組中啟動執行個體，您必須指定支援的執行個體類型。
- 標籤 — 您可以標記啟動範本，但無法標記啟動範本版本。
- 固定 – 啟動範本是不可變的。若要修改啟動範本，您必須建立新版本的啟動範本。

- 版本編號 – 啟動範本版本會依其建立順序進行編號。建立啟動範本版本時，您無法自行指定版本號碼。

IAMAmazon EC2 啟動模板所需的許可

您可以使用IAM權限來控制使用者是否可以列出、檢視、建立或刪除啟動範本或啟動範本版本。

> **Important**
>
> 您無法使用資源層級權限來限制使用者在建立啟動範本或啟動範本版本時可在啟動範本中指定的資源。因此，請確定只有受信任的管理員獲得建立啟動範本和啟動範本版本的權限。

您必須將使用啟動範本的任何人授與建立和存取啟動範本中指定之資源所需的權限。例如：

- 若要從共用私有 Amazon Machine 映像 (AMI) 啟動執行個體，使用者必須擁有AMI.
- 若要從現有快照建立含標籤的EBS磁碟區，使用者必須擁有快照的讀取權限，以及建立和標記磁碟區的權限。

目錄

- ec2：CreateLaunchTemplate
- ec2：DescribeLaunchTemplates
- ec2：DescribeLaunchTemplateVersions
- ec2：DeleteLaunchTemplate
- 控制版本控制許可
- 控制啟動範本上標籤的存取權

ec2：CreateLaunchTemplate

若要在主控台中建立啟動範本或使用APIs，主參與者必須具有IAM原則中的`ec2:CreateLaunchTemplate`權限。盡可能使用標籤來協助控制帳戶中啟動範本的存取權。

例如，下列IAM原則陳述式僅在範本使用指定的標籤時，才授與主參與者建立啟動範本的權限 (##=##).

```
{
    "Sid": "IAMPolicyForCreatingTaggedLaunchTemplates",
```

```
    "Action": "ec2:CreateLaunchTemplate",
    "Effect": "Allow",
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "aws:ResourceTag/purpose": "testing"
        }
    }
}
```

建立啟動範本的主體可能需要一些相關許可，例如：

- ec2: CreateTags — 要在`CreateLaunchTemplate`操作期間將標籤添加到啟動模板，`CreateLaunchTemplate`調用者必須具有IAM策略中的`ec2:CreateTags`許可。
- ec2: RunInstances — 若要從他們建立的啟動範本啟動EC2執行個體，主體也必須具有IAM政策中的`ec2:RunInstances`權限。

針對套用標籤的資源建立動作，使用者必須具有 `ec2:CreateTags` 許可。下列IAM原則陳述式會使用`ec2:CreateAction`條件索引鍵，讓使用者只能在的環境中建立標籤`CreateLaunchTemplate`。使用者無法為現有啟動範本或任何其他資源套用標籤。如需詳細資訊，請參閱授予在建立期間標記 Amazon EC2 資源的權限。

```
{
    "Sid": "IAMPolicyForTaggingLaunchTemplatesOnCreation",
    "Action": "ec2:CreateTags",
    "Effect": "Allow",
    "Resource": "arn:aws:ec2:region:account-id:launch-template/*",
    "Condition": {
        "StringEquals": {
            "ec2:CreateAction": "CreateLaunchTemplate"
        }
    }
}
```

建立啟動範本的使用IAM者不會自動擁有使用他們建立之啟動範本的權限。與任何其他主參與者一樣，啟動範本建立者必須透過IAM原則取得權限。如果IAM使用者想要從啟動範本啟動EC2執行個體，他們必須擁有該`ec2:RunInstances`權限。授與這些權限時，您可以指定使用者只能使用具有特定標籤或特定標籤的啟動範本IDs。您也可以指定呼叫的AMI資源層級權限，控制任何使用啟動範本的使用者在啟動執行個體時都可以參考和使用的資源和其他資源。`RunInstances`如需範例政策，請參閱 啟動範本。

ec2：DescribeLaunchTemplates

若要列出並檢視帳戶中的啟動範本，主參與者必須具有IAM策略中的`ec2:DescribeLaunchTemplates`權限。因為 `Describe` 動作不支援資源層級許可，您必須在沒有條件的情況下指定它們，且政策中資源元素的值必須為 `"*"`。

例如，下列IAM政策陳述式授予主參與者列出及檢視帳戶中所有啟動範本的權限。

```
{
    "Sid": "IAMPolicyForDescribingLaunchTemplates",
    "Action": "ec2:DescribeLaunchTemplates",
    "Effect": "Allow",
    "Resource": "*"
}
```

ec2：DescribeLaunchTemplateVersions

列出並檢視啟動範本的主參與者也應具備擷`ec2:DescribeLaunchTemplateVersions`取組成啟動範本之整個屬性集的權限。

若要列出並檢視帳戶中的啟動範本版本，主參與者必須具有IAM策略中的`ec2:DescribeLaunchTemplateVersions`權限。因為 `Describe` 動作不支援資源層級許可，您必須在沒有條件的情況下指定它們，且政策中資源元素的值必須為 `"*"`。

例如，下列IAM原則陳述式授予主參與者列出及檢視帳戶中所有啟動範本版本的權限。

```
{
    "Sid": "IAMPolicyForDescribingLaunchTemplateVersions",
    "Effect": "Allow",
    "Action": "ec2:DescribeLaunchTemplateVersions",
    "Resource": "*"
}
```

ec2：DeleteLaunchTemplate

> **Important**
>
> 在授予主體刪除資源的許可時，您應該小心謹慎。刪除啟動範本可能會導致依賴啟動範本的 AWS 資源失敗。

若要刪除啟動範本，主參與者必須具有IAM原則中的`ec2:DeleteLaunchTemplate`權限。盡可能使用以標籤為基礎的條件索引鍵以限制許可。

例如，下列IAM原則陳述式僅在範本具有指定的標籤時，才授予主參與者刪除啟動範本的權限 (*##*=*##*).

```
{
    "Sid": "IAMPolicyForDeletingLaunchTemplates",
    "Action": "ec2:DeleteLaunchTemplate",
    "Effect": "Allow",
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "aws:ResourceTag/purpose": "testing"
        }
    }
}
```

或者，您可以使用ARNs來識別套用IAM原則的啟動範本。

啟動範本具有下列項目ARN。

```
"Resource": "arn:aws:ec2:region:account-id:launch-template/lt-09477bcd97b0d310e"
```

您可以ARNs透過將它們包含在清單中來指定多個，也可以指定`"*"`不含`Condition`元素的`Resource`值，以允許主參與者刪除帳戶中的任何啟動範本。

控制版本控制許可

對於受信任的管理員，您可以使用類似下列範例的IAM原則，授予建立和刪除啟動範本版本的存取權，以及變更啟動範本的預設版本。

> **Important**
>
> 授與主參與者建立啟動範本版本或修改啟動範本的權限時，請務必小心。
>
> - 當您建立啟動範本版本時，會影響任何允許 Amazon EC2 代表您使用該`Latest`版本啟動執行個體的 AWS 資源。
> - 修改啟動範本時，您可以變更哪個版本，因此會影響允許 Amazon EC2 以此修改版本代表您啟動執行個體的任何 AWS 資源。`Default`

您還需要謹慎處理與模板版本互`Default`動`Latest`或啟動模板版本的 AWS 資源的方式，例如EC2艦隊和 Spot Fleet。當`Latest`或使用不同的啟動範本版本時`Default`，Amazon EC2 不會重新檢查使用者許可是否在啟動新執行個體以滿足叢集的目標容量時要完成的動作，因為沒有使用者與資源互動。 AWS 透過授與使用者呼叫 `CreateLaunchTemplateVersion` and 的權限 `ModifyLaunchTemplateAPIs`，如果使用者將叢集指向包含執行個體設定檔 (IAM角色的容器) 的其他啟動範本版本，則使用者也會實際上獲得該`iam:PassRole`權限。這表示使用者可能會更新啟動範本，以將IAM角色傳遞給執行個體，即使他們沒有`iam:PassRole`權限也是如此。在將許可授予可建立和管理啟動範本版本的使用者時保持謹慎，即可妥善管理此風險。

ec2：CreateLaunchTemplateVersion

若要建立啟動範本的新版本，主參與者必須具有IAM原則中啟動範本的`ec2:CreateLaunchTemplateVersion`權限。

例如，下列IAM原則陳述式僅在版本使用指定的標籤時，才授予主參與者建立啟動範本版本的權限 (*##*=*##*). 或者，您可以指定一或多個啟動範本ARNs，也可以指定`"*"`不含`Condition`元素的`Resource`值，以允許主參與者在帳戶中建立任何啟動範本的版本。

```
{
    "Sid": "IAMPolicyForCreatingLaunchTemplateVersions",
    "Action": "ec2:CreateLaunchTemplateVersion",
    "Effect": "Allow",
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "aws:ResourceTag/environment": "production"
        }
    }
}
```

ec2：DeleteLaunchTemplateVersion

> **Important**
>
> 一如往常，在授予委託人刪除資源的許可時，您應該小心謹慎。刪除啟動範本版本可能會導致依賴啟動範本版本的 AWS 資源失敗。

若要刪除啟動範本版本，主參與者必須具有IAM原則中啟動範本的`ec2:DeleteLaunchTemplateVersion`權限。

例如，下列IAM原則陳述式僅在版本使用指定的標籤時，才授予主參與者刪除啟動範本版本的權限 (*##=##*). 或者，您可以指定一或多個啟動範本ARNs，也可以指定`"*"`不含`Condition`元素的`Resource`值，以允許主參與者刪除帳戶中任何啟動範本的版本。

```
{
    "Sid": "IAMPolicyForDeletingLaunchTemplateVersions",
    "Action": "ec2:DeleteLaunchTemplateVersion",
    "Effect": "Allow",
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "aws:ResourceTag/environment": "production"
        }
    }
}
```

ec2：ModifyLaunchTemplate

若要變更與啟動範本相關聯的`Default`版本，主參與者必須具有IAM原則中啟動範本的`ec2:ModifyLaunchTemplate`權限。

例如，只有在啟動範本使用指定的標籤時，下列IAM原則陳述式才會授予主參與者修改啟動範本的權限 (*##=##*). 或者，您可以指定一或多個啟動範本ARNs，或者指定`"*"`不含`Condition`元素的`Resource`值，以允許主參與者修改帳戶中的任何啟動範本。

```
{
    "Sid": "IAMPolicyForModifyingLaunchTemplates",
    "Action": "ec2:ModifyLaunchTemplate",
    "Effect": "Allow",
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "aws:ResourceTag/environment": "production"
        }
    }
}
```

控制啟動範本上標籤的存取權

當資源為啟動範本時，您可以使用條件索引鍵來限制標記許可。例如，下列IAM原則只允許從指定帳戶和區域的啟動範本中移除具有*temporary*索引鍵的標籤。

```
{
    "Sid": "IAMPolicyForDeletingTagsOnLaunchTemplates",
    "Action": "ec2:DeleteTags",
    "Effect": "Allow",
    "Resource": "arn:aws:ec2:region:account-id:launch-template/*",
    "Condition": {
        "ForAllValues:StringEquals": {
            "aws:TagKeys": ["temporary"]
        }
    }
}
```

如需可用於控制可套用至 Amazon EC2 資源之標籤金鑰和值的條件金鑰的詳細資訊，請參閱控制對特定標籤的存取。

使用 Amazon EC2 啟動範本控制啟動 Amazon EC2 執行個體

您可以控制 Amazon EC2 執行個體的組態，方法是指定使用者只能在使用啟動範本時啟動執行個體，並且只能使用特定的啟動範本。您也可以控制哪些人員可以建立、修改、描述與刪除啟動範本和啟動範本版本。

使用啟動範本控制啟動參數

啟動範本可以包含所有或部分參數，以便在啟動時設定執行個體。不過，當您使用啟動範本啟動執行個體時，您可以覆寫在啟動範本中指定的參數。或者，您可以指定啟動範本中沒有的其他參數。

> **Note**
>
> 您無法在啟動期間移除啟動範本參數 (例如，您無法為參數指定空值)。若要移除參數，請在沒有參數的情況下建立新版本的啟動範本，並使用該版本來啟動執行個體。

若要啟動執行個體，使用者必須擁有使用`ec2:RunInstances`動作的權限。使用者也必須具備許可，才能建立或使用由該執行個體所建立或與其相關聯的資源。您可以使用 `ec2:RunInstances` 動作上

的資源層級許可，以控制使用者可指定的啟動參數。或者，您可以將使用啟動範本啟動執行個體的許可授予使用者。這可讓您在啟動範本而非IAM原則中管理啟動參數，並使用啟動範本做為啟動執行個體的授權工具。例如，您可以指定使用者只能使用啟動範本來啟動執行個體，而且使用者只能使用特定的啟動範本。您也可以控制使用者能在啟動範本中覆寫的啟動參數。如需範例政策，請參閱 啟動範本。

控制啟動範本的使用

根據預設， 使用者不具使用啟動範本的許可。您可以建立使用者政策，授予使用者建立、修改、描述與刪除啟動範本和啟動範本版本的許可。您也可以將資源層級許可套用至一些啟動範本動作，藉以控制使用者將特定資源用於那些動作的能力。如需詳細資訊，請參閱下列範例政策：範例：使用啟動範本。

將使用 `ec2:CreateLaunchTemplate` 和 `ec2:CreateLaunchTemplateVersion` 動作的許可授予使用者時請小心。您無法使用資源層級權限來控制使用者可以在啟動範本中指定哪些資源。若要限制用來啟動執行個體的資源，請確定您只將建立啟動範本和啟動範本版本的許可授予合適的管理員。

搭配EC2叢集或 Spot 叢集使用啟動範本時的重要安全性考量

若要使用啟動範本，您必須授予使用者許可，以便建立、修改、描述和刪除啟動範本及啟動範本版本。透過控制對 `ec2:CreateLaunchTemplate` 和 `ec2:CreateLaunchTemplateVersion` 動作的存取權，您可以控制誰可以建立啟動範本和啟動範本版本。您也可以透過控制 `ec2:ModifyLaunchTemplate` 動作的存取權來控制可修改啟動範本的對象。

> **Important**
>
> 如果EC2叢集或 Spot 叢集設定為使用「最新」或「預設」啟動範本版本，則叢集不會知道稍後是否變更為「最新」或「預設」，以指向不同的啟動範本版本。當 Latest 或 Default 使用EC2不同的啟動範本版本時，Amazon 在啟動新執行個體以滿足叢集的目標容量時，不會重新檢查要完成動作的許可。將許可授與可建立和管理啟動範本版本的使用者 (尤其是允許使用者變更「預設」啟動範本版本的 `ec2:ModifyLaunchTemplate` 動作) 時，請務必考量到這點。

透過授與使用者使用啟動範本EC2動作的權限APIs，如果使用者建立或更新EC2叢集或 Spot Fleet 以指向包含執行個體設定檔 (IAM角色的容器) 的不同啟動範本版本，該使用者也會實際上獲得該`iam:PassRole`權限。這表示使用者可能會更新啟動範本，以將IAM角色傳遞給執行個體，即使他們沒有`iam:PassRole`權限也是如此。如需詳細資訊和範例IAM政策，請參閱使用IAM者指南中的使用IAM角色將許可授與在 Amazon EC2 執行個體上執行的應用程式。

如需詳細資訊，請參閱 控制啟動範本的使用 和 範例：使用啟動範本。

創建一個 Amazon EC2 啟動模板

您可以透過為執行個體組態參數指定自己的值，或從現有啟動範本或 Amazon EC2 執行個體取得值，以建立 Amazon 啟動範本。EC2

您不需要為啟動範本中的每個參數指定值；您只需指定一個執行個體設定參數即可建立啟動範本。若要指出您選擇不指定的參數，請選取 [使用主控台時不要包含在啟動範本中]。使用命令列工具時，請勿加入參數以表示您選擇不在啟動範本中指定這些參數。

如果您要在啟動範本AMI中指定一個，您可以選取AMI或指定將指向執行個AMI體啟動時的「Systems Manager」參數。

使用啟動範本啟動執行個體時，會使用在啟動範本中指定的值來設定對應的執行處理參數。如果未在啟動範本中指定值，則會使用對應執行個體參數的預設值。

任務

- 透過指定參數建立啟動範本
- 從現有啟動範本建立啟動範本
- 從執行個體建立啟動範本
- 使用「Systems Manager」參數而不是 AMI ID

透過指定參數建立啟動範本

若要建立啟動範本，您必須指定啟動範本名稱和至少一個執行個體組態參數。

如需每個參數的描述，請參閱Amazon EC2 執行個體組態參數的參考。

Console

若要使用主控台建立啟動範本

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Launch Templates (啟動範本) 接著選擇 Create launch template (建立啟動範本)。
3. 在「啟動範本名稱與說明」下，執行下列動作：
   a. 針對 Launch template name (啟動範本名稱)，輸入啟動範本的描述性名稱。
   b. 在 Template version description (範本版本描述) 中，提供此啟動範本版本的簡短描述。

c. 若要在建立時標記啟動範本，請展開 [範本標籤]，選擇 [新增標籤]，然後輸入標籤索引鍵和值配對。為每個要新增的其他標籤重新選擇新增新的標籤。

> **Note**
>
> 要標記啟動執行個體時建立的資源，您必須在 Resource tags (資源標籤) 下指定標籤。如需詳細資訊，請參閱此程序中的步驟 9。

4. 在 [應用程式和作業系統映像 (Amazon Machine Image)] 下，您可以保持選取 [不包含在啟動範本中]，或選擇執行個體的作業系統 (OS)，然後選擇一個AMI。或者，您可以指定「Systems Manager」參數，而不是指定AMI。如需詳細資訊，請參閱使用「Systems Manager」參數而不是 AMI ID。

   AMI是包含啟動執行個體所需的作業系統和軟體的範本。

5. 在「執行個體類型」下，您可以選取「不要包含在啟動範本中」，或選取執行個體類型，或指定執行個體屬性，讓 Amazon 使用這些屬性EC2識別執行個體類型。

> **Note**
>
> 只有 Auto Scaling 群組、EC2叢集和 Spot 叢集使用啟動範本來啟動執行個體時，才支援指定執行個體屬性。如需詳細資訊，請參閱使用以屬性為基礎的執行個體類型選取建立 Auto Scaling 群組和。指定EC2叢集或 Spot 叢集的執行個體類型選取屬性
> 如果您計劃在啟動執行個體精靈中使用啟動範本，或搭配使用啟動範本 RunInstances API，就無法指定執行個體類型屬性。

   執行個體類型會決定執行個體所使用的硬體組態 (CPU記憶體、儲存空間和網路容量) 和主機電腦的大小。

   如果您不確定要選擇哪種執行個體類型，可以執行下列動作：

   - 選擇比較執行個體類型，依下列屬性來比較不同的執行個體類型：數目vCPUs、架構、記憶體數量 (GiB)、儲存空間量 (GB)、儲存區類型和網路效能。
   - 選擇 [取得建議]，從執行個體類型尋找工具取得執行EC2個體類型的指引和建議。如需詳細資訊，請參閱從EC2執行個體類型尋找工具取得。

> **Note**
>
> 如果您的使用時間少 AWS 帳戶 於 12 個月，您可以選擇 t2.micro 執行個體類型 (或 t2.micro 無法使用的區域中的 t3.micro 執行個體類型)，在免費方案EC2下使用 Amazon。如果某個執行個體類型符合免費方案的條件，則標記為 Free tier eligible (符合免費方案條件)。

6. 在 [key pair (登入)] 下，對於 [金鑰配對名稱]，請保持 [不包含在啟動範本中選取]，或選擇現有的金鑰配對，或建立新的金鑰配對。
7. 在 [網路設定] 下，您可以選取 [啟動範本中不包含]，也可以指定各種網路設定的值。
8. 在 [設定儲存區] 下，如果您AMI在啟動範本中指定了，則會AMI包含一或多個儲存磁碟區，包括根磁碟區 (磁碟區 1 (AMI根)。您可以選擇性地指定要附加至執行個體的其他磁碟區。若要新增磁碟區，請選擇 Add new volume (新增磁碟區)。
9. 在 [資源標籤] 底下，若要標記啟動執行個體時所建立的資源，請選擇 [新增標籤]，然後輸入標籤鍵和值配對。針對 Resource types (資源類型)，請指定建立時要標記的資源。您可以為所有資源指定相同的標籤，或者為不同的資源指定不同的標籤。為每個要新增的其他標籤重新選擇 Add tag (新增標籤)。

   您可以針對使用啟動範本時建立的以下資源指定標籤：

   - 執行個體
   - 磁碟區
   - Elastic Graphics
   - Spot 執行個體請求
   - 網路介面

   > **Note**
   >
   > 若要標記啟動範本本身，您必須在 Template tags (範本標籤) 下指定標籤。如需詳細資訊，請參閱此程序中的步驟 3。

10. 如需「進階」詳細資訊，請展開區段以檢視欄位，並選擇性地指定執行處理的任何其他參數。
11. 使用「摘要」面板來檢閱您的啟動範本組態。您可以通過選擇其鏈接導航到任何部分，然後進行任何必要的更改。

12. 當您準備好建立啟動範本時，請選擇 Create launch template (建立啟動範本)。

AWS CLI

下列範例會使用命create-launch-template令來建立具有指定名稱和執行個體組態的啟動範本。

```
aws ec2 create-launch-template \
    --launch-template-name TemplateForWebServer \
    --version-description WebVersion1 \
    --tag-specifications 'ResourceType=launch-
template,Tags=[{Key=purpose,Value=production}]' \
    --launch-template-data file://template-data.json
```

以下是指定執行個體組態的啟動範本資料的範JSON例。儲存JSON至檔案並將其包含在`--launch-template-data`參數中，如範例指令所示。

```
{
    "NetworkInterfaces": [{
        "AssociatePublicIpAddress": true,
        "DeviceIndex": 0,
        "Ipv6AddressCount": 1,
        "SubnetId": "subnet-7b16de0c"
    }],
    "ImageId": "ami-8c1be5f6",
    "InstanceType": "r4.4xlarge",
    "TagSpecifications": [{
        "ResourceType": "instance",
        "Tags": [{
            "Key":"Name",
            "Value":"webserver"
        }]
    }],
    "CpuOptions": {
        "CoreCount":4,
        "ThreadsPerCore":2
    }
}
```

下列為範例輸出。

```
{
```

```
    "LaunchTemplate": {
        "LatestVersionNumber": 1,
        "LaunchTemplateId": "lt-01238c059e3466abc",
        "LaunchTemplateName": "TemplateForWebServer",
        "DefaultVersionNumber": 1,
        "CreatedBy": "arn:aws:iam::123456789012:root",
        "CreateTime": "2017-11-27T09:13:24.000Z"
    }
}
```

PowerShell

下列範例會使用指New-EC2LaunchTemplate令程式，建立具有指定名稱和執行個體組態的啟動範本。

```
$launchTemplateData = [Amazon.EC2.Model.RequestLaunchTemplateData]@{
    ImageId = 'ami-8c1be5f6'
    InstanceType = 'r4.4xlarge'
    NetworkInterfaces = @(

 [Amazon.EC2.Model.LaunchTemplateInstanceNetworkInterfaceSpecificationRequest]@{
            AssociatePublicIpAddress = $true
            DeviceIndex = 0
            Ipv6AddressCount = 1
            SubnetId = 'subnet-7b16de0c'
        }
    )
    TagSpecifications = @(
        [Amazon.EC2.Model.LaunchTemplateTagSpecificationRequest]@{
            ResourceType = 'instance'
            Tags = [Amazon.EC2.Model.Tag]@{
                Key = 'Name'
                Value = 'webserver'
            }
        }
    )
    CpuOptions = [Amazon.EC2.Model.LaunchTemplateCpuOptionsRequest]@{
        CoreCount = 4
        ThreadsPerCore = 2
    }
}
$tagSpecificationData = [Amazon.EC2.Model.TagSpecification]@{
    ResourceType = 'launch-template'
```

```
    Tags = [Amazon.EC2.Model.Tag]@{
        Key = 'purpose'
        Value = 'production'
    }
}
New-EC2LaunchTemplate -LaunchTemplateName 'TemplateForWebServer' -VersionDescription
 'WebVersion1' -LaunchTemplateData $launchTemplateData -TagSpecification
 $tagSpecificationData
```

下列為範例輸出。

```
CreatedBy            : arn:aws:iam::123456789012:root
CreateTime           : 9/19/2023 16:57:55
DefaultVersionNumber : 1
LatestVersionNumber  : 1
LaunchTemplateId     : lt-01238c059eEXAMPLE
LaunchTemplateName   : TemplateForWebServer
Tags                 : {purpose}
```

從現有啟動範本建立啟動範本

您可以複製現有啟動範本，然後調整參數以建立新版啟動範本。但是，您只能在使用 Amazon EC2 控制台時執行此操作。 AWS CLI 不支援複製範本。如需每個參數的描述，請參閱Amazon EC2 執行個體組態參數的參考。

Console

若要從現有啟動範本建立啟動範本

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Launch Templates (啟動範本) 接著選擇 Create launch template (建立啟動範本)。
3. 針對 Launch template name (啟動範本名稱)，輸入啟動範本的描述性名稱。
4. 在 Template version description (範本版本描述) 中，提供此啟動範本版本的簡短描述。
5. 若要在建立時標記啟動範本，請展開 [範本標籤]，選擇 [新增標籤]，然後輸入標籤索引鍵和值配對。
6. 展開來源範本，並針對啟動範本名稱，選擇要以新啟動範本為基礎的啟動範本。
7. 針對 Source template version (來源範本版本)，選擇要做為新啟動範本基礎的啟動範本版本。

8. 視需要調整任何啟動參數，然後選擇 Create launch template (建立啟動範本)。

從執行個體建立啟動範本

您可以複製現有 Amazon EC2 執行個體的參數，然後調整參數以建立啟動範本。如需每個參數的描述，請參閱Amazon EC2 執行個體組態參數的參考。

Console

若要從執行個體建立啟動範本

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取例項，然後選擇 [動作]、[映像和範本]、[從執行個體建立範本]。
4. 提供名稱、描述和標籤，並視需要調整啟動參數。

> **Note**
> 當您從執行個體建立啟動範本時，範本中不會包含執行個體的網路介面IDs和 IP 位址。

5. 選擇 Create launch template (建立啟動範本)。

AWS CLI

您可以使用從現有執行個體建立啟動範本，方法是先從執行個體取得啟動範本資料，然後使用啟動範本資料建立啟動範本。 AWS CLI

若要從執行個體啟動範本資料

- 使用指get-launch-template-data令並指定執行個體 ID。您能以輸出做為基礎，用以建立新啟動範本或啟動範本版本。根據預設，輸出包括最上層 `LaunchTemplateData` 物件，而此物件無法於啟動範本資料中指定。使用 `--query` 選項排除此物件。

```
aws ec2 get-launch-template-data \
    --instance-id i-0123d646e8048babc \
    --query "LaunchTemplateData"
```

下列為範例輸出。

```
{
    "Monitoring": {},
    "ImageId": "ami-8c1be5f6",
    "BlockDeviceMappings": [
        {
            "DeviceName": "/dev/xvda",
            "Ebs": {
                "DeleteOnTermination": true
            }
        }
    ],
    "EbsOptimized": false,
    "Placement": {
        "Tenancy": "default",
        "GroupName": "",
        "AvailabilityZone": "us-east-1a"
    },
    "InstanceType": "t2.micro",
    "NetworkInterfaces": [
        {
            "Description": "",
            "NetworkInterfaceId": "eni-35306abc",
            "PrivateIpAddresses": [
                {
                    "Primary": true,
                    "PrivateIpAddress": "10.0.0.72"
                }
            ],
            "SubnetId": "subnet-7b16de0c",
            "Groups": [
                "sg-7c227019"
            ],
            "Ipv6Addresses": [
                {
                    "Ipv6Address": "2001:db8:1234:1a00::123"
                }
            ],
            "PrivateIpAddress": "10.0.0.72"
        }
    ]
}
```

例如，您可以將輸出直接寫入檔案：

```
aws ec2 get-launch-template-data \
    --instance-id i-0123d646e8048babc \
    --query "LaunchTemplateData" >> instance-data.json
```

使用啟動範本資料建立啟動範本

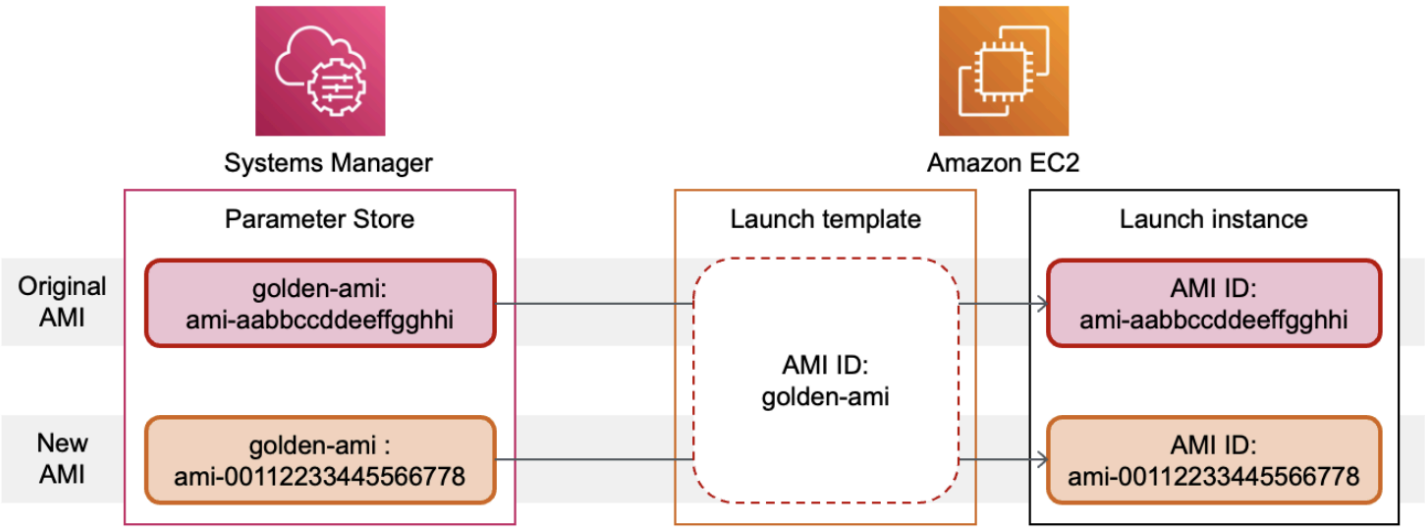
- 使用指create-launch-template令，使用上一個程序的輸出來建立啟動範本。如需使用建立啟動範本的詳細資訊 AWS CLI，請參閱透過指定參數建立啟動範本。

使用「Systems Manager」參數而不是 AMI ID

您可以指定參數，而不是在啟動範本中指 AWS Systems Manager 定 AMI ID。如果 AMI ID 變更，您可以透過更新「系統管理員參數存放區」中的「Systems Manager」參數，在一個位置更新 AMI ID。參數也可以與其他參數共用 AWS 帳戶。您可以在一個帳戶中集中存儲和管理AMI參數，並與需要引用它們的每個其他帳戶共享它們。透過使用 Systems Manager 參數，您可以在單一動作中更新所有啟動範本。

「Systems Manager」參數是您在「參數存放區」AWS Systems Manager 中建立的使用者定義鍵值對。參數存放區提供一個中央位置來存放應用程式組態值。

在下圖中，參數會先對映至「`golden-ami`參數存放區」AMI `ami-aabbccddeeffgghhi` 中的原始參數。在啟動範本中，AMIID 的值為`golden-ami`。使用此啟動範本啟動執行個體時，AMIID 會解析為`ami-aabbccddeeffgghhi`。稍後會AMI更新，產生新的 AMI ID。在參數存放區中，`golden-ami` 參數會映射至新的 `ami-00112233445566778`。啟動範本保持不變。使用此啟動範本啟動執行個體時，AMIID 會解析為新的`ami-00112233445566778`。



Systems Manager 參數格式 AMI IDs

使用啟動範本時，使用者定義的「Systems Manager」參數必須遵循下列格式來取AMI代 ID：

- 參數類型：`String`
- 參數資料類型：`aws:ec2:image`— 這可確保參數存放區驗證您輸入的值是否為 AMI ID 的正確格式。

若要取得有關為 AMI ID 建立有效參數的更多資訊，請參閱《AWS Systems Manager 使用指南》中的〈建立 Systems Manager 參數〉。

啟動範本中的 Systems Manager 參數格式

若要使用「Systems Manager」參數代替啟動範本中的 AMI ID，在啟動範本中指定參數時，必須使用下列其中一種格式：

若要參照公用參數：

- `resolve:ssm:`*`public-parameter`*

若要參照儲存在同一帳戶中的參數：

- `resolve:ssm:`*`parameter-name`*
- `resolve:ssm:`*`parameter-name`*`:`*`version-number`* - 版本編號本身是預設標籤
- `resolve:ssm:`*`parameter-name`*`:`*`label`*

若要參考從另一個參數共用的參數 AWS 帳戶：

- `resolve:ssm:parameter-ARN`
- `resolve:ssm:parameter-ARN:version-number`
- `resolve:ssm:parameter-ARN:label`

參數版本

Systems Manager 參數是版本化資源。更新參數時，會建立新的連續參數版本。Systems Manager 支援參數標籤，您可以將其映射至特定參數版本。

例如，`golden-ami` 參數可以有三個版本：1、2 和 3。您可以建立映射至版本 2 的參數標籤 `beta`，以及映射至版本 3 的參數標籤 `prod`。

在啟動範本中，可以使用下列其中一種格式來指定 `golden-ami` 參數的版本 3：

- `resolve:ssm:golden-ami:3`
- `resolve:ssm:golden-ami:prod`

指定版本或標籤是可選操作。如果未指定版本或標籤，則會使用最新版本的參數。

在啟動範本中指定 Systems Manager 參數

當您建立啟動範本或新版本的啟動範本時，您可以在啟動範本中指定 Systems Manager 參數，而不是 AMI ID。

Console

在啟動範本中指定 Systems Manager 參數

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Launch Templates (啟動範本) 接著選擇 Create launch template (建立啟動範本)。
3. 針對 Launch template name (啟動範本名稱)，輸入啟動範本的描述性名稱。
4. 在應用程式和作業系統映像 (Amazon 機器映像) 下，選擇瀏覽更多AMIs。
5. 選擇搜尋列右側的箭頭按鈕，然後選擇指定自訂值/Systems Manager 參數。
6. 在指定自訂值或 Systems Manager 參數對話方塊中，執行下列操作：

a. 對於 AMIID 或 Systems Manager 參數字串，請使用下列其中一種格式輸入「Systems Manager」參數名稱：

若要參照公用參數：

- **resolve:ssm:*public-parameter***

若要參照儲存在同一帳戶中的參數：

- **resolve:ssm:*parameter-name***
- **resolve:ssm:*parameter-name*:*version-number***
- **resolve:ssm:*parameter-name*:*label***

若要參考從另一個參數共用的參數 AWS 帳戶：

- **resolve:ssm:*parameter-ARN***
- **resolve:ssm:*parameter-ARN*:*version-number***
- **resolve:ssm:*parameter-ARN*:*label***

b. 選擇 Save (儲存)。

7. 視需要指定任何其他啟動範本參數，然後選擇建立啟動範本。

如需詳細資訊，請參閱透過指定參數建立啟動範本。

AWS CLI

在啟動範本中指定 Systems Manager 參數

- 使用create-launch-template指令建立啟動範本。若要指定AMI要使用的，請使用下列其中一種格式輸入「Systems Manager」參數名稱：

  若要參照公用參數：

  - **resolve:ssm:*public-parameter***

  若要參照儲存在同一帳戶中的參數：

  - **resolve:ssm:*parameter-name***

- **resolve:ssm:*parameter-name*:*version-number***
- **resolve:ssm:*parameter-name*:*label***

若要參考從另一個參數共用的參數 AWS 帳戶：

- **resolve:ssm:*parameter-ARN***
- **resolve:ssm:*parameter-ARN*:*version-number***
- **resolve:ssm:*parameter-ARN*:*label***

以下範例建立的啟動範本會指定下列項目：

- 啟動範本的名稱 (*TemplateForWebServer*)
- 啟動範本的標籤 (*purpose*=*production*)
- 執行處理組態的資料 (在JSON檔案中指定)：
  - 使AMI用的 (`resolve:ssm:`*golden-ami*)
  - 要啟動的執行個體類型 (*m5.4xlarge*)
  - 執行個體的標籤 (*Name*=*webserver*)

```
aws ec2 create-launch-template \
    --launch-template-name TemplateForWebServer \
    --tag-specifications 'ResourceType=launch-
template,Tags=[{Key=purpose,Value=production}]' \
    --launch-template-data file://template-data.json
```

下列範例JSON檔案包含執行個體組態的啟動範本資料。`ImageId` 的值為 Systems Manager 參數名稱，以所需格式 `resolve:ssm:`*golden-ami* 輸入。

```
{"LaunchTemplateData": {
    "ImageId": "resolve:ssm:golden-ami",
    "InstanceType": "m5.4xlarge",
    "TagSpecifications": [{
        "ResourceType": "instance",
        "Tags": [{
            "Key":"Name",
            "Value":"webserver"
        }]
```

```
        }]
      }
    }
```

確認啟動範本取得正確的 AMI ID

將「Systems Manager」參數解析為實際 AMI ID

使用指describe-launch-template-versions令並包括`--resolve-alias`參數。

```
aws ec2 describe-launch-template-versions \
    --launch-template-name my-launch-template \
    --versions $Default \
    --resolve-alias
```

回應包含的AMI識別碼`ImageId`。在此範例中，當使用此啟動範本啟動執行個體時，AMIID 會解析為`ami-0ac394d6a3example`。

```
{
    "LaunchTemplateVersions": [
        {
            "LaunchTemplateId": "lt-089c023a30example",
            "LaunchTemplateName": "my-launch-template",
            "VersionNumber": 1,
            "CreateTime": "2022-12-28T19:52:27.000Z",
            "CreatedBy": "arn:aws:iam::123456789012:user/Bob",
            "DefaultVersion": true,
            "LaunchTemplateData": {
                "ImageId": "ami-0ac394d6a3example",
                "InstanceType": "t3.micro",
            }
        }
    ]
}
```

相關資源

若要取得有關使用 Systems Manager 參數的更多資訊，請參閱「Systems Manager」文件中的下列參考資料。

- 如需如何查詢 Amazon 支援的AMI公用參數的詳細資訊EC2，請參閱呼叫AMI公用參數。

- 如需與其他 AWS 帳戶共用參數或透過共用參數的資訊 AWS Organizations，請參閱使用共用參數。
- 如需監控參數是否成功建立的相關資訊，請參閱 Amazon 機器映像的原生參數支援IDs。

限制

- 只有使用具有指定 Systems Manager 參數而取AMI代 ID 的啟動範本類型`instant`支援的EC2叢集。
- EC2類型為`maintain`和和 `request` Spot 叢集的叢集不支援使用已指定 Systems Manager 參數來取代 ID 的啟動範本。AMI對於類型為`maintain`和的EC2叢集`request`，以及對於 Spot 叢集，如果您在啟動範本AMI中指定，則必須指定 AMI ID。
- 如果您在EC2叢集中使用以屬性為基礎的執行個AMI體選取項，則無法指定 Systems Manager 參數來取代 ID。使用以屬性為基礎的例證選取時，您必須指定 AMI ID。
- Amazon EC2 Auto Scaling 提供其他限制。如需詳細資訊，請AWS Systems Manager 參閱 Amazon EC2 Auto Scaling 使用者指南中的使用參數而非啟動範本。AMI IDs

修改啟動範本 (管理啟動範本版本)

啟動範本是不可變的；啟動範本建立後便無法再修改。但是，您可以建立新版本的啟動範本，來包含您需要的任何變更。

您可以建立不同版本的啟動範本、設定預設版本、描述啟動範本版本，以及刪除不再需要的版本。

任務

- 建立啟動範本版本
- 設定預設啟動範本版本
- 描述啟動範本版本

建立啟動範本版本

當您建立啟動範本版本時，可以指定新的啟動參數，或使用現有版本做為新版本的基礎。如需每個參數的描述，請參閱Amazon EC2 執行個體組態參數的參考。

Console

若要建立啟動範本版本

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。

2. 在導覽窗格中，選擇 Launch Templates (啟動範本)。
3. 選取啟動範本，然後選取 Actions (動作)、Modify template (Create new version) (修改範本 (建立新版本))。
4. 在 Template version description (範本版本描述) 中，輸入此啟動範本版本的描述。
5. (選取性) 展開 Source template (來源範本)，並選取啟動範本的版本，以作為新啟動範本版本的基礎。新啟動範本版本會繼承此啟動範本版本的啟動參數。
6. 視需要修改啟動參數。
7. 選擇 Create launch template (建立啟動範本)。

AWS CLI

若要建立啟動範本版本

- 使用指create-launch-template-version令。您可以指定要做為新版本基礎的來源版本。新版本會繼承此版本的啟動參數，但您可以使用 `--launch-template-data` 來覆寫參數。下列範例會根據啟動範本的版本 1 建立新版本，並指定不同的 AMI ID。

```
aws ec2 create-launch-template-version \
    --launch-template-id lt-0abcd290751193123 \
    --version-description WebVersion2 \
    --source-version 1 \
    --launch-template-data "ImageId=ami-c998b6b2"
```

PowerShell

使用指New-EC2LaunchTemplateVersion令程式。您可以指定要做為新版本基礎的來源版本。新版本會繼承此版本的啟動參數，但您可以使用 `LaunchTemplateData` 來覆寫參數。下列範例會根據啟動範本的版本 1 建立新版本，並指定不同的 AMI ID。

```
New-EC2LaunchTemplateVersion `
    -LaunchTemplateId lt-0abcd290751193123 `
    -VersionDescription WebVersion2 `
    -SourceVersion 1 `
    -LaunchTemplateData (
        New-Object `
            -TypeName Amazon.EC2.Model.RequestLaunchTemplateData `
            -Property @{ImageId = 'ami-c998b6b2'}
```

```
    )
```

設定預設啟動範本版本

您可以設定啟動範本的預設版本。當您從啟動範本啟動執行個體而且未指定版本時，會使用預設版本的參數來啟動執行個體。

Console

若要設定預設啟動範本版本

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Launch Templates (啟動範本)。
3. 選取啟動範本，然後選取 Actions (動作)、Set default version (設定預設版本)。
4. 對於 Template version (範本版本)，請選取要設為預設版本的版本號碼，然後選取 Set as default version (設為預設版本)。

AWS CLI

若要設定預設啟動範本版本

- 使用指modify-launch-template令並指定您要設定為預設的版本。

```
aws ec2 modify-launch-template \
    --launch-template-id lt-0abcd290751193123 \
    --default-version 2
```

PowerShell

使用 Edit-EC2LaunchTemplateCmdlet 並指定您要設定為預設值的版本。

```
Edit-EC2LaunchTemplate `
    -LaunchTemplateId lt-0abcd290751193123 `
    -DefaultVersion 2
```

描述啟動範本版本

您可以使用主控台，檢視所選啟動範本的所有版本，或取得其最新或預設版本符合特定版本號碼之啟動範本的清單。使用 AWS CLI，您可以描述指定啟動範本的所有版本、個別版本或某個範圍的版本。您也可以說明帳戶中所有啟動範本的所有最新版本或所有預設版本。

Console

若要描述啟動範本版本

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Launch Templates (啟動範本)。
3. 您可以檢視特定啟動範本版本，或取得其最新或預設版本符合特定版本號碼之啟動範本的清單。
   - 若要檢視某個版本的啟動範本：請選取啟動範本。在 Versions (版本) 索引標籤上，從 Version (版本) 中選取某個版本以檢視其詳細資料。
   - 若要取得其最新版本符合特定版本號碼之所有啟動範本的清單：請從搜尋列中選擇 Latest version (最新版本)，然後選擇版本號碼。
   - 若要取得其預設版本符合特定版本號碼之所有啟動範本的清單：請從搜尋列中選擇 Default version (預設版本)，然後選擇版本號碼。

AWS CLI

若要描述啟動範本版本

- 使用指describe-launch-template-versions令並指定版本號碼。在下面的例子中，版本*1*和 *3* 指定 。

```
aws ec2 describe-launch-template-versions \
    --launch-template-id lt-0abcd290751193123 \
    --versions 1 3
```

若要描述帳戶中所有最新和預設啟動範本版本

- 使用指describe-launch-template-versions令並指定`$Latest`$Default、或兩者。您必須省略呼叫中的啟動範本 ID 和名稱。您無法指定版本號碼。

```
aws ec2 describe-launch-template-versions \
    --versions "$Latest,$Default"
```

PowerShell

若要描述啟動範本版本

- 使用指Get-EC2TemplateVersion令程式並指定版本號碼。在下面的例子中，版本*1*和 *3* 指定 。

```
Get-EC2TemplateVersion `
    -LaunchTemplateId lt-0abcd290751193123 `
    -Version 1,3
```

若要描述帳戶中所有最新和預設啟動範本版本

- 使用指Get-EC2TemplateVersion令程式並指定`$Latest`$Default`、或兩者。您必須省略呼叫中的啟動範本 ID 和名稱。您無法指定版本號碼。

```
Get-EC2TemplateVersion `
    -Version '$Latest','$Default'
```

刪除啟動範本或啟動範本版本

如果您不再需要啟動範本，即可將其刪除。刪除啟動範本會刪除它的所有版本。如果您只想刪除特定版本的啟動範本，您可以在保留啟動範本的其他版本的同時刪除。

刪除啟動範本或啟動範本版本不會影響您從啟動範本啟動的任何執行個體。

刪除啟動範本及其所有版本

如果您不再需要啟動範本 (包括其所有版本)，您可以刪除啟動範本。刪除啟動範本會刪除它的所有版本。

Console

若要刪除啟動範本及其所有版本

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Launch Templates (啟動範本)。
3. 選取啟動範本，然後選取 Actions (動作)、Delete template (刪除範本)。
4. 輸入 **Delete** 以確認刪除，然後選擇 Delete (刪除)。

AWS CLI

若要刪除啟動範本及其所有版本

使用 delete-launch-template(AWS CLI) 指令並指定啟動範本。

```
aws ec2 delete-launch-template --launch-template-id lt-01238c059e3466abc
```

PowerShell

若要刪除啟動範本及其所有版本

使用 Remove-EC2LaunchTemplate(AWS Tools for PowerShell) 指令並指定啟動範本。如果省略 `-Force`，PowerShell 會提示進行確認。

```
Remove-EC2LaunchTemplate -LaunchTemplateId lt-0123456789example -Force
```

刪除啟動範本版本

如果您不再需要啟動範本版本，可以將其刪除。

考量事項

- 版本號碼在刪除之後就無法取代。
- 您無法刪除啟動範本的預設版本；您必須先指派不同的版本作為預設值。如果預設版本是啟動範本的唯一版本，則須刪除整個啟動範本。
- 使用主控台時，您一次可以刪除一個啟動範本版本。使用時 AWS CLI，您最多可以在單一要求中刪除 200 個啟動範本版本。若要在單個請求中刪除超過 200 個版本，您可以刪除啟動範本，而這也會刪除啟動範本的所有版本。

Console

若要刪除啟動範本版本

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Launch Templates (啟動範本)。
3. 選取啟動範本，然後選取 Actions (動作)、Delete template version (刪除範本版本)。
4. 選取要刪除的版本，然後選取 Delete (刪除)。

AWS CLI

若要刪除啟動範本版本

- 使用指delete-launch-template-versions令並指定要刪除的版本號碼。您在單個請求中最多可刪除 200 個啟動範本版本。

```
aws ec2 delete-launch-template-versions \
    --launch-template-id lt-0abcd290751193123 \
    --versions 1
```

PowerShell

使用指Remove-EC2TemplateVersion令程式並指定要刪除的版本號碼。您在單個請求中最多可刪除 200 個啟動範本版本。

```
Remove-EC2TemplateVersion `
    -LaunchTemplateId lt-0abcd290751193123 `
    -Version 1
```

啟動 Amazon EC2 實例

執行個體是 AWS 雲端中的虛擬伺服器。您可以從 Amazon 機器映像（AMI）啟動實例。提AMI供執行個體的作業系統、應用程式伺服器和應用程式。

當您註冊時 AWS，您可以開始使用 Amazon EC2 免費使用 AWS 免費方案. 您可以使用免費方`t2.micro`案啟動並免費使用`t2.micro`執行個體 12 個月 (在無法使用的區域中，您可以使用免費

方案下的`t3.micro`執行個體)。在執行個體執行時，您的執行個體或使用量會計入免費方案限制的費用，即使執行個體仍處於閒置狀態也一樣。如需詳細資訊，請參閱 Amazon EC2 定價。

當您啟動執行個體時，您可以在與下列其中一個資源相關聯的子網中啟動執行個體：

- 可用區域 — 此選項為預設選項。
- 本機區域 — 若要在本機區域中啟動執行個體，您必須選擇加入本機區域，然後在區域中建立子網路。如需詳細資訊，請參閱開始使用 Local Zones。
- Wavelength 區域 — 若要在「Wavelength 區域」中啟動執行個體，您必須選擇加入「Wavelength 區域」，然後在區域中建立子網路。如需如何在「Wavelength 區域」中啟動執行個體的詳細資訊，請參閱開始使用 AWS Wavelength。
- 前哨-要在 Outpost 中啟動實例，您必須創建一個前哨。如需如何建立前哨的詳細資訊，請參閱開始使用 AWS Outposts。

啟動您的執行個體之後，您便可以連線及使用它。開始時，執行個體的狀態為 `pending`。當執行個體的狀態為 `running` 時，執行個體便會開始開機。在您連線到執行個體前可能需要短暫的時間。請注意，裸機執行個體類型可能需要更長的時間來啟動。

執行個體會收到一個公開DNS名稱，您可以用來從網際網路連絡執行個體。執行個體也會收到一個私有DNS名稱，其中的其他執行個體VPC可以用來聯絡執行個體。

當您完成執行個體時，為避免產生不必要的成本，請務必將其終止。如需詳細資訊，請參閱終止 Amazon EC2 實例。

下列方法是您可以啟動執行個體的一些方法。

方法	工具	文件
使用啟動執行處理精靈來指定啟動參數。	Amazon EC2 遊戲	使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體
建立啟動範本，並從啟動範本啟動執行個體。	Amazon EC2 遊戲	使用 EC2 Amazon 啟動模板EC2啟動 Amazon 實例
使用現有例證做為基礎。	Amazon EC2 遊戲	使用現有 Amazon EC2 執行個體的參數啟動 Amazon EC2 執行個體

方法	工具	文件
使用AMI您從中購買的 AWS Marketplace。	Amazon EC2 遊戲	從一個啟動 Amazon EC2 實例 AWS Marketplace AMI
使用您AMI指定的。	AWS CLI	啟動、列出和關閉 Amazon EC2 執行個體 AWS CLI
使用您AMI指定的。	AWS Tools for Windows PowerShell	使用 Windows 啟動 Amazon EC2 實例 PowerShell
使用EC2叢集跨不同EC2執行個體類型和可用區域佈建容量，以及隨需執行個體、預留執行個體和 Spot 執行個體購買選項。	AWS CLI	EC2艦隊和現貨艦隊
使用 AWS CloudFormation 範本指定執行個體。	AWS CloudFormation	AWS:::EC2: AWS CloudFormation 用戶指南中的實例
使用特定語言 AWS SDK來啟動執行個體。	AWS SDK	AWS SDK對於. NET AWS SDK對於 C ++ AWS SDK為了去 AWS SDK對於爪哇 AWS SDK對於 JavaScript AWS SDK對於 PHP V3 AWS SDK對於 Python AWS SDK對於紅寶石 V3

Amazon EC2 執行個體組態參數的參考

Amazon EC2 主控台中的啟動執行個體精靈和啟動範本提供設定 Amazon EC2 執行個體的所有參數。

除了 key pair 之外，啟動執行個體精靈會為每個參數提供預設值。您可以接受任何或所有預設值，或使用自己的值設定執行個體。建立啟動範本時，參數是選用的。如果您使用啟動範本來啟動執行處理，則啟動範本中指定的參數會覆寫啟動執行處理精靈中的預設值。啟動範本中未指定的任何參數都會預設為啟動執行個體精靈所提供的值。

這些參數會分組在啟動執行個體精靈和啟動範本中。下列說明會根據主控台中的參數群組來呈現。

執行個體組態的參數

- 名稱和標籤
- 應用程式和作業系統映像 (Amazon Machine Image)
- 執行個體類型
- 金鑰對 (登入)
- Network settings (網路設定)
- 設定儲存
- 進階詳細資訊
- Summary

名稱和標籤

執行個體名稱是一個標籤，其中鍵是 Name (名稱)，而值是您指定的名稱。您可以標記執行個體、磁碟區和網路介面。對於 Spot 執行個體，您只能標記 Spot 執行個體請求。如需標籤的相關資訊，請參閱標記您的 Amazon EC2 資源。

指定執行個體名稱和其他標籤是選用的。

- 對於 Name (名稱)，輸入執行個體的描述性名稱。如果您未指定名稱，則可以透過其 ID 來標識執行個體，該 ID 將在您啟動執行個體時自動產生。
- 若要新增其他標籤，請選擇 Add additional tags (新增其他標籤)。選取 Add tag (新增標籤)，然後輸入鍵和值，然後選取要標記的資源類型。為每個要新增的其他標籤重新選擇 Add tag (新增標籤)。

您只能在啟動執行個體時指定執行個體名稱。您無法在建立啟動範本時為執行個體命名，但可以為啟動執行個體時建立的資源新增標籤。

應用程式和作業系統映像 (Amazon Machine Image)

Amazon 機器映像檔 (AMI) 包含建立執行個體所需的資訊。例如，AMI可能包含充當網頁伺服器所需的軟體，例如 Linux、Apache 和您的網站。

你可以找到一個合適的AMI如下。使用每個尋找選項時AMI，您都可以選擇「取消」(位於右上角) 以返回啟動執行個體精靈，而無需選擇AMI。

搜尋列

若要搜尋所有可用的項目AMIs，請在AMI搜尋列中輸入關鍵字，然後按 Enter 鍵。若要選取AMI，請選擇 「選取」。

Recents (最近)

您最近使用過的AMIs。

選擇 [最近啟動] 或 [目前使用中]，然後從 Amazon 機器映像 (AMI) 中選取一個AMI。

我的 AMIs

您擁有的私AMIs人或與AMIs您共享的私人。

選擇「我擁有」或「與我共用」，然後從 Amazon 機器映像 (AMI) 中選取一個AMI。

Quick Start

AMIs按操作系統（OS）分組，以幫助您快速入門。

首先選擇您需要的操作系統，然後從 Amazon 機器映像（AMI）中選擇一個AMI. 若要選取符AMI合免費方案資格的方案，請確定該免費方案標示AMI為符合資格的免費方案。

瀏覽更多 AMIs

選擇「瀏覽更多」AMIs 以瀏覽完整AMI目錄。

- 若要搜尋所有可用的項目AMIs，請在搜尋列中輸入關鍵字，然後按 Enter 鍵。
- 若要使AMI用「Systems Manager」參數來尋找，請選擇搜尋列右側的箭頭按鈕，然後選擇「依 Systems Manager 搜尋」參數。如需詳細資訊，請參閱AMI使用「Systems Manager」參數尋找。
- 若要依類別搜尋，請選擇 [快速入門] AMIs、[我的 AMIs] 或 [社群 AMIs]。AWS Marketplace AMIs

  這 AWS Marketplace 是一個在線商店，您可以在其中購買運行的軟件 AWS，包括AMIs。如需有關從啟動執行個體的詳細資訊 AWS Marketplace，請參閱從一個啟動 Amazon EC2 實例 AWS Marketplace AMI。在社群中AMIs，您可以發AMIs現 AWS 社群成員已提供給其他人使用。AMIsAmazon 或經過驗證的合作夥伴會標示為「已驗證供應商」
- 若要篩選清單AMIs，請在畫面左側調整結果下選取一或多個核取方塊。篩選條件選項會有所不同，取決於選取的搜尋類別。

- 檢查列出的每個根設備類型AMI。請注意哪AMIs些是您需要的類型：ebs（由 Amazon 支持 EBS）或實例存儲（由實例存儲支持）。如需詳細資訊，請參閱Root device type (根設備類型)。
- 檢查每個列出的虛擬化類型AMI。請注意哪AMIs些是您需要的類型：hvm 或半虛擬。例如，某些執行個體類型需要HVM。如需 Linux 虛擬化類型的詳細資訊，請參閱虛擬化類型。
- 檢查每個列出的引導模式AMI。請注意，AMIs使用您需要的啟動模式：舊版 BIOS，uefi 或 UEFI 首選。如需詳細資訊，請參閱使用 Amazon EC2 開機模式執行個體啟動行為。
- 選擇符合AMI您需求的項目，然後選擇 「選取」。

變更時的警告 AMI

當您啟動執行個體時，如果您修改任何與所選磁碟區或安全群組相關聯的組態AMI，然後選擇不同的磁碟區或安全群組AMI，則會開啟一個視窗，警告您目前的某些設定將會遭到變更或移除。您可以檢閱對安全群組和磁碟區做出的變更。此外，您可以檢視要新增和刪除的磁碟區，或者僅檢視待新增的磁碟區。建立啟動範本時，不會出現此警告。

執行個體類型

執行個體類型定義執行個體的硬體組態和大小。較大的執行個體類型擁有更多CPU記憶體。如需詳細資訊，請參閱 Amazon EC2 執行個體類型。

- 執行個體類型：確定執行個體類型與您指定的執行AMI個體類型相容。如需詳細資訊，請參閱Amazon EC2 實例類型。

  免費方案 — 如果您的使用年限少 AWS 帳戶 於 12 個月，您可以透過選取 t2.micro 執行個體類型 (或 t2.micro 無法使用的區域中的 t3.micro 執行個體類型)，在免費方案EC2下使用 Amazon。如果某個執行個體類型符合免費方案的條件，則標記為 Free tier eligible (符合免費方案條件)。
- 比較執行個體類型：您可以依照下列屬性來比較不同的執行個體類型：數量vCPUs、架構、記憶體數量 (GiB)、儲存容量 (GB)、儲存區類型和網路效能。
- 取得建議：您可以從執行個體類型尋找工具取得執行個體類型的EC2指引和建議。如需詳細資訊，請參閱從EC2執行個體類型尋找工具取得。
- (僅限啟動範本) 進階：若要指定執行個體屬性，並讓 Amazon 使用這些屬性EC2識別執行個體類型，請選擇 [進階]，然後選擇 [指定執行個體類型屬性]。
  - 數目 vCPUs：輸入您運算需求的 vCPUs 最小與最大數目。若要表示沒有限制，請輸入下限 **0**，並將上限保留為空白。
  - Amount of memory (MiB) (記憶體數量 (MiB))：輸入運算需求的記憶體數量 (MiB) 下限和上限。若要表示沒有限制，請輸入下限 **0**，並將上限保留為空白。

- 展開 Optional instance type attributes (選用執行個體類型屬性)，然後選擇 Add attribute (新增屬性)，以更詳細地表達您的運算需求。如需有關每個屬性的資訊，請InstanceRequirementsRequest參閱 Amazon EC2 API 參考中的。
- Resulting instance types (產生的執行個體類型)：您可以預覽符合所指定屬性的執行個體類型。若要排除執行個體類型，請選擇 Add attribute (新增屬性)，然後從 Attribute (屬性) 清單中，選擇 Excluded instance types (排除的執行個體類型)。從 Attribute value (屬性值) 清單中，選取要排除的執行個體類型。

金鑰對 (登入)

針對 Key pair name (金鑰對名稱)，選擇現有的金鑰對，或選擇 Create new key pair (建立新的金鑰對) 以建立新的金鑰對。如需詳細資訊，請參閱Amazon EC2 密鑰對和 Amazon EC2 實例。

> **Important**
>
> 如果您選擇不使用 key pair 繼續 (不建議) 選項，則除非您選擇設定為允許使用者以其他方式登入AMI的方式，否則您將無法連線至執行個體。

Network settings (網路設定)

視需要設定網路設定。

- (僅限啟動執行個體精靈) VPC：VPC為您的執行個體選擇現有的。您可以選擇預設值VPC或VPC您建立的。如需詳細資訊，請參閱the section called “Virtual Private Cloud (VPC)”。
- Subnet (子網)：您可以在與可用區域、Local Zone、Wavelength 區域或 Outpost 相關聯的子網中啟動執行個體。

  若要在可用區域中啟動執行個體，請選取要在當中啟動執行個體的子網。若要建立新的子網路，請選擇「建立新子網路」以前往 Amazon VPC 主控台。完成後，請返回啟動執行個體精靈並選擇 Refresh (重新整理) 圖示，載入清單中的子網。

  若要在IPv6唯一子網路中啟動執行個體，執行個體必須建立在 Nitro System 上。

  在 Local Zone 中啟動執行個體，選取您在 Local Zone 中建立的子網。

  若要在 Outpost 中啟動執行個體，請在與 Outpost 相關聯VPC的子網路中選取一個子網路。

- (僅限啟動執行個體精靈) 自動指派公用 IP：指定執行個體是否接收公用位IPv4址。根據預設，預設子網路中的執行個體會接收公用IPv4位址，而非預設子網路中的執行個體則不會接收公用位址。您可以選取 Enable (啟用) 或 Disable (停用) 覆寫子網的預設設定。如需詳細資訊，請參閱 公共IPv4地址。
- Firewall (security groups) (防火牆 (安全群組))：使用安全群組定義執行個體的防火牆規則。這些規則會指定應交付至您執行個體的傳入網路流量。所有其他流量都會遭到忽略。如需安全群組的詳細資訊，請參閱適用於EC2執行個體的 Amazon EC2 安全群組。

  如果新增網路介面，您必須在網路介面中指定相同的安全群組。

  選取或建立安全群組，如下所示：

  - 若要為您的現有安全性群組選取VPC，請選擇選取現有的安全性群組，然後從一般安全性群組中選取您的安全性群組。
  - 若要為您的建立新的安全性群組VPC，請選擇 [建立安全性群組]。啟動執行個體精靈會自動定義 launch-wizard-x 安全群組，並提供以下核取方塊以快速新增安全群組規則：

    (Linux) 允許SSH流量來自 — 建立輸入規則以允許您透過 SSH (連接埠 22) 連線至執行個體。

    (Windows) 允許RDP流量來自 — 建立輸入規則，讓您透過 RDP (通訊埠 3389) 連線至執行個體。

    指定流量來自 Anywhere (任意位置)、Custom (自訂) 還是 My IP (我的 IP)。

    允許來自網際網路的HTTPs流量 — 建立開啟連接埠 443 (HTTPS) 的輸入規則，以允許來自任何地方的網際網路流量。如果您的執行個體將是 Web 伺服器，則需要此規則。

    允許來自網際網路的HTTP流量 — 建立開啟連接埠 80 (HTTP) 的輸入規則，以允許來自任何地方的網際網路流量。如果您的執行個體將是 Web 伺服器，則需要此規則。

    您可以編輯這些規則並新增符合您需求的規則。

    若要編輯或新增規則，請選擇 Edit (編輯) (位於右上角)。若要新增規則，請選擇 Add security group rule (新增安全群組規則)。對於 Type (類型)，選取網路流量類型。Protocol (通訊協定) 欄位會自動填入要對網路流量開啟的通訊協定。對於 Source type (來源類型)，選取來源類型。若要讓啟動執行個體精靈或啟動範本新增電腦的公用 IP 位址，請選擇 [我的 IP]。不過，如果您透過ISP或從防火牆後方連線而沒有靜態 IP 位址，則需要找出用戶端電腦所使用的 IP 位址範圍。

> **Warning**
>
> 允許所有 IP 地址 (`0.0.0.0/0`) 訪問您的實例的規則，SSH或者如果您RDP正在短暫啟動測試實例，並且將很快停止或終止它，但對於生產環境來說是不安全的，則可以接受該規則。建議您僅授權特定 IP 地址或地址範圍存取您的執行個體。

- Advanced network configuration (進階網路組態) – 僅在選擇子網時才能使用。

  網路介面

  - Device index (設備索引)：網路卡的索引。主要網路介面必須指派給網路卡索引 0。某些執行個體類型支援多個網路卡。
  - 網路介面：選取 [新增介面] 讓 Amazon EC2 建立新的介面，或選取現有的可用網路介面。
  - Description (描述)：(選用) 新網路介面的描述。
  - Subnet (子網)：要建立新網路介面的子網。對主要網路介面 (`eth0`) 而言，這是要啟動執行個體的子網。如果您已輸入 `eth0` 的現有網路介面，則會在網路介面所在的子網中啟動執行個體。

> **Note**
>
> 若要將EC2執行個體啟動至IPv6唯一子網路，您必須使用 AWS Nitro System 上建置的執行個體。

> **Note**
>
> 啟動IPv6僅執行個體時，可DHCPv6能不會立即為執行個體提供IPv6DNS名稱伺服器。在此初始延遲期間，執行個體可能無法解析公有網域。
> 對於在 Amazon Linux 2 上執行的執行個體，如果您想要立即使用IPv6DNS名稱伺服器更新 /etc/resolv.conf 檔案，請在啟動時執行下列雲端初始化指令：
>
> ```
> #cloud-config
> bootcmd:
> - /usr/bin/sed -i -E 's,^nameserver\s+[\.[:digit:]]+$,nameserver
>  fd00:ec2::253,' /etc/resolv.conf
> ```
>
> 另一種選擇是更改配置文件並重新映像文件，AMI以便在啟動時立即具有IPv6DNS名稱服務器地址。

- 安全性群組：要與網路介面關聯的一或多個安全性群組。
- (僅限啟動範本) 自動指派公用 IP：指定執行個體是否接收公開位IPv4址。依預設，預設子網路中的執行個體會接收公用IPv4位址，而非預設子網路中的執行個體則不會接收公用位址 您可以選取 Enable (啟用) 或 Disable (停用) 覆寫子網的預設設定。如需詳細資訊，請參閱公共IPv4地址。
- 主要 IP：子網路範圍內的私人位IPv4址。保留空白，讓 Amazon 為您EC2選擇私人IPv4地址。
- 次要 IP：子網路範圍內的其他私人IPv4位址。選擇「手動指派」，然後輸入IPv4地址。選擇 [新增 IP] 以新增其他位IPv4址。或者，選擇「自動指派」並輸入值，以指出 Amazon 為您EC2選擇的IPv4地址數量。
- （IPv6僅）IPv6IPs：來自IPv6子網路範圍的位址。選擇「手動指派」，然後輸入IPv6地址。選擇 [新增 IP] 以新增其他位IPv6址。或者，選擇「自動指派」並輸入值，以指出 Amazon 為您EC2選擇的IPv6地址數量。
- IPv4前綴：網絡接口的IPv4前綴。選擇「手動指定」，然後輸入IPv4字首。或者，選擇「自動指派」並輸入值，以指出 Amazon 為您EC2選擇的IPv4首碼數目。
- IPv6前綴：網絡接口的IPv6前綴。選擇「手動指定」，然後輸入IPv6字首。或者，選擇「自動指派」並輸入值，以指出 Amazon 為您EC2選擇的IPv6首碼數目。
- (雙堆疊和IPv6唯一) 指派主要 IPv6 IP：當您將執行個體啟動至雙堆疊或IPv6唯一子網路時，您可以指出該執行個體是否具有主要IPv6位址。這有助於防止流量中斷到執行個體或網路介面。如果您依賴此執行個體的IPv6地址不會變更，並且 Amazon EC2 選擇與網路界面相關聯的IPv6地址作為主要IPv6地址，請選擇是。您稍後無法移除主要IPv6地址。當您將某個IPv6GUA位址啟用為主要位址時IPv6，第一個位址IPv6GUA會成為主要IPv6位址，直到執行個體終止或網路介面中斷連線為止。如果您有多個地IPv6址與連接到執行個體的網路界面相關聯，並且讓 Amazon EC2 指派主IPv6地址，則與網路界面相關聯的第一個IPv6GUA地址會成為主要IPv6地址。
- 終止時刪除：指出刪除執行個體時是否刪除網路介面。
- Elastic Fabric Adapter：指出網路介面是否為 Elastic Fabric Adapter。如需詳細資訊，請參閱適用於 Amazon 上的機器學習HPC工作負載的彈性光纖 EC2。
- 網路卡索引：網路卡的索引。主要網路介面必須指派給網路卡索引 0。某些執行個體類型支援多個網路卡。
- ENA快遞：ENAExpress 由可 AWS 擴展的可靠數據報（SRD）技術提供支持。SRD技術使用數據包噴塗機制來分配負載並避免網絡擁塞。啟用 ENA Express 允許支援的執行個體在可能的情況下，SRD在常規TCP流量之上進行通訊。除非您從清單中選取 [啟用] 或 [停用]，否則啟動執行處理精靈或啟動範本不包含執行處理的 ENA Express 組態。
- ENA快速 UDP：如果您已啟用 ENA Express，您可以選擇性地將其用於UDP流量。除非您選取 [啟用] 或 [停用]，否則啟動執行處理精靈或啟動範本不包含執行處理的 ENA Express 組態。

選擇 [新增網路介面] 以新增其他網路介面。您可新增的網路介面數取決於選取執行個體類型支援的數量。其他網路介面可以位於相同子網路的不同子網路中，VPC或位於您擁有VPC的不同子網路中(只要子網路與您的執行個體位於相同的可用區域)。如果您選擇新增位於另一個子網路的其他網VPC路介面，當您選取子網路時，您會看到「多重VPC子網路」選項。如果您在另一個子網路中選取子網路VPC，「多重」VPC 標籤會顯示在您新增的網路介面旁邊。這可讓您在不同的網路和安全性VPCs設定之間建立多重主目錄執行個體。請注意，如果您ENI從另一個附加VPC，則必須從ENI中選擇一個安全性群組VPC。

如需詳細資訊，請參閱彈性網絡介面。如果您指定多個網路介面，您的執行個體將無法接收公用IPv4位址。此外，如果您為 eth0 指定現有的網路介面，則無法使用自動指派公用 IP 覆寫子網路的公用IPv4設定。如需詳細資訊，請參閱在執行個體啟動期間指派公用IPv4地址。

設定儲存

AMI您選取的包括一或多個儲存磁碟區，包括根磁碟區。您可以指定要連接至執行個體的其他磁碟區。

(僅限「啟動執行處理精靈」) 您可以使用「簡單」或「進階」檢視。使用 Simple (簡單) 檢視，您可以指定磁碟區的大小和類型。若要指定所有磁碟區參數，請選擇卡片右上方的 Advanced (進階) 檢視。

使用 Advanced (進階) 檢視，您可以設定每個磁碟區，如下所示：

- 儲存類型：選取要與執行個體建立關聯的 Amazon EBS 或執行個體存放區磁碟區。此清單中可用的磁碟區類型取決於您選擇的執行個體類型。有關更多信息，請參閱Amazon EC2 執行個體的執行個體存放臨時區塊儲和 Amazon EBS 卷。
- Device name (設備名稱)：從磁碟區可用的設備名稱清單中選取。
- Snapshot (快照)：選取要從中還原磁碟區的快照。您可以在 Snapshot (快照) 欄位中輸入文字，搜尋可用的共享和公有快照。
- 大小 (GiB)：對於EBS磁碟區，您可以指定儲存大小。如果您選取符合免費方案資格的AMI和執行個體，請記住，若要保留在免費方案內，您必須保持在總儲存空間的 30 GiB 以下。
- 磁碟區類型：對於EBS磁碟區，請選取磁碟區類型。如需詳細資訊，請參閱 Amazon EBS使用者指南中的 Amazon EBS 磁碟區類型。
- IOPS：如果您已選取已佈建的IOPSSSD磁碟區類型，則可以輸入磁碟區可支援的每秒 I/O 作業數目 (IOPS)。
- 終止時刪除：對於 Amazon EBS 磁碟區，選擇是以在執行個體終止時刪除磁碟區，或選擇否保留磁碟區。如需詳細資訊，請參閱在執行個體終止時保留資料。

- 已加密：如果執行個體類型支援EBS加密，您可以選擇 [是] 啟用磁碟區的加密功能。如果您在此區域中預設啟用加密，則會為您啟用預設加密。如需詳細資訊，請參閱 Amazon EBS使用者指南中的 Amazon EBS 加密。
- KMS金鑰：如果您針對已加密選取 [是]，則必須選取用來加密磁碟區的客戶管理金鑰。如果在此區域中預設啟用加密，則會為您選取預設客戶受管金鑰。您可以選取不同的金鑰，或指定您建立ARN的任何客戶管理金鑰。
- 檔案系統：將 Amazon EFS 或 Amazon FSx 檔案系統掛接至執行個體。如需掛接 Amazon EFS 檔案系統的詳細資訊，請參閱使用 Amazon EFS 與 Amazon EC2 Linux 實例。如需掛接 Amazon FSx 檔案系統的詳細資訊，請參閱 使用 Amazon FSx 與 Amazon EC2 實例

進階詳細資訊

針對 Advanced Details (進階詳細資訊)，展開此區段來檢視欄位，指定執行個體的其他參數。

- (僅限啟動執行個體精靈) 網域加入 AWS Directory Service 目錄：選取啟動後要加入執行個體的目錄 (網域)。如果您選取網域，則必須選取具有所需權限的IAM角色。如需有關網域加入的詳細資訊，請參閱將 Amazon EC2 Linux 執行個體無縫加入您的 AWS 受管 Microsoft AD 目錄 (Linux 執行個體) 和無縫地將 Amazon EC2 Windows 執行個體加入您的 AWS 受管 Microsoft AD 目錄 (Windows 執行個體)。
- IAM執行個體設定檔：選取要與IAM執行個體關聯的執行個體設定檔。這是IAM角色的容器。如需詳細資訊，請參閱IAMAmazon 的角色 EC2。
- Hostname type (主機名稱類型)：選取執行個體的客體作業系統主機名稱是否要包含資源名稱或 IP 名稱。如需詳細資訊，請參閱Amazon EC2 執行個體主機名稱。
- DNS主機名稱：決定對資源名稱或 IP 名稱（取決於您為「主機名稱」類型選取的內容）的DNS查詢是否會以位IPv4址（A 記錄）、IPv6地址（AAAA記錄）或兩者進行回應。如需詳細資訊，請參閱Amazon EC2 執行個體主機名稱。
- 執行個體自動復原：啟用時，如果系統狀態檢查失敗，就會復原執行個體。此設定在啟動時預設會針對支援的執行個體類型啟用。如需詳細資訊，請參閱設定簡化的自動復原。
- Shutdown behavior (關機行為)：選取執行個體是否應於關機時停止或終止。如需詳細資訊，請參閱變更執行個體啟動的關機行為。
- Stop - Hibernate behavior (停止 - 休眠行為)：若要啟用休眠，請選取 Enable (啟用)。此欄位只在您的執行個體符合休眠先決條件時才可使用。如需詳細資訊，請參閱 休眠您的 Amazon EC2 實例。
- Termination protection (終止保護)：若要防止意外性的終止，請選取 Enable (啟用)。如需詳細資訊，請參閱啟用終止保護。

- Stop protection (停止保護)：若要防止意外性的停止，請選擇 Enable (啟用)。如需詳細資訊，請參閱啟用停止保護。
- 詳細 CloudWatch 監控：選擇啟用以開啟使用 Amazon 對執行個體的詳細監控 CloudWatch。需支付額外費用。如需詳細資訊，請參閱使用監控執行個體 CloudWatch。
- 彈性 GPU：Amazon 彈性圖形於 2024 年 1 月 8 日終止使用壽命。對於需要圖形加速的工作負載，我們建議您使用 Amazon EC2 G4ad、G4dn 或 G5 執行個體。
- 彈性推論：可連接至例證的EC2CPU彈性推論加速器。如需詳細資訊，請參閱 Amazon Elastic Inference 開發人員指南中的 Working with Amazon Elastic Inference。

> **Note**
>
> 自 2023 年 4 月 15 日起，不 AWS 會將新客戶加入 Amazon Elastic Inference (EI)，並協助目前客戶將工作負載遷移到提供更優惠價格和效能的選項。2023 年 4 月 15 日之後，新客戶將無法在 Amazon、Amazon 或 Amazon 使用 Amazon EI 加速器啟動執行個體。SageMaker ECS EC2但是，在過去 30 天內至少使用過一次 Amazon EI 的客戶將被視為目前客戶，並且可以繼續使用該服務。

- Credit specification (額度規格)：選擇 Unlimited (無限制)，只要有需要，就讓應用程式爆量超過基線。此欄位僅對 T 執行個體有效。可能需支付額外費用。如需詳細資訊，請參閱爆量效能執行個體。
- 放置群組：指定要在其中啟動例證的放置群組。可以選取現有的置放群組或建立新的置放群組。並非所有執行個體類型都支援在置放群組中啟動執行個體。如需詳細資訊，請參閱Amazon EC2 執行個體的放置群組。
- EBS-最佳化執行個體：針對 Amazon 最佳化的執行個體EBS使用最佳化的組態堆疊，並為 Amazon EBS I/O 提供額外的專用容量；如果執行個體類型支援此功能，請選擇啟用以啟用。需支付額外費用。如需詳細資訊，請參閱the section called “EBS最佳化”。
- 購買選項：選擇 Spot 執行個體以 Spot 價格請求競價型執行個體，以隨需價格上限，然後選擇自訂競價型執行個體選項以變更預設競價型執行個體設定。您可以設定最高價 (不建議)，並變更請求類型、請求持續時間和中斷行為。如果您沒有請求 Spot 執行個體，Amazon 預設會EC2啟動隨需執行個體。如需詳細資訊，請參閱管理 Spot 執行個體。
- 容量保留：指定是否要在任何開啟的容量保留 (Open (開啟))、特定容量保留 (Target by ID (按 ID 分類的目標)) 或容量保留群組 (Target by group (按群組分類的目標)) 中啟動執行個體。若要指定不應使用容量保留，請選擇 None (無)。如需詳細資訊，請參閱 使用現有的 容量預留 啟動執行個體。
- Tenancy (租用)：選擇在共享硬體 (Shared (共享))、隔離的、專用硬體 (Dedicated (專用)) 或專用執行個體 (Dedicated host (專用執行個體)) 上執行您的執行個體。如果您選擇將執行個體啟動至專用執

行個體，則可以指定是否將執行個體啟動至主機資源群組，或是以特定專用執行個體為目標。可能需支付額外費用。如需詳細資訊，請參閱 Amazon EC2 專用實例 和 Amazon EC2 專用主機。

- RAM磁碟識別碼：(僅對半虛擬 (PV) 有效AMIs) 選取執行個體的RAM磁碟。如果您選取了核心，您可能需要選取含有驅動程式的特定RAM磁碟來支援該核心。
- 核心 ID：(僅對半虛擬 (PV) 有效AMIs) 為執行個體選取核心。
- 硝基隔離區：可讓您從 Amazon 執行個體建立隔離的執行環境 (稱為隔離區)。EC2選取「啟用」以啟用「 AWS 硝基隔離區」的執行個體。如需詳細資訊，請參閱什麼是 AWS 硝基飛地？ 在AWS 硝基飛地用戶指南。
- 授權組態：您可以根據指定的授權配置啟動執行個體，以追蹤您的授權使用情況。如需詳細資訊，請參閱《AWS License Manager 使用者指南》中的建立授權組態。
- 指定CPU選項：在啟動執行個體精靈中，只有在選取的執行個體類型支援指定選CPU項時，才會顯示此欄位。選擇「指定CPU選項」以指定啟動 vCPUs 期間的自訂數目。設定每個核心的CPU核心和執行緒數目。如需詳細資訊，請參閱CPUAmazon EC2 實例的選項。
- 可存取的中繼資料：您可以啟用或停用執行個體中繼資料服務 (IMDS) 的存取權。如需詳細資訊，請參閱設定新執行個體的執行個體中繼資料選項。
- 中繼資料IPv6端點：您可以讓執行個體使用IMDSIPv6位址`[fd00:ec2::254]`擷取執行個體中繼資料。只有當您將 AWS Nitro System 上建置的執行個體啟動到IPv6支援的子網路 (或IPv6僅限雙堆疊) 時，才能使用此選項。如需有關擷取執行個體中繼資料的詳細資訊，請參閱 訪問實例的實EC2例元數據。
- 中繼資料版本：如果您啟用存取權IMDS，您可以在要求執行個體中繼資料時，選擇要求使用執行個體中繼資料服務版本 2。如需詳細資訊，請參閱設定新執行個體的執行個體中繼資料選項。
- 中繼資料回應躍點限制：如果啟用IMDS，您可以為中繼資料 Token 設定允許的網路躍點數目。如需詳細資訊，請參閱設定新執行個體的執行個體中繼資料選項。
- Allow tags in metadata (允許中繼資料中的標籤)：如果選取 Enable (啟用)，則執行個體將允許從其中繼資料存取其所有標籤。如果未指定任何值，則預設情況下，不允許存取執行個體中繼資料中的標籤。如需詳細資訊，請參閱允許存取執行個體中繼資料中的標籤。
- User data (使用者資料)：您可以指定使用者資料，以在啟動期間設定執行個體或執行組態指令碼。如需 Linux 執行個體之使用者資料的詳細資訊，請參閱使用使用者資料輸入啟動執行個EC2體時執行指令。如需 Windows 執行個體之使用者資料的詳細資訊，請參閱〈〉Amazon 如何EC2處理 Windows 實例的用戶數據。

Summary

使用 Summary (摘要) 面板，以指定要啟動的執行個體數量、檢閱您的執行個體組態，以及啟動您的執行個體。

- Number of instances (執行個體數目)：輸入要啟動的執行個體數目。所有執行個體都會以相同的組態啟動。

> **Tip**
>
> 為了確保執行個體能更快啟動，請將大型請求分成較小的批次。例如，分別建立五個啟動請求，每個請求各啟動 100 個執行個體，而不要以單一請求啟動 500 個執行個體。

- (選擇性) 如果您指定多個執行個體，為了確保維持正確的執行個體數目以處理應用程式的需求，您可以選擇考慮使用 EC2 Auto Scaling 來建立啟動範本和 Auto Scaling 群組。Auto Scaling 會根據您的規格擴展群組中的執行個體數目。如需詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南。

> **Note**
>
> 如果 Amazon EC2 Auto Scaling 將 Auto Scaling 群組中的執行個體標記為運作狀態不佳，則執行個體會自動排定在終止並啟動另一個執行個體的情況下進行更換，而您遺失原始執行個體上的資料。如果您停止或重新開機執行個體，或另一個事件將此執行個體標記為「狀況不良」，則執行個體會標記為「狀況不良」。如需詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南中的 Auto Scaling 群組中執行個體的運作 Health 態檢查。

- 檢閱執行個體的詳細資訊，並進行任何必要的變更。您可以直接導覽至某個區段，方法是在 Summary (摘要) 面板中選擇其連結。
- 當您準備好啟動執行個體時，請選擇 Launch instance (啟動執行個體)。

使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體

您可以使用 Amazon EC2 主控台中的啟動執行個體精靈啟動 Amazon EC2 執行個體。精靈提供啟動參數的預設值，您可以接受或修改這些參數以符合您的需求。唯一未指定的參數是 key pair。如果您選擇接受預設值，只需選取 key pair 即可快速啟動執行個體。

> **Important**
>
> 執行個體處於狀態時，即使執行個體仍處於閒置狀`running`態，您也會向執行個體產生費用。不過，如果您符合免費方案的資格，則可能不會產生任何費用。如需詳細資訊，請參閱追蹤您的 Amazon 免費方案使用情況 EC2。

如需啟動執行個體精靈中每個參數的說明，請參閱Amazon EC2 執行個體組態參數的參考。

主題

- 快速啟動執行個體
- 使用定義的參數啟動執行個體

快速啟動執行個體

若要快速設定執行個體以進行測試，請遵循下列步驟。您將選取作業系統和金鑰對，並接受預設值。除了金鑰對之外，啟動執行個體精靈還會提供所有參數的預設值。您可以接受任何或全部預設值，或者透過為每個參數指定自己的值來設定執行個體。

如需啟動執行個體精靈中每個參數的說明，請參閱Amazon EC2 執行個體組態參數的參考。

使用啟動執行個體精靈快速啟動執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在畫面頂端的導覽列中，會顯示目前的「 AWS 區域」(例如，美國東部 (俄亥俄))。如有需要，請選取要在其中啟動執行個體的其他「區域」。
3. 在 Amazon 主EC2控台儀表板中，選擇啟動執行個體。
4. (選用) 在 Name and tags (名稱和標籤) 下，針對 Name (名稱)，輸入執行個體的描述性名稱。
5. 在 Application and OS Images (Amazon Machine Image) (應用程式和作業系統映像 (Amazon Machine Image)) 下，選擇 Quick Start (快速入門)，然後選擇執行個體的作業系統。
6. 在 Key pair (login) (金鑰對 (登入)) 下，針對 Key pair name (金鑰對名稱)，選擇現有的金鑰對或建立新的金鑰對。
7. 在 Summary (摘要) 面板中，選擇 Launch instance (啟動執行個體)。

使用定義的參數啟動執行個體

如果您要啟動將在生產環境中使用的執行個體，則需要設定執行個體以符合您的需求。如需啟動執行個體精靈中每個參數的說明，請參閱Amazon EC2 執行個體組態參數的參考。

使用啟動執行處理精靈定義所有啟動參數來啟動執行處理

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在畫面頂端的導覽列中，會顯示目前的「 AWS 區域」(例如，美國東部 (俄亥俄))。如有需要，請選取要在其中啟動執行個體的其他「區域」。
3. 在 Amazon 主EC2控台儀表板中，選擇啟動執行個體。
4. (選擇性) 在 [名稱和標籤] 底下，針對 [名稱] 輸入執行個體的描述性名稱，以便您輕鬆追蹤它。

   執行個體名稱是一個標籤，其中鍵是 Name (名稱)，而值是您指定的名稱。
5. 在「應用程式和作業系統映像 (Amazon Machine Image)」下，選擇執行個體的作業系統 (OS)，然後選擇一個AMI。

   AMI是包含啟動執行個體所需的作業系統和軟體的範本。
6. 在 Instance type (執行個體類型)，選擇執行個體類型。

   執行個體類型會決定執行個體所使用的硬體設定 (CPU記憶體、儲存空間和網路容量) 和主機電腦的大小。

   如果您不確定要選擇哪種執行個體類型，可以執行下列動作：

   - 選擇比較執行個體類型，依下列屬性來比較不同的執行個體類型：數目vCPUs、架構、記憶體數量 (GiB)、儲存空間量 (GB)、儲存區類型和網路效能。
   - 選擇 [取得建議]，從執行個體類型尋找工具取得執行EC2個體類型的指引和建議。如需詳細資訊，請參閱從EC2執行個體類型尋找工具取得。

> **Note**
>
> 如果您的使用時間少 AWS 帳戶 於 12 個月，您可以選擇 t2.micro 執行個體類型 (或 t2.micro 無法使用的區域中的 t3.micro 執行個體類型)，在免費方案EC2下使用 Amazon。如果某個執行個體類型符合免費方案的條件，則標記為 Free tier eligible (符合免費方案條件)。

7. 在 Key pair (login) (金鑰對 (登入)) 下，針對 Key pair name (金鑰對名稱)，選擇現有的金鑰對或建立新的金鑰對。如果您不需要金鑰組即可連線至執行個體，您可以選擇不使用 key pair 繼續 (不建議)。
8. 在 [網路設定] 下方，如果您要啟動測試執行個體，則可以保留預設值。如果您要啟動生產執行個體，最佳做法是使用您定義的網路設定和安全性群組來控制進出執行個體的流量。
9. 在 [設定儲存空間] 底下，您可以保留預設值或指定其他儲存區。AMI您選取的包括一或多個儲存磁碟區，包括根磁碟區。您可以指定要連接至執行個體的其他磁碟區。

   您可以使用 Simple (簡單) 或 Advanced (進階) 檢視。使用 Simple (簡單) 檢視，您可以指定磁碟區的大小和類型。若要指定所有磁碟區參數，請選擇卡片右上方的 Advanced (進階) 檢視。
10. 如需「進階」詳細資訊，請展開區段以檢視欄位，並為您的執行個體指定任何其他參數。
11. 在「摘要」面板中，您可以執行下列動作：

    a. 指定要啟動的執行個體數目。
    b. 檢閱您的執行個體設定，並選擇區段連結，直接導覽至區段。
    c. 當您準備好啟動執行個體時，請選擇 Launch instance (啟動執行個體)。

    如果執行個體無法啟動或狀態立即進入 `terminated` 而非 `running`，請參閱解決 Amazon EC2 執行個體啟動問題。
12. (選用) 您可以建立執行個體的帳單提醒。在確認畫面上的 Next Steps (後續步驟) 下方，選擇 Create billing alerts (建立帳單提醒)，然後遵循指示進行。您還可以在啟動執行個體後建立帳單提醒。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的建立帳單警示以監控估計 AWS 費用。

使用 EC2 Amazon 啟動模板EC2啟動 Amazon 實例

Amazon EC2 啟動範本會儲存執行個體啟動參數，因此您不必在每次啟動執行個體時指定它們。

數個執行個體啟動服務可以選擇性地在啟動執行個體時使用啟動範本，而對於其他服務 (例如EC2叢集)，除非使用啟動範本，否則無法啟動執行個體。本主題說明如何在使用啟動執行個體精靈、Amazon EC2 Auto Scaling、EC2叢集和 Spot 叢集EC2啟動執行個體時使用啟動範本。

如需啟動範本 (包括如何建立啟動範本) 的詳細資訊，請參閱在 Amazon 啟動範本中存放執行個體EC2啟動參數。

主題

- 使用啟動範本啟動 Amazon EC2 執行個體
- 使用啟動範本在 Amazon EC2 Auto Scaling 群組中啟動執行個體
- 使用啟動範本啟動EC2艦隊
- 使用啟動範本啟動競價型叢集

使用啟動範本啟動 Amazon EC2 執行個體

您可以使用啟動範本中包含的參數來啟動 Amazon EC2 執行個體。在選取啟動範本之後，但在啟動執行個體之前，您可以修改啟動參數。

使用啟動範本所啟動的執行個體會自動獲指派兩個具有索引鍵 `aws:ec2launchtemplate:id` 和 `aws:ec2launchtemplate:version` 的標籤。您無法移除或編輯這些標籤。

Console

若要使用啟動範本啟動執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 使用下列其中一個選項來選取啟動範本：
   - 在 Amazon EC2 主控台儀表板中，選擇 Launch 執行個體旁邊的向下箭頭，選擇從範本啟動執行個體，然後針對來源範本選取啟動範本。
   - 在導覽窗格中，選擇「啟動範本」，選取啟動範本，然後選擇「動作」>「從範本啟動例項」。
3. 對於 Source template version (來源範本版本)，請選取要使用的啟動範本版本。
4. (選擇性) 您可以修改任何啟動參數的值。如果您不修改值，則會使用啟動範本所定義的值。如果啟動範本中未指定任何值，則會使用參數的預設值。
5. 在「摘要」面板中，對於「實體數目」，指定要啟動的實體數目。
6. 選擇啟動執行個體。

   如果執行個體無法啟動或狀態立即進入 `terminated` 而非 `running`，請參閱解決 Amazon EC2 執行個體啟動問題。

AWS CLI

若要從啟動範本啟動執行個體

- 使用 run-instances 命令，並指定 `--launch-template` 參數。選擇指定要使用的啟動範本版本。如果您未指定版本，即使用預設版本。

```
aws ec2 run-instances \
    --launch-template LaunchTemplateId=lt-0abcd290751193123,Version=1
```

- 若要覆寫啟動範本參數，請在 run-instances 命令中指定參數。下列範例會覆寫啟動範本中指定的執行個體類型 (如果有的話)。

```
aws ec2 run-instances \
    --launch-template LaunchTemplateId=lt-0abcd290751193123 \
    --instance-type t2.small
```

- 如果您指定屬複雜結構一部分的巢狀參數，則會使用啟動範本中所指定的複雜結構以及您指定的其他巢狀參數，來啟動執行個體。

  在下列範例中，會使用 *Owner*=*TeamA* 標籤以及啟動範本中指定的其他標籤，來啟動執行個體。如果啟動範本的現有標籤包含 *Owner* 索引鍵，則會將值取代為 *TeamA*。

```
aws ec2 run-instances \
    --launch-template LaunchTemplateId=lt-0abcd290751193123 \
    --tag-specifications "ResourceType=instance,Tags=[{Key=Owner,Value=TeamA}]"
```

  在下列範例中，執行個體會以裝置名稱的磁碟區啟動 */##/xvdb* 以及在啟動範本中指定的任何其他區塊裝置對應。如果啟動範本已定義的現有磁碟區 */##/xvdb*，其值會被指定的值取代。

```
aws ec2 run-instances \
    --launch-template LaunchTemplateId=lt-0abcd290751193123 \
    --block-device-mappings "DeviceName=/dev/
xvdb,Ebs={VolumeSize=20,VolumeType=gp2}"
```

如果執行個體無法啟動或狀態立即進入 `terminated` 而非 `running`，請參閱解決 Amazon EC2 執行個體啟動問題。

PowerShell

使用 AWS Tools for PowerShell從啟動範本啟動執行個體

- 使用指New-EC2Instance令並指定-LaunchTemplate參數。選擇指定要使用的啟動範本版本。如果您未指定版本，即使用預設版本。

```
Import-Module AWS.Tools.EC2
New-EC2Instance `
    -LaunchTemplate (
        New-Object -TypeName Amazon.EC2.Model.LaunchTemplateSpecification -
Property @{
            LaunchTemplateId = 'lt-0abcd290751193123';
            Version          = '4'
    }
)
```

- 若要覆寫啟動範本參數，請在指New-EC2Instance令中指定參數。下列範例會覆寫啟動範本中指定的執行個體類型 (如果有的話)。

```
Import-Module AWS.Tools.EC2
New-EC2Instance `
    -InstanceType t4g.small `
    -LaunchTemplate (
        New-Object -TypeName Amazon.EC2.Model.LaunchTemplateSpecification -
Property @{
            LaunchTemplateId = 'lt-0abcd290751193123';
            Version          = '4'
    }
)
```

- 如果您指定屬複雜結構一部分的巢狀參數，則會使用啟動範本中所指定的複雜結構以及您指定的其他巢狀參數，來啟動執行個體。

  在下列範例中，會使用 *Owner*=*TeamA* 標籤以及啟動範本中指定的其他標籤，來啟動執行個體。如果啟動範本的現有標籤包含 *Owner* 索引鍵，則會將值取代為 *TeamA*。

```
Import-Module AWS.Tools.EC2
New-EC2Instance `
    -InstanceType t4g.small  `
    -LaunchTemplate (
```

```
        New-Object -TypeName Amazon.EC2.Model.LaunchTemplateSpecification -
Property @{
            LaunchTemplateId = 'lt-0abcd290751193123';
            Version          = '4'
        }
) `
    -TagSpecification (
        New-Object -TypeName Amazon.EC2.Model.TagSpecification -Property @{
            ResourceType = 'instance';
            Tags         = @(
                @{key = "Owner"; value = "TeamA" },
                @{key = "Department"; value = "Operations" }
            )
        }
)
```

在下列範例中，執行個體會以裝置名稱的磁碟區啟動 */##/xvdb* 以及在啟動範本中指定的任何其他區塊裝置對應。如果啟動範本已定義的現有磁碟區 */##/xvdb*，其值會被指定的值取代。

```
Import-Module AWS.Tools.EC2
New-EC2Instance `
    -InstanceType t4g.small  `
    -LaunchTemplate (
        New-Object -TypeName Amazon.EC2.Model.LaunchTemplateSpecification -
Property @{
            LaunchTemplateId = 'lt-0abcd290751193123';
            Version          = '4'
    }
) `
    -BlockDeviceMapping  (
        New-Object -TypeName Amazon.EC2.Model.BlockDeviceMapping -Property @{
            DeviceName = '/dev/xvdb';
            EBS        = (
                New-Object -TypeName Amazon.EC2.Model.EbsBlockDevice -Property @{
                    VolumeSize = 25;
                    VolumeType = 'gp3'
                }
            )
        }
)
```

如果執行個體無法啟動或狀態立即進入 `terminated` 而非 `running`，請參閱解決 Amazon EC2 執行個體啟動問題。

使用啟動範本在 Amazon EC2 Auto Scaling 群組中啟動執行個體

您可以建立 Auto Scaling 群組，並指定要用於群組的啟動範本。當 Amazon EC2 Auto Scaling 在自動擴展群組中啟 Auto Scaling 執行個體時，它會使用關聯啟動範本中定義的啟動參數。

使用啟動範本建立 Auto Scaling 群組之前，您必須先建立包含在 Auto Scaling 群組中啟動執行個體所需參數的啟動範本。某些參數是必需的，例如 IDAMI，某些參數無法與「Auto Scaling 例」群組搭配使用。主控台提供指引，協助您建立可與 Amazon EC2 Auto Scaling 搭配使用的範本。

若要使用主控台建立具有啟動範本的 Auto Scaling 群組

- 如需相關指示，請參閱 Amazon Auto Scaling 使用者指南中的使用啟動範本建立 EC2 Auto Scaling 群組。

若要使用啟動範本建立或更新「Auto Scaling」群組 AWS CLI

- 使用create-auto-scaling-group或指update-auto-scaling-group令並指定`--launch-template`參數。

如需詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南中的以下主題：

- 為「Auto Scaling」群組建立啟動範本
- 使用進階設定建立啟動範本
- 使用 AWS Command Line Interface (AWS CLI) 建立和管理啟動範本的範例 — 提供示範如何使用各種參數組合建立啟動範本的範例。
- 使用啟動範本建立 Auto Scaling 群組
- 更新 Auto Scaling 群組

使用啟動範本啟動EC2艦隊

建立EC2叢集請求時，需要啟動範本。Amazon 完EC2成EC2叢集請求時，會使用相關啟動範本中定義的啟動參數。您可以覆寫啟動範本中指定的部分參數。如需詳細資訊，請參閱建立EC2艦隊。

若要使用啟動範本建立EC2叢集 AWS CLI

- 使用 create-fleet 命令。使用 `--launch-template-configs` 參數指定啟動範本以及啟動範本的任何覆寫。

使用啟動範本啟動競價型叢集

建立競價型叢集請求時，啟動範本是選用的。如果您不使用啟動範本，您可以手動指定啟動參數。如果您使用啟動範本，Amazon 在EC2滿足 Spot 叢集請求時，會使用關聯啟動範本中定義的啟動參數。您可以覆寫啟動範本中指定的部分參數。如需詳細資訊，請參閱建立 Spot 機群。

若要使用啟動範本建立競價型叢集請求

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選擇 Request Spot Instances (要求 Spot 執行個體)。
4. 在 Launch parameters (啟動參數) 下，選擇 Use a launch template (使用啟動範本)。
5. 針對 Launch template (啟動範本)，選擇啟動範本，然後從右邊的欄位選擇啟動範本版本。
6. 透過在此畫面上選取不同的選項，設定 Spot 機群。如需選項的詳細資訊，請參閱 使用已定義的參數建立 Spot Fleet 請求 (主控台)。
7. 當您準備好建立 Spot 機群時，選擇 Launch (啟動)。

若要使用啟動範本建立競價型叢集請求

- 使用指request-spot-fleet令。使用 `LaunchTemplateConfigs` 參數指定啟動範本以及啟動範本的任何覆寫。

使用現有 Amazon EC2 執行個體的參數啟動 Amazon EC2 執行個體

Amazon 主EC2控台提供與此類似的 Launch 選項，可讓您使用目前的執行個體做為啟動其他執行個體的基礎。此選項會自動將所選執行個體的特定組態詳細資訊填入 Amazon EC2 啟動執行個體精靈。

考量事項

- 我們不會複製您的執行個體；我們只會複寫部分的組態詳細資訊。若要建立執行個體的副本，請先從中建立執行個體，然後AMI從中啟動更多執行個體AMI。建立啟動範本，以確保您使用相同的啟動詳細資訊來啟動執行個體。

- 目前的執行個體必須處於 `running` 狀態。

已複製的詳細資訊

下列組態詳細資訊會從選取的執行個體複製到啟動執行個體精靈：

- AMI身份證
- 執行個體類型
- 可用區域，或所選執行個體所在的VPC和子網路
- 公共IPv4地址。如果選取的執行個體目前擁有公開IPv4位址，則無論選取執行個體的預設公用IPv4位址設定為何，新執行個體都會收到公開IPv4位址。如需有關公用IPv4位址的詳細資訊，請參閱公共IPv4地址。
- 置放群組 (如適用)
- IAM與執行個體相關聯的角色 (如果適用)
- 關機行為設定 (停止或終止)
- 終止保護設定 (true 或 false)
- CloudWatch 監視 (啟用或停用)
- Amazon EBS 優化設置（真或假）
- 租賃設定 VPC (如果啟動到共用或專用)
- 核心 ID 和RAM磁碟 ID (如果適用)
- 使用者資料 (若指定)
- 與執行個體相關聯的標籤 (如適用)
- 與執行個體相關聯的安全群組
- [視窗執行個體] 關聯資訊。若選取的執行個體與組態檔相關聯，則相同的檔案便會自動與新的執行個體相關聯。若組態檔包含加入的網域組態，新的執行個體便會加入相同網域。如需有關加入網域的詳細資訊，請參閱管理指南中的 Windows EC2 執行個體無縫加入 AWS 受管理的AWS Directory Service Microsoft AD 活動目錄。

未複製的詳細資訊

下列組態詳細資訊不是從您選取的執行個體複製的。相反地，精靈會套用其預設設定或行為：

- 網路介面數 – 預設為一個網路介面，即主要網路介面 (eth0)。
- 儲存 — 預設儲存組態由AMI和執行個體類型決定。

依據現有執行個體的操作方式，啟動更多執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後選取動作、映像和範本、啟動更多相似執行個體。
4. 啟動執行個體精靈隨即開啟。透過在此畫面上選取不同的選項，對執行個體組態進行任何必要的變更。

   當您準備好啟動執行個體時，請選擇 Launch instance (啟動執行個體)。
5. 如果執行個體無法啟動或狀態立即進入 `terminated` 而非 `running`，請參閱解決 Amazon EC2 執行個體啟動問題。

從一個啟動 Amazon EC2 實例 AWS Marketplace AMI

您可以使用 Amazon EC2 主控台或命令列工具從執行個體訂閱 AWS Marketplace AMI並啟動執行個體。如需有關的更多資訊 AWS Marketplace AMIs，請參閱AMIs在 Amazon EC2 實例中支付 AWS Marketplace。

若要AMI在啟動後取消訂閱，您必須先終止從啟動的所有執行個體AMI。如需詳細資訊，請參閱管理您的 AWS Marketplace 訂閱。

New console

若要使用 Amazon EC2 主控台從執 AWS Marketplace AMI行個體啟動

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在 Amazon 主EC2控台儀表板中，選擇啟動執行個體。
3. (選用) 在 Name and tags (名稱和標籤) 下，針對 Name (名稱)，輸入執行個體的描述性名稱。
4. 在 [應用程式和作業系統映像 (Amazon Machine Image)] 下，選擇 [瀏覽更多]AMIs，然後選擇索AWS Marketplace AMIs引標籤。AMI通過瀏覽類別或使用搜索功能找到合適的。若要選取產品，請選取 Select (選取)。
5. 隨即開啟一個視窗，其中顯示您所選取產品的概觀。您可以檢視定價資訊，以及廠商已提供的任何其他資訊。準備就緒後，請選擇下列其中一個按鈕：
   - 執行個體啟動時訂閱 — 當您選擇 Launch 執行個體 (在步驟 10) 時即會開始訂閱。
   - 立即訂閱 — 您的訂閱立即開始。訂閱正在進行中時，您可以繼續執行此程序中的步驟來設定執行個體。如果您的信用卡詳細資訊有任何問題，系統會要求您更新帳戶詳細資訊。

> **Note**
>
> 在您啟動執行個體之前，不需要支付使用產品的費用AMI。當您選取執行個體類型時，請記下每個受支援執行個體類型的定價。可能也需支付產品的額外稅金。

6. 針對 Instance type (執行個體類型)，選取執行個體的執行個體類型。執行個體類型定義執行個體的硬體組態和大小。
7. 在 Key pair (login) (金鑰對 (登入)) 下，針對 Key pair name (金鑰對名稱)，選擇現有的金鑰對或建立新的金鑰對。
8. 在 [網路設定] 下，針對 [防火牆 (安全性群組)]，記下根據廠商對產品的規格建立的新安全性群組。安全性群組可能包含允許 Linux 上所有IPv4位址 (`0.0.0.0/0`) 存取 SSH (連接埠 22) 或 Windows 上 RDP (通訊埠 3389) 的規則。建議您調整這些規則，改為僅允許特定地址或某範圍的地址透過這些連接埠存取您的執行個體。
9. 您可以使用畫面上的其他欄位來設定執行個體、新增儲存以及新增標籤。如需有關您可設定之不同選項的資訊，請參閱 Amazon EC2 執行個體組態參數的參考。
10. 在「摘要」面板的「軟體映像」(AMI) 下，檢查您要AMI從中啟動執行個體的詳細資料。同時檢查您指定的其他組態詳細資訊。當您準備好啟動執行個體時，請選擇 Launch instance (啟動執行個體)。
11. 根據您已訂閱的產品，執行個體可能需要幾分鐘以上的時間啟動。如果您在步驟 5 選擇「執行個體啟動時訂閱」，則在執行個體啟動之前，您會先訂閱產品。如果您的信用卡詳細資訊有任何問題，系統會要求您更新帳戶詳細資訊。顯示啟動確認頁面時，請選擇 View all instances (檢視所有執行個體) 以前往 Instances (執行個體) 頁面。

> **Note**
>
> 只要您的執行個體處於 `running` 狀態，就會向您收取訂閱費用，即使它處於閒置狀態。如果您的執行個體已停止，仍可能會向您收取儲存體的費用。

12. 當您的執行個體處於 `running` 狀態時，即可連線。若要執行此操作，請選取清單中的執行個體，然後選取 Connect (連線)，並選取連線選項。如需連線至執行個體的詳細資訊，請參閱Connect 至您的EC2執行個體。

> **Important**
>
> 請仔細查看供應商的使用說明，因為您可能需要使用特定的使用者名稱連線至執行個體。如需有關存取訂閱的資訊，請參閱 管理您的 AWS Marketplace 訂閱。

13. 如果執行個體無法啟動或狀態立即進入 `terminated` 而非 `running`，請參閱解決 Amazon EC2 執行個體啟動問題。

Old console

若要使用 Amazon EC2 主控台從執 AWS Marketplace AMI行個體啟動

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 從 Amazon EC2 儀表板中，選擇啟動實例。
3. 在選擇 Amazon 電腦映像 (AMI) 頁面上，選擇左側的AWS Marketplace類別。AMI通過瀏覽類別或使用搜索功能找到合適的。選取 Select (選取) 來選取產品。
4. 對話方塊會顯示選取之產品的概觀。您可以檢視定價資訊，以及廠商已提供的任何其他資訊。當您就緒時，請選擇 Continue (繼續)。

> **Note**
>
> 使用產品啟動執行個體之前，不會向您收取使用產品的費用AMI。請記下每個支援之執行個體類型的定價，因為系統會提示您於精靈的下一頁選取執行個體類型。可能也需支付產品的額外稅金。

5. 在 Choose an Instance Type (選取執行個體類型) 頁面上，選取要啟動之執行個體的硬體組態和大小。完成後，請選擇 Next: Configure Instance Details (下一步：設定執行個體詳細資訊)。
6. 在精靈的後續頁面上，您可以設定執行個體、新增儲存體，以及新增標籤。如需您可設定之不同選項的詳細資訊，請參閱使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體。在您到 Configure Security Group (設定安全群組) 頁面前，請皆選擇 Next (下一步)。

   此精靈會根據廠商的產品規格來建立新的安全群組。安全性群組可能包含允許所有IPv4位址 (`0.0.0.0/0`) 存取 Linux 上 SSH (連接埠 22) 或 Windows 上 RDP (連接埠 3389) 的規則。建議您調整這些規則，改為僅允許特定地址或某範圍的地址透過這些連接埠存取您的執行個體。

準備就緒後，請選擇 Review and Launch (檢閱和啟動)。

7. 在 [檢閱執行個體啟動] 頁面上，查看您要AMI從中啟動執行個體的詳細資料，以及您在精靈中設定的其他設定詳細資料。準備就緒後，請選取 Launch (啟動) 選取或建立金鑰對，並啟動執行個體。
8. 根據您已訂閱的產品，執行個體可能需要幾分鐘以上的時間啟動。您要先訂閱產品，才能啟動執行個體。如果您的信用卡詳細資訊有任何問題，系統會要求您更新帳戶詳細資訊。顯示啟動確認頁面時，請選擇 View Instances (檢視執行個體) 前往 Instances (執行個體) 頁面。

> **Note**
>
> 只要您的執行個體正在執行，就會向您收取訂閱費用，即使執行個體處於閒置狀態亦同。如果您的執行個體已停止，仍可能會向您收取儲存體的費用。

9. 當您的執行個體處於 `running` 狀態時，即可連線。若要執行此作業，請選取清單中的執行個體，然後選取 Connect (連線)。請遵循對話方塊中的說明。如需連線至執行個體的詳細資訊，請參閱Connect 至您的EC2執行個體。

> **Important**
>
> 請仔細檢查供應商的使用說明，因為您可能需要使用特定的使用者名稱登入執行個體。如需存取訂閱詳細資訊的詳細資訊，請參閱管理您的 AWS Marketplace 訂閱。

10. 如果執行個體無法啟動或狀態立即進入 `terminated` 而非 `running`，請參閱解決 Amazon EC2 執行個體啟動問題。

從 AWS Marketplace AMI使用指令行工具啟動例證的步驟

若要使用命令列工具從 AWS Marketplace 產品啟動執行個體，請先確定您已訂閱產品。然後，您可以使用下列方法啟動含有產品 AMI ID 的執行個體：

方法	文件
AWS CLI	請使用 run-instance 指令，或參閱下列主題以取得詳細資訊：啟動執行個體。
AWS Tools for Windows PowerShell	使用命New-EC2Instance令，或參閱下列主題以取得詳細資訊：使用 Windows 啟動 Amazon EC2 執行個體 PowerShell

方法	文件
查詢 API	使用RunInstances請求。

Connect 至您的EC2執行個體

您的 Amazon EC2 執行個體是 AWS 雲端中的虛擬伺服器。若要登入執行個體，您必須建立與執行個體的連線。連線到執行個體的方式取決於執行個體的作業系統，以及您用來連線至執行個體的電腦上的作業系統。您可以從下列選項來選擇。

連線選項

- 使用用SSH戶端 Connect 至您的 Linux 執行個體
- 使用 Pu Connect 到您的 Linux 執行個體 TTY
- 使用用RDP戶端 Connect 至您的 Windows 執行個體
- 使用 Fleet Manager 連線到 Windows 執行個體
- 使用工作階段管理員進行連接
- 使 Connect「執EC2行個體 Connect
- 使用EC2執行個體連線端點連

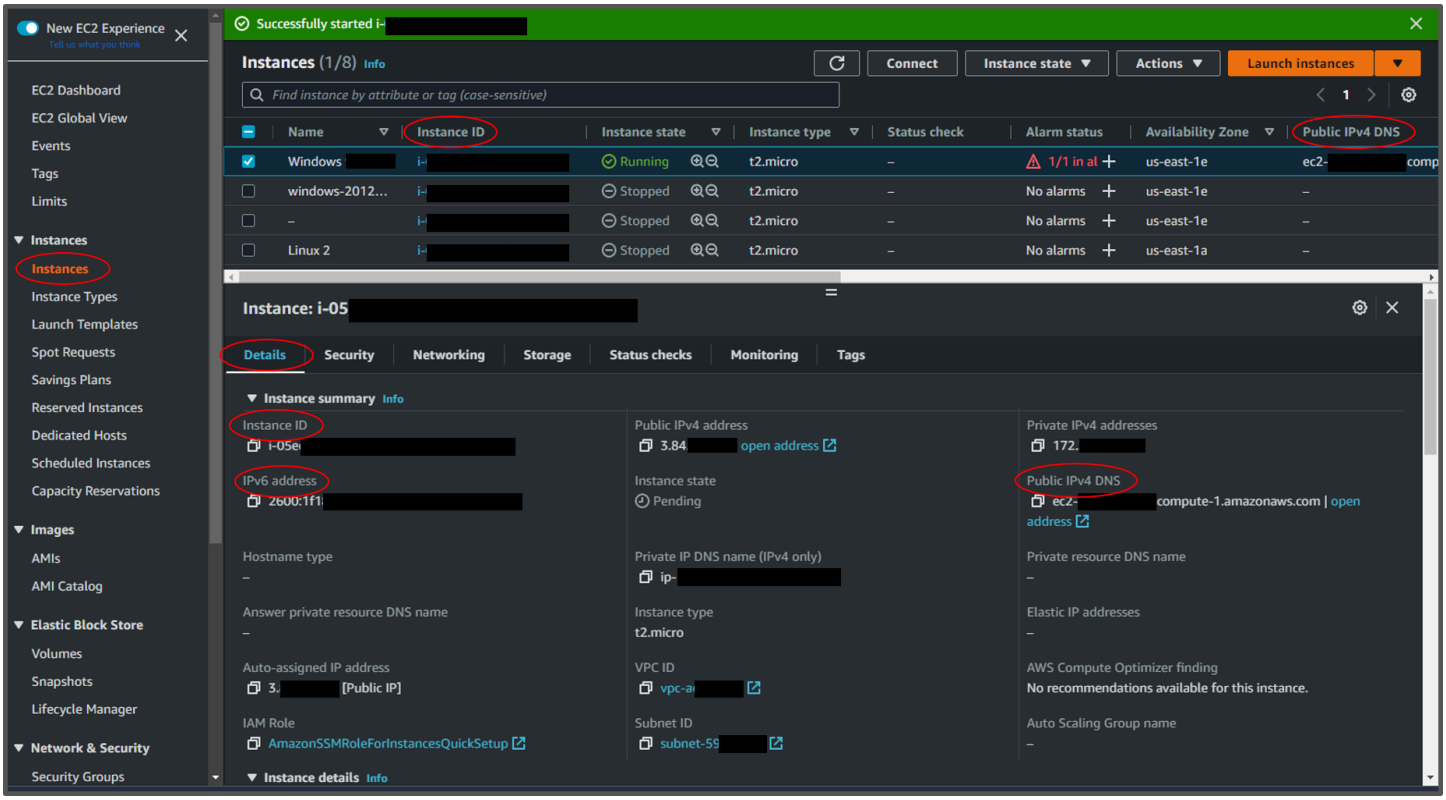
以下是一般先決條件。請注意，您選擇的連線選項可能還有其他必要條件。

一般先決條件

- 請確認您的執行個體已通過狀態檢查。執行個體可能需要幾分鐘的時間才能準備好接受連線要求。如需詳細資訊，請參閱檢視狀態檢查。
- 獲取所需的實例詳細信息.
- 找出私有金鑰並設定許可.
- (選用) 取得執行個體指紋.

獲取所需的實例詳細信息

若要準備連線到執行個體，請從 Amazon EC2 主控台或使用命令列取得下列資訊。



- 取得執行個體的公用DNS名稱。

  您可以從 Amazon EC2 主控台取得執行個體的公開DNS資訊。檢查「執行處理」窗格的「公用」IPv4 DNS 欄。如果此欄位為隱藏，請選擇畫面右上角的設定圖示 ( )，然後選取 [公開] IPv4 DNS。您也可以DNS在 [執行個體] 窗格的 [執行個體資訊] 區段中找到公開資訊。當您在 Amazon EC2 主控台的「執行個體」窗格中選取執行個體時，該執行個體的相關資訊將顯示在頁面的下半部。在「詳細資料」標籤下，尋找「公開」IPv4 DNS。

  如果您願意，您可以使用描述實例 (AWS CLI) 或 Get-EC2Instance(AWS Tools for Windows PowerShell) 命令。

  如果未顯示「公IPv4DNS用」，請確認執行個體狀態為「執行中」，而且您尚未在私有子網路中啟動執行個體。如果您使用啟動執行個體精靈啟動執行個體，則可能已編輯網路設定下的自動指派公有 IP 欄位，並將值變更為停用。如果停用自動指派公有 IP 選項，則執行個體在啟動時不會被指派公有 IP 地址。

- （IPv6僅實例）獲取實例的IPv6地址。

  如果您為執行個體指派IPv6位址，您可以選擇性地使用執行個體的位址，而非公用IPv6位IPv4址或公用IPv4DNS主機名稱來連線至執行個體。您的本機電腦必須具有IPv6位址，且必須設定才

能使用IPv6。您可以從 Amazon EC2 主控台取得執行個體的IPv6位址。檢查「執行個體」窗格的IPv6IPs欄。或者，您可以在實例信息部分中找到IPv6地址。當您在 Amazon EC2 主控台的「執行個體」窗格中選取執行個體時，該執行個體的相關資訊將顯示在頁面的下半部。在 [詳細資料] 索引標籤下，尋找IPv6位址。

如果您願意，您可以使用描述實例 (AWS CLI) 或 Get-EC2Instance(AWS Tools for Windows PowerShell) 命令。如需有關的更多資訊IPv6，請參閱IPv6地址。

- (Linux 執行個體) 取得執行個體的使用者名稱。

  您可以使用使用者帳戶的使用者名稱或用來啟動執行個體的預設使AMI用者名稱來連線至執行個體。

  - 取得使用者帳戶的使用者名稱。

    如需如何建立使用者帳戶的詳細資訊，請參閱管理 Amazon EC2 Linux 執行個體上的系統使用者。

  - 取得您用來啟動執行AMI個體的預設使用者名稱。
    - CentOS — `centos` 或 `ec2-user`
    - Debian — `admin`
    - 軟呢帽 — `fedora` 或 `ec2-user`
    - RHEL— `ec2-user` 或 `root`
    - SUSE— `ec2-user` 或 `root`
    - Ubuntu 的 — `ubuntu`
    - Oracle – `ec2-user`
    - 比特納米 — `bitnami`
    - 洛基 Linux — `rocky`
    - 其他 — 請與AMI供應商聯絡

找出私有金鑰並設定許可

您必須知道私密金鑰檔案的位置，才能使用的 Linux 執行個體進行初始連線，SSH或使用 Windows 執行個體RDP。對於SSH連線，您必須設定檔案權限，以便只有您可以讀取私密金鑰。

如需使用 Amazon 時金鑰配對如何運作的相關資訊EC2，請參閱Amazon EC2 密鑰對和 Amazon EC2 實例。

- 找到私密金鑰。

取得當您啟動執行個體時所指定的金鑰對的 .pem 檔案在您電腦中的位置的完整路徑。如需詳細資訊，請參閱the section called “識別啟動時指定的公有金鑰”。

如果找不到私密金鑰檔案，請參閱 我的私有金鑰遺失。如何連線到我的執行個體？

(Linux 執行個體) 如果您使用 Pu 連線至執行個體，TTY而且需要將.pem檔案轉換為.ppk，請參閱使用 P 轉換您的私鑰 uTTYgen。

- (Linux 執行個體) 設定私密金鑰的權限，以便只有您可以讀取它。
  - 從 macOS 或 Linux 進行連線

    如果您打算使用 macOS 或 Linux 電腦上的用SSH戶端連線至 Linux 執行個體，請使用下列指令來設定私密金鑰檔案的權限，以便只有您才能讀取該檔案。

    ```
    chmod 400 key-pair-name.pem
    ```

    如果您未設定這些許可，則無法使用此金鑰對來連接至您的執行個體。如需詳細資訊，請參閱錯誤：未受保護的私有金鑰檔案。
  - 從 Windows 進行連線

    開啟檔案總管並在 .pem 檔案上按一下滑鼠右鍵。選取屬性 > 安全性索引標籤，然後選擇進階。選擇停用繼承。移除目前使用者以外的所有使用者存取權。

(選用) 取得執行個體指紋

為了保護自己免受 man-in-the-middle 攻擊，您可以驗證顯示的指紋來驗證即將連線的執行個體的真實性。如果您是從第三方AMI提供的公開執行個體啟動執行個體，則驗證指紋非常有用。

任務概觀

首先，從執行個體取得執行個體指紋。然後，當您連線至執行個體並提示您驗證指紋時，請將您在此程序中取得的指紋與顯示的指紋進行比較。如果指紋不相符，可能是有人試圖 man-in-the-middle 攻擊。如果相符，您就能很有信心地連線至您的執行個體。

取得執行個體指紋的先決條件

- 執行個體必不得處於 pending 狀態。只有在執行個體的第一次開機完成後，才能使用指紋。
- 您必須是執行個體擁有者才能取得主控台輸出。

- 有多種方法可以獲取實例指紋。如果您要使用 AWS CLI，則必須將其安裝在您的本機電腦上。若要取得有關安裝的資訊 AWS CLI，請參閱《AWS Command Line Interface 使用指南》 AWS Command Line Interface中的〈安裝〉。

取得執行個體指紋

在步驟 1 中，您將獲得控制台輸出，其中包括實例指紋。在步驟 2 中，您會在主控台輸出中找到執行個體指紋。

1. 使用下列其中一種方法取得主控台輸出。

   Console

   1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
   2. 從左側瀏覽器中，選擇「執行環境」。
   3. 選取執行個體，然後選擇 [動作]、[監控和疑難排解]、[取得系統記錄]。

   AWS CLI

   在您的本機電腦上 (不在您要連線的執行個體上)，使用 get-console-output(AWS CLI) 指令。如果輸出很大，可以將輸出傳送至文字檔案，這可能更易於讀取。請注意，當您使用 AWS 區域 時，您必須明確指定 AWS CLI，或透過設定預設「區域」(Region) 來指定。若要取得有關如何設定或指定區域的資訊，請參閱《AWS Command Line Interface 使用者指南》 AWS CLI中的〈配置〉。

   ```
   aws ec2 get-console-output --instance-id instance_id --query Output --output
    text > temp.txt
   ```

2. 在控制台輸出中，找到位於下的實例（主機）指紋BEGIN SSH HOST KEY FINGERPRINTS。可能有幾個實例指紋。當您連線至執行個體時，它只會顯示其中一個指紋。

   確切的輸出可能會因作業系統、AMI版本以及是否 AWS 建立金鑰配對而有所不同。下列為範例輸出。

   ```
   ec2:#############################################################
   ec2: -----BEGIN SSH HOST KEY FINGERPRINTS-----
   ec2: 256 SHA256:l4UB/neBad9tvkgJf1QZWxheQmR59WgrgzEimCG6kZY no comment (ECDSA)
   ec2: 256 SHA256:kpEa+rw/Uq3zxaYZN8KT501iBtJOIdHG52dFi66EEfQ no comment (ED25519)
   ec2: 2048 SHA256:L8l6pepcA7iqW/jBecQjVZClUrKY+o2cHLI0iHerbVc no comment (RSA)
   ```

```
ec2: -----END SSH HOST KEY FINGERPRINTS-----
ec2: ##############################################################
```

> **Note**
>
> 當您連線至執行個體時，您會參考此指紋。

使用 Connect 到您的 Linux 執行個體 SSH

您可以使用多種方式連接到 Linux 執行個體SSH。某些方法取決於您連接的本機電腦的作業系統。其他方法是以瀏覽器為基礎，例如「EC2執行個體 Connect」或「 AWS Systems Manager 工作階段管理員」，可從任何電腦上使用。您可以使用連線SSH到 Linux 執行個體並執行指令，或用SSH於在本機電腦和執行個體之間傳輸檔案。

使用連線到 Linux 執行個體之前SSH，請先完成下列先決條件：

- 請確認您的執行個體已通過狀態檢查。執行個體可能需要幾分鐘的時間才能準備好接受連線要求。如需詳細資訊，請參閱檢視狀態檢查。
- 確保與執行個體相關聯的安全群組允許來自 IP 位址的傳入SSH流量。如需詳細資訊，請參閱從您電腦連線到執行個體的規則。
- 獲取所需的實例詳細信息.
- 找出私有金鑰並設定許可.
- (選用) 取得執行個體指紋.

然後，從下列其中一個選項中選擇，以使用連線到 Linux 執行個體SSH。

- 使 Connect 用SSH戶端連線
- 使用 Pu Connect TTY
- 使用傳輸檔案 SCP

如果您無法連線至執行個體且需要疑難排解協助，請參閱解決連線到 Amazon EC2 Linux 執行個體的問題。

使用用SSH戶端 Connect 至您的 Linux 執行個體

您可以使用安全殼層 (SSH) 從本機電腦連線到 Linux 執行個體。如需其他選項的更多資訊，請參閱Connect 至您的EC2執行個體。

> **Note**
>
> 如果您在嘗試連線至執行個體時收到錯誤訊息，請確認執行個體符合所有 SSH連線先決條。如果其符合所有先決條件，但您仍然無法連線至 Linux 執行個體，請參閱 解決連線到 Amazon EC2 Linux 執行個體的問題。

目錄

- SSH連線先決條
- 使用用SSH戶端 Connect 至您的 Linux 執行個體

SSH連線先決條

在您可以使用連線到 Linux 執行個體之前SSH，請先完成下列工作。

完成一般先決條件。

- 請確認您的執行個體已通過狀態檢查。執行個體可能需要幾分鐘的時間才能準備好接受連線要求。如需詳細資訊，請參閱檢視狀態檢查。
- 獲取所需的實例詳細信息.
- 找出私有金鑰並設定許可.
- (選用) 取得執行個體指紋.

允許來自 SSH IP 位址的入站流量。

確保與執行個體相關聯的安全群組允許來自 IP 位址的傳入SSH流量。如需詳細資訊，請參閱從您電腦連線到執行個體的規則。

在您的本機電腦上安裝SSH用戶端 (如有需要)。

根據預設，您的本機電腦可能已安裝SSH用戶端。您可以通過在終端窗口中輸入以下命令來驗證這一點。如果您的計算機無法識別該命令，則必須安裝SSH客戶端。

```
ssh
```

以下是一些適用於 Windows 的可能選項。如果您的電腦執行不同的作業系統，請參閱該作業系統的文件以取得用SSH戶端選項。

SSH在視窗上安裝開啟

SSH在 Windows 上安裝「開啟」之後，您可以使用 Windows 電腦連線到 Linux 執行個體SSH。開始之前，請確定您符合下列需求。

Windows 版本

您電腦上的視窗版本必須是視窗伺服器 2019 或更新版本。

對於早期版本的視窗，請下載並安裝 Win32 打開SSH代替。

PowerShell 要求

若要使用在 Windows 作業系統SSH上安裝「開啟」 PowerShell，您必須執行 5.1 或更新PowerShell版本，且您的帳戶必須是內建管理員群組的成員。`$PSVersionTable.PSVersion`從運行 PowerShell 以檢查您的PowerShell 版本。

若要檢查您是否是內建管理員群組的成員，請執行下列 PowerShell 命令：

```
(New-Object
 Security.Principal.WindowsPrincipal([Security.Principal.WindowsIdentity]::GetCurrent())).Is
```

如果您是內建管理員群組的成員，則輸出為 `True`。

若要使用安裝開啟 SSH Windows 版 PowerShell，請執行下列 PowerShell 命令。

```
Add-WindowsCapability -Online -Name OpenSSH.Client~~~~0.0.1.0
```

下列為範例輸出。

```
Path          :
Online        : True
RestartNeeded : False
```

要使用SSH從 Windows 中卸載打開 PowerShell，請運行以下 PowerShell 命令。

```
Remove-WindowsCapability -Online -Name OpenSSH.Client~~~~0.0.1.0
```

下列為範例輸出。

```
Path          :
Online        : True
RestartNeeded : True
```

安裝視窗子系統 (WSL)

WSL在 Windows 上安裝之後，您可以使用 Linux 命令列工具 (例如用SSH戶端) 從 Windows 電腦連線到 Linux 執行個體。

請遵循中的說明進行在 Windows 執行個體上安裝適用於 Linux 的EC2視窗子系統 如果您按照微軟的安裝指南中的說明進行操作，他們會安裝 Linux 的 Ubuntu 發行版。如果您願意，您可以安裝不同的 Linux 發行版本。

在WSL終端機視窗中，將.pem檔案 (針對您在啟動時為執行個體指定的金鑰組) 從 Windows 複製到 WSL。請注意連線至執行個體時WSL要使用的.pem檔案完整路徑。如需有關如何指定 Windows 硬碟路徑的資訊，請參閱如何存取 C 磁碟機？。

```
cp /mnt/<Windows drive letter>/path/my-key-pair.pem ~/WSL-path/my-key-pair.pem
```

如需解除安裝 Linux 版 Windows 子系統的相關資訊，請參閱如何解除安裝WSL發行版本？ 。

使用用SSH戶端 Connect 至您的 Linux 執行個體

請遵循下列程序，使用用SSH戶端連線至 Linux 執行個體。

使用用SSH戶端連線至執行個體

1. 在電腦上開啟終端機視窗。
2. 使用指ssh令連線至執行個體。您需要在先決條件中收集的執行個體相關詳細資訊。例如，您需要私鑰（.pem文件）的位置，用戶名和公共DNS名稱或IPv6地址。以下是範例命令。
   - (公用DNS) 若要使用公用DNS名稱，請輸入下列命令。

     ```
     ssh -i /path/key-pair-name.pem instance-user-name@instance-public-dns-name
     ```

   - (IPv6) 或者，如果您的執行個體有IPv6位址，請輸入下列指令以使用該IPv6位址。

     ```
     ssh -i /path/key-pair-name.pem instance-user-name@instance-IPv6-address
     ```

以下是回應範例。

```
The authenticity of host 'ec2-198-51-100-1.compute-1.amazonaws.com (198-51-100-1)'
 can't be established.
ECDSA key fingerprint is l4UB/neBad9tvkgJf1QZWxheQmR59WgrgzEimCG6kZY.
Are you sure you want to continue connecting (yes/no)?
```

3. (選擇性) 確認安全性警示中的指紋與指紋相符。如果這些指紋不匹配，可能是有人試圖 man-in-the-middle 攻擊。如果它們符合，請繼續下一個步驟。如需詳細資訊，請參閱取得執行個體指紋。
4. 輸入 **yes**。

   您會看到如下的回應：

```
Warning: Permanently added 'ec2-198-51-100-1.compute-1.amazonaws.com' (ECDSA) to
 the list of known hosts.
```

使用 Pu Connect 到您的 Linux 執行個體 TTY

您可以使用 Pu TTY (一個免費的 Windows 用SSH戶端) 連線到您的 Linux 執行個體。

如果您執行的是 Windows Server 2019 或更新版本，我們建議您使用開放原始碼連線工具SSH，使用SSH通訊協定進行遠端登入。

> **Note**
>
> 如果您在嘗試連線至執行個體時收到錯誤訊息，請確認執行個體符合所有 SSH連線先決條。如果其符合所有先決條件，但您仍然無法連線至 Linux 執行個體，請參閱 解決連線到 Amazon EC2 Linux 執行個體的問題。

目錄

- 必要條件
- （可選）使用 P 轉換您的私鑰 uTTYgen
- 連接至您的 Linux 執行個體

必要條件

使用 Pu 連線到 Linux 執行個體之前TTY，請先完成下列工作。

完成一般先決條件。

- 請確認您的執行個體已通過狀態檢查。執行個體可能需要幾分鐘的時間才能準備好接受連線要求。如需詳細資訊，請參閱 檢視狀態檢查。
- 獲取所需的實例詳細信息.
- 找出私有金鑰並設定許可.
- (選用) 取得執行個體指紋.

允許來自 SSH IP 位址的入站流量。

確保與執行個體相關聯的安全群組允許來自 IP 位址的傳入SSH流量。如需詳細資訊，請參閱 從您電腦連線到執行個體的規則。

在本地計算機TTY上安裝 Pu（如果需要）。

TTY從 Pu 下載頁面TTY下載並安裝 Pu。如果您已經TTY安裝了早期版本的 Pu，我們建議您下載最新版本。務必安裝整個套件。

使用 P 將您的私鑰轉換為PPK格式uTTYgen。

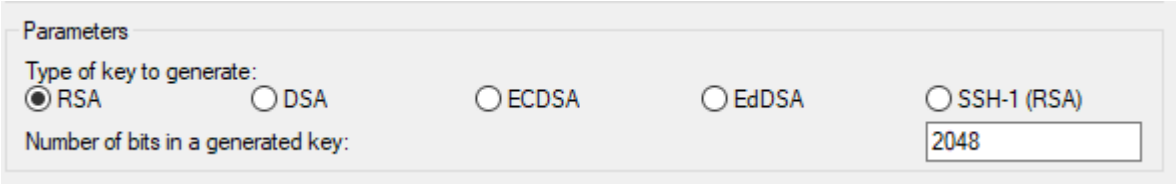
您必須為啟動執行個體時指定的 key pair 指定私密金鑰。如果您以 .pem 格式建立私密金鑰，則必須將其轉換為PPK檔案以與 Pu 搭配使用。TTY找出私密金鑰 (.pem 檔案)，然後依照中的步驟執行。使用 P 轉換您的私鑰 uTTYgen

（可選）使用 P 轉換您的私鑰 uTTYgen

Pu 本身TTY不支持SSH密鑰PEM格式。Pu TTY 提供了一個名為 P 的工具uTTYgen，它將PEM密鑰轉換為 Pu 所需的PPK格式TTY。如果您使用PEM格式而不是格式創建密鑰，則必須將私鑰（.pem 文件）轉換為此格式（.ppk 文件）以與 Pu 一起使用。PPK TTY

將私密金鑰從轉換PEM為PPK格式

1. 從「開始」功能表中選擇「所有程式」、「Pu」TTY、「P」uTTYgen。
2. 在要產生的金鑰類型下，選擇RSA。如果您的 P 版本uTTYgen 不包含此選項，請選擇 SSH-2 RSA。



3. 選擇載入。依預設，P 只uTTYgen 會顯示副檔名為的檔案`.ppk`。若要尋找您的 `.pem` 檔案，請選擇顯示所有類型之檔案的選項。

4. 為您在啟動執行個體時所指定的金鑰對選取 `.pem` 檔案，然後選取 Open (開啟)。P uTTYgen 會顯示已成功匯入`.pem`檔案的通知。選擇確定。
5. 若要TTY以 Pu 可使用的格式儲存金鑰，請選擇 [儲存私密金鑰]。P uTTYgen 會顯示有關儲存金鑰而不使用複雜密碼的警告。選擇 Yes (是)。

> **Note**
>
> 私有金鑰上的複雜密碼是額外一層保護。即使有人探索到您的私有金鑰，在沒有複雜密碼的情況下就無法使用您的私有金鑰。使用複雜密碼的缺點是提高自動化的難度，因為需要人力介入登入執行個體，或將檔案複製到執行個體。

6. 為您用於金鑰對的金鑰指定相同名稱 (例如，`key-pair-name`)，然後選擇 Save (儲存)。Pu TTY 會自動添加`.ppk`文件擴展名。

您的私鑰現在採用了正確的格式，可與 Pu 一起使用TTY。您現在可以使用 Pu TTY 的用SSH戶端連線到執行個體。

連接至您的 Linux 執行個體

請使用下列程序，使用 Pu 連線到您的 Linux 執行個體TTY。需要您為私有金鑰建立的 `.ppk` 檔案。如需詳細資訊，請參閱上一節中的（可選）使用 P 轉換您的私鑰 uTTYgen。如果您在嘗試連線至執行個體時遇到錯誤，請參閱 解決連線到 Amazon EC2 Linux 執行個體的問題。

上次測試的版本 — Pu TTY .78

使用 Pu 連線至執行個體 TTY

1. 啟動 PuTTY（從「開始」菜單中搜索 Pu，TTY然後選擇「打開」）。

2. 在 Category (類別) 窗格中選擇 Session (工作階段)，然後填妥下列欄位：

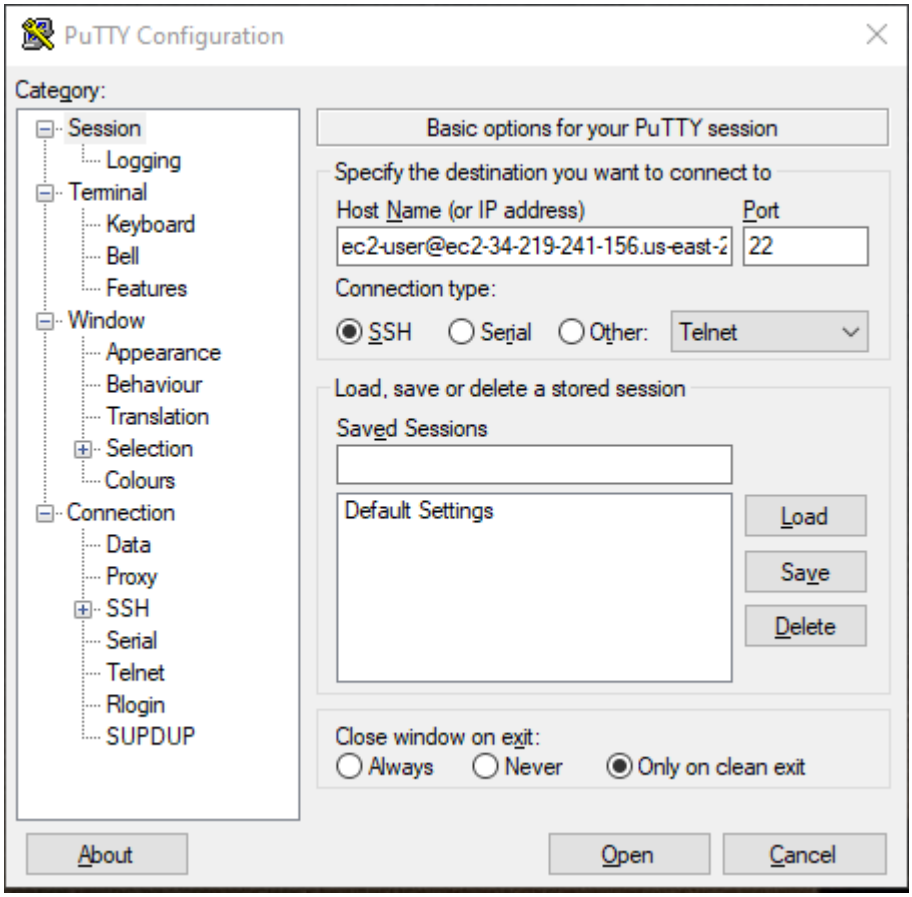
   a. 在 Host Name (主機名稱) 方塊中，執行下列其中一項作業：

      - (公用DNS) 若要使用執行個體的公開DNS名稱進行連線，請輸入 *instance-user-name*@*instance-public-dns-name*.
      - (IPv6) 或者，如果您的執行個體有IPv6位址，要使用執行個體的IPv6位址進行連線，請輸入 *instance-user-name*@*instance-IPv6-address*.

      如需如何取得執行個體的使用者名稱，以及執行個體的公開DNS名稱或IPv6地址的詳細資訊，請參閱獲取所需的實例詳細信息。

   b. 確保 Port (連接埠) 值是 22。

   c. 在「連線類型」下，選取SSH。



3. （可選）您可以TTY將 Pu 配置為定期自動發送「keepalive」數據，以使會話保持活動狀態。這對避免您執行個體因工作階段閒置而中斷連線很有幫助。在類別窗格中選擇連線，然後在保持連線之間的秒數中輸入所需間隔。例如，如果您的工作階段在閒置 10 分鐘後中斷連線，請輸入 180，將 Pu TTY 設定為每 3 分鐘傳送一次保存資料。

4. 在「類別」窗格中，展開「連線」和「驗證」。SSH選擇憑證。
5. 在要驗證的私有金鑰檔案旁，選擇瀏覽。在選取私有金鑰檔案對話方塊中，選取您為金鑰對產生的 .ppk 檔案。您可以按兩下檔案，或在選取私有金鑰檔案對話方塊中選擇開啟。
6. (選用) 如果您打算稍後再次連線到此執行個體，可以儲存工作階段資訊以供日後使用。在類別窗格中，選擇工作階段。在儲存的工作階段中輸入工作階段的名稱，然後選擇儲存。
7. 若要連線至執行個體，請選擇開啟。
8. 如果這是您第一次連線至此執行個體，Pu TTY 會顯示安全性警示對話方塊，詢問您是否信任要連線的主機。

   a. (選用) 確認安全提醒對話方塊中的指紋是否符合您之前在(選用) 取得執行個體指紋中取得的指紋。如果這些指紋不相符，可能是有人試圖攻擊 man-in-the-middle「」。如果它們符合，請繼續下一個步驟。

   b. 選擇 Accept (接受)。隨即開啟視窗，您會連線到您的執行個體。

> **Note**
>
> 如果您在將私密金鑰轉換為 Pu TTY 格式時指定了複雜密碼，則必須在登入執行個體時提供該複雜密碼。

如果您在嘗試連線至執行個體時遇到錯誤，請參閱 解決連線到 Amazon EC2 Linux 執行個體的問題。

使用下列方式將檔案傳輸到 Linux 執行 SCP

在本機電腦和 Linux 執行個體之間傳輸檔案的一種方法是使用安全複製通訊協定 (SCP)。本節說明如何使用傳輸檔案SCP。此程序與使用連接至執行個體的程序類似SSH。

使用連線到 Linux 執行個體之前SCP，請先完成下列工作：

- 完成一般先決條件。
  - 請確認您的執行個體已通過狀態檢查。執行個體可能需要幾分鐘的時間才能準備好接受連線要求。如需詳細資訊，請參閱檢視狀態檢查。
  - 獲取所需的實例詳細信息.
  - 找出私有金鑰並設定許可.
  - (選用) 取得執行個體指紋.
- 允許來自 SSH IP 位址的入站流量。

確保與執行個體相關聯的安全群組允許來自 IP 位址的傳入SSH流量。如需詳細資訊，請參閱從您電腦連線到執行個體的規則。

- 安裝用SCP戶端。

  默認情況下，大多數 Linux，Unix 和蘋果計算機都包含一個SCP客戶端。如果您沒有，Open SSH 專案會提供包括SCP用戶端在內的完整SSH工具套件的免費實作。如需詳細資訊，請參閱https://www.openssh.com。

下列程序會逐步引導您使用執行個體的公開DNS名稱SCP來傳輸檔案，或使用IPv6位址 (如果執行個體有的話) 來傳輸檔案。

用SCP於在電腦和執行個體之間傳輸檔案

1. 決定來源檔案在電腦上的位置，以及執行個體上的目的地路徑。在下列範例中，私密金鑰檔案的名稱是`key-pair-name.pem`，要傳輸的檔案是`my-file.txt`，執行個體的使用者名稱為ec2-user，執行個體的公開DNS名稱為`instance-public-dns-name`，而執行個體的IPv6位址為`instance-IPv6-address`。
   - (公用DNS) 若要將檔案傳輸到執行個體上的目的地，請從電腦輸入下列指令。

     ```
     scp -i /path/key-pair-name.pem /path/my-file.txt ec2-user@instance-public-dns-name:path/
     ```

   - (IPv6) 若要將檔案傳輸到執行個體上的目的地 (如果執行個體有IPv6位址)，請從您的電腦輸入下列指令。地IPv6址必須用方括號（[ ]）括起來，必須轉義（\）。

     ```
     scp -i /path/key-pair-name.pem /path/my-file.txt ec2-user@\[instance-IPv6-address\]:path/
     ```

2. 如果您尚未使用連線至執行個體SSH，您會看到類似下列的回應：

   ```
   The authenticity of host 'ec2-198-51-100-1.compute-1.amazonaws.com (10.254.142.33)'
   can't be established.
   RSA key fingerprint is 1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f.
   Are you sure you want to continue connecting (yes/no)?
   ```

   (選用) 您可以選用驗證安全性警示中的指紋是否符合執行個體指紋。如需詳細資訊，請參閱(選用) 取得執行個體指紋。

輸入 **yes**。

3. 如果傳輸成功，則回應會和下列內容相似：

```
Warning: Permanently added 'ec2-198-51-100-1.compute-1.amazonaws.com' (RSA)
to the list of known hosts.
my-file.txt                                   100%   480     24.4KB/s   00:00
```

4. 若要向另一個方向傳輸檔案 (從 Amazon EC2 執行個體到電腦)，請反轉主機參數的順序。例如，您可以`my-file.txt`從EC2執行個體傳輸到本機電腦上的目的地`my-file2.txt`，如下列範例所示。

   - (公用DNS) 若要將檔案傳輸到電腦上的目的地，請從電腦輸入以下指令。

     ```
     scp -i /path/key-pair-name.pem ec2-user@instance-public-dns-name:path/my-
     file.txt path/my-file2.txt
     ```

   - (IPv6) 如果執行個體有IPv6位址，若要將檔案傳輸到電腦上的目的地，請從您的電腦輸入下列指令。地IPv6址必須用方括號（[ ]）括起來，必須轉義（\）。

     ```
     scp -i /path/key-pair-name.pem ec2-user@\[instance-IPv6-address\]:path/my-
     file.txt path/my-file2.txt
     ```

管理 Amazon EC2 Linux 執行個體上的系統使用者

每個 Linux 執行個體都會使用預設 Linux 系統使用者來啟動。您可以將使用者新增至執行個體，並刪除使用者。

對於預設使用者，預設使用者名稱是由您啟動執行個體時所指定的使用者名稱決定。AMI

> **Note**
>
> 預設情況下，密碼驗證和根登入會停用，而 sudo 會啟用。若要登入執行個體，您必須建立金鑰對。如需有關登入的詳細資訊，請參閱使用 Connect 到您的 Linux 執行個體 SSH。
> 您可以允許執行個體的密碼驗證和根登入。如需詳細資訊，請參閱適用於您作業系統的文件。

> **Note**
>
> Linux 系統用戶不應與IAM用戶混淆。如需詳細資訊，請參閱《使IAM用指南》中的〈IAM使用者〉

目錄

- 預設使用者名稱
- 考量事項
- 建立使用者
- 移除使用者

預設使用者名稱

執行個體的預設使用者名稱取決於您啟動EC2執行個體時所指定的使用者名稱。AMI

預設的使用者名稱為：

- 對於 Amazon LinuxAMI，用戶名是`ec2-user`.
- 就 CentOS 而言AMI，使用者名稱為`centos`或`ec2-user`。
- 對於 Debian 來說AMI，使用者名稱是`admin`.
- 對於 Fedora 而言AMI，使用者名稱是`fedora`或`ec2-user`。
- 對於 a RHELAMI，使用者名稱為`ec2-user`或`root`。
- 對於 a SUSEAMI，使用者名稱為`ec2-user`或`root`。
- 對於 UbuntuAMI，使用者名稱為`ubuntu`.
- 對於 OracleAMI，使用者名稱為`ec2-user`。
- 對於 BitnamiAMI，用戶名是. `bitnami`

> **Note**
>
> 若要尋找其他 Linux 發行版的預設使用者名稱，請洽詢提AMI供者。

考量事項

許多應用程式都適合使用預設使用者。不過，您可以選擇新增使用者，讓個人能有自己的檔案和工作空間。此外，為新使用者建立使用者也會比授予多名 (可能缺乏經驗) 預設使用者的存取權更安全，因為若預設使用者使用不當，可能會對系統造成重大傷害。如需詳細資訊，請參閱保護EC2執行個體的秘訣。

若要讓使用者透過 Linux 系統使用者SSH存取您的EC2執行個體，您必須與使用者共用SSH金鑰。或者，您也可以使用EC2執行個體 Connect，為使用者提供存取權，而不需要共用和管理SSH金鑰。如需詳細資訊，請參閱使用執行個體 Connect Connect 到 Linux EC2 執行個體。

建立使用者

首先建立使用者，然後新增允許使用者連線到執行個體並登入執行個體的SSH公開金鑰。

> **Important**
>
> 在此程序的步驟 1 中，您會建立新的 key pair。由於 key pair 的功能類似於密碼，因此安全地處理它至關重要。如果您為使用者建立 key pair，您必須確保私密金鑰已安全地傳送給他們。或者，使用者也可以透過建立自己的 key pair 來完成步驟 1 和 2，將私密金鑰保持在其電腦上的安全性，然後傳送公開金鑰給您以完成步驟 3 的程序。

建立使用者

1. 建立新的金鑰對。您必須將 `.pem` 檔案提供給您要為其建立使用者的使用者。他們必須使用此檔案來連接到執行個體。
2. 從您在上一個步驟中建立的金鑰對中，擷取公開金鑰。

```
$ ssh-keygen -y -f /path_to_key_pair/key-pair-name.pem
```

此命令會傳回公有金鑰，如下列範例所示。

```
ssh-rsa
 AAAAB3NzaC1yc2EAAAADAQABAAABAQClKsfkNkuSevGj3eYhCe53pcjqP3maAhDFcvBS7O6Vhz2ItxCih
+PnDSUaw+WNQn/mZphTk/a/gU8jEzoOWbkM4yxyb/wB96xbiFveSFJuOp/
d6RJhJOI0iBXrlsLnBItntckiJ7FbtxJMXLvvwJryDUilBMTjYtwB+QhYXUMOzce5Pjz5/
i8SeJtjnV3iAoG/cQk+0FzZqaeJAAHco
+CY/5WrUBkrHmFJr6HcXkvJdWPkYQS3xqC0+FmUZofz221CBt5IMucxXPkX4rWi
+z7wB3RbBQoQzd8v7yeb7OzlPnWOyN0qFU0XA246RA8QFYiCNYwI3f05p6KLxEXAMPLE
```

3. 連線到執行個體。
4. 使用 adduser 命令建立使用者並將其新增到系統 (在 `/etc/passwd` 檔案中具有項目)。此命令也會為使用者建立群組和主目錄。在此範例中，使用者命名為 *newuser*。

   - AL2Amazon 和亞馬遜系統 2

     使用 AL2 023 和 Amazon Linux 2 時，系統會在預設情況下停用密碼身份驗證的情況下建立使用者。

     ```
     [ec2-user ~]$ sudo adduser newuser
     ```

   - Ubuntu

     包含 `--disabled-password` 參數，以建立已停用密碼驗證的使用者。

     ```
     [ubuntu ~]$ sudo adduser newuser --disabled-password
     ```

5. 切換到新的使用者，以便您建立的目錄和檔案可擁有適當的所有權。

   ```
   [ec2-user ~]$ sudo su - newuser
   ```

   此提示會從 `ec2-user` 變更為 *newuser*，表示您已將 shell 工作階段切換至新使用者。

6. 將SSH公開金鑰新增至使用者。首先在使用者的主目錄中建立SSH金鑰檔案的目錄，然後建立金鑰檔案，最後將公開金鑰貼入金鑰檔案，如下列子步驟所述。

   a. 在 `.ssh` 主目錄中建立 *newuser* 目錄，然後將其檔案許可變更為 `700` (只有擁有者能讀取、寫入和開啟目錄)。

      ```
      [newuser ~]$ mkdir .ssh
      ```

      ```
      [newuser ~]$ chmod 700 .ssh
      ```

      > **Important**
      > 若沒有這些明確的檔案許可，使用者將無法登入。

   b. 在 `.ssh` 目錄中建立名為 `authorized_keys` 的檔案，然後將其檔案許可變更為 `600` (只有擁有者能讀取和寫入檔案)。

```
[newuser ~]$ touch .ssh/authorized_keys
```

```
[newuser ~]$ chmod 600 .ssh/authorized_keys
```

> **Important**
>
> 若沒有這些明確的檔案許可，使用者將無法登入。

c. 使用您喜愛的文字編輯器 (例如 vim 或 nano) 來開啟 `authorized_keys` 檔案。

```
[newuser ~]$ nano .ssh/authorized_keys
```

將您在步驟 2 中擷取到的公開金鑰貼到檔案中，並儲存這些變更。

> **Important**
>
> 請確定將此公開金鑰貼成連續一行。此公開金鑰絕對不可分成多行。

使用者現在應該能使用對應到您為 `authorized_keys` 檔案新增之公有金鑰的私有金鑰，在您的執行個體上登入 *`newuser`* 使用者。如需連接至 Linux 執行個體之不同方法的詳細資訊，請參閱使用 Connect 到您的 Linux 執行個體 SSH。

移除使用者

若不再需要使用者，您可以移除該使用者，使該使用者無法再被使用。

使用 userdel 命令來移除系統的使用者。當指定 `-r` 參數時，也會同時刪除使用者的主目錄和郵件多工緩衝處理。若要保留使用者的主目錄和郵件多工緩衝處理，請忽略 `-r` 參數。

```
[ec2-user ~]$ sudo userdel -r olduser
```

使用 Connect 至您的 Windows 執行個體 RDP

您可以使用遠程桌面連接到從大多數 Windows Amazon 機器映像（AMIs）創建的 Amazon EC2 實例。遠端桌面使用遠端桌面通訊協定 (RDP) 連線至執行個體並使用您的執行個體，就如同您使用前方電腦 (本機電腦) 的電腦一樣。其適用於多數的 Windows 版本，也適用於 Mac OS。

Windows Server 作業系統的授權允許為了管理用途而同時有兩個遠端連線。Windows Server 的授權包含在您的 Windows 執行個體價格中。如果您需要兩個以上的同時遠端連線，您必須購買遠端桌面服務 (RDS) 授權。如果您嘗試第三個連線，就會出現錯誤。

> **Tip**
>
> 如果您需要連線到執行個體，以便對 AWS Nitro 系統上建置的執行個體，進行開機、網路設定和其他問題的疑難排解，您可以使用 EC2實例的序列主控台。

目錄

- 使用用RDP戶端 Connect 至您的 Windows 執行個體
- 使用 Fleet Manager 連線到 Windows 執行個體
- 將檔案傳輸到 Windows 執行個體，使用 RDP

使用用RDP戶端 Connect 至您的 Windows 執行個體

您可以使用用RDP戶端連線到 Windows 執行個體，如下所示。

> **Tip**
>
> 或者，您也可以使用系統管理員叢集管理員或執行個體連線端點連線到 Windows EC2 執行個體。

必要條件

您必須符合下列先決條件，才能使用用RDP戶端連線到 Windows 執行個體。

- 完成一般先決條件。
  - 請確認您的執行個體已通過狀態檢查。執行個體可能需要幾分鐘的時間才能準備好接受連線要求。如需詳細資訊，請參閱檢視狀態檢查。
  - 獲取所需的實例詳細信息.
  - 找出私有金鑰並設定許可.
  - (選用) 取得執行個體指紋.
- 安裝用RDP戶端。

- (視窗) 視窗預設包含用RDP戶端。若要確認，請在命令提示視窗下輸入 mstsc。如果電腦無法辨識此命令，請參閱 Windows 首頁並搜尋下載 Microsoft 遠端桌面應用程式。
- （macOS X）從 Mac 應用程序商店下載 Microsoft 遠程桌面應用程序。
- (Linux) 使用雷米娜。
- 允許來自 RDP IP 位址的入站流量。

  確保與執行個體相關聯的安全群組允許來自 IP 位址的傳入RDP流量。如需詳細資訊，請參閱從您電腦連線到執行個體的規則。

擷取管理員密碼

如果您將執行個體加入網域，您可以使用來自的網域認證連線至執行個體 AWS Directory Service。在遠端桌面登入畫面上，請使用管理員的完整使用者名稱 (例如，**`corp.example.com\Admin`**) 和此帳戶的密碼，而不是使用本機電腦名稱和產生的密碼。

若要使用連線至 Windows 執行個體RDP，您必須擷取初始管理員密碼，然後在連線至執行個體時輸入此密碼。在執行個體啟動後，需要幾分鐘的時間處理才能提供密碼。

管理員帳戶的預設使用者名稱取決於中包含的作業系統 (OS) 語言AMI。若要確定正確的使用者名稱，請識別作業系統AMI的語言，然後選擇對應的使用者名稱。例如，對於英語操作系統，用戶名是`Administrator`，對於法國操作系統來說`Administrateur`，對於葡萄牙語操作系統而言，它是`Administrador`。如果作業系統的語言版本沒有相同語言的使用者名稱，請選擇使用者名稱`Administrator (Other)`。如需詳細資訊，請參閱 Microsoft TechNet Wiki 中的 Windows 系統管理員帳戶的本地化名稱。

擷取初始管理員密碼

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後選取 Connect (連線)。
4. 在 [Connect 至執行個體] 頁面上，選擇用RDP戶端索引標籤。
5. 在使用者名稱中，選擇管理員帳戶的預設使用者名稱。您選擇的使用者名稱必須與您用來啟動執行個體的作業系統 (OS) 語言相符。AMI如果沒有與您的作業系統相同語言的使用者名稱，請選擇「管理員 (其他)」。
6. 選擇取得密碼。
7. 在 [取得 Windows 密碼] 頁面上，執行下列動作：

a. 選擇「上傳私密金鑰檔案」，然後瀏覽至您在啟動執行個體時指定的私密金鑰 (`.pem`) 檔案。選取檔案並選取 Open (開啟)，將檔案的完整內容複製至此視窗。

b. 選擇「解密密碼」。[取得 Windows 密碼] 頁面會關閉，且執行個體的預設管理員密碼會顯示在 [密碼] 底下，取代先前顯示的 [取得密碼] 連結。

c. 複製密碼並將其保存在安全的地方。必須使用此密碼來連接至執行個體。

連線至 Windows 執行個體

下列程序會使用 Windows 的遠端桌面連線用戶端 (MSTSC)。如果您使用不同的用RDP戶端，請下載RDP檔案，然後參閱用RDP戶端的說明文件，瞭解建立RDP連線的步驟。

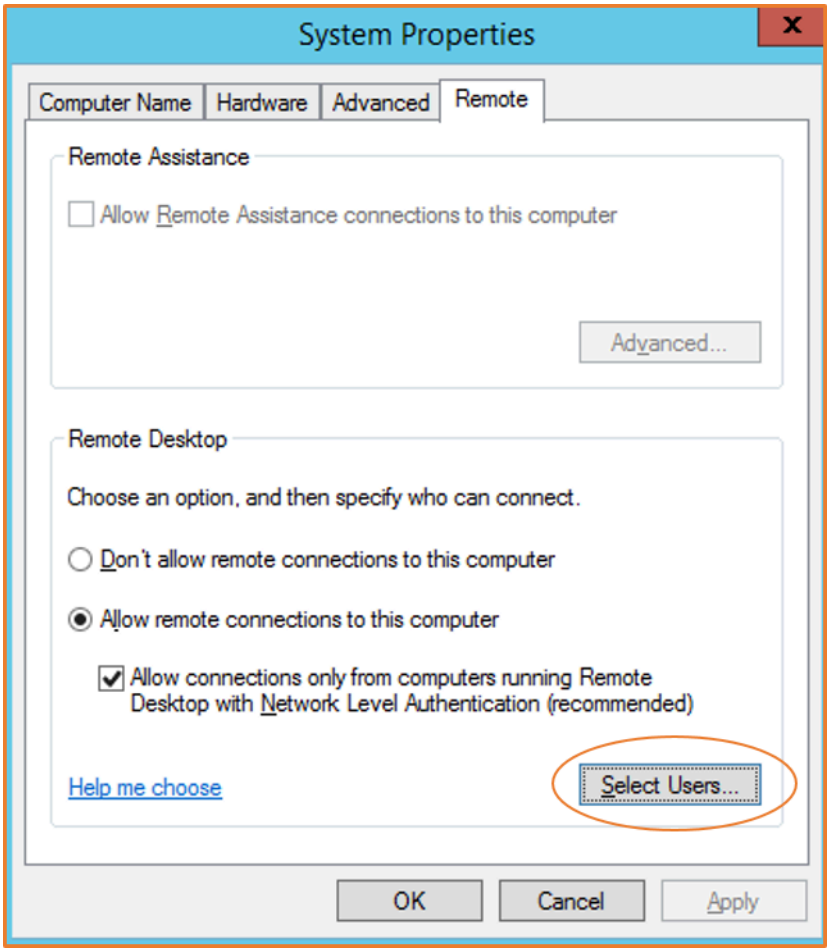
使用用RDP戶端連線至 Windows 執行個體

1. 在 [Connect 至執行個體] 頁面上，選擇 [下載遠端桌面檔案]。檔案下載完成時，請選擇「取消」返回「執行處理」頁面。RDP檔案即會下載至您的`Downloads`資料夾。
2. 執行`mstsc.exe`以開啟用RDP戶端。
3. 展開 [顯示選項]，選擇 [開啟]，然後從`Downloads`資料夾中選取 .rdp 檔案。
4. 根據預設，電腦是執行個體的公用IPv4DNS名稱，使用者名稱是系統管理員帳戶。若要使用來連線至執行個體，IPv6請將執行個體的公用IPv4DNS名稱取代為其IPv6位址。檢閱預設設定並視需要進行變更。
5. 選擇連線。如果您收到遠端連線發行者未知的警告，請選擇「Connect 線」以繼續。
6. 輸入您先前儲存的密碼，然後選擇 [確定]。
7. 由於自我簽署憑證的性質，您可能會收到安全憑證無法驗證的警告。執行以下任意一項：
   - 如果您信任憑證，請選擇 [是] 連線至您的執行個體。
   - [Windows] 在繼續之前，請將證書的指紋與系統日誌中的值進行比較，以確認遠端電腦的身份。選擇檢視憑證，然後從詳細資料索引標籤中選擇指紋。將此值與 [動作]、[監視及疑難排解]、[取得系統記錄] `RDPCERTIFICATE-THUMBPRINT` 中的值進行比較。
   - [Mac OS X] 繼續之前，請先將憑證的指紋與系統記錄檔中的值進行比較，以確認遠端電腦的身分識別。選擇顯示憑證，展開詳細資料，然後選擇SHA1指紋。將此值與 [動作]、[監視及疑難排解]、[取得系統記錄] `RDPCERTIFICATE-THUMBPRINT` 中的值進行比較。
8. 如果RDP連線成功，RDP用戶端會顯示 Windows 登入畫面，然後顯示 Windows 桌面。如果您收到錯誤訊息，請參閱the section called “遠端桌面無法連線到遠端電腦”。RDP連線完成後，您可以關閉用RDP戶端。

設定使用者帳戶

在您透過連線至執行個體之後RDP，建議您執行下列工作：

- 變更預設的管理員密碼。您可以在登入執行個體後變更密碼，就像使用任何其他執行 Windows 伺服器的電腦一樣。
- 在執行個體上建立其他擁有管理員權限的使用者。如果您忘記管理員密碼或管理員帳戶發生問題，這可作為保護措施。新使用者必須具備可遠端存取執行個體的許可。在 Windows 桌面或檔案總管上，以滑鼠右鍵按一下This PC (本機)，選取內容，以開啟系統內容。選取遠端設定，然後選取選取使用者，將使用者新增至遠端桌面使用者群組。



使用 Fleet Manager 連線到 Windows 執行個體

您可以使用叢集管理員功能 AWS Systems Manager，使用遠端桌面通訊協定 (RDP) 連線到 Windows 執行個體，並在中的相同頁面上顯示最多四個 Windows 執行個體 AWS Management Console。您可以直接從 Amazon EC2 主控台的執行個體頁面連線到叢集管理員遠端桌面中的第一個執行個體。如需

有關 Fleet Manager 的詳細資訊，請參閱《AWS Systems Manager 使用者指南》中的使用遠端桌面連線到受管節點。

如果您使用叢集管理員進行連線，則不需要特別允許來自 IP 位址的傳入RDP流量。Fleet Manager 會為您處理。

必要條件

在嘗試使用叢集管理員連線至執行個體之前，您必須先設定環境。若要取得更多資訊，請參閱《AWS Systems Manager 使用指南》中的〈設定環境〉。

若要使用叢集管理員連線至 Windows 執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 從導覽窗格中選擇 Instances (執行個體)。
3. 選取執行個體，然後選取 Connect (連線)。
4. 在用RDP戶端索引標籤上，針對 [連線類型] 選擇 [使用叢集管理員 Connect 線]
5. 選擇 Fleet Manager 遠端桌面。這將開啟 AWS Systems Manager 主控台中的 Fleet Manager Remote Desktop (Fleet Manager 遠端桌面) 頁面。
6. 輸入您的認證，然後選擇 [Connect]。
7. 如果RDP連線成功，叢集管理員會顯示 Windows 桌面平台。當您完成工作階段時，請選擇「動作」，「結束工作階段」。

如需詳細資訊，請參閱使用指南中的使用遠端桌面連線至 Windows Server 受管理執行個體。AWS Systems Manager

將檔案傳輸到 Windows 執行個體，使用 RDP

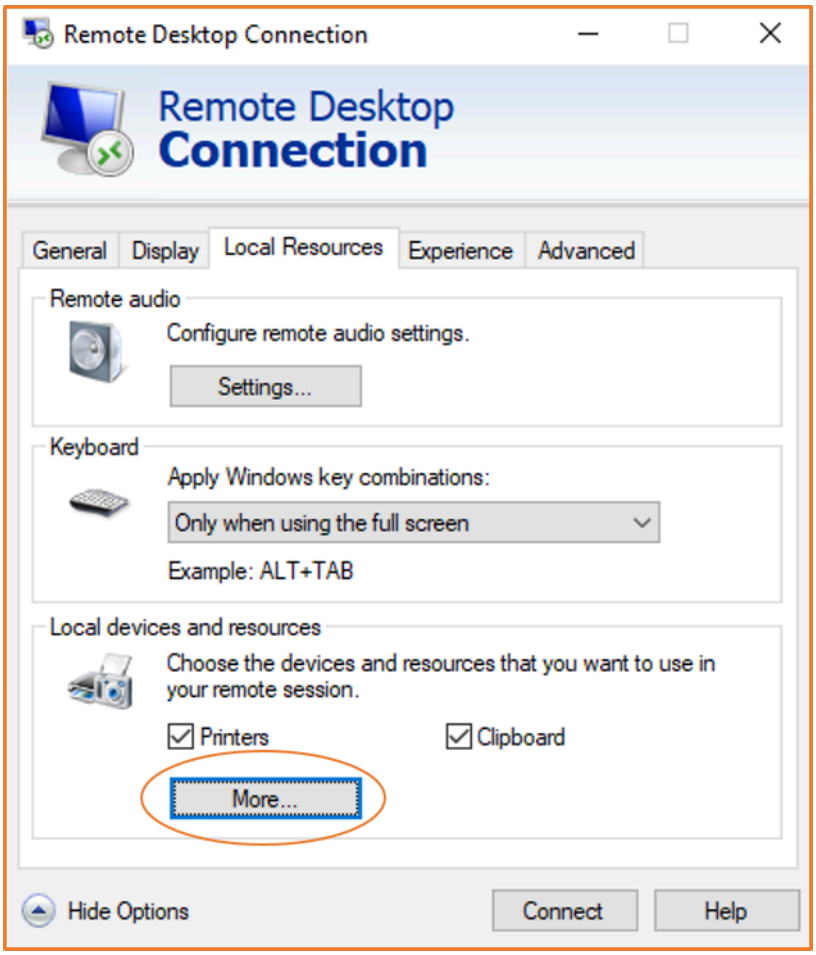
您可用與任何 Windows 伺服器相同的使用方式來使用 Windows 執行個體。例如，您可以使用 Microsoft 遠端桌面連線 (RDP) 軟體的本機檔案共用功能，在 Windows 執行個體和本機電腦之間傳輸檔案。您可以存取硬碟、磁碟機、可攜式媒體磁碟DVD機和對應網路磁碟機上的本機檔案。

如要從 Windows 執行個體存取本機檔案，您必須透過將遠端工作階段磁碟機映射至您本機磁碟機來啟用本機檔案共享功能。這些步驟略有不同，取決於您的本機電腦作業系統是 Windows 還是 macOS X。

Windows

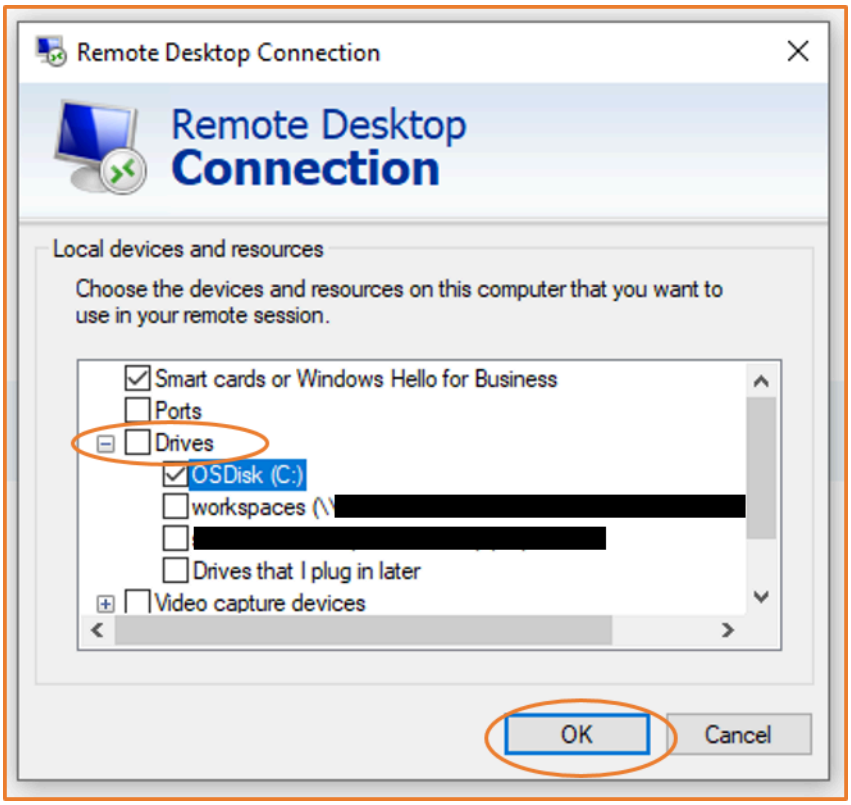
如要將遠端工作階段磁碟機映射至您本機 Windows 電腦的本機磁碟機

1. 開啟遠端桌面連線用戶端。
2. 選擇 Show Options (顯示選項)。
3. 將執行個體主機名稱新增至 [電腦] 欄位，並將使用者名稱新增至 [使用者名稱] 欄位，如下
   a. 在連線設定下，選擇開啟... ，然後瀏覽至您從 Amazon EC2 主控台下載的RDP捷徑檔案。檔案包含識別執行個體的 Public IPv4 DNS 主機名稱，以及管理員使用者名稱。
   b. 選取檔案，然後選取 Open (開啟)。「電腦」和「使用者名稱」欄位會填入RDP捷徑檔案中的值。
   c. 選擇 Save (儲存)。
4. 選擇 Local Resources (本機資源) 標籤。
5. 在 Local Devices and resources (本機裝置和資源) 下，選擇 More... (更多...)



6. 開啟 Drives (磁碟機) 並選取本機磁碟機，以映射到 Windows 執行個體。

7. 選擇 OK (確定)。

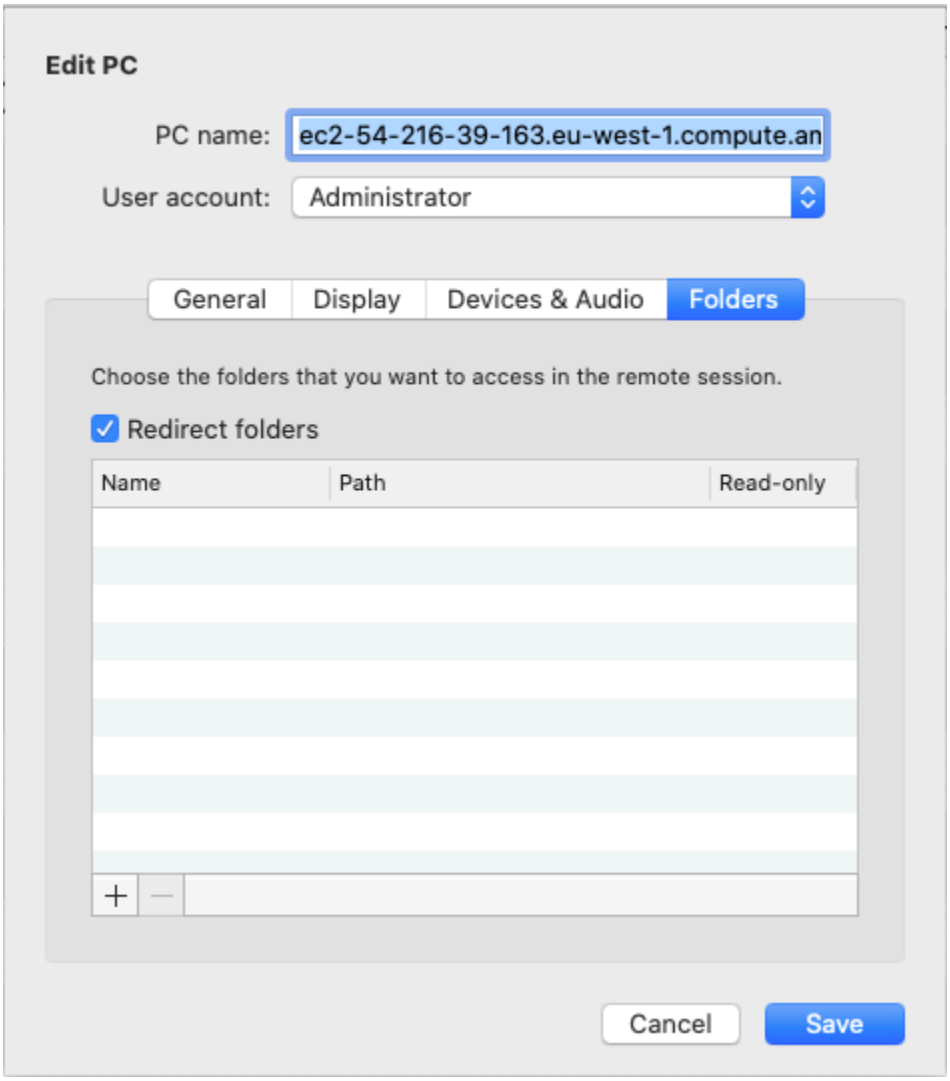


8. 選擇 Connect (連接) 以連接到您的 Windows 執行個體。

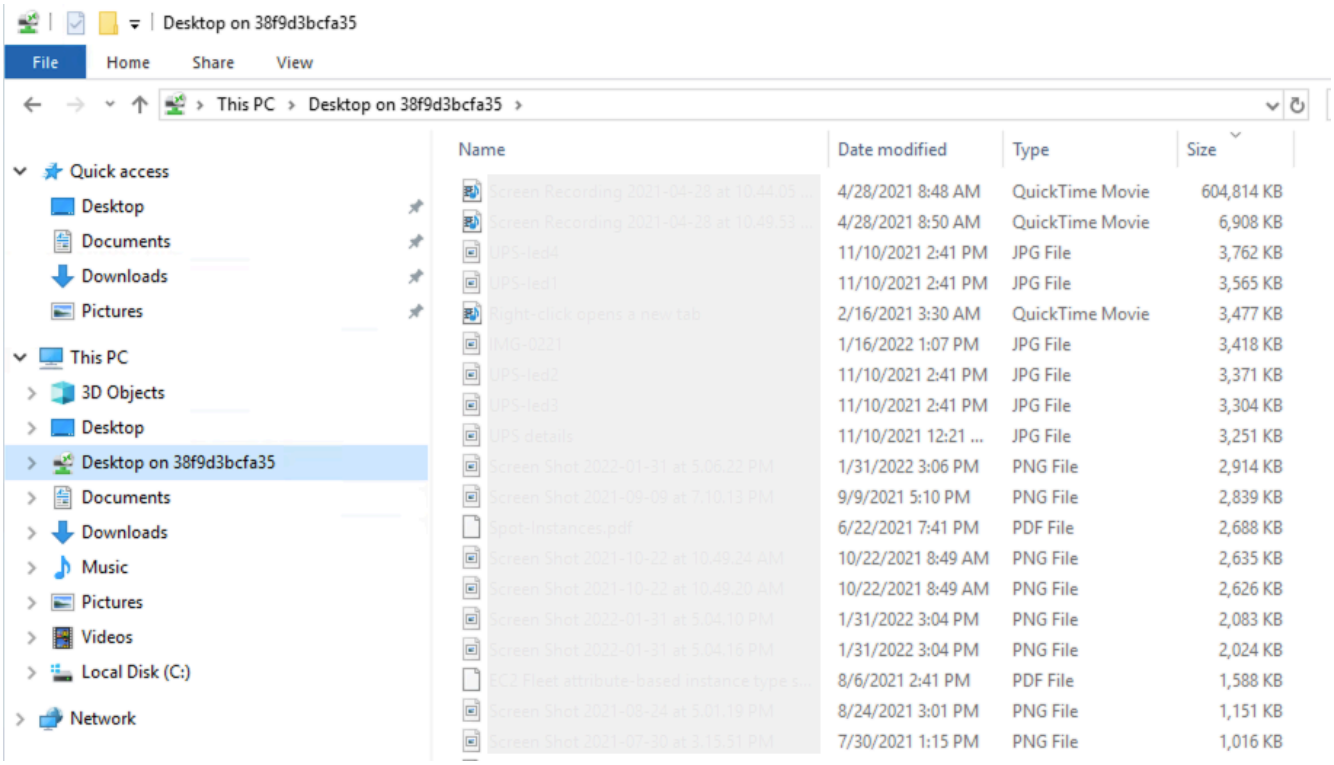
macOS X

如要將遠端工作階段磁碟機映射至本機 macOS X 電腦上的本機資料夾

1. 開啟遠端桌面連線用戶端。
2. 瀏覽至您從 Amazon EC2 主控台下載的RDP檔案 (當您最初連線至執行個體時)，然後將其拖曳至遠端桌面連線用戶端。
3. 在RDP檔案上按一下右鍵，然後選擇 「編輯
4. 選取 Folders (資料夾) 索引標籤，然後選取 Redirect folders (重新引導資料夾) 核取方塊。



5. 選擇左下方的 + 圖示，瀏覽至要映射的資料夾，然後選擇 Open (開啟)。為每個要映射的資料夾重複此步驟。
6. 選擇 Save (儲存)。
7. 選擇 Connect (連接) 以連接到您的 Windows 執行個體。系統會提示您輸入密碼。
8. 在執行個體上的檔案總管中，展開 This PC (此 PC)，然後找出您可從中存取本機檔案的共用資料夾。於下列螢幕擷取畫面中，本機電腦上的 Desktop (桌面) 資料夾已映射至執行個體上的遠端工作階段磁碟機。



如需在 Mac 電腦上使本機裝置可用於遠端工作階段的詳細資訊，請參閱 Mac 用戶端入門。

使用工作階段管理員 Connect 到 Amazon EC2 執行個體

工作階段管理員是一EC2項全受管 AWS Systems Manager 功能，可透過互動式、按一下瀏覽器型殼層或透過. AWS CLI您可以使用工作階段管理員，搭配您帳戶中的執行個體啟動工作階段。工作階段啟動之後，您可以在執行個體上執行互動式指令，就像對任何其他連線類型執行一樣。如需工作階段管理員的詳細資訊，請參閱《AWS Systems Manager 使用者指南》中的 AWS Systems Manager 工作階段管理員。

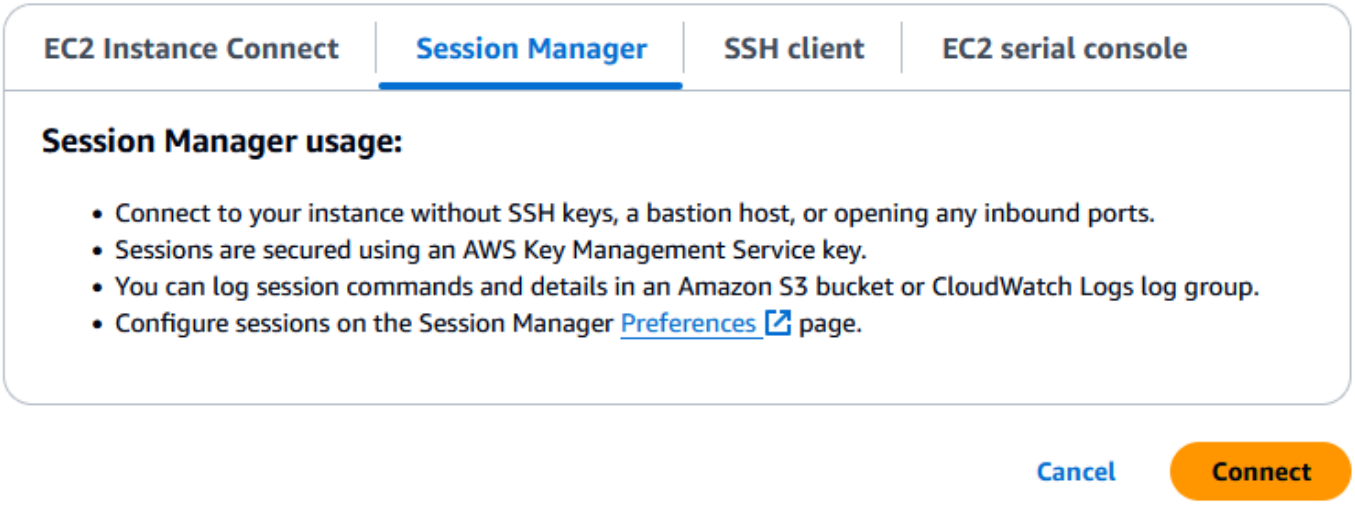
必要條件

嘗試使用工作階段管理員連線至執行個體之前，必須先完成必要的設定步驟。例如，執行個體必須由管理，SSM且必須具有 A mazonSSMManaged InstanceCore 原則的附加IAM角色。如需詳細資訊，請參閱設定工作階段管理員。

使用 Amazon 主控台上的工作階段管理器連接到 Amazon EC2 EC2 執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。

3. 選取執行個體，然後選取 Connect (連線)。
4. 對於連線方法，請選擇工作階段管理員。
5. 選擇「Connect」以啟動工作階段。



故障診斷

如果您收到未授權執行一或多個 Systems Manager 動作的錯誤訊息 (`ssm:command-name`)，則必須更新政策以允許您從 Amazon EC2 主控台啟動工作階段。如需詳細資訊和指示，請參閱《AWS Systems Manager 使用指南》中的工作階段管理員快速入門預設IAM原則。

使用執行個體 Connect Connect 到 Linux EC2 執行個體

Amazon EC2 執行個體 Connect 提供一種透過安全殼層 (SSH) 連接到 Linux 執行個體的安全方式。透過EC2執行個體 Connect，您可以使用 AWS Identity and Access Management (IAM) 原則和主參與者控制執行個體的SSH存取，不需要共用和管理SSH金鑰。使用「EC2執行個體連線」的所有連線要求都會記錄到，以 AWS CloudTrail便您可以稽核連線要求。

您可以使用EC2執行個體 Connect，透過 Amazon EC2 主控台或您選擇的用SSH戶端連接到執行個體。

當您使用執行個體 Connect Connect 至EC2執行個體時，執行個體連線會將SSH公開金鑰API推送至執行個體中繼資料，並保留 60 秒。附加至您的使用者的IAM政策授權您的使用者將公開金鑰推送至執行個體中繼資料。SSH常駐程式會使用`AuthorizedKeysCommand`和 `AuthorizedKeysCommandUser` (在安裝執行個體 Connect 時設定) 從執行個體中繼資料查詢公開金鑰以進行驗證，並將您連線至執行個體。

> **Tip**
>
> EC2執行個體連線是連線到 Linux 執行個體的其中一個選項。如需有關其他選項的詳細資訊，請參閱 使用 Connect 到您的 Linux 執行個體 SSH。若要連線至 Windows 執行個體，請參閱使用 Connect 至您的 Windows 執行個體 RDP。

定價

EC2執行個體 Connect 無需額外費用。

區域可用性

EC2所有執行個體 Connect 都可使用 AWS 區域。Local Zones 不支援此功能。

目錄

- 教學課程：使用執行個體連線完成 Connect 至執行個體EC2所需的設定
- EC2執行個體 Connect 的必要
- 授與EC2執行個體 Connect 的IAM權限
- 在執行個EC2體安裝EC2執行個體 Connect
- 使 Connect「執EC2行個體 Connect
- 解除安裝EC2實例 Connect

如需討論如何使用執行個體連線改善防禦主機安全性的部落格文章，請參閱使用 Amazon EC2 EC2 執行個體 Connect 保護堡壘主機。

教學課程：使用執行個體連線完成 Connect 至執行個體EC2所需的設定

若要使用 Amazon EC2 主控台中的EC2執行個體 Connect 連線到執行個體，您首先需要完成先決條件組態，以便成功連線到執行個體。本教學課程的目的是引導您完成工作，以完成必要條件組態。

教程概述

在本教學課程中，您將完成以下四項工作：

- 工作 1：授與使用EC2執行個體 Connect 所需的權限

首先，您將建立一個IAM政策，其中包含允IAM許您將公開金鑰推送至執行個體中繼資料的權限。您會將此原則附加至您的IAM身分識別 (使用者、使用者群組或角色)，以便您的IAM身分取得這些權限。

- 工作 2：允許從EC2執行個體 Connect 服務傳入您的執行個體的傳入流量

  然後，您將建立一個安全群組，允許從執行個EC2體 Connect 服務傳輸到執行個體的流量。當您使用 Amazon EC2 主控台中的EC2執行個體 Connect 連線到執行個體時，這是必要的。

- 工作 3：啟動執行個體

  然後，您將使用 In EC2 stance Connect 預先安裝AMI的執行個體啟動EC2執行個體，然後新增在上一個步驟中建立的安全性群組。

- 工作 4：Connect 至執行個體

  最後，您將使用 Amazon EC2 主控台中的EC2執行個體 Connect 來連線到執行個體。如果您可以連線，則可以確定您在工作 1、2 和 3 中完成的先決條件組態是否成功。

工作 1：授與使用EC2執行個體 Connect 所需的權限

當您使用執行個體 Connect Connect 至EC2執行個體時，EC2執行個體連線會將SSH公開金鑰API推送至執行個體中繼資料，並保留 60 秒。您需要將IAM原則附加至您的IAM身分識別 (使用者、使用者群組或角色)，才能授與將公開金鑰推送至執行個體中繼資料的必要權限。

任務目標

您將建立授與將公開金鑰推送至執行個體之權限的IAM原則。要允許的特定動作為`ec2-instance-connect:SendSSHPublicKey`。您還必須允許該`ec2:DescribeInstances`動作，以便在 Amazon EC2 主控台中檢視和選取執行個體。

建立原則之後，您會將原則附加至您的IAM身分識別 (使用者、使用者群組或角色)，以便您的IAM身分取得權限。

您將建立如下設定的原則：

```
{
    "Version": "2012-10-17",
    "Statement": [{
            "Effect": "Allow",
            "Action": "ec2-instance-connect:SendSSHPublicKey",
            "Resource": "*"
```

```
        },
        {
            "Effect": "Allow",
            "Action": "ec2:DescribeInstances",
            "Resource": "*"
        }
    ]
}
```

> **Important**
>
> 本教學課程中建立的IAM原則是高度寬鬆的原則；它可讓您使用任何使用者名稱連線至任何執行個體。AMI我們正在使用這個高度寬鬆的政策來保持教程簡單，並專注於本教程所教學的特定配置。不過，在生產環境中，我們建議您將IAM原則設定為提供最低權限權限。如需IAM原則範例，請參閱授與EC2執行個體 Connect 的IAM權限。

建立並附加可讓您使用EC2執行個體 Connect 至執行個體的IAM原則

1. 首先建立原IAM則
   a. 在開啟IAM主控台https://console.aws.amazon.com/iam/。
   b. 在導覽窗格中，選擇政策。
   c. 選擇 Create policy (建立政策)。
   d. 在 [指定權限] 頁面上，執行下列動作：
      i. 在「服務」中，選擇「EC2執行個體 Connect
      ii. 在「允許的動作」下，在搜尋欄位中開始輸入**send**以顯示相關動作，然後選取「SendSSHPublic Key」。
      iii. 在資源下，選擇全部。對於生產環境，我們建議您指定執行個體ARN，但在本教學課程中，您允許所有執行個體。
      iv. 選擇 [新增更多權限]。
      v. 對於「服務」，請選擇EC2。
      vi. 在「允許的動作」下，在搜尋欄位中開始輸入**describein**以顯示相關動作，然後選取DescribeInstances。
      vii. 選擇 Next (下一步)。
   e. 在「檢閱並建立」頁面上，執行下列動作：

i. 針對 Policy Name (政策名稱)，請輸入政策的名稱。

ii. 選擇 建立政策。

2. 然後將政策附加到您的身份

a. 在IAM主控台的導覽窗格中，選擇 [原則]。

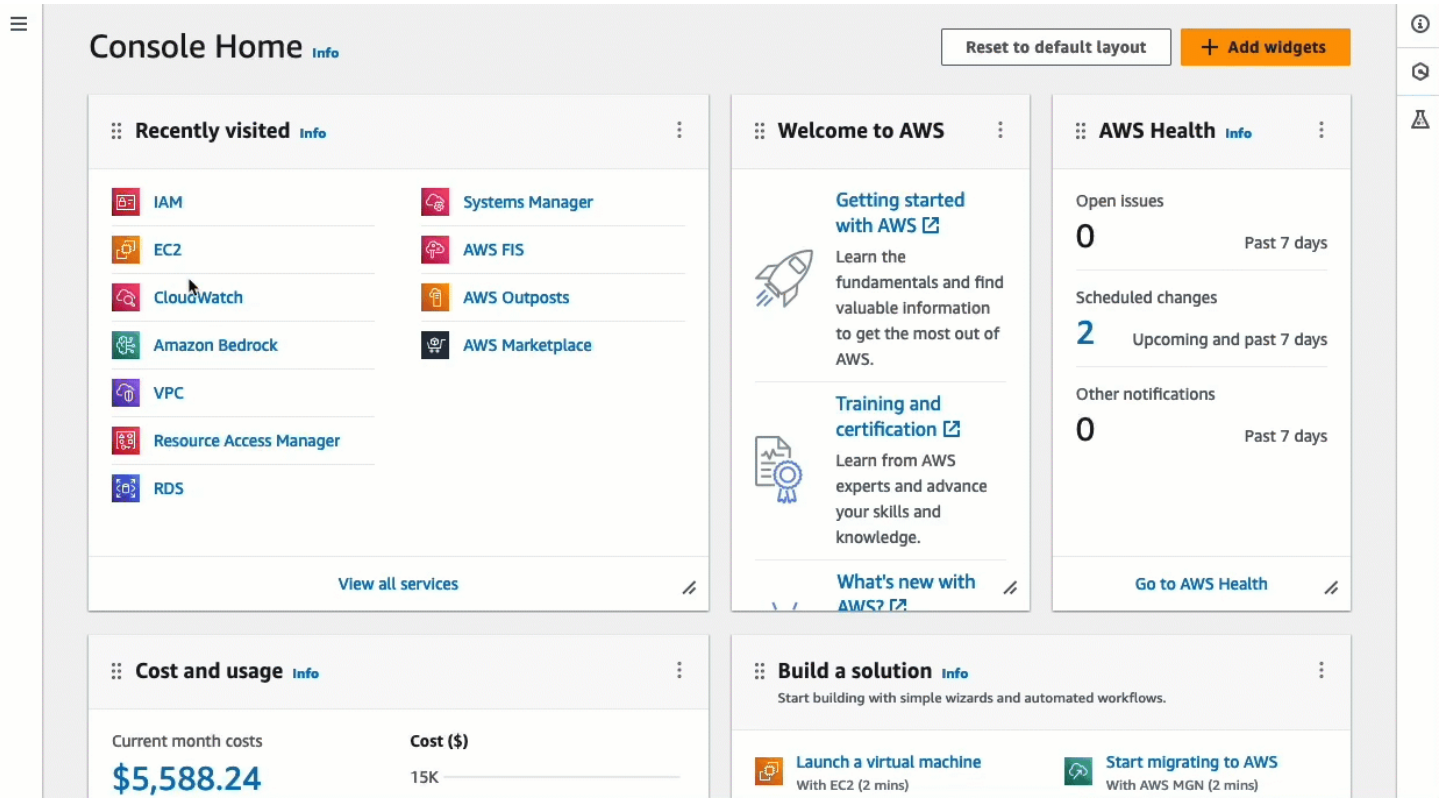
b. 在策略清單中，選取您建立之策略名稱旁邊的選項按鈕。您可以使用搜尋方塊來篩選政策清單。

c. 選擇 Actions (動作)、Attach (連接)。

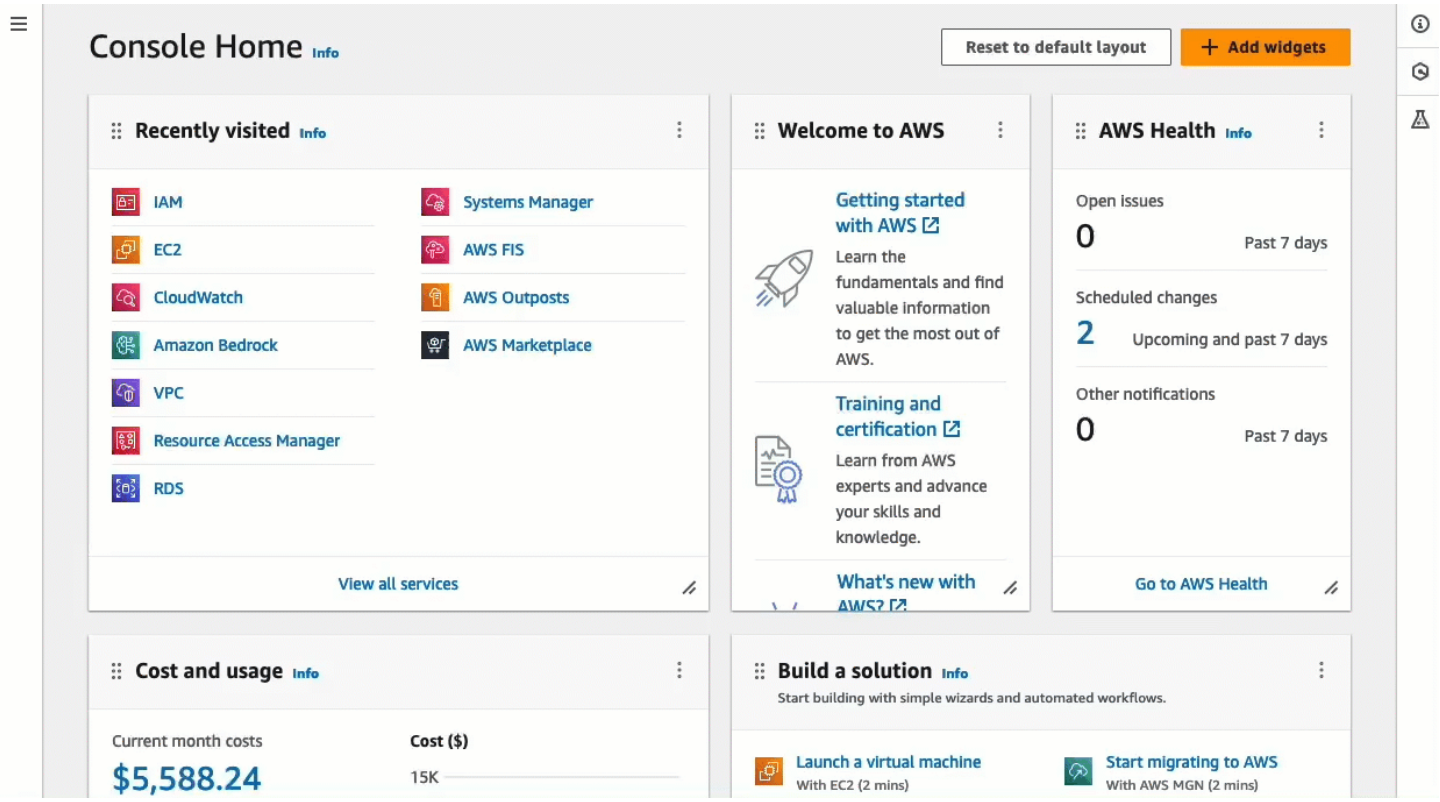
d. 在IAM實體下，選取身分 (使用者、使用者群組或角色) 旁邊的核取方塊。您可以使用搜尋方塊來篩選實體清單。

e. 選擇連接政策。

檢視動畫：建立IAM原則



檢視動畫：附加IAM原則



工作 2：允許從EC2執行個體 Connect 服務傳入您的執行個體的傳入流量

當您在 Amazon EC2 主控台中使用EC2執行個體 Connect 連線到執行個體時，必須允許 Connect 執行個體的流量是來自執行個EC2體連接服務的流量。這與從本機電腦連線到執行個體不同；在這種情況下，您必須允許從本機電腦傳輸到執行個體的流量。若要允許來自EC2執行個體 Connect 服務的流量，您必須建立一個安全群組，以允許來自EC2執行個體 Connect 服務 IP 位址範圍的輸入SSH流量。

AWS 服務的 IP 位址範圍可在 https://ip-ranges.amazonaws.com/ip 範圍 .json 取得。EC2執行個體 Connect IP 位址範圍由識別`"service": "EC2_INSTANCE_CONNECT"`。

任務目標

首先，您會找到執行個體所`EC2_INSTANCE_CONNECT` AWS 區域 在的 IP 位址範圍。然後，您將建立一個安全性群組，允許來自該 IP 位址範圍的SSH通訊埠 22 上的輸入流量。

建立允許從執行個EC2體 Connect 服務傳入流量到執行個體的安全性群組

1. 首先取得EC2執行個體 Connect 服務的 IP 位址範圍

   a. 在 https://ip-ranges.amazonaws.com/ AWSip 範圍 .json 處開啟 IP 位址範圍JSON檔案。

b. 選擇「原始資料」。

c. 尋找執行個EC2_INSTANCE_CONNECT體所 AWS 區域 在的 IP 位址範圍。您可以使用瀏覽器搜尋欄位來搜尋服務EC2_INSTANCE_CONNECT，並繼續搜尋，直到找到執行個體所在的區域為止。

例如，如果您的執行個體位於美國東部 (維吉尼亞北部us-east-1) () 區域，則該區域的 IP EC2_INSTANCE_CONNECT 位址範圍為18.206.107.24/29。

> **Note**
> 每個 IP 位址範圍都不同 AWS 區域。

d. 複製出現在旁邊的 IP 位址範圍ip_prefix。您將在稍後的程序中使用此 IP 位址範圍。

如需有關下載 AWS IP 位址範圍JSON檔案和依服務篩選的詳細資訊，請參閱 Amazon VPC 使用者指南中的 AWS IP 位址範圍。

2. 然後使用輸入規則建立安全群組，以允許來自複製 IP 位址範圍的流量

a. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。

b. 在導覽窗格中，選擇 Security Groups (安全群組)。

c. 選擇 Create Security Group (建立安全群組)。

d. 在 Basic details (基本詳細資料) 下，執行下列動作：

i. 在「安全性群組名稱」中，輸入有意義的安全性群組名稱。

ii. 在說明中，為您的安全性群組輸入有意義的說明。

e. 在「輸入規則」下，執行下列動作：

i. 選擇新增規則。

ii. 對於「類型」，選擇SSH。

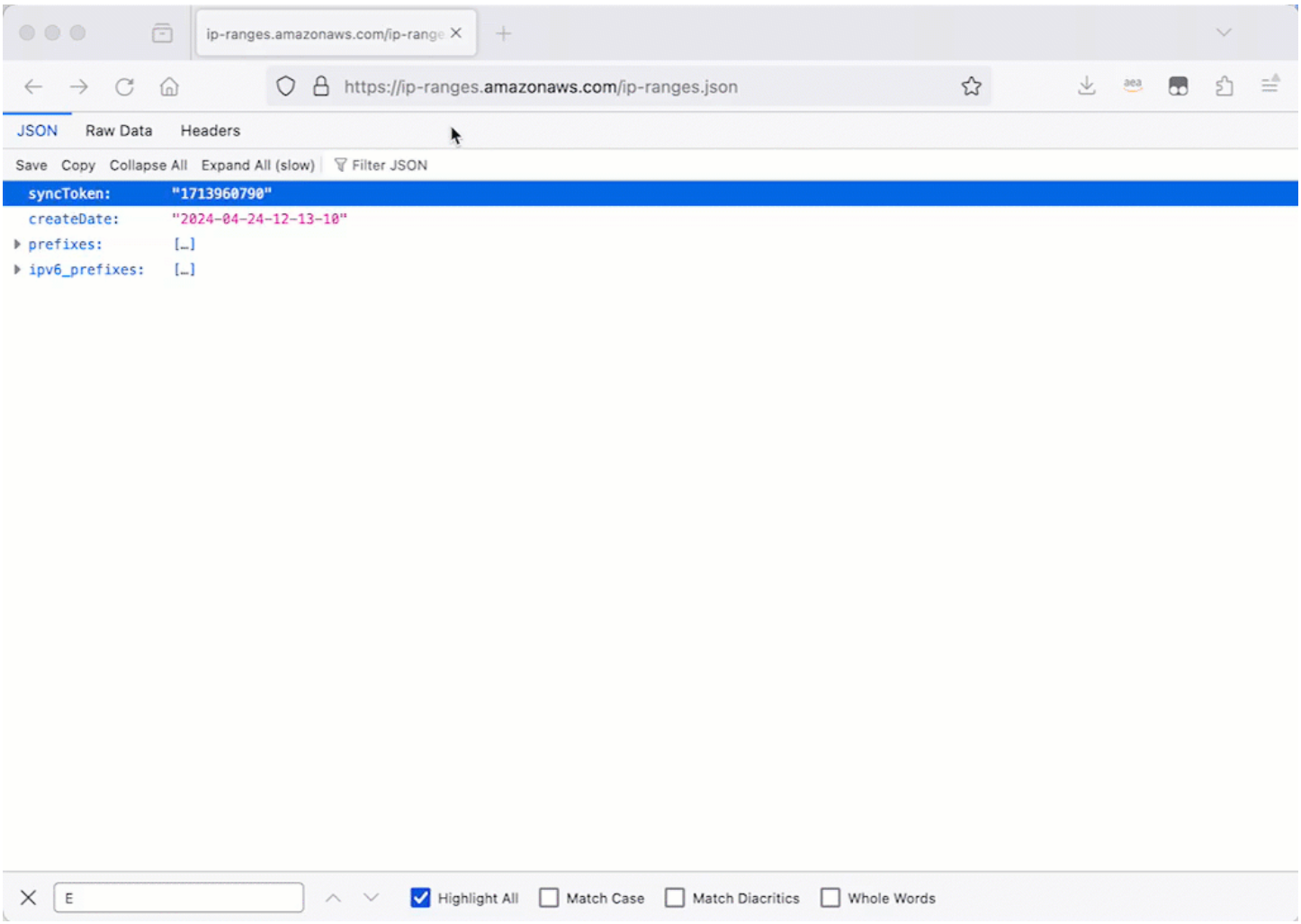
iii. 對於來源，請保留自訂。

iv. 在 [來源] 旁邊的欄位中，貼上您先前在此程序中複製之EC2執行個體 Connect 服務的 IP 位址範圍。

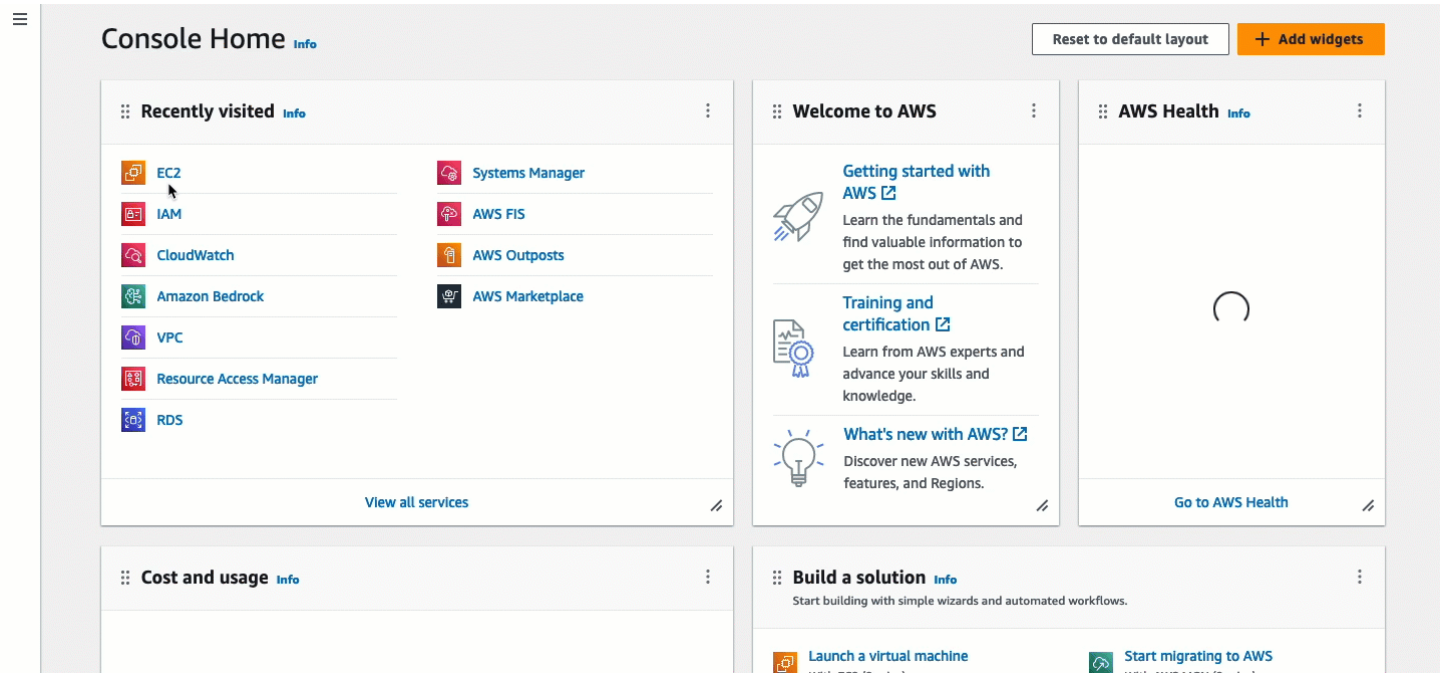
例如，如果您的執行個體位於美國東部 (維吉尼亞北部) (us-east-1) 區域，請將下列 IP 位址範圍貼到欄位中：18.206.107.24/29

f. 選擇建立安全群組。

檢視動畫：取得特定區域EC2執行個體 Connect 的 IP 位址範圍



檢視動畫：設定安全性群組



工作 3：啟動執行個體

啟動執行個體時，您必須指定包AMI含啟動執行個體所需資訊的執行個體。您可以選擇啟動執行個體，不論是否已預先安裝EC2執行個體 Connect。在這項工作中，我們會指定EC2執行AMI個體 Connect 預先安裝的。

如果您在未預先安裝執行EC2個體 Connect 的情況下啟動執行個體，而且想要使用EC2執行個體連線連線至執行個體，則需要執行其他設定步驟。這些步驟不在本教學課程的範圍內。

任務目標

您將使用 Amazon Linux 2023 啟動執行個體AMI，該執行個體已預先安裝在EC2執行個體 Connect 中。您還將指定先前建立的安全群組，以便在 Amazon EC2 主控台中使用EC2執行個體 Connect 連接到執行個體。由於您將使用 EC2 Instance Connect 連線至執行個體 (將公開金鑰推送至執行個體的中繼資料)，因此在啟動執行個體時不需要指定SSH金鑰。

啟動可在 Amazon EC2 主控台中使用EC2執行個體 Connect 進行連線的執行個體

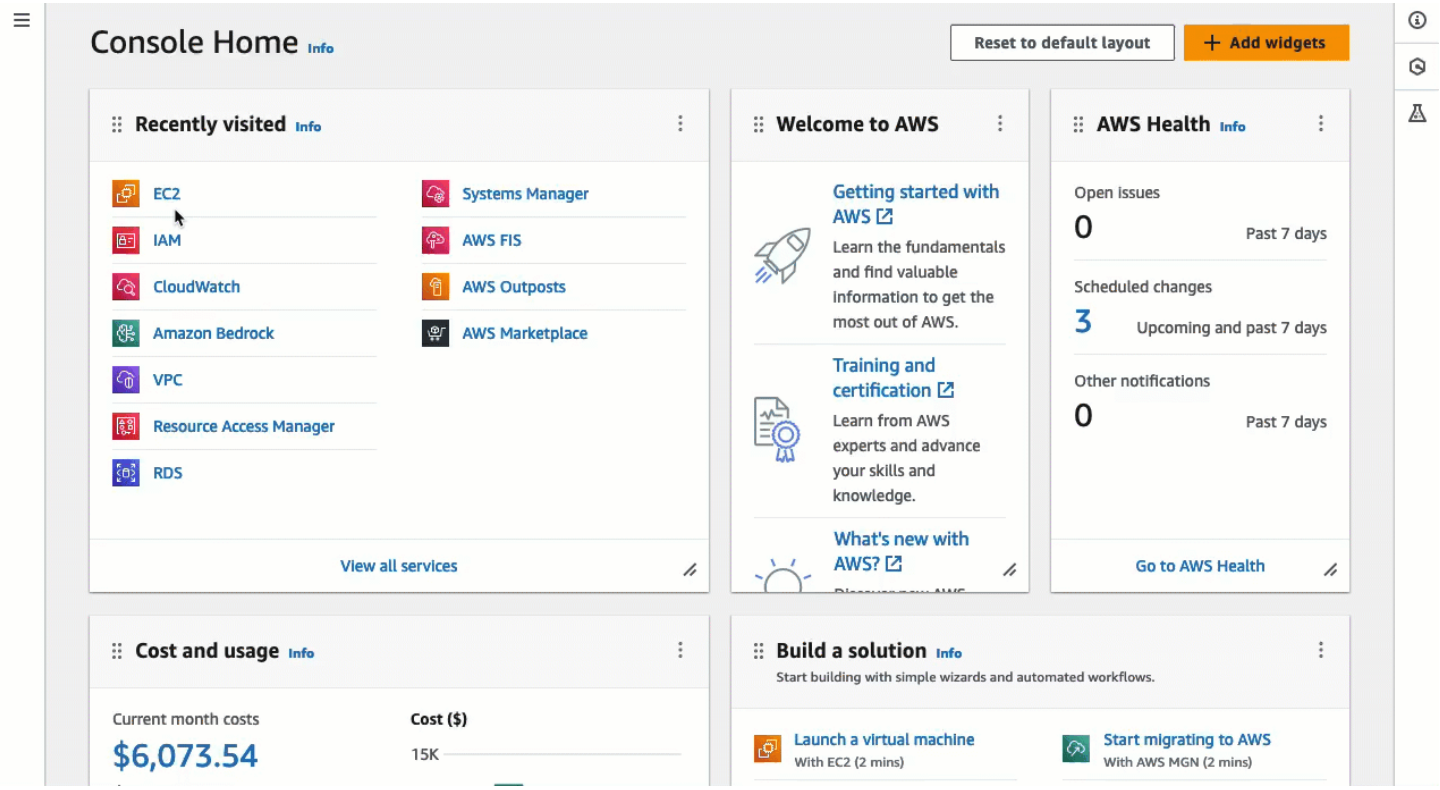
1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在畫面頂端的導覽列中，會顯示目前的「區域」(AWS Region) (例如，愛爾蘭)。選取要在其中啟動執行個體的區域。這個選擇很重要，因為您建立了允許特定區域流量的安全性群組，因此您必須選取要在其中啟動執行個體的相同區域。

3. 在 Amazon 主EC2控台儀表板中，選擇啟動執行個體。
4. (選用) 在 Name and tags (名稱和標籤) 下，針對 Name (名稱)，輸入執行個體的描述性名稱。
5. 在應用程式和作業系統映像 (Amazon 機器映像) 下，選擇快速入門。依預設會選取 Amazon Linux。在 Amazon 機器映像（AMI）下，默認情況下選擇了 Amazon Linux 2023 AMI。保留此工作的預設選項。
6. 在「例證類型」下，對於「例證類型」，保留預設選取項，或選擇不同的例證類型。
7. 在 key pair (登入) 下，對於金鑰配對名稱，選擇不使用金鑰配對繼續 (不建議)。當您使用EC2執行個體連線 Connect 線至執行個體時，EC2Instance Connect 會將 key pair 推送至執行個體的中繼資料，而這個 key pair 用於連線。
8. 在Network settings (網路設定) 中，執行下列操作：
   a. 對於自動指派公用 IP，請離開 [啟用]。

      > **Note**
      > 若要在 Amazon EC2 主控台中使用EC2執行個體 Connect 連線到執行個體，執行個體必須具有公開IPv4地址。

   b. 對於「防火牆（安全群組）」，選擇 「選取現有安全群組」。
   c. 在 [一般安全性群組] 底下，選擇您先前建立的安全性群組。
9. 在 Summary (摘要) 面板中，選擇 Launch instance (啟動執行個體)。

檢視動畫：啟動您的執行個體



工作 4：Connect 至執行個體

當您使用執行個體 Connect Connect 至EC2執行個體時，EC2執行個體連線會將SSH公開金鑰API推送至執行個體中繼資料，並保留 60 秒。SSH常駐程式會使用`AuthorizedKeysCommand`和`AuthorizedKeysCommandUser`查詢執行個體中繼資料中的公開金鑰以進行驗證，並將您連線至執行個體。

任務目標

在此任務中，您將使用 Amazon EC2 主控台中的執行個EC2體 Connect 連接到執行個體。如果您已完成先決條件工作 1、2 和 3，連線應該會成功。

連線至執行個體的步驟

請使用下列步驟連線至您的執行個體。若要檢視步驟的動畫，請參閱檢視動畫：Connect 至您的執行個體。

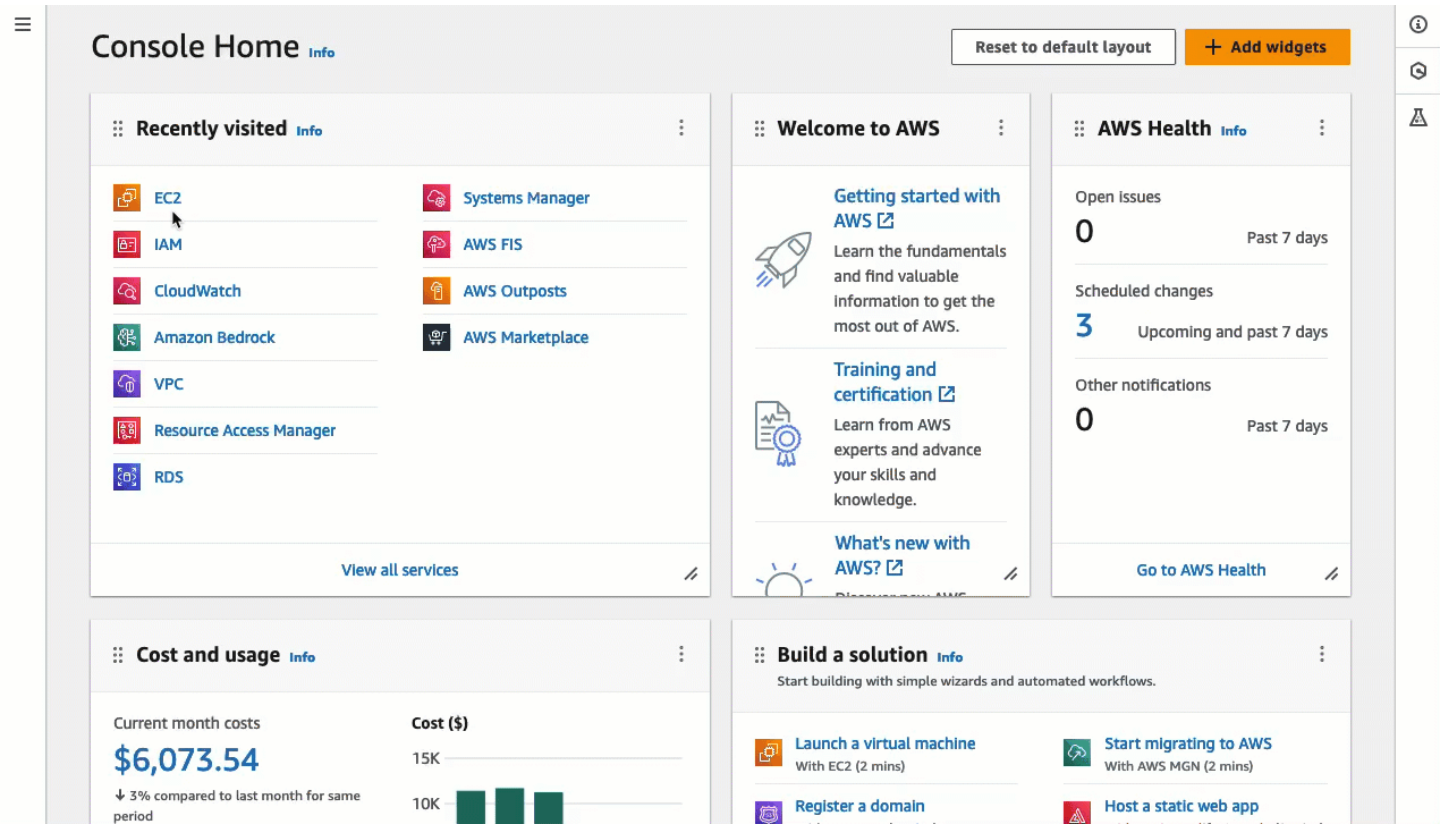
使用 Amazon EC2 主控台中的EC2執行個體 Connect 來連接執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。

2. 在畫面頂端的導覽列中，會顯示目前的「區域」(AWS Region) (例如，愛爾蘭)。選取執行個體所在的區域。
3. 在導覽窗格中，選擇 Instances (執行個體)。
4. 選取執行個體並選擇 [Connect]。
5. 選擇「EC2執行處理 Connect」標籤。
6. 針對「連線類型」，請選擇「使用EC2執行個體 Connect 線
7. 選擇連線。

   終端機視窗會在瀏覽器中開啟，且您已連線至執行個體。

檢視動畫：Connect 至您的執行個體



EC2執行個體 Connect 的必要

以下是安裝和使用「EC2執行個體 Connect」的先決條件：

- 安裝EC2實例 Connect
- 確保網路連線
- 允許入站SSH流量

- 授予許可
- 在本機電SSH腦上安裝用戶端
- 滿足用戶名要求

安裝EC2實例 Connect

若要使用「EC2執行個體 Connect 線」Connect 至執行個體，執行個體必須已安裝EC2執行個體連線。您可以使用執行個體 Connect 預先安裝AMI的執行個體來啟動EC2執行個體，也可以在支援的情況下啟動執行個體 Connect 安裝EC2執行個體連線AMIs。如需詳細資訊，請參閱在執行個EC2體安裝EC2執行個體 Connect。

確保網路連線

執行個體可設定為允許使用者透過網際網路或執行個體的私有 IP 地址連線至您的執行個體。視使EC2用者使用執行個體 Connect 至執行個體的方式而定，您必須設定下列網路存取權：

- 如果使用者將透過網際網路連線到執行個體，則該執行個體必須具有公有 IP 地址並位於公有子網路中。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的啟用網際網路存取。
- 如果您的使用者將透過執行個體的私有 IP 位址連線至您的執行個體，則您必須建立與您的私人網路連線VPC，例如使用 AWS Direct Connect AWS Site-to-Site VPN、或VPC對等連線，以便使用者可以連線到執行個體的私有 IP 位址。

如果您的執行個體沒有公開IPv4位址，而且您不想如上所述設定網路存取權，則可以將執行個EC2體 Connect 端點視為執行個EC2體 Connect 的替代方案。透過EC2執行個體 Connect 端點，您可以使用執行個體連線至執行個體，SSH或RDP即使執行個體沒有公開IPv4位址也是如此。如需詳細資訊，請參閱使用 Amazon EC2 主控台 Connect 到您的 Linux 執行個體。

允許入站SSH流量

確保與執行個體相關聯的安全性群組允許來自 IP 位址或網路的連接埠 22 上的輸入SSH流量。預設的預設安全性群組預設VPC不允許傳入SSH流量。根據預設，啟動執行個體精靈建立的安全性群組允許傳入SSH流量。如需詳細資訊，請參閱從您電腦連線到執行個體的規則。

EC2執行個體 Connect 會使用特定的 IP 地址範圍，以瀏覽器為基礎的SSH連線至執行個體 (當使用者使用 Amazon EC2 主控台連線至執行個體時)。如果您的使用者將使用 Amazon EC2 主控台連線到執行個體，請確保與您的執行個體關聯的安全群組允許來自 IP 地址範圍的輸入SSH流量`EC2_INSTANCE_CONNECT`。若要識別位址範圍，請下載提供的JSON檔案， AWS 並篩選EC2執行

個體 Connect 的子集，`EC2_INSTANCE_CONNECT`做為服務值。這些 IP 位址範圍之間有所不同 AWS 區域。如需有關下載JSON檔案和依服務篩選的詳細資訊，請參閱 Amazon VPC 使用者指南中的 AWS IP 位址範圍。

授予許可

您必須將必要的權限授與每位將IAM使用EC2執行個體 Connect 線連線的使用者。如需詳細資訊，請參閱授與EC2執行個體 Connect 的IAM權限。

在本機電SSH腦上安裝用戶端

如果您的使用者將使用連線SSH，他們必須確保其本機電腦具有用SSH戶端。

根據預設，使用者的本機電腦很可能已安裝用SSH戶端。他們可以通過在命令行輸入ssh來檢查SSH客戶端。如果他們的本地計算機無法識別該命令，則可以安裝SSH客戶端。如需在 Linux 或 macOS X 上安裝SSH用戶端的詳細資訊，請參閱 http://www.openssh.com。如需有關在 Windows 10 上安裝SSH用戶端的詳細資訊，請參閱SSH在視窗中開啟。

如果您的使用者只使用 Amazon EC2 主控台連線至執行個體，則無需在本機電腦上安裝用戶SSH端。

滿足用戶名要求

使用「EC2執行個體 Connect」連線至執行個體時，使用者名稱必須符合下列需求：

- 第一個字元：必須是字母 (`A-Z`,`a-z`)、數字 (`0-9`) 或底線 (`_`)
- 後續字元：可以是字母 (`A-Z`、`a-z`)、數字 (`0-9`) 或 `@` `.` `_` `-` 這些字元
- 長度下限：1 個字元
- 長度上限：31 個字元

授與EC2執行個體 Connect 的IAM權限

若要使用執行個體連線連線至EC2執行個體，您必須建立IAM原則，授與使用者下列動作和條件的權限：

- `ec2-instance-connect:SendSSHPublicKey` 動作 – 准許將公有金鑰推送至執行個體。
- `ec2:osuser` 條件 – 指定可將公有金鑰推送至執行個體的 OS 使用者名稱。使用您用來啟動執行AMI個體的預設使用者名稱。AL2023 和 Amazon Linux 2 的默認用戶名是`ec2-user`，對於 Ubuntu 而言，它是`ubuntu`。

- `ec2:DescribeInstances`動作 — 使用EC2主控台時需要，因為包裝函式會呼叫此動作。使用者可能已有從另一個政策呼叫此動作的許可。

請考慮限制特定EC2執行個體的存取權。否則，具有`ec2-instance-connect:SendSSHPublicKey`動作權限IAM的所有主參與者都可以連線至所有EC2實例。您可以指定資源ARNs或使用資源標籤作為條件鍵來限制存取。

如需詳細資訊，請參閱 Amazon EC2 執行個體 Connect 的動作、資源和條件金鑰。

如需有關建立IAM策略的資訊，請參閱IAM使用指南中的建立IAM策略。

允許使用者連線至特定執行個體

下列IAM政策授與連線至特定執行個體 (由其資源所識別) 的權限ARNs。

在下列範例IAM原則中，會指定下列動作和條件：

- 此`ec2-instance-connect:SendSSHPublicKey`動作會授與使用者連線至資源指定的兩個執行個體的權限ARNs。若要授ARNs與使用者連線至所有EC2執行個體的權限，請使用`*`萬用字元取代資源。
- 此`ec2:osuser`條件僅授予連線至執行個體的權限 *`ami-username`* 在連線時指定。
- 指定 `ec2:DescribeInstances` 動作，准許使用主控台的使用者連線至執行個體。如果您的使用者只會使用用SSH戶端連線至您的執行個體，您可以省略`ec2:DescribeInstances`。請注意，這些`ec2:Describe*`API動作不支援資源層級權限。因此，在 `Resource` 元素中必須包含 `*` 萬用字元。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ec2-instance-connect:SendSSHPublicKey",
            "Resource": [
                "arn:aws:ec2:region:account-id:instance/i-1234567890abcdef0",
                "arn:aws:ec2:region:account-id:instance/i-0598c7d356eba48d7"
            ],
            "Condition": {
                "StringEquals": {
                    "ec2:osuser": "ami-username"
                }
```

```
            }
        },
        {
            "Effect": "Allow",
            "Action": "ec2:DescribeInstances",
            "Resource": "*"
        }
    ]
}
```

允許使用者連線至具有特定標籤的執行個體

以屬性為基礎的存取控制 (ABAC) 是一種授權策略，根據可附加至使用者和 AWS 資源的標籤來定義權限。您可使用資源標籤來控制對執行個體的存取。如需有關使用標籤來控制 AWS 資源存取權的詳細資訊，請參閱《使IAM用指南》中的〈控制 AWS 資源存取〉。

在下列範例IAM原則中，`ec2-instance-connect:SendSSHPublicKey`動作會授與使用者連線至任何執行個體 (以資源中的*萬用字元表示ARN) 的權限，但該執行個體具有 key= `tag-key` 和 value `tag-value` = 的資源標籤。

指定 `ec2:DescribeInstances` 動作，准許使用主控台的使用者連線至執行個體。如果您的使用者只會使用用SSH戶端連線至您的執行個體，您可以省略`ec2:DescribeInstances`。請注意，這些`ec2:Describe*`API動作不支援資源層級權限。因此，在 `Resource` 元素中必須包含 `*` 萬用字元。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ec2-instance-connect:SendSSHPublicKey",
            "Resource": "arn:aws:ec2:region:account-id:instance/*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/tag-key": "tag-value"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": "ec2:DescribeInstances",
            "Resource": "*"
        }
    ]
```

```
}
```

在執行個EC2體安裝EC2執行個體 Connect

若要使用「執行個體 Connect」Connect 至 Linux EC2 執行個體，執行個體必須安裝EC2執行個體連線。安裝EC2執行個體 Connect 會設定執行個體上的SSH常駐程式。

如需EC2執行個體 Connect 套件的詳細資訊，請參閱網站上的 aws/aws-ec2-instance-connect-config 。 GitHub

> **Note**
>
> 如果您已設定SSH驗證的`AuthorizedKeysCommand`和`AuthorizedKeysCommandUser`設定，則EC2執行個體 Connect 安裝將不會更新這些設定。因此，您無法使用EC2執行個體 Connect。

安裝先決條

安裝EC2執行個體 Connect 之前，請確定您符合下列先決條件。

- 確認執行個體使用下列其中一項：
  - Amazon Linux 2 在 2.0.20190618 之前的版本
  - AL2023 最小AMI或 Amazon ECS 優化 AMI
  - CentOS Stream 8 和 9
  - macOS 索諾瑪之前 14.2.1, 文圖拉之前 13.6.3, 和蒙特雷之前 12.7.2
  - 紅帽企業版 (RHEL) 8 和 9
  - Ubuntu 16.04 和 18.04

  > **Tip**
  >
  > 如果您使用更新版本的 Amazon Linux，macOS 索諾瑪，文圖拉或蒙特雷或 Ubuntu 啟動實例，則它已預先安裝了EC2實例 Connect，因此您無需自行安裝。

- 確認EC2執行個體 Connect 的一般先決條件。

  如需詳細資訊，請參閱EC2執行個體 Connect 的必要。
- 確認使用本機電腦上的用SSH戶端連線至執行個體的先決條件。

如需詳細資訊，請參閱使用 Connect 到您的 Linux 執行個體 SSH。

- 取得執行個體的 ID。

  您可以使用 Amazon EC2 主控台 (從「執行個體 ID」欄) 取得執行個體的 ID。如果您願意，您可以使用描述實例 (AWS CLI) 或 Get-EC2Instance(AWS Tools for Windows PowerShell) 命令。

手動安裝EC2實例 Connect

> **Note**
>
> 如果您使用下列其中一種方式啟動執行個EC2體AMIs，則會預先安裝執行個體 Connect，您可以略過此程序：
>
> - AL2標準 AMI
> - Amazon Linux 2 2.0.20190618 或更新版本
> - macOS 索諾瑪 14.2.1 或更高版本
> - macOS 文圖拉 13.6.3 或更高版本
> - macOS 蒙特雷 12.7.2 或更高版本
> - Ubuntu 20.04 或更新版本

根據EC2執行個體的作業系統，使用下列其中一個程序來安裝執行個體 Connect。

Amazon Linux 2

在使用 Amazon Linux 2 啟動的執行個體上安EC2裝執行個體 Connect

1. 使用 Connect 至您的執行個體SSH。

   將以下命令中的範例值取代為您自己的值。使用啟動執行個體時指派給執行個體的 SSH key pair，以及您用來啟動執行個體的預設使用者名稱。AMI對於 Amazon Linux 2，默認的用戶名是`ec2-user`.

   ```
   $ ssh -i my_ec2_private_key.pem ec2-user@ec2-a-b-c-d.us-
   west-2.compute.amazonaws.com
   ```

   如需連線至執行個體的詳細資訊，請參閱使用用SSH戶端 Connect 至您的 Linux 執行個體。

2. 在EC2執行個體上安裝執行個體 Connect 套件。

```
[ec2-user ~]$ sudo yum install ec2-instance-connect
```

您應會在 /opt/aws/bin/ 資料夾中看到三個新指令碼：

```
eic_curl_authorized_keys
eic_parse_authorized_keys
eic_run_authorized_keys
```

3. (選擇性) 確認EC2執行個體 Connect 已成功安裝在執行個體上。

```
[ec2-user ~]$ sudo less /etc/ssh/sshd_config
```

EC2如果AuthorizedKeysCommand和AuthorizedKeysCommandUser行包含下列值，則已成功安裝「執行處理 Connect 線」：

```
AuthorizedKeysCommand /opt/aws/bin/eic_run_authorized_keys %u %f
AuthorizedKeysCommandUser ec2-instance-connect
```

- AuthorizedKeysCommand 會設定 eic_run_authorized_keys 指令碼來從執行個體中繼資料查閱金鑰
- AuthorizedKeysCommandUser 會將系統使用者設為 ec2-instance-connect

> **Note**
>
> 如果您先前已設定AuthorizedKeysCommand和AuthorizedKeysCommandUser，則EC2執行個體 Connect 安裝將不會變更值，且您將無法使用EC2執行個體 Connect。

CentOS

在以 CentOS 啟動的EC2執行個體上安裝執行個體 Connect

1. 使用 Connect 至您的執行個體SSH。

將以下命令中的範例值取代為您自己的值。使用啟動執行個體時指派給執行個體的 SSH key pair，以及您用來啟動執行個體的預設使用者名稱。AMI對於 CentOS 而言，預設的使用者名稱為`centos`或`ec2-user`。

```
$ ssh -i my_ec2_private_key.pem centos@ec2-a-b-c-d.us-
west-2.compute.amazonaws.com
```

如需連線至執行個體的詳細資訊，請參閱使用用SSH戶端 Connect 至您的 Linux 執行個體。

2. 如果您使用HTTP或HTTPS代理伺服器，您必須在目前的 shell 工作階段中設定`http_proxy`或`https_proxy`環境變數。

   如果您不使用代理伺服器，則可以略過此步驟。

   - 對於HTTP代理服務器，請運行以下命令：

     ```
     $ export http_proxy=http://hostname:port
     $ export https_proxy=http://hostname:port
     ```

   - 對於HTTPS代理服務器，請運行以下命令：

     ```
     $ export http_proxy=https://hostname:port
     $ export https_proxy=https://hostname:port
     ```

3. 執行下列命令，在EC2執行個體上安裝執行個體 Connect 套件。

   CentOS 的EC2實例 Connect 配置檔案是以 Red Hat P RPM ackage 件管理員 (RPM) 套件提供，而 CentOS 8 和 CentOS 9 以及在 Intel/(x86_64) 或 AMD () 上執行的執行個體類型則有不同的套件。ARM AArch64

   針對您的作業系統和CPU架構使用命令區塊。

   - CentOS 8

     英特尔/AMD(x86_64)

     ```
     [ec2-user ~]$ mkdir /tmp/ec2-instance-connect
     [ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-
     west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect.rhel8.rpm -o /
     tmp/ec2-instance-connect/ec2-instance-connect.rpm
     ```

```
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-
west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect-
selinux.noarch.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect-
selinux.rpm
[ec2-user ~]$ sudo yum install -y /tmp/ec2-instance-connect/ec2-instance-
connect.rpm /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
```

ARM (AArch64)

```
[ec2-user ~]$ mkdir /tmp/ec2-instance-connect
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-
west-2.amazonaws.com/latest/linux_arm64/ec2-instance-connect.rhel8.rpm -o /
tmp/ec2-instance-connect/ec2-instance-connect.rpm
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-
west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect-
selinux.noarch.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect-
selinux.rpm
[ec2-user ~]$ sudo yum install -y /tmp/ec2-instance-connect/ec2-instance-
connect.rpm /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
```

- CentOS 9

英特尔/AMD(x86_64)

```
[ec2-user ~]$ mkdir /tmp/ec2-instance-connect
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-
west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect.rpm -o /tmp/ec2-
instance-connect/ec2-instance-connect.rpm
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-
west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect-
selinux.noarch.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect-
selinux.rpm
[ec2-user ~]$ sudo yum install -y /tmp/ec2-instance-connect/ec2-instance-
connect.rpm /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
```

ARM (AArch64)

```
[ec2-user ~]$ mkdir /tmp/ec2-instance-connect
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-
west-2.amazonaws.com/latest/linux_arm64/ec2-instance-connect.rpm -o /tmp/ec2-
instance-connect/ec2-instance-connect.rpm
```

```
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-
west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect-
selinux.noarch.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect-
selinux.rpm
[ec2-user ~]$ sudo yum install -y /tmp/ec2-instance-connect/ec2-instance-
connect.rpm /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
```

您應會在 `/opt/aws/bin/` 資料夾中看到以下新指令碼：

```
eic_run_authorized_keys
```

4. (選擇性) 確認EC2執行個體 Connect 已成功安裝在執行個體上。

   - CentOS 8：

     ```
     [ec2-user ~]$ sudo less /lib/systemd/system/ssh.service.d/ec2-instance-
     connect.conf
     ```

   - CentOS 9：

     ```
     [ec2-user ~]$ sudo less /etc/ssh/sshd_config.d/60-ec2-instance-connect.conf
     ```

   EC2如果`AuthorizedKeysCommand`和`AuthorizedKeysCommandUser`行包含下列值，則已成功安裝「執行處理 Connect 線」：

   ```
   AuthorizedKeysCommand /opt/aws/bin/eic_run_authorized_keys %u %f
   AuthorizedKeysCommandUser ec2-instance-connect
   ```

   - `AuthorizedKeysCommand` 會設定 `eic_run_authorized_keys` 指令碼來從執行個體中繼資料查閱金鑰
   - `AuthorizedKeysCommandUser` 會將系統使用者設為 `ec2-instance-connect`

> **Note**
>
> 如果您先前已設定`AuthorizedKeysCommand`和`AuthorizedKeysCommandUser`，則EC2執行個體 Connect 安裝將不會變更值，且您將無法使用EC2執行個體 Connect。

macOS

在以 macOS 啟動的EC2執行個體上安裝執行個體 Connect

1. 使用 Connect 至您的執行個體SSH。

   將以下命令中的範例值取代為您自己的值。使用啟動執行個體時指派給執行個體的 SSH key pair，以及您用來啟動執行個體的預設使用者名稱。AMI對於 macOS 執行個體，預設使用者名稱為`ec2-user`。

   ```
   $ ssh -i my_ec2_private_key.pem ec2-user@ec2-a-b-c-d.us-
   west-2.compute.amazonaws.com
   ```

   如需連線至執行個體的詳細資訊，請參閱使用用SSH戶端 Connect 至您的 Linux 執行個體。

2. 使用以下命令更新 Homebrew。更新將列出 Homebrew 已知的軟體。EC2執行個體 Connect 套件是透過 macOS 執行個體上的自製軟體提供。 如需詳細資訊，請參閱更新 Mac 實例上的操作系統和軟件。

   ```
   [ec2-user ~]$ brew update
   ```

3. 在EC2執行個體上安裝執行個體 Connect 套件。這會安裝軟體，並設定 sshd 以使用該軟體。

   ```
   [ec2-user ~]$ brew install ec2-instance-connect
   ```

   您應會在 `/opt/aws/bin/` 資料夾中看到以下新指令碼：

   ```
   eic_run_authorized_keys
   ```

4. (選擇性) 確認EC2執行個體 Connect 已成功安裝在執行個體上。

```
[ec2-user ~]$ sudo less /etc/ssh/sshd_config.d/60-ec2-instance-connect.conf
```

EC2如果AuthorizedKeysCommand和AuthorizedKeysCommandUser行包含下列值，則已成功安裝「執行處理 Connect 線」：

```
AuthorizedKeysCommand /opt/aws/bin/eic_run_authorized_keys %u %f
AuthorizedKeysCommandUser ec2-instance-connect
```

- AuthorizedKeysCommand 會設定 eic_run_authorized_keys 指令碼來從執行個體中繼資料查閱金鑰
- AuthorizedKeysCommandUser 會將系統使用者設為 ec2-instance-connect

> **Note**
>
> 如果您先前已設定AuthorizedKeysCommand和AuthorizedKeysCommandUser，則EC2執行個體 Connect 安裝將不會變更值，且您將無法使用EC2執行個體 Connect。

RHEL

在使用 RHEL 啟動的執行個體上安裝EC2執行個體 Connect (RHEL)

1. 使用 Connect 至您的執行個體SSH。

   將以下命令中的範例值取代為您自己的值。使用啟動執行個體時指派給執行個體的 SSH key pair，以及您用來啟動執行個體的預設使用者名稱。AMI對於RHEL，預設使用者名稱為ec2-user或root。

   ```
   $ ssh -i my_ec2_private_key.pem ec2-user@ec2-a-b-c-d.us-
   west-2.compute.amazonaws.com
   ```

   如需連線至執行個體的詳細資訊，請參閱使用用SSH戶端 Connect 至您的 Linux 執行個體。

2. 如果您使用HTTP或HTTPS代理伺服器，您必須在目前的 shell 工作階段中設定http_proxy或https_proxy環境變數。

如果您不使用代理伺服器，則可以略過此步驟。

- 對於HTTP代理服務器，請運行以下命令：

```
$ export http_proxy=http://hostname:port
$ export https_proxy=http://hostname:port
```

- 對於HTTPS代理服務器，請運行以下命令：

```
$ export http_proxy=https://hostname:port
$ export https_proxy=https://hostname:port
```

3. 執行下列命令，在EC2執行個體上安裝執行個體 Connect 套件。

的「執行個EC2體 Connect」組態檔案RHEL會以 Red Hat Package 件管理員 (RPM) 套件提供，其中包含 RHEL 8 與 RHEL 9 的RPM套件，以及在 Intel/ AMD (x86_64) 或 () 上執行的執行個體類型的不同套件。ARM AArch64

針對您的作業系統和CPU架構使用命令區塊。

- RHEL8

  英特尔/AMD(x86_64)

```
[ec2-user ~]$ mkdir /tmp/ec2-instance-connect
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-
west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect.rhel8.rpm -o /
tmp/ec2-instance-connect/ec2-instance-connect.rpm
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-
west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect-
selinux.noarch.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect-
selinux.rpm
[ec2-user ~]$ sudo yum install -y /tmp/ec2-instance-connect/ec2-instance-
connect.rpm /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
```

  ARM (AArch64)

```
[ec2-user ~]$ mkdir /tmp/ec2-instance-connect
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-
west-2.amazonaws.com/latest/linux_arm64/ec2-instance-connect.rhel8.rpm -o /
tmp/ec2-instance-connect/ec2-instance-connect.rpm
```

```
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-
west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect-
selinux.noarch.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect-
selinux.rpm
[ec2-user ~]$ sudo yum install -y /tmp/ec2-instance-connect/ec2-instance-
connect.rpm /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
```

- RHEL9

  英特尔/AMD(x86_64)

```
[ec2-user ~]$ mkdir /tmp/ec2-instance-connect
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-
west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect.rpm -o /tmp/ec2-
instance-connect/ec2-instance-connect.rpm
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-
west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect-
selinux.noarch.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect-
selinux.rpm
[ec2-user ~]$ sudo yum install -y /tmp/ec2-instance-connect/ec2-instance-
connect.rpm /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
```

  ARM (AArch64)

```
[ec2-user ~]$ mkdir /tmp/ec2-instance-connect
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-
west-2.amazonaws.com/latest/linux_arm64/ec2-instance-connect.rpm -o /tmp/ec2-
instance-connect/ec2-instance-connect.rpm
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-
west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect-
selinux.noarch.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect-
selinux.rpm
[ec2-user ~]$ sudo yum install -y /tmp/ec2-instance-connect/ec2-instance-
connect.rpm /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
```

您應會在 `/opt/aws/bin/` 資料夾中看到以下新指令碼：

```
eic_run_authorized_keys
```

4. (選擇性) 確認EC2執行個體 Connect 已成功安裝在執行個體上。

- 對於 RHEL 8：

```
[ec2-user ~]$ sudo less /lib/systemd/system/ssh.service.d/ec2-instance-connect.conf
```

- 對於 RHEL 9：

```
[ec2-user ~]$ sudo less /etc/ssh/sshd_config.d/60-ec2-instance-connect.conf
```

EC2如果`AuthorizedKeysCommand`和`AuthorizedKeysCommandUser`行包含下列值，則已成功安裝「執行處理 Connect 線」：

```
AuthorizedKeysCommand /opt/aws/bin/eic_run_authorized_keys %u %f
AuthorizedKeysCommandUser ec2-instance-connect
```

- `AuthorizedKeysCommand` 會設定 `eic_run_authorized_keys` 指令碼來從執行個體中繼資料查閱金鑰
- `AuthorizedKeysCommandUser` 會將系統使用者設為 `ec2-instance-connect`

> **Note**
>
> 如果您先前已設定`AuthorizedKeysCommand`和`AuthorizedKeysCommandUser`，則EC2執行個體 Connect 安裝將不會變更值，且您將無法使用EC2執行個體 Connect。

Ubuntu

在使用 Ubuntu 16.04 或更新版本啟動的執行個體上安EC2裝執行個體 Connect

1. 使用 Connect 至您的執行個體SSH。

   將以下命令中的範例值取代為您自己的值。使用啟動執行個體時指派給執行個體的 SSH key pair，並使用您用來啟動執行個體的預設使用者名稱。AMI對於 UbuntuAMI，使用者名稱為`ubuntu`.

```
$ ssh -i my_ec2_private_key.pem ubuntu@ec2-a-b-c-d.us-
west-2.compute.amazonaws.com
```

如需連線至執行個體的詳細資訊，請參閱使用用SSH戶端 Connect 至您的 Linux 執行個體。

2. (選擇性) 確定您的執行個體擁有最新版本的 Ubuntu AMI。

   執行下列命令以更新您的執行個體上的所有套件。

```
ubuntu:~$ sudo apt-get update
```

```
ubuntu:~$ sudo apt-get upgrade
```

3. 在EC2執行個體上安裝執行個體 Connect 套件。

```
ubuntu:~$ sudo apt-get install ec2-instance-connect
```

   您應會在 `/usr/share/ec2-instance-connect/` 資料夾中看到三個新指令碼：

```
eic_curl_authorized_keys
eic_parse_authorized_keys
eic_run_authorized_keys
```

4. (選用) 確認 Instance Connect 是否已順利安裝在您的執行個體上。

```
ubuntu:~$ sudo less /lib/systemd/system/ssh.service.d/ec2-instance-connect.conf
```

   EC2如果`AuthorizedKeysCommand`和`AuthorizedKeysCommandUser`行包含下列值，則已成功安裝「執行處理 Connect 線」：

```
AuthorizedKeysCommand /usr/share/ec2-instance-connect/eic_run_authorized_keys %
%u %%f
AuthorizedKeysCommandUser ec2-instance-connect
```

   - `AuthorizedKeysCommand` 會設定 `eic_run_authorized_keys` 指令碼來從執行個體中繼資料查閱金鑰
   - `AuthorizedKeysCommandUser` 會將系統使用者設為 `ec2-instance-connect`

> **Note**
>
> 如果您先前已設定`AuthorizedKeysCommand`和`AuthorizedKeysCommandUser`，則EC2執行個體 Connect 安裝將不會變更值，且您將無法使用EC2執行個體 Connect。

使 Connect「執EC2行個體 Connect

下列指示說明如何透過 Amazon EC2 主控台、或用SSH戶端使用EC2執行個體 Connect 連線到 AWS CLI Linux 執行個體。

要求

開始之前，請務必檢閱先決條件。

連線選項

- 使用 Amazon EC2 控制台 Connect
- 使用 Connect AWS CLI
- 使 Connect 您自己的金鑰和用SSH戶端連線
- 疑難排解

使用 Amazon EC2 控制台 Connect

您可以透過 Amazon EC2 主控台使用EC2執行個體 Connect 連線到執行個體。執行個體 Connect 會處理權限。

需求

若要使用 Amazon EC2 主控台進行連線，執行個體必須具有公開IPv4地址。如果執行個體只有IPv6位址，您可以使用 ec2 執 AWS CLI行個體連線指令進行連線。

使用 Amazon EC2 主控台連線到您的執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。

3. 選取執行個體，然後選取 Connect (連線)。
4. 選擇「EC2執行處理 Connect」標籤。
5. 針對「連線類型」，請選擇「使用EC2執行個體 Connect 線
6. 對於使用者名稱，驗證使用者名稱
7. 選擇連線以開啟終端視窗。

使用 Connect AWS CLI

您可以使用 ec2-執行個體連線 AWS CLI 指令，透過用戶端連線至執行個體。SSH

如果您未指定連線類型，EC2Instance Connect 會嘗試連線至執行個體，如下所示：

- 使用公用IPv4位址進行 Connect 線。
- 如果沒有公開IPv4位址，請使用私人位IPv4址和EC2執行個體連線端點進行連線
- 如果沒有私人IPv4位址或沒有EC2執行個體 Connect 端點，請使用該IPv6位址進行連線。

要求

您必須準備好使用 AWS CLI 版本 2。如需詳細資訊，請參閱安裝或更新至最新版本的 AWS CLI.

連線類型

`auto` (default)

會CLI嘗試以下列順序和對應的連線類型，使用執行個體的 IP 位址進行連線：

- 公開IPv4：`direct`
- 私人IPv4：`eice`
- IPv6: `direct`

`direct`

會CLI嘗試依下列順序使用執行個體的 IP 位址進行連線 (不會透過EC2執行個體連線端點連線)：

- 公眾 IPv4
- IPv6
- 私人 IPv4

`eice`

一CLI律使用執行個體的私人IPv4位址。

> **Note**
>
> 將來，我們可能會變更 `auto` 連線類型的行為。若要確保使用您想要的連線類型，建議您明確地將 `--connection-type` 設定為 `direct` 或 `eice`。

當您使用執行個體 Connect Connect 至EC2執行個體時，EC2執行個體連線會將SSH公開金鑰API推送至執行個體中繼資料，並保留 60 秒。附加至您的使用者的IAM政策授權您的使用者將公開金鑰推送至執行個體中繼資料。

使用執行個體 ID 連線到執行個體

如果您只知道執行個體 ID，而且想要讓「EC2執行個體 Connect 線」決定連線至執行個體時要使用的連線類型，請使用 ec2-instance connect 指CLI令並指定ssh參數和執行個體 ID。

```
aws ec2-instance-connect ssh --instance-id i-1234567890example
```

> **Tip**
>
> 如果在使用此命令時出現錯誤，請確定您使用的是 AWS CLI 版本 2，因為該ssh參數僅在此主要版本中可用。我們也建議您定期更新至第 2 版的最新次要 AWS CLI 版本，以存取最新功能。如需詳細資訊，請參閱《使用指南》中的AWS Command Line Interface 〈關於 AWS CLI 版本 2〉。

使用執行個體 ID 和執行個體連線端點連線至EC2執行個體

如果您想要透過執行個體 Connect 線端點連線至EC2執行個體，請使用上述指令，並指定具有`eice`值的`--connection-type`參數。

```
aws ec2-instance-connect ssh --instance-id i-1234567890example --connection-type eice
```

使用執行個體 ID 和您自己的私有金鑰檔案連線至執行個體

如果您想要使用自己的私密金鑰透過EC2執行個體 Connect Endpoint 連線至執行個體，請指定執行個體 ID 和私密金鑰檔案的路徑。不包括 *file://* 在路徑中；下列範例會失敗：*file:///path/to/key*.

```
aws ec2-instance-connect ssh --instance-id i-1234567890example --private-key-file /
path/to/key.pem
```

使 Connect 您自己的金鑰和用SSH戶端連線

您可以使用自己的SSH金鑰，並在使用執行個體 Connect 時，從您選擇的用SSH戶端連線至EC2執行個體API。這可讓您從 Instance Connect 發佈公有的金鑰至執行個體。此連線方法適用於具有公有和私有 IP 地址的執行個體。

要求

- 金鑰對的需求
  - 支援的類型：RSA(開啟SSH和SSH2) 和 ED25519
  - 支援的長度：2048 和 4096
  - 如需詳細資訊，請參閱使用第三方工具建立 key pair，並將公開金鑰匯入 Amazon EC2。
- 連線至只有私人 IP 位址的執行個體時，您啟動SSH工作階段的本機電腦必須具有連線至EC2執行個體 Connect 線服務端點 (才能將SSH公開金鑰推送至執行個體)，以及與執行個體私有 IP 位址的網路連線才能建立SSH工作階段。EC2執行個體 Connect 服務端點可透過網際網路或 AWS Direct Connect 公用虛擬介面存取。若要連線至執行個體的私有 IP 位址，您可以利用AWS Direct ConnectAWS Site-to-Site VPN、或VPC對等服務。

使用您自己的金鑰和任何用SSH戶端連線至執行個體

1. (選擇性) 產生新的SSH私密金鑰和公開金鑰

   您可以使用以下命令生成新的SSH私鑰`my_key`和`my_key.pub`公鑰：

   ```
   ssh-keygen -t rsa -f my_key
   ```

2. 將您的SSH公鑰推送到實例

   使用指send-ssh-public-key令將SSH公開金鑰推送至執行個體。如果您使用 AL2 023 或 Amazon Linux 2 啟動執行個體，則的預設使用者名稱為。AMI `ec2-user`如果您使用 Ubuntu 啟動執行個體，則預設的使用者名稱AMI為`ubuntu`。

下列範例將公開金鑰推送至特定可用區域的特定執行個體，若要認證 `ec2-user`。

```
aws ec2-instance-connect send-ssh-public-key \
    --region us-west-2 \
    --availability-zone us-west-2b \
    --instance-id i-001234a4bf70dec41EXAMPLE \
    --instance-os-user ec2-user \
    --ssh-public-key file://my_key.pub
```

3. 使用私有金鑰連線至執行個體

   使用 ssh 命令，來使用私有金鑰連接至執行個體，之後系統會將公有金鑰從執行個體中繼資料中移除 (在系統移除金鑰前您有 60 秒的時間)。指定與公開金鑰對應的私密金鑰、用來啟動執行個體的AMI預設使用者名稱，以及執行個體的公開DNS名稱 (如果透過私人網路連線，請指定私人DNS名稱或 IP 位址)。新增該 `IdentitiesOnly=yes` 選項以確保連線只使用 ssh 組態中的檔案和指定的金鑰。

```
ssh -o "IdentitiesOnly=yes" -i my_key ec2-
user@ec2-198-51-100-1.compute-1.amazonaws.com
```

疑難排解

如果您在嘗試連線至執行個體時遇到錯誤，請參閱下文：

- 解決連線到 Amazon EC2 Linux 執行個體的問題
- 如何疑難排解使用執行個體連 Connect 連線到EC2EC2執行個體的問題？

解除安裝EC2實例 Connect

若要停用EC2執行個體 Connect 線，請連線至 Linux 執行個體，然後解除安裝在作業系統上安裝的`ec2-instance-connect`套件。如果`sshd`組態與安裝EC2執行個體 Connect 時的設定相符，解除安裝`ec2-instance-connect`也會移除設`sshd`定。如果您在安裝「EC2執行個體 Connect」之後修改了`sshd`組態，則必須手動更新。

Amazon Linux

您可以在 AL2 023 和 Amazon 2 2.0.20190618 或更新版本 (其中已預先設定EC2執行個體連線) 上解除安裝執行個體 Connect。EC2

解除安裝EC2使用 Amazon Linux 啟動的執行個體上的執行個體 Connect

1. 使用 Connect 至您的執行個體SSH。指定啟動執行個體時用於執行個體的 SSH key pair，以及 AL2 023 或 Amazon Linux 2 的預設使用者名稱AMI，也就是`ec2-user`說。

   例如，下列ssh命令會使用 key pair 連線至具有公用DNS名稱`ec2-a-b-c-d.us-west-2.compute.amazonaws.com`的執行個體`my_ec2_private_key.pem`。

   ```
   $ ssh -i my_ec2_private_key.pem ec2-user@ec2-a-b-c-d.us-west-2.compute.amazonaws.com
   ```

2. 使用 yum 指令，解除安裝 `ec2-instance-connect` 套件。

   ```
   [ec2-user ~]$ sudo yum remove ec2-instance-connect
   ```

Ubuntu

在使用 Ubuntu 啟動的EC2執行個體上解除安裝執行個體 Connect AMI

1. 使用 Connect 至您的執行個體SSH。指定啟動執行個體時用於執行個體的 SSH key pair，以及 Ubuntu 的預設使用者名稱AMI，也就是說`ubuntu`。

   例如，下列ssh命令會使用 key pair 連線至具有公用DNS名稱`ec2-a-b-c-d.us-west-2.compute.amazonaws.com`的執行個體`my_ec2_private_key.pem`。

   ```
   $ ssh -i my_ec2_private_key.pem ubuntu@ec2-a-b-c-d.us-west-2.compute.amazonaws.com
   ```

2. 使用 apt-get 指令，解除安裝 `ec2-instance-connect` 套件。

   ```
   ubuntu:~$ sudo apt-get remove ec2-instance-connect
   ```

使用執行個體 Connect 端點 Connect 至EC2執行個體

EC2執行個體 Connect Endpoint 可讓您從網際網路安全地連線至執行個體，而無需使用防禦主機，或要求您的虛擬私有雲端 (VPC) 具有直接的網際網路連線。

優勢

- 您可以連線至執行個體，而不需要執行個體擁有公開IPv4位址。 AWS 所有公共IPv4地址的費用，包括與執行中執行個體和彈性 IP IPv4 地址相關聯的公共地址。如需詳細資訊，請參閱 Amazon VPC 定價頁面上的公開IPv4位址索引標籤。
- 您可以從網際網路連線到執行個體，而不需要透過網際網路閘道VPC直接連線到網際網路。
- 您可以控制建立和使用EC2執行個體 Connect 端點的存取權，以便使用IAM策略和權限連線至執行個體。
- 連線至執行個體的所有嘗試 (無論是成功還是失敗) 都會記錄到CloudTrail。

定價

使用EC2執行個體 Connect 端點無需額外費用。如果您使用EC2執行個體 Connect 端點連線到不同可用區域中的執行個體，則跨可用區域的資料傳輸需支付額外費用。

目錄

- 運作方式
- 考量事項
- 授與使用EC2執行個體 Connect 端點的權限
- EC2執行個體 Connect 端點的安全群組
- 建立EC2執行個體 Connect 端點
- 使用執行個體 Connect 端點 Connect 到 Amazon EC2 EC2 執行個體
- 透過EC2執行個體 Connect 線端點建立的記錄
- 刪除EC2執行個體 Connect 端點
- EC2執行個體連線端點的服務連結角色
- EC2執行個體 Connect 端點的配額

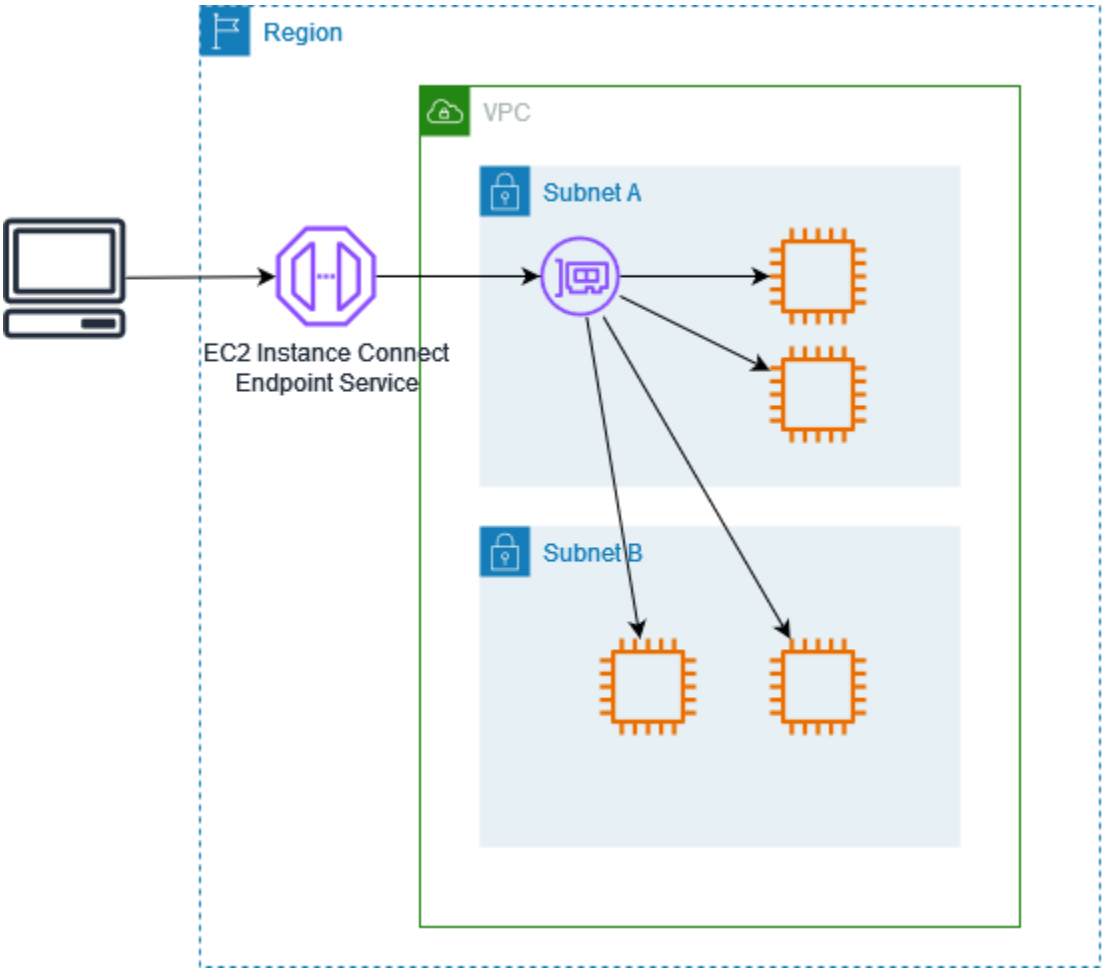
運作方式

EC2執行個體 Connect 端點是識別身分識別TCP代理伺服器。EC2執行個體 Connect 端點服務會使用您IAM實體的認證，建立從電腦到端點的私人通道。流量在到達您的VPC.

您可以設定其他安全群組規則，以限制執行個體的輸入流量。例如，您可以使用輸入規則，僅允許來自EC2執行個體連線端點的管理 Connect 埠流量。

您可以設定路由表規則，以允許端點連線到的任何子網路中的任何執行個體VPC。

下圖顯示使用者如何使用執行個體 Connect 端點從網際網路連線到其執行個EC2體。首先，在子網路 A 中建立 In EC2stance Connect Endpoint。我們為子網路中的端點建立網路介面，該介面可做為中傳送至執行個體的流量的入口點。VPC如果子網路 B 的路由表允許來自子網路 A 的流量，則您可以使用端點連線子網路 B 中的執行個體。



考量事項

在開始之前，請考慮下列事項。

- EC2執行個體 Connect 端點專門用於管理流量使用案例，而非大量資料傳輸。系統會限流大量資料傳輸。
- 您的執行個體必須擁有IPv4地址 (私人或公用)。EC2執行個體 Connect 端點不支援使用IPv6位址連線至執行個體。
- （Linux 實例）如果您使用自己的 key pair，則可以使用任何 Linux AMI。否則，您的執行個體必須安裝EC2執行個體 Connect。如需其中AMIs包括EC2執行個體 Connect，以及如何將其安裝在其他支援的資訊AMIs，請參閱安裝EC2實例 Connect。

- 您可以在建立EC2執行個體 Connect 端點時將安全群組指派給該群組。否則，我們會使用預設的安全性群組VPC。EC2執行個體 Connect 端點的安全群組必須允許輸出流量到目的地執行個體。如需詳細資訊，請參閱EC2執行個體 Connect 端點的安全群組。
- 您可以設定EC2執行個體 Connect 端點，以便在將要求傳送至執行個體時保留用戶端的來源 IP 位址。否則，網路介面的 IP 位址會變成所有傳入流量的用戶端 IP 位址。
  - 如果您開啟用戶端 IP 保留，執行個體的安全群組必須允許來自用戶端的流量。此外，執行個體必須與EC2執行個體 Connect 端點位於VPC相同的位置。
  - 如果您關閉用戶端 IP 保留，執行個體的安全性群組必須允許來自VPC. 此為預設值。
  - 下列執行個體類型不支援用戶端 IP 保留：C1 CG1、CG2、、G1、M2 HI1、M2、M3 和 T1。如果您開啟用戶端 IP 保留功能，並嘗試使用執行個體 Connect 線端點連線到具有上述其中一種執行個體類型的EC2執行個體，則連線會失敗。
  - 當流量透過傳輸閘道路由時，不支援用戶端 IP 保留。
- 當您建立EC2執行個體 Connect 端點時，會在 AWS Identity and Access Management (IAM) 中為 Amazon EC2 服務自動建立服務連結角色。Amazon EC2 使用服務連結角色在您的帳戶中佈建網路界面，這是建立EC2執行個體 Connect 端點時所需要的。如需詳細資訊，請參閱EC2執行個體連線端點的服務連結角色。
- 每個子網路VPC和每個子網路只能建立 1 EC2 個執行個體 Connect 端點。如需詳細資訊，請參閱EC2執行個體 Connect 端點的配額。如果您需要在同一個不同的可用區域中建立另一個EC2執行個體 Connect 端點VPC，則必須先刪除現有的EC2執行個體 Connect 端點。否則，您將收到配額錯誤。
- 每個EC2執行個體 Connect 線端點最多可支援 20 個同時連線。
- 已建立TCP連線的最長持續時間為 1 小時 (3,600 秒)。您可以指定IAM策略中允許的持續時間上限，最長可達 3,600 秒。如需詳細資訊，請參閱使用執行個EC2體連線端點連線至執行個體的權限。

授與使用EC2執行個體 Connect 端點的權限

根據預設，IAM實體沒有建立、描述或修改EC2執行個體 Connect 端點的權限。IAM管理員可以建立IAM策略，以授與對所需資源執行特定動作所需的權限。

如需有關建立IAM策略的資訊，請參閱IAM使用指南中的建立IAM策略。

下列範例策略顯示您可以控制使用者對於EC2執行個體 Connect 端點所擁有的權限。

範例

- 建立、說明和刪除EC2執行個體 Connect 端點的權限

- 使用執行個EC2體連線端點連線至執行個體的權限
- 僅從特定 IP 位址範圍連線的權限

建立、說明和刪除EC2執行個體 Connect 端點的權限

若要建立EC2執行個體 Connect 端點，使用者需要下列動作的權限：

- `ec2:CreateInstanceConnectEndpoint`
- `ec2:CreateNetworkInterface`
- `ec2:CreateTags`
- `iam:CreateServiceLinkedRole`

若要說明和刪除EC2執行個體 Connect 端點，使用者需要下列動作的權限：

- `ec2:DescribeInstanceConnectEndpoints`
- `ec2:DeleteInstanceConnectEndpoint`

您可以建立一個策略，授與在所有子網路中建立、描述和刪除EC2執行個體 Connect 端點的權限。或者，您只能將子網路指定ARNs為允許的子網路`Resource`或使用`ec2:SubnetID`條件索引鍵，來限制指定子網路的動作。您也可以使用 `aws:ResourceTag` 條件索引鍵來明確允許或拒絕具有特定標籤的端點建立。如需詳細資訊，請參閱《IAM使用指南》IAM中的「策略和權限」。

範例IAM政策

在下列範例IAM策略中，`Resource`區段會授與建立和刪除所有子網路中端點 (以星號 (*) 指定的權限。這些`ec2:Describe*`API動作不支援資源層級權限。因此，在 `Resource` 元素中必須包含 `*` 萬用字元。

```
{
    "Version": "2012-10-17",
    "Statement": [{
            "Sid": "GrantAllActionsInAllSubnets",
            "Action": [
                "ec2:CreateInstanceConnectEndpoint",
                "ec2:DeleteInstanceConnectEndpoint",
                "ec2:CreateNetworkInterface",
                "ec2:CreateTags",
                "iam:CreateServiceLinkedRole"
```

```
            ],
            "Effect": "Allow",
            "Resource": "arn:aws:ec2:region:account-id:subnet/*"
        },
        {
            "Action": [
                "ec2:CreateNetworkInterface"
            ],
            "Effect": "Allow",
            "Resource": "arn:aws:ec2:::security-group/*"
        },
        {
            "Sid": "DescribeInstanceConnectEndpoints",
            "Action": [
                "ec2:DescribeInstanceConnectEndpoints"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

使用執行個EC2體連線端點連線至執行個體的權限

此`ec2-instance-connect:OpenTunnel`動作會授與建立執行個體TCP連線的權限，以透過執行個EC2體 Connect 線端點進行連線。您可以指定要使用的「EC2執行個體 Connect 端點」。或者，`Resource`帶有星號 (*) 的可讓使用者使用任何可用的EC2執行個體 Connect 端點。您也可以根據是否存在作為條件索引鍵的資源標籤來限制執行個體的存取。

條件

- `ec2-instance-connect:remotePort`— 執行個體上可用來建立連線的TCP連接埠。使用此條件索引鍵時，嘗試連線到政策中指定的連接埠以外的任何其他連接埠上的執行個體會導致失敗。
- `ec2-instance-connect:privateIpAddress`— 與您要建立連線的執行個體相關聯的目標私有 TCP IP 位址。您可以指定單一 IP 位址，例如`10.0.0.1/32`，或到IPs的範圍CIDRs，例如`10.0.1.0/28`。使用此條件金鑰時，嘗試連線至具有不同私有 IP 位址或超出CIDR範圍的執行個體會導致失敗。
- `ec2-instance-connect:maxTunnelDuration`— 已建立TCP連線的最長持續時間。單位為秒，持續時間範圍為至少 1 秒至最多 3,600 秒 (1 小時)。如果未指定條件，則預設持續時間會設為 3,600 秒 (1 小時)。嘗試連線至執行個體時間超過IAM原則中指定的持續時間或超過預設上限時間會導致失敗。連線會在指定的持續時間後中斷。

如果maxTunnelDuration在IAM原則中指定，且指定的值小於 3,600 秒 (預設值)，則您必須在連線至執行個體時--max-tunnel-duration在命令中指定。如需有關如何連線至執行個體的資訊，請參閱 使用執行個體 Connect 端點 Connect 到 Amazon EC2 EC2 執行個體。

您也可以授與使用者存取權，以根據執行個體連線端點上是否存在資源標籤來建立EC2執行個體的 Connect 線。如需詳細資訊，請參閱《IAM使用指南》IAM中的「策略和權限」。

對於 Linux 執行個體，此ec2-instance-connect:SendSSHPublicKey動作會授與將公開金鑰推送至執行個體的權限。ec2:osuser 條件指定可將公有金鑰推送至執行個體的 OS (作業系統) 使用者名稱。使用您用來啟動執行AMI個體的預設使用者名稱。如需詳細資訊，請參閱授與EC2執行個體 Connect 的IAM權限。

範例IAM政策

下列範例IAM原則允許IAM主體僅使用指定的執行個體 Connect 線端點 (由指定的端點 ID 識別) 連線至EC2執行個體eice-123456789abcdef。只有在滿足所有條件的情況下，才能成功建立連接。

> **Note**
>
> 這些ec2:Describe*API動作不支援資源層級權限。因此，在 Resource 元素中必須包含 * 萬用字元。

Linux

本範例會評估執行個體的連線是否在 —port 22 (SSH) 上建立，執行個體的私有 IP 位址是否位於 10.0.1.0/31 (介於10.0.1.0和之間10.0.1.1) 的範圍內，且小maxTunnelDuration於或等於3600秒。連線會在 3600 秒 (1 小時) 後中斷。

```
{
    "Version": "2012-10-17",
    "Statement": [{
            "Sid": "EC2InstanceConnect",
            "Action": "ec2-instance-connect:OpenTunnel",
            "Effect": "Allow",
            "Resource": "arn:aws:ec2:region:account-id:instance-connect-
endpoint/eice-123456789abcdef",
            "Condition": {
                "NumericEquals": {
                    "ec2-instance-connect:remotePort": "22"
```

```
                },
                "IpAddress": {
                    "ec2-instance-connect:privateIpAddress": "10.0.1.0/31"
                },
                "NumericLessThanEquals": {
                    "ec2-instance-connect:maxTunnelDuration": "3600"
                }
            }
        },
        {
            "Sid": "SSHPublicKey",
            "Effect": "Allow",
            "Action": "ec2-instance-connect:SendSSHPublicKey",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "ec2:osuser": "ami-username"
                }
            }
        },
        {
            "Sid": "Describe",
            "Action": [
                "ec2:DescribeInstances",
                "ec2:DescribeInstanceConnectEndpoints"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

Windows

此範例會評估執行個體的連線是否在連接埠 3389 (RDP) 上建立，執行個體的私有 IP 位址是否位於 `10.0.1.0/31` (介於`10.0.1.0`和之間`10.0.1.1`) 的範圍內，且小`maxTunnelDuration`於或等於`3600`秒。連線會在 3600 秒 (1 小時) 後中斷。

```
{
    "Version": "2012-10-17",
    "Statement": [{
            "Sid": "EC2InstanceConnect",
            "Action": "ec2-instance-connect:OpenTunnel",
```

```
            "Effect": "Allow",
            "Resource": "arn:aws:ec2:region:account-id:instance-connect-
endpoint/eice-123456789abcdef",
            "Condition": {
                "NumericEquals": {
                    "ec2-instance-connect:remotePort": "3389"
                },
                "IpAddress": {
                    "ec2-instance-connect:privateIpAddress": "10.0.1.0/31"
                },
                "NumericLessThanEquals": {
                    "ec2-instance-connect:maxTunnelDuration": "3600"
                }
            }
        },
        {
            "Sid": "Describe",
            "Action": [
                "ec2:DescribeInstances",
                "ec2:DescribeInstanceConnectEndpoints"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

僅從特定 IP 位址範圍連線的權限

下列範例IAM原則允許IAM主體在從原則中指定的 IP 位址範圍內的 IP 位址連線到執行個體的情況下，連線至執行個體。如果IAM主體呼叫`OpenTunnel`的 IP 位址不在內 `192.0.2.0/24` (此原則中的範例 IP 位址範圍)，則回應為`Access Denied`。若要取得更多資訊，請參閱《IAM使用指南》*aws:SourceIp*中的。

```
{
    "Version": "2012-10-17",
    "Statement": [{
            "Effect": "Allow",
            "Action": "ec2-instance-connect:OpenTunnel",
            "Resource": "arn:aws:ec2:region:account-id:instance-connect-
endpoint/eice-123456789abcdef",
            "Condition": {
```

```
                "IpAddress": {
                    "aws:SourceIp": "192.0.2.0/24"
                },
                "NumericEquals": {
                    "ec2-instance-connect:remotePort": "22"
                }
            }
        },
        {
            "Sid": "SSHPublicKey",
            "Effect": "Allow",
            "Action": "ec2-instance-connect:SendSSHPublicKey",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "ec2:osuser": "ami-username"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeInstances",
                "ec2:DescribeInstanceConnectEndpoints"
            ],
            "Resource": "*"
        }
    ]
}
```

EC2執行個體 Connect 端點的安全群組

安全群組負責控制允許到達和離開其關聯資源的流量。例如，除非與EC2執行個體相關聯的安全群組特別允許，否則我們拒絕進出 Amazon 執行個體的流量。

下列範例說明如何為執行個體 Connect 端點和目標執行個體設定安全群組規則。EC2

範例

- EC2執行個體 Connect 端點安全性群組
- 目標例項安全性群組規則

EC2執行個體 Connect 端點安全性群組

EC2執行個體 Connect 端點的安全群組規則必須允許目標執行個體的輸出流量離開端點。您可以將執行個體安全性群組或的IPv4位址範圍指定VPC為目的地。

端點的流量源自EC2執行個體 Connect 端點服務，無論端點安全性群組的輸入規則為何，都允許此流量。若要控制誰可以使用EC2執行個體 Connect 線端點連線至執行個體，請使用IAM政策。如需詳細資訊，請參閱使用執行個EC2體連線端點連線至執行個體的權限。

輸出規則範例：安全性群組參照

下列範例使用安全性群組參考，這表示目的地是與目標執行個體相關聯的安全性群組。此規則允許從端點傳輸到使用此安全性群組的所有執行個體的輸出流量。

通訊協定	目的地	連接埠範圍	註解
TCP	*ID of instance security group*	22	允許與執行個體安全群組相關聯的所有執行個體輸出SSH流量

輸出規則範例：IPv4位址範圍

下列範例允許輸出流量傳送至指定IPv4位址範圍。執行個體的IPv4位IPv4址是從其子網路指派的，因此您可以使用VPC.

通訊協定	目的地	連接埠範圍	註解
TCP	*VPC IPv4 CIDR*	22	允許輸出SSH流量 VPC

目標例項安全性群組規則

目標執行個體的安全群組規則必須允許來自執行個體 Connect 端點的輸入流量。EC2您可以指定端點安全性群組或IPv4位址範圍作為來源。如果您指定位IPv4址範圍，則來源取決於用戶端 IP 保留是關閉還是開啟。如需詳細資訊，請參閱考量事項。

由於安全群組是可設定狀態的，因此VPC無論執行個體安全性群組的輸出規則為何，都允許回應流量離開。

輸入規則範例：安全性群組參考

下列範例使用安全性群組參考，這表示來源是與端點相關聯的安全性群組。此規則允許從端點SSH傳送到使用此安全群組的所有執行個體的輸入流量，無論用戶端 IP 保留是開啟還是關閉。如果沒有的其他輸入安全群組規則SSH，則執行個體只接受來自端點的SSH流量。

通訊協定	來源	連接埠範圍	註解
TCP	*ID of endpoint security group*	22	允許來自與端點安全群組關聯之資源的輸入SSH流量

輸入規則範例：用戶端 IP 保留關閉

下列範例允許來自指定IPv4位址範圍的輸入SSH流量。由於用戶端 IP 保留已關閉，因此來源位IPv4址是端點網路介面的位址。端點網路介面的位址是從其子網路指派的，因此您可以使用的IPv4位址範圍來允許連線到中的所有執行個體VPC。VPC

通訊協定	來源	連接埠範圍	註解
TCP	*VPC IPv4 CIDR*	22	SSH允許來自 VPC

輸入規則範例：用戶端 IP 保留

下列範例允許來自指定IPv4位址範圍的輸入SSH流量。由於用戶端 IP 保留已開啟，因此來源位IPv4址是用戶端的位址。

通訊協定	來源	連接埠範圍	註解
TCP	*Public IPv4 address range*	22	允許來自指定用戶端IPv4位址範圍的輸入流量

建立EC2執行個體 Connect 端點

您可以建立EC2執行個體 Connect 線端點，以允許與執行個體的安全連線。

建立EC2執行個體 Connect 端點後，就無法修改它。相反地，您必須刪除EC2執行個體 Connect 端點，然後使用您需要的設定建立新的端點。

必要條件

您必須擁有必要的IAM權限才能建立EC2執行個體 Connect 端點。如需詳細資訊，請參閱建立、說明和刪除EC2執行個體 Connect 端點的權限。

共用子網路

您可以在與您共用的子網路中建立EC2執行個體 Connect 端點。您無法使用VPC擁有者在與您共用的子網路中建立的EC2執行個體 Connect 端點。

使用主控台建立端點

請使用下列程序來建立EC2執行個體 Connect 端點。

建立EC2執行個體 Connect 端點

1. 在打開 Amazon VPC 控制台https://console.aws.amazon.com/vpc/。
2. 在左側導覽窗格中選擇 Endpoints (端點)。
3. 選擇 [建立端點]，然後依下列方式指定端點設定：
   a. (選用) 針對名稱標籤，請輸入端點的名稱。
   b. 在「服務」類別中，選擇「EC2執行個體 Connect 端
   c. 對於 VPC，選取具VPC有目標例證的。
   d. (選擇性) 若要保留用戶端 IP 位址，請展開其他設定，然後選取核取方塊。否則，預設會使用端點網路介面做為用戶端 IP 位址。
   e. (選用) 針對安全群組，選取要與端點產生關聯的安全群組。否則，預設為使用的預設安全性群組VPC。如需詳細資訊，請參閱EC2執行個體 Connect 端點的安全群組。
   f. 針對子網路，請選取要在其中建立端點的子網路。
   g. (選用) 若要新增標籤，請選擇 Add new tag (新增標籤)，然後輸入標籤的鍵和值。
4. 檢閱您的設定，然後選擇 [建立端點]。

   端點的初始狀態為「擱置中」。您必須等到端點狀態為「可用」，才能使用此端點連線至執行個體。這可能需要幾分鐘的時間。

5. 若要使用端點連線至執行個體，請參閱連接至執行個體。

使用建立端點 AWS CLI

使用指create-instance-connect-endpoint令建立「EC2執行個體 Connect 端點」。

必要條件

安裝 AWS CLI 版本 2 並使用您的憑據進行配置。若要取得更多資訊，請參閱《AWS Command Line Interface 使用者指南》 AWS CLI中的〈安裝或更新至最新版本〉 AWS CLI和〈規劃〉。或者，在其預先驗證的 AWS CLI shell 中打開 AWS CloudShell 並運行命令。

建立端點

使用下列命令為指定子網路中的EC2執行個體 Connect 端點建立端點網路介面。

```
aws ec2 create-instance-connect-endpoint --subnet-id subnet-0123456789example
```

下列為範例輸出。

```
 {
        "OwnerId": "111111111111",
        "InstanceConnectEndpointId": "eice-0123456789example",
        "InstanceConnectEndpointArn": "arn:aws:ec2:us-east-1:111111111111:instance-
connect-endpoint/eice-0123456789example",
        "State": "create-complete",
        "StateMessage": "",
        "DnsName": "eice-0123456789example.0123abcd.ec2-instance-connect-endpoint.us-
east-1.amazonaws.com",
        "FipsDnsName": "eice-0123456789example.0123abcd.fips.ec2-instance-connect-
endpoint.us-east-1.amazonaws.com",
        "NetworkInterfaceIds": [
            "eni-0123abcd"
        ],
        "VpcId": "vpc-0123abcd",
        "AvailabilityZone": "us-east-1a",
        "CreatedAt": "2023-04-07T15:43:53.000Z",
        "SubnetId": "subnet-0123abcd",
        "PreserveClientIp": false,
        "SecurityGroupIds": [
            "sg-0123abcd"
        ],
```

```
        "Tags": []
}
```

監視建立狀態

State 欄位的初始值為 create-in-progress。在您可使用此端點連線至執行個體之前，請等待狀態變為 create-complete。使用命describe-instance-connect-endpoints令監視「EC2執行個體 Connect 端點」的狀態。-query 參數會將結果篩選至欄位State。

```
aws ec2 describe-instance-connect-endpoints --instance-connect-endpoint-
ids eice-0123456789example --query InstanceConnectEndpoints[*].State --output text
```

下列為範例輸出。

```
create-complete
```

使用執行個體 Connect 端點 Connect 到 Amazon EC2 EC2 執行個體

您可以使用EC2執行個體 Connect 端點連接到支援SSH或的 Amazon EC2 執行個體RDP。

目錄

- 必要條件
- 疑難排解

必要條件

- 您必須擁有必要的IAM權限才能 Connect 至EC2執行個體連線端點。如需詳細資訊，請參閱使用執行個EC2體連線端點連線至執行個體的權限。
- EC2執行個體 Connect 端點必須處於可用 (主控台) 或 create-complete (AWS CLI) 狀態。如果您的EC2執行個體 Connect 端點沒有VPC，您可以建立一個。如需詳細資訊，請參閱建立EC2執行個體 Connect 端點。
- 您的執行個體必須擁有IPv4地址 (私人或公用)。EC2執行個體 Connect 端點不支援使用IPv6位址連線至執行個體。
- (Linux 執行個體) 若要使用EC2主控台 Connect 至執行個體，或使用 Connect 並CLI讓EC2執行個體連線處理暫時金鑰，您的執行個體必須已EC2安裝執行個體連線。如需詳細資訊，請參閱安裝EC2實例 Connect。

- 確定執行個體的安全性群組允許來自執行個EC2體 Connect 端點的輸入SSH流量。如需詳細資訊，請參閱目標例項安全性群組規則。

使用 Amazon EC2 主控台 Connect 到您的 Linux 執行個體

您可以使用 Amazon EC2 主控台連線到執行個體，如下所示。

使用以瀏覽器為基礎的用戶端連線至執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，選擇 [Connect]。
4. 選擇「EC2執行處理 Connect」標籤。
5. 針對「連線類型」，請選擇「使用EC2執行個體 Connect 線端點
6. 在「EC2執行個體 Connect 端點」中，選擇「EC2執行個體 Connect 端點」的 ID。
7. 對於使用者名稱，如果您AMI用來啟動執行個體的使用者名稱不是`ec2-user`，請輸入正確的使用者名稱。
8. 在通道持續時間上限 (秒) 中，輸入允許的SSH連線持續時間上限。

   持續時間必須符合IAM政策中指定的任何`maxTunnelDuration`條件。如果您無權存取IAM原則，請連絡您的系統管理員。
9. 選擇連線。這會為您的執行個體開啟終端機視窗。

使用 Connect 到您的 Linux 執行個體 SSH

您可以使用連線SSH到 Linux 執行個體，並使用`open-tunnel`指令建立私有通道。可以在單一連線或多重連線模式下使用 `open-tunnel`。

如需使用連線 AWS CLI 至執行個體的相關資訊SSH，請參閱使用 Connect AWS CLI。

下列範例使用 Open SSH。您可以使用任何其他支援 Proxy 模式的用SSH戶端。

單一連接

若要使用SSH和**`open-tunnel`**指令僅允許單一連線至執行個體

使用`ssh`和命open-tunnel AWS CLI 令如下。`-o` 代理命令包含 `open-tunnel` 命令，它可建立執行個體的私有通道。

```
ssh -i my-key-pair.pem ec2-user@i-0123456789example \
    -o ProxyCommand='aws ec2-instance-connect open-tunnel --instance-
id i-0123456789example'
```

對於：

- `-i` – 指定用來啟動執行個體的金鑰對。
- *`ec2-user`*`@`*`i-0123456789example`*— 指定用來啟動執行AMI個體的使用者名稱，以及執行個體ID。
- `--instance-id` – 指定要連線之執行個體的 ID。或者，指定 `%h`，其可從使用者中擷取執行個體ID。

多重連線

若要允許多個連線至執行個體，請先執行open-tunnel AWS CLI 命令以開始接聽新的TCP連線，然後使ssh用建立新的TCP連線和私人通道至您的執行個體。

若要使用和**`open-tunnel`**指令允許多個連線至您SSH的執行個體

1. 執行下列命令，開始偵聽本機電腦上指定TCP連接埠上的新連線。

   ```
   aws ec2-instance-connect open-tunnel \
       --instance-id i-0123456789example \
       --local-port 8888
   ```

   預期的輸出結果

   ```
   Listening for connections on port 8888.
   ```

2. 在新的終端機視窗中，執行下列ssh命令，為執行個體建立新TCP連線和私人通道。

   ```
   ssh -i my-key-pair.pem ec2-user@localhost -p 8888
   ```

   預期輸出 – 在第一個終端視窗中，您會看到以下內容：

   ```
   [1] Accepted new tcp connection, opening websocket tunnel.
   ```

   您可能會看到以下內容：

```
[1] Closing tcp connection.
```

使用 Connect 到您的 Linux 執行個體 AWS CLI

如果您只知道執行個體 ID，則可以使用 ec2-instance connect AWS CLI 指令，透過用戶端連線至執行個體。SSH若要取得有關使用 ec2 執行個體連線指令的更多資訊，請參閱〈〉。使用 Connect AWS CLI

必要條件

安裝 AWS CLI 版本 2 並使用您的憑據進行配置。若要取得更多資訊，請參閱《AWS Command Line Interface 使用者指南》 AWS CLI中的〈安裝或更新至最新版本〉 AWS CLI和〈規劃〉。或者，在其預先驗證的 AWS CLI shell 中打開 AWS CloudShell 並運行命令。

使用執行個體 ID 和執行個體連線端點連線至EC2執行個體

如果您只知道例證 ID，請使用 ec2-instance connect 指CLI令，並使用值指定指`ssh`令、例證 ID 和`--connection-type`參數。`eice`

```
aws ec2-instance-connect ssh --instance-id i-1234567890example --connection-type eice
```

> **Tip**
> 如果使用此命令時出現錯誤，請確定您使用的是 AWS CLI 版本 2。該`ssh`參數僅在 AWS CLI 版本 2 中可用。如需詳細資訊，請參閱《使用指南》中的AWS Command Line Interface 〈關於 AWS CLI 版本 2〉。

使用執行個體 Connect 端點 Connect 到 Windows EC2 執行個體

您可以透過EC2執行個體 Connect 線端點使用遠端桌面通訊協定 (RDP)，連線到沒有公開IPv4位址或公開DNS名稱的 Windows 執行個體。

使用用RDP戶端連線至您的 Windows 執行個體

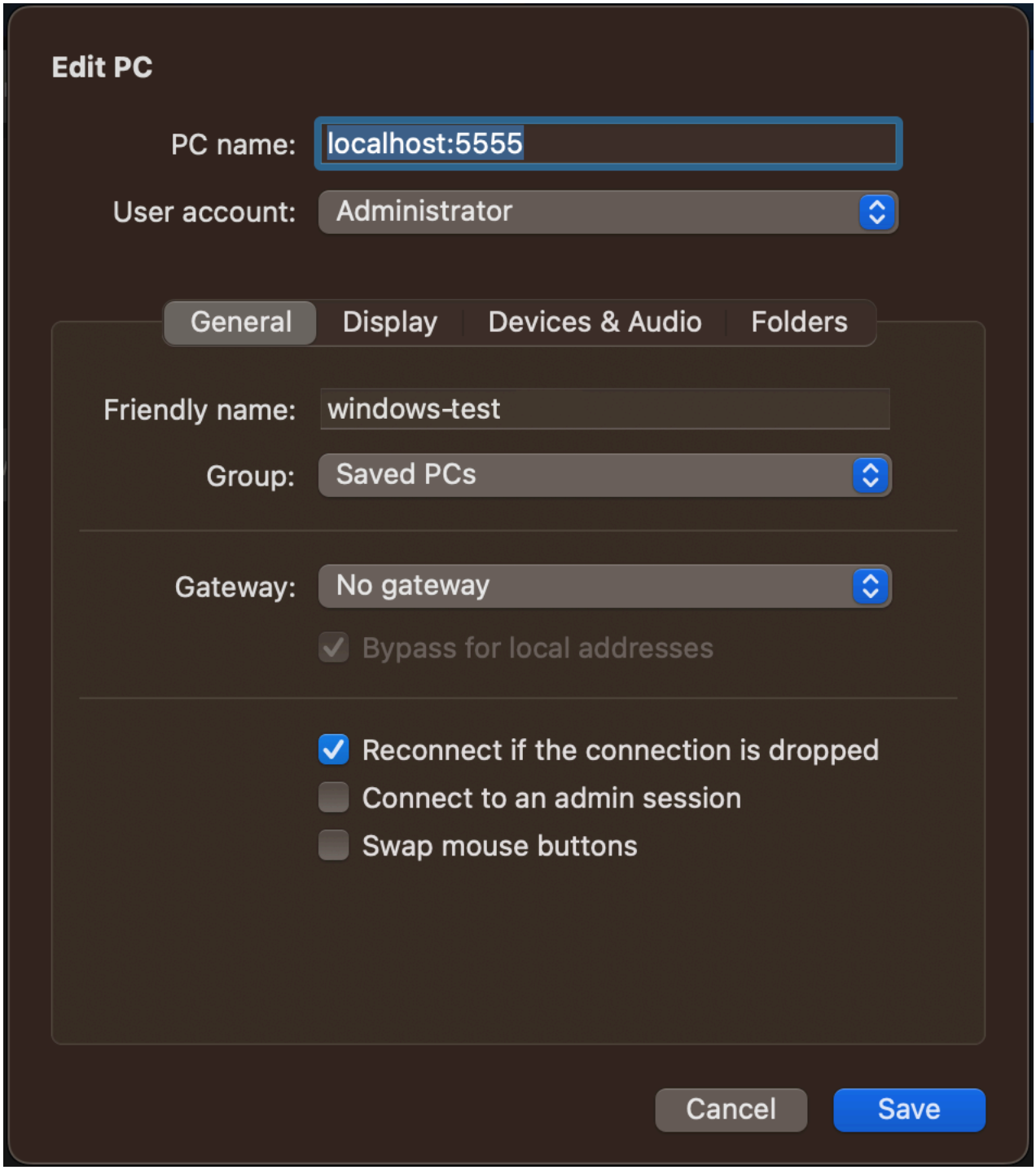
1. 完成使用 Connect 至您的 Windows 執行個體中的步驟 1 — 8 RDP。在步驟 8 下載RDP桌面檔案後，您會收到「無法連線」訊息，這是預期的訊息，因為您的執行個體沒有公用 IP 位址。

2. 執行下列命令，建立執行個體所VPC在的私人通道。 `--remote-port`必須是3389因為依預設RDP使用連接埠 3389。

```
aws ec2-instance-connect open-tunnel \
    --instance-id i-0123456789example \
    --remote-port 3389 \
    --local-port any-port
```

3. 在「下載」資料夾中，找到您下載的RDP桌面檔案，然後將其拖曳至用RDP戶端視窗。
4. 以滑鼠右鍵按一下RDP桌面檔案，然後選擇
5. 在 [編輯電腦] 視窗中，針對電腦名稱 (要連線的執行個體) 輸入`localhost:`*`local-port`*，其中*`local-port`*使用與您在步驟 2 中指定的相同值，然後選擇 [儲存]。

   請注意，編輯電腦視窗的以下螢幕截取畫面來自 Mac 中的 Microsoft 遠端桌面。如果使用 Windows 用戶端，則該視窗可能會有所不同。



6. 在RDP用戶端中，以滑鼠右鍵按一下電腦 (您剛設定的電腦)，然後選擇 [Connect] 以連線至您的執行個體。
7. 出現提示時，輸入管理員帳戶的解密密碼。

疑難排解

使用下列資訊協助診斷並修正使用執行個EC2體連線端點連線執行個體時可能遇到的問題。

無法連線至執行個體

以下是您可能無法連線至執行個體的常見原因。

- 安全群組 — 檢查指派給EC2執行個體 Connect 端點和執行個體的安全群組。有關所需安全群組規則的詳細資訊，請參閱 EC2執行個體 Connect 端點的安全群組。
- 執行個體狀態：驗證您的執行個體處於 `running` 狀態。
- 金鑰對：如果用來連線的命令需要私有金鑰，請確認執行個體具有公有金鑰，且您擁有對應的私有金鑰。
- IAM權限 — 確認您擁有必要的IAM權限。如需詳細資訊，請參閱授與使用EC2執行個體 Connect 端點的權限。

如需 Linux 執行個體的詳細疑難排解秘訣，請參閱解決連線到 Amazon EC2 Linux 執行個體的問題。如需 Windows 執行個體的疑難排解提示，請參閱the section called “視窗實例RDP問題”。

ErrorCode: AccessDeniedException

如果您收到錯`AccessDeniedException`誤訊息，且`maxTunnelDuration`條件已在IAM原則中指定，請務必在連線至執行個體時指定`--max-tunnel-duration`參數。如需有關此參數的詳細資訊，請參閱《AWS CLI 命令參考》中的 open-tunnel。

透過EC2執行個體 Connect 線端點建立的記錄

您可以使用 AWS CloudTrail 記錄記錄來記錄資源作業和稽核透過EC2執行個體 Connect 線端點建立的連線。

如需 AWS CloudTrail 搭配 Amazon 使用的詳細資訊EC2，請參閱使用記錄 Amazon EC2 API 呼叫 AWS CloudTrail。

記錄EC2執行個體 Connect 端點API呼叫與 AWS CloudTrail

EC2執行個體 Connect 端點資源作業會記錄 CloudTrail 為管理事件。進行下列API呼叫時，活動會記錄為 CloudTrail 事件歷史記錄中的事件：

- `CreateInstanceConnectEndpoint`
- `DescribeInstanceConnectEndpoints`

- `DeleteInstanceConnectEndpoint`

您可以查看，搜索和下載最近的事件 AWS 帳戶。如需詳細資訊，請參閱AWS CloudTrail 使用指南中的檢視具有 CloudTrail 事件歷程記錄的事件。

用於稽核 AWS CloudTrail 使用執行個體連線端點連線至執行個EC2體的使用者

透過EC2執行個體連線端點嘗試執行個體的 Connect 線會記錄 CloudTrail 在事件歷程記錄中。透過執行個體連線端點初始化EC2執行個體的 Connect 線時，連線會記錄為使用`eventName`的 CloudTrail 管理事件`OpenTunnel`。

您可以建立將 CloudTrail 事件路由到目標的 Amazon EventBridge 規則。如需詳細資訊，請參閱 Amazon EventBridge 使用者指南。

以下是已登入之`OpenTunnel`管理事件的範例 CloudTrail。

```
{
     "eventVersion": "1.08",
     "userIdentity": {
         "type": "IAMUser",
         "principalId": "ABCDEFGONGNOMOOCB6XYTQEXAMPLE",
         "arn": "arn:aws:iam::1234567890120:user/IAM-friendly-name",
         "accountId": "123456789012",
         "accessKeyId": "ABCDEFGUKZHNAW4OSN2AEXAMPLE",
         "userName": "IAM-friendly-name"
     },
     "eventTime": "2023-04-11T23:50:40Z",
     "eventSource": "ec2-instance-connect.amazonaws.com",
     "eventName": "OpenTunnel",
     "awsRegion": "us-east-1",
     "sourceIPAddress": "1.2.3.4",
     "userAgent": "aws-cli/1.15.61 Python/2.7.10 Darwin/16.7.0 botocore/1.10.60",
     "requestParameters": {
         "instanceConnectEndpointId": "eici-0123456789EXAMPLE",
         "maxTunnelDuration": "3600",
         "remotePort": "22",
         "privateIpAddress": "10.0.1.1"
     },
     "responseElements": null,
     "requestID": "98deb2c6-3b3a-437c-a680-03c4207b6650",
     "eventID": "bbba272c-8777-43ad-91f6-c4ab1c7f96fd",
     "readOnly": false,
```

```
     "resources": [{
         "accountId": "123456789012",
         "type": "AWS::EC2::InstanceConnectEndpoint",
         "ARN": "arn:aws:ec2:us-east-1:123456789012:instance-connect-endpoint/
eici-0123456789EXAMPLE"
     }],
     "eventType": "AwsApiCall",
     "managementEvent": true,
     "recipientAccountId": "123456789012",
     "eventCategory": "Management"
}
```

刪除EC2執行個體 Connect 端點

當您完成EC2執行個體 Connect 端點時，您可以將其刪除。

您必須擁有必要的IAM權限才能建立EC2執行個體 Connect 端點。如需詳細資訊，請參閱建立、說明和刪除EC2執行個體 Connect 端點的權限。

當您使用主控台刪除EC2執行個體 Connect 端點時，該端點會進入 [刪除] 狀態。如果刪除成功，則刪除的端點將不再顯示。如果刪除失敗，狀態為`delete-failed`，狀態訊息會提供失敗原因。

當您使用刪除EC2執行個體 Connect 端點時 AWS CLI，它將進入`delete-in-progress`狀態。如果刪除成功，它將進入`delete-complete`狀態。如果刪除失敗，則狀態為`delete-failed`並`StateMessage`提供失敗原因。

Console

刪除EC2執行個體 Connect 端點

1. 在打開 Amazon VPC 控制台https://console.aws.amazon.com/vpc/。
2. 在左側導覽窗格中選擇 Endpoints (端點)。
3. 選取端點。
4. 選擇「動作」、「刪除VPC端點」
5. 出現確認提示時，請按一下 **delete**。
6. 選擇 刪除 。

AWS CLI

刪除EC2執行個體 Connect 端點

使用指delete-instance-connect-endpoint AWS CLI 令並指定要刪除之EC2執行個體 Connect 端點的 ID。

```
aws ec2 delete-instance-connect-endpoint --instance-connect-endpoint-
id eice-03f5e49b83924bbc7
```

下列為範例輸出。

```
{
    "InstanceConnectEndpoint": {
        "OwnerId": "111111111111",
        "InstanceConnectEndpointId": "eice-0123456789example",
        "InstanceConnectEndpointArn": "arn:aws:ec2:us-east-1:111111111111:instance-
connect-endpoint/eice-0123456789example",
        "State": "delete-in-progress",
        "StateMessage": "",
        "NetworkInterfaceIds": [],
        "VpcId": "vpc-0123abcd",
        "AvailabilityZone": "us-east-1d",
        "CreatedAt": "2023-02-07T12:05:37+00:00",
        "SubnetId": "subnet-0123abcd"
    }
}
```

EC2執行個體連線端點的服務連結角色

Amazon EC2 使用 AWS Identity and Access Management （IAM）服務鏈接角色。服務連結角色是直接連結至 Amazon EC2 的一種獨特IAM角色類型。服務連結角色由 Amazon 預先定義，EC2並包含所有必要的許可，以便 Amazon EC2 可以代表您呼叫其他 AWS 服務 角色。如需詳細資訊，請參閱IAM使用指南中的服務連結角色。

EC2執行個體連線端點的服務連結角色權限

Amazon EC2 使AWSServiceRoleForEC2InstanceConnect用在您的帳戶中建立和管理EC2執行個體 Connect 端點所需的網路界面。

服AWSServiceRoleForEC2InstanceConnect務連結角色會信任下列服務擔任該角色：

- `ec2-instance-connect.amazonaws.com`

AWSServiceRoleForEC2InstanceConnect服務連結角色使用受管政策 EC2 InstanceConnectEndpoint。若要檢視此政策的許可，請參閱AWS 受管政策參考 InstanceConnectEndpoint中的 EC2。

您必須設定權限，才能允許IAM實體 (例如使用者、群組或角色) 建立、編輯或刪除服務連結角色。如需詳細資訊，請參閱IAM使用指南中的服務連結角色權限。

為EC2執行個體連線端點建立服務連結角色

您不需要手動建立 服務連結角色。當您建立EC2執行個體 Connect 端點時，Amazon EC2 會為您建立服務連結角色。

編輯執行個EC2體連線端點的服務連結角色

EC2執行個體 Connect 端點不允許您編輯AWSServiceRoleForEC2InstanceConnect服務連結角色。

刪除執行個EC2體連線端點的服務連結角色

如果您不再需要使用EC2執行個體 Connect 端點，建議您刪除AWSServiceRoleForEC2InstanceConnect服務連結角色。

您必須先刪除所有EC2執行個體 Connect 端點資源，才能刪除服務連結角色。

若要刪除服務連結角色，請參閱IAM使用指南中的刪除服務連結角色。

EC2執行個體 Connect 端點的配額

您的每項 AWS 服務都 AWS 帳戶 有預設配額 (先前稱為限制)。除非另有說明，否則每個配額都是區域特定規定。

您 AWS 帳戶 有下列與EC2執行個體 Connect 端點相關的配額。

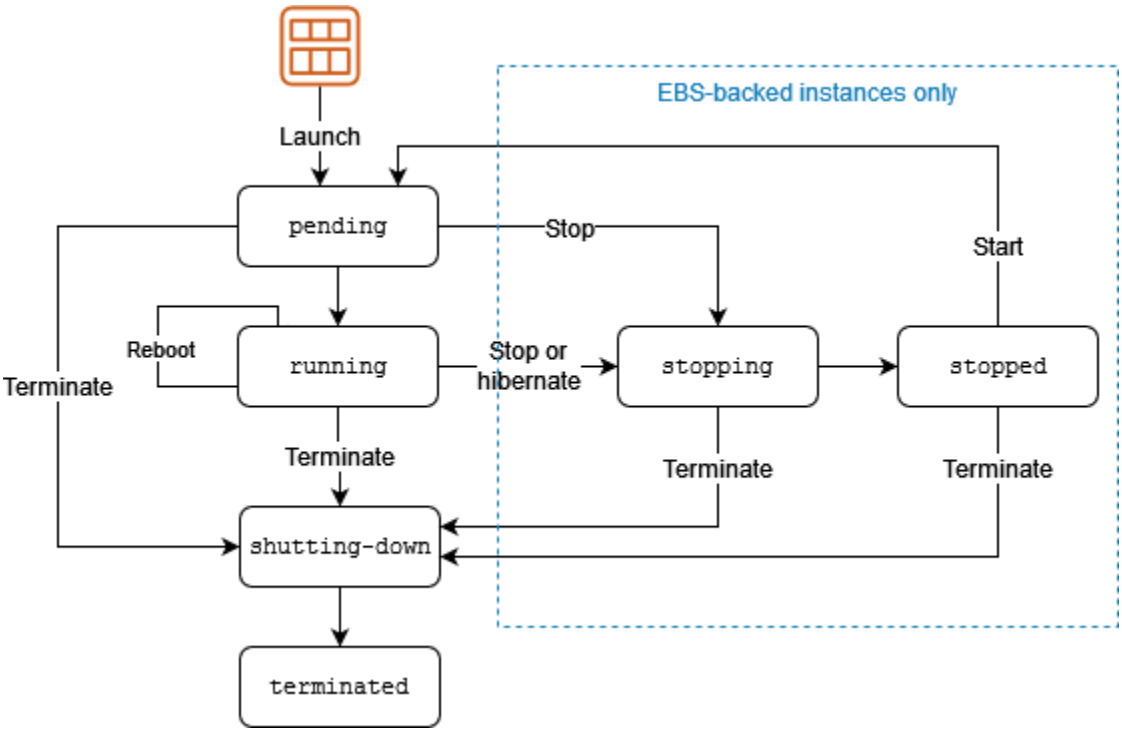
名稱	預設	可調整
每EC2 AWS 帳戶 個執行個體 Connect 端點的最大數目 AWS 區域	5	否
每EC2個執行個體 Connect 端點的最大數目 VPC	1	否

名稱	預設	可調整
每個子網路的最大EC2執行個體 Connect 端點數	1	否
每個執行個體連線端點的同時連線EC2數目上限	20	否

Amazon EC2 實例狀態更改

從您啟動 Amazon EC2 執行個體到終止的那一刻，Amazon 執行個體就會經過不同的狀態。

下圖顯示執行個體狀態之間的轉換。



您可以在執行個體狀態變更時收到通知。如需詳細資訊，請參閱the section called “狀態變更事件”。

依執行個體狀態計費

下表提供每個執行個體狀態的簡短說明，並指出是否要計費執行個體用量。某些 AWS 資源 (例如 Amazon EBS 磁碟區和彈性 IP 地址) 會產生費用，無論執行個體的狀態為何。如需詳細資訊，請參閱 AWS Billing 使用者指南中的避免非預期的費用。

執行個體狀態	描述	執行個體用量計費
pending	執行個體準備進入 running 狀態。當執行個體啟動時，或者在處於 stopped 狀態之後啟動時，將進入 pending 狀態。	不計費
running	執行個體正在執行中且可供使用。	計費
stopping	執行個體已準備停止。	不計費 Note 如果您將執行個體休眠，則會在執行個體處於stopping狀態時向您收費。
stopped	執行個體已關閉且無法使用。執行個體可以隨時啟動。	不計費
shutting down	執行個體準備終止。	不計費
terminated	執行個體已永久刪除且無法啟動。	不計費 Note 適用於已終止執行個體的預留執行個體，根據其付款選項計費至期滿。如需詳細資訊，請參閱 Amazon 預留執行個體EC2概觀

待處理執行

當您啟動執行個體時，其將進入 pending 狀態。您在啟動時指定的執行個體類型決定託管執行個體的電腦硬體。我們使用您在啟動時指定的 Amazon 機器映像 (AMI) 來啟動執行個體。執行個體便已準備就緒，可供使用時，其將變為 running 狀態。接著您可連接到執行中的執行個體，並以一般使用電腦的方式使用。

您的執行個體轉換為 running 狀態後，會針對該執行個體執行的每秒計費 (最少為一分鐘)，即使執行個體保持閒置而且您未連接它仍會計費。

停止的執行

如果執行個體無法進行狀態檢查或未如預期般執行應用程式，而且執行個體的根磁碟區是 Amazon EBS 磁碟區，您可以停止並啟動執行個體以嘗試修正問題。

停止執行個體時，其會變成 stopping 狀態，然後再變成 stopped 狀態。執行個體若為 stopped，您不需支付使用量或資料傳輸費用。任何 Amazon EBS 卷的存儲都會產生費用。當您的執行個體變為 stopped 狀態時，您可修改執行個體的部分屬性，包括執行個體類型。

當您啟動執行個體時，它會進入 pending 狀態，執行個體會移至新的主機電腦 (但有時會保留在目前的主機上)。停止並啟動執行個體時，連接原本主機電腦的執行個體儲存體磁碟區內的所有資料可能都會遺失。

您的執行個體會保留其私有位IPv4址，這表示與私有位址或網路介面相關聯的彈性 IP 位IPv4址仍會與您的執行個體保持關聯。如果您的執行個體有IPv6地址，則會保留該IPv6位址。

每次將執行個體從 stopped 轉換為 running 時，只要執行個體在執行就會按秒收費，每次啟動執行個體至少會收費一分鐘。

如需停止和啟動執行個體的詳細資訊，請參閱：停止並啟動 Amazon EC2 執行個體。

休眠執行個體

當您將執行個體休眠時，我們會通知作業系統執行休眠 (suspend-to-disk)，這樣會將執行個體記憶體 (RAM) 中的內容儲存到 Amazon EBS 根磁碟區。我們會保留執行個體的 Amazon EBS 根磁碟區和任何連接的 Amazon EBS 資料磁碟區。啟動執行個體時，Amazon EBS 根磁碟區會還原到先前的狀態，並重新載入RAM內容。先前連接的資料磁碟區會重新連接，且執行個體會保留其執行個體 ID。

執行個體休眠時會變成 stopping 狀態，然後再變成 stopped 狀態。當執行個體處於 stopped 狀態時，不收取休眠執行個體的使用費，但當執行個體處於 stopping 狀態時，我們會收費，這與停止

執行個體但不休眠的情況不同。我們不會針對資料傳輸費用收取使用費，但我們會針對任何 Amazon EBS 磁碟區的儲存收取費用，包括RAM資料儲存。

當您啟動休眠的執行個體時，它會進入 `pending` 狀態，而且我們會將執行個體移至新的主機電腦 (但有時候會保留在目前的主機上)。

您的執行個體會保留其私有位IPv4址，這表示與私有位址或網路介面相關聯的彈性 IP 位IPv4址仍會與您的執行個體建立關聯。如果您的執行個體有IPv6地址，它會保留其IPv6位址。

如需詳細資訊，請參閱休眠您的 Amazon EC2 實例。

將執行個體

您可以使用 Amazon EC2 主控台、命令列工具和 Amazon 重新啟動執行個體EC2API。建議您使用 Amazon EC2 重新啟動執行個體，而不是從執行個體執行作業系統重新啟動命令。

重新啟動執行個體等同於重新啟動作業系統。執行個體會保留在相同的主機電腦上，並維護其公開DNS名稱、私有 IP 位址，以及執行個體儲存磁碟區上的任何資料。重新啟動通常需要幾分鐘時間才能完成，但重新啟動實際需要時間仍取決於執行個體組態。

重新啟動執行個體不會啟動新的執行個體計費期間；會繼續以秒計費，但不會收取其他的最少一分鐘收費。

如需詳細資訊，請參閱重新啟動您的執行個體。

終止的實例

當您決定不再需要執行個體時，您可終止執行個體。執行個體的狀態變更為 `shutting-down` 或 `terminated` 時刻起，該執行個體便停止收取費用。

如果您啟用終止保護，就無法使用主控台CLI、或終止執行個體API。

終止執行個體後，仍會短暫顯示於主控台，之後該項目會自動刪除。您也可以使用和來描述終止的執行個CLI體API。資源 (例如標籤) 會逐漸與該終止的執行個體取消關聯，過一段時間後，將不再顯示於該已終止的執行個體。您無法連接或復原已終止的執行個體。

每個 Amazon EBS 支援的執行個體都支援`InstanceInitiatedShutdownBehavior`屬性，該屬性可控制執行個體在您從執行個體本身啟動關機時停止或終止 (例如，在 Linux 上使用shutdown命令)。預設行為是停止執行個體。您可在執行個體運作中或停止時，修改此屬性設定。

每個 Amazon EBS 磁碟區都支援該`DeleteOnTermination`屬性，該屬性可控制當您終止磁碟區連接到的執行個體時是刪除還是保留磁碟區。預設值是刪除根裝置磁碟區並保留任何其他EBS磁碟區。

如需詳細資訊，請參閱終止 Amazon EC2 實例。

實例狀態之間的差異

下表摘要列出正在重新啟動、正在停止、正在休眠和正在終止執行個體之間的主要差異。

特性	重新開機	停止/啟動 (僅EBS限 Amazon 支援的執行個體)	休眠 (僅EBS限 Amazon 支援的執行個體)	終止
主機電腦	執行個體保持在同一個主機電腦上	我們將執行個體移至新的主機電腦 (但有時候會保留在目前主機上)。	我們將執行個體移至新的主機電腦 (但有時候會保留在目前主機上)。	無
私人和公共IPv4地址	這些地址將保持不變	執行個體會保留其私人IPv4位址。執行個體會取得新的公用位IPv4址，除非它有彈性 IP 位址，在停止/啟動期間不會變更。	執行個體會保留其私人IPv4位址。執行個體會取得新的公用位IPv4址，除非它有彈性 IP 位址，在停止/啟動期間不會變更。	無
彈性 IP 位址 (IPv4)	彈性 IP 地址仍與執行個體關聯	彈性 IP 地址仍與執行個體關聯	彈性 IP 地址仍與執行個體關聯	彈性 IP 地址將從執行個體解除關聯
IPv6地址	執行個體會保留其IPv6位址	執行個體會保留其IPv6位址	執行個體會保留其IPv6位址	無
執行個體存放區磁碟區	資料將保留	資料將清除	資料將清除	資料將清除
根設備磁碟區	磁碟區將保留	磁碟區將保留	磁碟區將保留	磁碟區預設為刪除

特性	重新開機	停止/啟動 (僅EBS限 Amazon 支援的執行個體)	休眠 (僅EBS限 Amazon 支援的執行個體)	終止
RAM(記憶體內容)	被刪除 RAM	被刪除 RAM	儲RAM存到根磁碟區上的檔案	被刪除 RAM
帳單	執行個體帳單時間不會變更	執行個體的狀態變更為 `stopping` 時刻起，該執行個體便停止收取費用。每次執行個體從 `stopped` 轉移為 `running` 時，我們都會啟動新的執行個體計費期間，每次啟動執行個體時最少收費一分鐘。	執行個體的狀態為 `stopping` 時會產生費用，但執行個體的狀態為 `stopped` 時即停止產生費用。每次執行個體從 `stopped` 轉移為 `running` 時，我們都會啟動新的執行個體計費期間，每次啟動執行個體時最少收費一分鐘。	一旦執行個體的狀態變更為，就不再產生費用 `shutting-down`

作業系統關機命令每次都會終止執行個體後端執行個體。您可以控制作業系統關閉命令是停止還是終止 Amazon EBS 支援的執行個體。如需詳細資訊，請參閱變更執行個體啟動的關機行為。

停止並啟動 Amazon EC2 執行個體

如果執行個體的根裝置有 Amazon EBS 磁碟區，您可以停止並啟動執行個體。當您停止執行個體時，執行個體會關閉。啟動執行個體時，通常會將其移轉至新的基礎主機電腦，並指派新的公用IPv4位址。

當您停止執行個體時，該執行個體不會刪除。如果您決定不再需要使用執行個體時，可將其終止。如需詳細資訊，請參閱終止 Amazon EC2 實例。如果要讓執行個體休眠以儲存執行個體記憶體 (RAM) 中的內容，請參閱休眠您的 Amazon EC2 實例。有關執行個體生命週期操作之間的區別，請參閱 :實例狀態之間的差異。

目錄

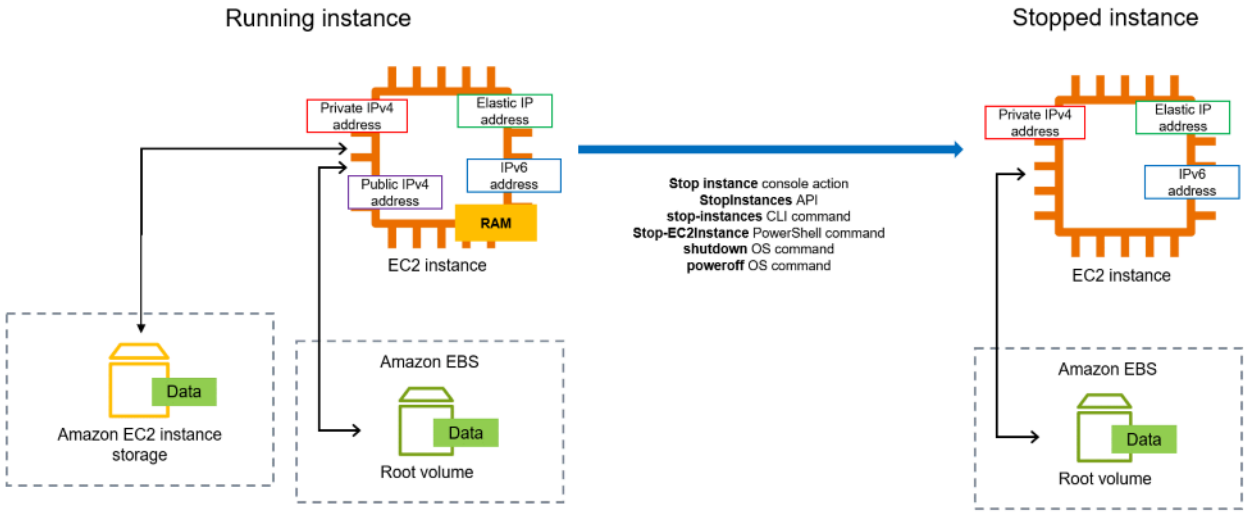
- EC2執行個體如何停止和啟動運作
- 手動停止和啟動執行個體
- 自動停止及啟動執行個體

- 尋找所有執行中和停止的執行個體
- 為EC2執行個體啟用停止保護

EC2執行個體如何停止和啟動運作

當您停止執行個體時，變更會在執行個體的作業系統層級註冊，部分資源會遺失，而且部分資源會持續存在。啟動執行個體時，會在執行個體層級註冊變更。

下圖顯示 Amazon EC2 執行個體停止時遺失的內容，以及持續存在的內容。執行個體停止時，如果彈性 IP 位IPv4址與執行個體沒有關聯，則會遺失任何連接的執行個體儲存磁碟區和儲存在這些磁碟區上的資料RAM、儲存在執行個體上的資料，以及指派的公用位址。執行個體會保留指派的私有IPv4地址、與執行個體關聯的彈性 IP 地址、任何IPv6地址，以及任何連接的 Amazon EBS 磁碟區以及這些磁碟區上的資料。



若您停止執行個體會如何

在作業系統層級註冊的變更

- 要API求會傳送按鈕按下事件給客座端。
- 由於按鈕事件而停止各種系統服務。正常關機是由 Hypervisor 的ACPI關機按鈕按下事件觸發。
- ACPI啟動關閉。
- 當順利關機程序結束時，執行個體就會關機。OS 關機時間無法設定。
- 如果執行個體作業系統在幾分鐘內沒有完全關閉，便會執行硬關機。
- 執行個體會停止執行。
- 執行個體的狀態會變成 `stopping` 然後再變成 `stopped`。

- [Auto Scaling] 如果您的執行個體位於 Auto Scaling 群組中、執行個體處於`running`非 Amazon EC2 狀態或狀態檢查的狀態變成時`impaired`，Amazon EC2 Auto Scaling 會將執行個體視為運作狀態不良並進行取代。如需詳細資訊，請參閱 Amazon Auto Scaling 使用者指南中的自 Au EC2 to Scaling 執行個體運作 Health 態檢查。
- [Windows 執行個體] 停止並啟動 Windows 執行個體時，啟動代理程式會在執行個體上執行任務，例如變更任何連接 Amazon EBS 磁碟區的磁碟機代號。若要取得有關這些預設值以及如何變更它們的更多資訊，請參閱〈〉the section called “EC2Launch v2”。

資源丟失

- 資料儲存在RAM.
- 存放在執行個體儲存體磁碟區的資料。
- Amazon 在啟動或啟動時EC2自動指派給執行個體的公開IPv4地址。若要保留永不變更的公用位IPv4址，您可以將彈性 IP 位址與執行個體建立關聯。

持續存在的資源

- 任何連接的 Amazon EBS 卷。
- 存放在連接的 Amazon EBS 磁碟區上的資料。
- 私人IPv4地址。
- IPv6地址。
- 與執行個體有關聯的彈性 IP 地址。請注意，當執行個體停用時，系統會針對相關聯的彈性 IP 地址向您收費。

如需停止 Mac 執行個體時會發生什麼情況的詳細資訊，請參閱停止或終止您的 Amazon EC2 Mac 實例。

啟動執行個體時會出現的情況

在作業系統層級註冊的變更

- 大多數情況下，執行個體會遷移至新的基礎主機電腦 (不過有時候會保留在目前的主機上，例如當執行個體是配置到專用執行個體組態中的主機時)。
- 如果執行個體設定為接收公共IPv4地址，Amazon EC2 會為執行個體指派新的公共IPv4地址。若要保留永不變更的公用位IPv4址，您可以將彈性 IP 位址與執行個體建立關聯。

測試應用程式對停止和啟動的回應

您可以用 AWS Fault Injection Service 來測試執行個體停止和啟動時應用程式的回應方式。如需詳細資訊，請參閱《AWS Fault Injection Service 使用者指南》https://docs.aws.amazon.com/fis/latest/userguide。

與執行個體停止和啟動相關的成本

停止和啟動執行個體的相關成本如下。

停止 — 一旦執行個體狀態變更為 `shutting-down` 或 `terminated`，執行個體就不會再產生費用。您不需為已停止的執行個體支付使用量或資料傳輸費用。EBS存儲 Amazon 存儲卷產生費用。

啟動 — 每次啟動已停止的執行個體時，我們會收取至少一分鐘的使用費。如果超過一分鐘，只會依使用的秒數收費。例如，如果您執行執行個體 20 秒便停止，我們會收取完整一分鐘的費用。如果您執行執行個體 3 分 40 秒，則只會收取 3 分 40 秒的使用費。

手動停止和啟動執行個體

您可以停止並啟動 Amazon EBS 支援的執行個體 (具有EBS根裝置的執行個體)。您無法使用實例存儲根設備停止和啟動實例。

> **Warning**
>
> 停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。停止執行個體之前，請確認您已將所需的任何資料從執行個體存放磁碟區複製到持續性儲存，例如 Amazon EBS 或 Amazon S3。

Console

停止並啟動 Amazon EBS 支援的執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側導覽窗格中，選擇 [執行個體]，然後選取執行個體。
3. 在 [儲存空間] 索引標籤上，確認根裝置類型為EBS。否則，您無法停止執行個體。
4. 選擇 Instance state (執行個體狀態)、Stop instance (停止執行個體)。如果此選項已停用，則執行個體已停止或其根設備是執行個體存放磁碟區。
5. 出現確認提示時，請選擇 Stop (停止)。停止執行個體可能需要幾分鐘。

6. 若要啟動停止的執行個體，請選取執行個體，然後選取執行個體狀態、啟動執行個體。
7. 執行個體進入 `running` 狀態可能需要數分鐘。
8. 如果您停止了 Amazon EBS 支援的執行個體，且該執行個體在狀態中顯示為「卡住」`stopping` 狀態，您可以強制停止執行該執行個體。如需詳細資訊，請參閱解決 Amazon EC2 執行個體停止問題。

Command line

必要條件

確認執行個體的根裝置是否為EBS磁碟區。例如，執行描述執行個體 AWS CLI 命令，並確認`RootDeviceType`是`ebs`否為。`instance-store`

停止並啟動 Amazon EBS 支援的執行個體

請使用以下其中一個命令：

- AWS CLI—stop-instances 與 start-instances。
- AWS Tools for PowerShell— Stop-EC2Instance和 Start-EC2Instance.
- OS 命令—您可以使用 shutdown 或 poweroff 命令啟動關機作業。使用 OS 命令時，執行個體預設會停止。您可以變更這項預設動作，讓執行個體改為終止。如需詳細資訊，請參閱變更執行個體啟動的關機行為。

[Linux 執行個體] 使用執行個體的 OS halt 指令不會啟動關機。如果您使用該halt命令，實例不會終止; 相反，它放置 inHLT，從而暫停CPU操作。CPU執行個體會維持執行中狀態。

自動停止及啟動執行個體

您可以使用下列服務自動停止及啟動執行個體：

執行個體排程器 AWS

您可以使用「執行個體排程器」 AWS 來自動EC2執行個體的啟動和停止。如需詳細資訊，請參閱如何搭配使用執行個體排程器 CloudFormation 來排程EC2執行個體？ 請注意，這會額外收費。

AWS Lambda 和 Amazon 的 EventBridge 規則

您可以使用 Lambda 和 EventBridge 規則按排程停止和啟動執行個體。如需詳細資訊，請參閱如何使用 Lambda 定期停止和啟動 Amazon EC2 執行個體？

Amazon EC2 Auto Scaling

為確保您擁有正確數量的 Amazon EC2 執行個體可用來處理應用程式的負載，請建立 Auto Scaling 群組。 Amazon EC2 Auto Scaling 確保您的應用程式始終具有處理流量需求的適當容量，並且只在需要執行個體時啟動執行個體來節省成本。請注意， Amazon EC2 Auto Scaling 會終止不需要的執行個體，而不是停止。若要設定 Auto Scaling 群組，請參閱開始使用 Amazon EC2 Auto Scaling。

尋找所有執行中和停止的執行個體

您可以使用 Amazon EC2 全域檢視，在單一頁面 AWS 區域 上找到所有執行中和已停止的執行個體。此功能對清查和查找遺忘的執行個體尤其有用。如需有關如何建立全域檢視的資訊，請參閱 使用 Amazon EC2 全球視圖查看跨區域的資源。

為EC2執行個體啟用停止保護

若要避免執行個體意外停止，您可啟用執行個體的停止保護功能。停止保護還可以保護您的執行個體免遭意外終止。

Amazon 的`DisableApiStop`屬性EC2`ModifyInstanceAttribute`API控制實例是否可以通過使用 Amazon EC2 控制台 AWS CLI，或 Amazon 停止EC2API。您可在執行個體啟動時、運作中或停止時，設定此屬性值。

考量事項

- 啟用停止保護功能無法防止您在使用作業系統命令 (例如 shutdown 或 poweroff) 從執行個體啟動關機作業，而意外停止執行個體。
- 啟用停止防護並不會防 AWS 止在排程事件停止執行個體時停止執行個體。
- 啟用停止保護並不會防止 Amazon EC2 Auto Scaling 在執行個體運作狀態不良或擴充事件期間終止執行個體。您可以使用執行個體停止保護功能，控制 Auto Scaling 群組是否能在縮減時終止特定的執行個體。
- 停止保護不僅可防止執行個體意外停止，還可避免在使用主控台 AWS CLI、或時意外終止API。但是此功能不會自動設定 `DisableApiTermination` 屬性。請注意，當`DisableApiStop`屬性設定為時`false`，`DisableApiTermination`屬性設定會決定是否可以使用主控台 AWS CLI、或終止執行個體API。如需詳細資訊，請參閱 終止 Amazon EC2 實例。
- 您無法為執行個體儲存體所備份的執行個體啟用停止保護。
- 您無法為 Spot 執行個體啟用停止保護。

- 當您啟用或停用停止保護時，Amazon 會EC2API遵循最終一致性模型。這表示執行命令來設定停止保護屬性的結果，可能不會立即顯示在您執行的所有後續命令中。如需詳細資訊，請參閱 Amazon EC2 開發人員指南中的最終一致性。

停止保護任務

- 在執行個體啟動時啟用停止保護
- 針對執作中或已停止的執行個體啟用停止保護
- 針對執作中或已停止的執行個體停用停止保護

在執行個體啟動時啟用停止保護

您可以在執行個體啟動時，使用下列其中一種方法來啟用停止保護。

Console

在執行個體啟動時啟用停止保護

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在儀表板上，選擇啟動執行個體。
3. 在新版啟動執行個體精靈中設定執行個體。
4. 在精靈中，在進階詳細資訊下的停止保護選擇啟用，即可啟用停止保護功能。

AWS CLI

在執行個體啟動時啟用停止保護

使用執行個體指 AWS CLI 令啟動執行個體，並指定參數`disable-api-stop`。

```
aws ec2 run-instances \
    --image-id ami-a1b2c3d4e5example \
    --instance-type t3.micro \
    --key-name MyKeyPair \
    --disable-api-stop \
    ...
```

針對執作中或已停止的執行個體啟用停止保護

您可以在執行個體為執行中或已停止狀態時，使用下列其中一種方法來啟用停止保護。

Console

針對執作中或已停止的執行個體啟用停止保護

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側導覽窗格中，選擇執行個體。
3. 選取執行個體，然後選擇動作、執行個體設定、變更停止保護。
4. 勾選 Enable (啟用) 核取方塊，然後選取 Save (儲存)。

AWS CLI

針對執作中或已停止的執行個體啟用停止保護

使用指modify-instance-attribute AWS CLI 令並指定`disable-api-stop`參數。

```
aws ec2 modify-instance-attribute \
    --instance-id i-1234567890abcdef0 \
    --disable-api-stop
```

針對執作中或已停止的執行個體停用停止保護

您可以使用下列其中一種方法，來對執行中或已停止的執行個體停用停止保護。

Console

針對執作中或已停止的執行個體停用停止保護

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側導覽窗格中，選擇執行個體。
3. 選取執行個體，然後選取 Actions (動作)、Instance Settings (執行個體設定)、Change Stop Protection (變更停止保護)。
4. 清除 Enable (啟用) 核取方塊，然後選擇 Save (儲存)。

AWS CLI

針對執作中或已停止的執行個體停用停止保護

使用指modify-instance-attribute AWS CLI 令並指定no-disable-api-stop參數。

```
aws ec2 modify-instance-attribute \
    --instance-id i-1234567890abcdef0 \
    --no-disable-api-stop
```

休眠您的 Amazon EC2 實例

當您將執行個體休眠時，Amazon 會EC2通知作業系統執行休眠 () suspend-to-disk。休眠功能可將執行個體記憶體 (RAM) 中的內容儲存至 Amazon 彈性區塊存放區 (AmazonEBS) 根磁碟區。Amazon 會保EC2留執行個體的EBS根磁碟區和任何連接的EBS資料磁碟區。啟動執行個體時：

- EBS根磁碟區會還原到先前的狀態
- 內RAM容被重新加載
- 恢復先前在執行個體上執行的程序
- 先前連接的資料磁碟區會重新連接，且執行個體會保留其執行個體 ID

只有當您的執行個體已啟用休眠並符合休眠必要條件，才能讓執行個體進入休眠。

如果執行個體或應用程式需要花費長時間來引導並建置記憶體使用量，以完全發揮功效，則您可以使用休眠來「預先培養」執行個體。若要「預先培養」執行個體，您需要執行下列作業：

1. 在已啟用休眠的情況下將其啟動。
2. 使其進入所需的狀態。
3. 讓其休眠，以便準備好可在需要時恢復為所需狀態。

當休眠的執行個體處於stopped狀態時，執行個體使用量不需支付費用，或在將其內容傳輸至EBS根磁碟區時進行資料傳輸。RAM您需支付任何EBS磁碟區的儲存費用，包括RAM內容的儲存空間。

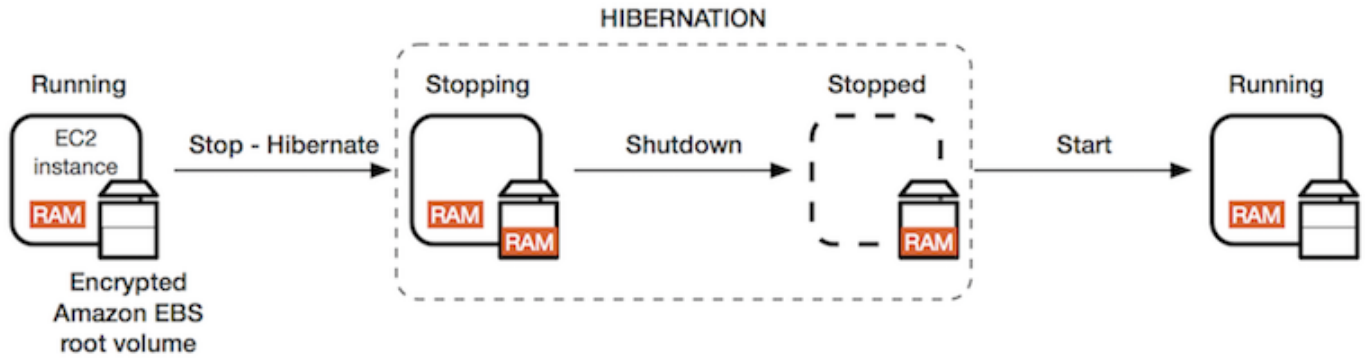
如果您認為不再需要某個執行個體，您可以隨時予以終止，即使該執行個體處於 stopped (休眠) 狀態也可以終止。如需詳細資訊，請參閱終止 Amazon EC2 實例。

目錄

- Amazon EC2 實例休眠的工作原理
- Amazon EC2 執行個體休眠的先決條件
- AMI將 Linux 設定為支援休眠
- 啟用 Amazon EC2 執行個體的休眠功能
- 在執行個體KASLR上停用 (僅限 Ubuntu)
- 休眠 Amazon EC2 實例
- 啟動休眠狀態的 Amazon 執行個體 EC2
- 疑難排解 Amazon EC2 執行個體休眠

Amazon EC2 實例休眠的工作原理

下圖顯示執行個EC2體休眠程序的基本概觀。



當您休眠執行個體時會發生什麼情況

當您休眠執行個體時，會發生下列情況：

- 執行個體會移至 `stopping` 狀態。Amazon EC2 信號操作系統執行休眠（suspend-to-disk）。休眠狀態會凍結所有處理程序，將內容儲存RAM到EBS根磁碟區，然後執行定期關機。
- 完成關機之後，執行個體會進入 `stopped` 狀態。
- 所有EBS磁碟區都會保持連接至執行個體，且其資料仍然存在，包括儲存的內容RAM。
- 任何 Amazon EC2 執行個體存放區磁碟區都會保持連接至執行個體，但執行個體存放區磁碟區上的資料會遺失。
- 當您的執行個體變為 `stopped` 狀態時，您可修改執行個體的部分屬性，包括執行個體類型或大小。
- 在大部分的情況中，執行個體會在啟動時轉移到新的基礎主機電腦。這也是您在停止並啟動執行個體時會發生的情況。

- 啟動執行個體時，執行個體會啟動，作業系統會RAM從EBS根磁碟區讀取的內容，然後解除凍結程序以恢復其狀態。
- 執行個體會保留其私人IPv4位址和任何IPv6位址。執行個體啟動時，執行個體會繼續保留其私有IPv4地址和任何IPv6位址。
- Amazon EC2 發布了公共IPv4地址。啟動執行個體時，Amazon EC2 會為執行個體指派新的公共IPv4地址。
- 執行個體仍會保持與彈性 IP 地址的關聯。我們會針對和已休眠執行個體建立關聯的任何彈性 IP 地址向您收費。

如需休眠與重新開機、停止和終止之間差異的資訊，請參閱實例狀態之間的差異。

限制

- 讓執行個體休眠時，任何執行個體存放區磁碟區上的資料都會遺失。
- (Linux 執行個體) 您無法讓擁有超過 150 GB 的 RAM Linux 執行個體進行休眠。
- (視窗執行個體) 您無法讓擁有超過 16 GB 的 RAM Windows 執行個體休眠。
- 如果您建立快照，或AMI從休眠或啟用休眠的執行個體建立，您可能無法連線至從快照建立的AMI或從快照建立的新執行個體啟動AMI的執行個體。
- (僅限競價型執行個體) 如果 Amazon EC2 將您的競價型執行個體休眠，則只有 Amazon EC2 可以恢復您的執行個體。如果是您讓 Spot 執行個體休眠 (使用者啟動的休眠)，您可以恢復執行個體。只有在有可用的容量，且 Spot 價格低於或等於您指定的最高價格時，才能恢復休眠的 Spot 執行個體。
- 您無法休眠 Auto Scaling 群組中或 Amazon ECS 使用的執行個體。如果您的執行個體位於 Auto Scaling 群組中，而您嘗試將其休眠，Amazon EC2 Auto Scaling 服務會將停止的執行個體標示為運作狀態不良，並可能會終止該執行個體並啟動替換執行個體。如需詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南中的 Auto Scaling 群組中執行個體的運作 Health 態檢查。
- 您無法在啟用「UEFI安全開機」的情況下，將設定為以UEFI模式開機的執行個體進行休眠。
- 如果您休眠啟動到 容量預留 的執行個體，容量預留 不確保休眠執行個體可以在您嘗試啟動之後恢復。
- 如果啟用了聯邦資訊處理標準 (FIPS) 模式，則無法讓使用 5.10 以下核心的執行個體休眠。
- 我們不支援讓執行個體休眠超過 60 天。若要保留執行個體 60 天以上，您必須啟動休眠的執行個體、停止執行個體並將其啟動。
- 我們持續透過升級與安全修補程式來更新我們的平台，可能會與現有的已休眠執行個體衝突。在進行需要啟動已休眠執行個體的重大更新時，我們會通知您，以便我們執行關機或重新啟動來套用必要的升級和安全修補程式。

讓 Spot 執行個體休眠的考量

- 如果「您」讓 Spot 執行個體休眠，只要容量可用且 Spot 價格低於或等於您指定的最高價格，就可以重新啟動該執行個體。
- 如果 Amazon 將您的競價型執行個EC2體休眠：
  - 只有 Amazon 才EC2能恢復您的執行個體。
  - 當容量可用且 Spot 價格小於或等於您指定的最高價格時，Amazon 會EC2恢復休眠的競價型執行個體。
  - Amazon 將 Spot 執行個體EC2休眠之前，您會在休眠開始前兩分鐘收到中斷通知。

  如需詳細資訊，請參閱Spot 執行個體中斷。

Amazon EC2 執行個體休眠的先決條件

您可以在啟動隨需執行個體或 Spot 執行個體時啟用休眠支援。您無法在現有執行個體上啟用休眠功能，無論執行中或已停止。如需詳細資訊，請參閱啟用執行個體休眠。

休眠執行個體的需求

- AWS 區域
- AMIs
- 例證族群
- 實例RAM大小
- 根磁碟區類型
- 根磁碟區大小
- 根磁碟區加密
- EBS磁碟區類型
- Spot 執行個體請求

AWS 區域

您可以在所 AWS 區域有實例中使用休眠。

AMIs

您必須使用支HVMAMI援休眠的。以下AMIs支持休眠：

Linux AMIs

AMIs適用於 Intel 和AMD執行個體類型

- AL2023 年AMI已發佈二零二三年九月二十日或之後
- Amazon Linux 2 AMI 發布 9 月 29 日或更高版本
- Amazon Linux 發布 AMI 2013 年 11 月 16 日或更高版本
- CentOS 第 8 版 AMI [1]（需要額外的配置）
- Fedora 版本 34 或更新版本 AMI [1]（需要額外的配置）
- RHEL (RHEL) 9 AMI [1] (需要額外的設定)
- RHEL (RHEL) 8 AMI [1] (需要額外的組態設定)
- AMI發布序列號 20230 LTS 303 或更高版本 [2]
- Ubuntu 20.04 LTS (焦點弗薩) AMI 發布序列號 20210820 或更高版本 [2]
- Ubuntu 18.04 LTS (仿生海狸) AMI 發布與序列號
- Ubuntu 16.04LTS（異度施洛斯）AMI[2]（需要額外的配置）

AMIs對於重力彈例證類型

- AL2023 AMI (64 位元臂) 已發佈 2024 年 7 月 1 日或更新版本
- Amazon Linux 2AMI（64 位手臂）發布 2024.06.20 或更高版本
- Ubuntu 22.04.2LTS（64 位手臂）（卡米水母）AMI發布了序列號 20240701 或更高版本 [2]
- Ubuntu 20.04LTS（64 位手臂）（焦點支架）AMI發布了序列號 20240701 或更高版本 [2]

[1] 對於 CentOS、Fedora 和 Red Hat Enterprise Linux，只有以 Nitro 為基礎的執行個體才支援休眠。

[2] 我們建議KASLR在使用 Ubuntu 22.04.2LTS（賈米水母），Ubuntu 20.04（焦點弗薩），Ubuntu 18.04LTS（仿生海狸）和 Ubuntu 16.04LTS（異常施留斯）的實例禁用。LTS如需詳細資訊，請參閱在執行個體KASLR上停用 (僅限 Ubuntu)。

[3] 針對 Ubuntu 16.04 LTS (異階版)AMI，執行個體類型不支援休眠。`t3.nano`由於 Ubuntu (Xenial Xerus) 已於 2021 年 4 月終止支援，因此不會提供任何修補程式。如果你想使用`t3.nano`實例類型，我們建議您升級到 Ubuntu 22.04.2LTS（傑米水母），Ubuntu 20.04LTS（焦點弗薩）AMI，或 Ubuntu 18.04（仿生海狸）。LTS AMI

Support Ubuntu 18.04LTS（仿生海狸）和 Ubuntu 16.04LTS（特亞爾施洛斯）已經到了生命的盡頭。

若要將您自己的設定AMI為支援休眠，請參閱AMI將 Linux 設定為支援休眠。

即將推出其他版本的 Ubuntu 及其他作業系統的支援。

視窗 AMIs

- 視窗伺服器 2022 年AMI發佈
- 視窗伺服器 2019 年 9 AMI 月 11 日或更新版本發佈
- 視窗伺服器 2016 AMI 發佈 9 月 11 日或更新版本
- 視窗伺服器 2012 R2 AMI 發佈 11 年 9 月 11 日或更新版本
- 視窗伺服器 2012 AMI 發佈 9 月 11 日或更新版本

例證族群

您必須使用支援休眠的執行個體系列。

- 一般用途:立方米, M4, M5, M5 安培, M5 公升, M5 D, M6 克, M6gD, M6 標識, M7 克, M7 吉米, M7-彈性, T2, T3, T3A, T4G
- 計算優化：C3，C4，C5，C5，C6 克，C6gD，C6 指數，C6 I，C6 標識，C7 克，C7 克，C7gD，C7i，C7 我-柔性
- 記憶體最佳化：R3、R4、R5、R5、R5 阿、R5、R5 D、R6 克、R6gD、R7 克、R7 克、R7 吉、R7 地安
- 儲存最佳化：I3、i3en

Nitro 執行個體 — 不支援裸機執行個體。

查看特定區域中支援休眠的可用執行個體類型

可用的執行個體類型因區域而異。若要查看區域中支援休眠的可用執行個體類型，請搭配`--region`參數使用describe-instance-types命令。包含 `--filters` 參數以將結果範圍限定為支援休眠的執行個體類型，以及包含 `--query` 參數以將輸出範圍限定為 `InstanceType` 的值。

```
aws ec2 describe-instance-types --filters Name=hibernation-supported,Values=true --
query "InstanceTypes[*].[InstanceType]" --output text | sort
```

範例輸出

```
c3.2xlarge
c3.4xlarge
c3.8xlarge
c3.large
c3.xlarge
c4.2xlarge
c4.4xlarge
c4.8xlarge
...
```

實例RAM大小

Linux 執行個體 — 必須小於 150 GB。

視窗執行個體 — 最多可容納 16 GB。若要讓 T3 或 T3A 視窗執行個體休眠，我們建議您至少使用 1 GB 的執行個體。RAM

根磁碟區類型

根磁碟區必須是磁EBS碟區，而不是執行個體儲存磁碟區。

根磁碟區大小

根磁碟區必須足夠大，才能儲存RAM內容並容納您預期的使用情況，例如作業系統或應用程式。如果您啟用休眠，則會在啟動時在根磁碟區上配置空間來儲存. RAM

根磁碟區加密

根磁碟區必須加密，以確保在休眠時保護記憶體中的敏感內容。當RAM資料移至EBS根磁碟區時，資料一律會加密。根磁碟區的加密會在執行個體啟動時執行。

使用下列三個選項之一來確定根磁碟區是加密的EBS磁碟區：

- EBS預設情況下加密 — 您可以預設啟用EBS加密，以確保在您 AWS 帳戶中建立的所有新EBS磁碟區都已加密。以此方式，您可以為您的執行個體啟用休眠，而不在執行個體啟動時指定加密意圖。如需詳細資訊，請參閱預設啟用加密。
- EBS「單一步驟」加密 — 您可以從未加密啟EBS動已加密支援的EC2執行個體，AMI並同時啟用休眠功能。如需詳細資訊，請參閱對 EBS 後端 AMI 使用加密。
- 已加密 AMI — 您可以使用EBS加密來啟動執行個體AMI來啟用加密。如果您AMI沒有加密的根快照，您可以將其複製到新的AMI並請求加密。如需詳細資訊，請參閱 複製期間加密未加密的映像 和 複製 AMI。

EBS磁碟區類型

EBS磁碟區必須使用下列其中一種磁EBS碟區類型：

- 一般用途SSD（gp2和gp3）
- 已佈建 IOPS SSD (io1和io2)

如果您選擇佈建的IOPSSSD磁碟區類型，您必須佈建適當的EBS磁碟區，IOPS以達到最佳的休眠效能。如需詳細資訊，請參閱 Amazon EBS使用者指南中的 Amazon EBS 磁碟區類型。

Spot 執行個體請求

對於 Spot 執行個體，下列需求適用：

- Spot 執行個體請求類型必須為 persistent。
- 您無法在 Spot 執行個體請求中指定啟動群組。

AMI將 Linux 設定為支援休眠

下列 Linux AMIs 可支援 Amazon EC2 執行個體休眠，前提是您完成本節所述的其他設定步驟。

需要設定其他組態：

- Amazon Linux 2 最小AMI版本發布
- 2019.08.29 前發佈的 Amazon Linux 2
- 2018.11.16 前發佈的 Amazon Linux
- CentOS 版本 8 或更新版本
- Fedora 版本 34 或更新版本
- Red Hat Enterprise Linux 版本 8 或 9
- 在序列號 20210 LTS 820 之前發布的 Ubuntu 20.04（焦點福薩）
- Ubuntu 18.04 (Bionic Beaver) 在序號 20190722.1 前發佈的版本
- Ubuntu 16.04 (Xenial Xerus)

如需支援休眠功AMIs能且不需要額外設定的 Linux 和 Windows，請參閱AMIs。

如需詳細資訊，請參閱更新 Amazon Linux 2 執行個體上的執行個體軟體。

Amazon Linux 2 最小AMI版本發布

配置 Amazon Linux 2 最小版本AMI發布 2019.08.29 或更高版本以支持休眠

1. 從儲存庫安裝 `ec2-hibinit-agent` 套件。

```
[ec2-user ~]$ sudo yum install ec2-hibinit-agent
```

2. 重新啟動服務。

```
[ec2-user ~]$ sudo systemctl start hibinit-agent
```

2019.08.29 前發佈的 Amazon Linux 2

若要設定在 2019 年 8 月 29 日之前AMI發行的 Amazon Linux 2 以支援休眠

1. 更新核心至 `4.14.138-114.102` 或更新版本。

```
[ec2-user ~]$ sudo yum update kernel
```

2. 從儲存庫安裝 `ec2-hibinit-agent` 套件。

```
[ec2-user ~]$ sudo yum install ec2-hibinit-agent
```

3. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

4. 確認核心版本已更新為 `4.14.138-114.102` 或更新版本。

```
[ec2-user ~]$ uname -a
```

5. 停止執行個體並建立AMI. 如需詳細資訊，請參閱創建一個 Amazon 支EBS持 AMI。

2018.11.16 前發佈的 Amazon Linux

設定在 2016 年 11 月 16 日之前AMI發行的 Amazon Linux 以支援休眠

1. 更新核心至 `4.14.77-70.59` 或更新版本。

```
[ec2-user ~]$ sudo yum update kernel
```

2. 從儲存庫安裝 `ec2-hibinit-agent` 套件。

```
[ec2-user ~]$ sudo yum install ec2-hibinit-agent
```

3. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

4. 確認核心版本已更新為 `4.14.77-70.59` 或更高版本。

```
[ec2-user ~]$ uname -a
```

5. 停止執行個體並建立AMI. 如需詳細資訊，請參閱創建一個 Amazon 支EBS持 AMI。

CentOS 版本 8 或更新版本

設定 CentOS 版本 8 或更新版本AMI以支援休眠

1. 更新核心至 `4.18.0-305.7.1.el8_4.x86_64` 或更新版本。

```
[ec2-user ~]$ sudo yum update kernel
```

2. 安裝適用於企業 Linux 的 Fedora 額外套件 (EPEL) 軟體庫。

```
[ec2-user ~]$ sudo yum install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
```

3. 從儲存庫安裝 `ec2-hibinit-agent` 套件。

```
[ec2-user ~]$ sudo yum install ec2-hibinit-agent
```

4. 啟用休眠代理程式以在開機時開始。

```
[ec2-user ~]$ sudo systemctl enable hibinit-agent.service
```

5. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

6. 確認核心版本已更新為 `4.18.0-305.7.1.el8_4.x86_64` 或更新版本。

```
[ec2-user ~]$ uname -a
```

Fedora 版本 34 或更新版本

若要設定 Fedora 第 34 版或更新版本AMI以支援休眠

1. 更新核心至 `5.12.10-300.fc34.x86_64` 或更新版本。

```
[ec2-user ~]$ sudo yum update kernel
```

2. 從儲存庫安裝 `ec2-hibinit-agent` 套件。

```
[ec2-user ~]$ sudo dnf install ec2-hibinit-agent
```

3. 啟用休眠代理程式以在開機時開始。

```
[ec2-user ~]$ sudo systemctl enable hibinit-agent.service
```

4. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

5. 確認核心版本已更新為 `5.12.10-300.fc34.x86_64` 或更新版本。

```
[ec2-user ~]$ uname -a
```

Red Hat Enterprise Linux 版本 8 或 9

若要設定 RHEL 8 或 9 AMI 來支援休眠

1. 更新核心至 `4.18.0-305.7.1.el8_4.x86_64` 或更新版本。

```
[ec2-user ~]$ sudo yum update kernel
```

2. 安裝適用於企業 Linux 的 Fedora 額外套件 (EPEL) 軟體庫。

   RHEL版本八：

```
[ec2-user ~]$ sudo yum install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
```

RHEL版本九：

```
[ec2-user ~]$ sudo yum install https://dl.fedoraproject.org/pub/epel/epel-release-latest-9.noarch.rpm
```

3. 從儲存庫安裝 `ec2-hibinit-agent` 套件。

```
[ec2-user ~]$ sudo yum install ec2-hibinit-agent
```

4. 啟用休眠代理程式以在開機時開始。

```
[ec2-user ~]$ sudo systemctl enable hibinit-agent.service
```

5. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

6. 確認核心版本已更新為 `4.18.0-305.7.1.el8_4.x86_64` 或更新版本。

```
[ec2-user ~]$ uname -a
```

在序列號 20210 LTS 820 之前發布的 Ubuntu 20.04（焦點福薩）

若要設定序號 20210820 之前AMI發行的 Ubuntu 20.04 LTS (焦點位置)，以支援休眠

1. 將 grub2 更新 linux-aws-kernel 至`5.8.0-1038.40`或更新版本，並將 grub2 更新至`2.04-1ubuntu26.13`或更新版本。

```
[ec2-user ~]$ sudo apt update
[ec2-user ~]$ sudo apt dist-upgrade
```

2. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

3. 確認核心版本已更新為 `5.8.0-1038.40` 或更新版本。

```
[ec2-user ~]$ uname -a
```

4. 確認 grub2 版本已更新為 `2.04-1ubuntu26.13` 或更新版本。

```
[ec2-user ~]$ dpkg --list | grep grub2-common
```

Ubuntu 18.04 (Bionic Beaver) 在序號 20190722.1 前發佈的版本

若要設定序號之前LTSAMI發行的 Ubuntu 18.04，以支援休眠

1. 更新核心至 `4.15.0-1044` 或更新版本。

```
[ec2-user ~]$ sudo apt update
[ec2-user ~]$ sudo apt dist-upgrade
```

2. 從儲存庫安裝 `ec2-hibinit-agent` 套件。

```
[ec2-user ~]$ sudo apt install ec2-hibinit-agent
```

3. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

4. 確認核心版本已更新為 `4.15.0-1044` 或更新版本。

```
[ec2-user ~]$ uname -a
```

Ubuntu 16.04 (Xenial Xerus)

若要設定 Ubuntu 16.04 LTS 以支援休眠功能，您必須安裝linux-aws-hwe 核心套件 4.15.0-1058 或更新版本，以及 ec2-劫持代理程式。

> **Important**
>
> Canonical 支援 `linux-aws-hwe` 核心套件。對於 Ubuntu 16.04 的標準支持將於 2021 年 4 月LTS結束，該軟件包不再收到定期更新。但是，此套件會收到其他安全性更新，直到 2024 年結束 Extended Security Maintenance 支援為止。有關更多信息，請參閱 Ubuntu 16.04 的 Amazon EC2 休眠LTS現在可在規範 Ubuntu 博客上使用。

我們建議您升級到 Ubuntu 20.04LTS（焦點弗薩）AMI或 Ubuntu 18.04LTS（仿生海狸）。AMI

若要設定 Ubuntu 16.04 LTS AMI 以支援休眠

1. 更新核心至 `4.15.0-1058-aws` 或更新版本。

```
[ec2-user ~]$ sudo apt update
[ec2-user ~]$ sudo apt install linux-aws-hwe
```

2. 從儲存庫安裝 `ec2-hibinit-agent` 套件。

```
[ec2-user ~]$ sudo apt install ec2-hibinit-agent
```

3. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

4. 確認核心版本已更新為 `4.15.0-1058-aws` 或更新版本。

```
[ec2-user ~]$ uname -a
```

啟用 Amazon EC2 執行個體的休眠功能

若要讓執行個體休眠，必須在啟動執行個體時先為其啟用休眠。

> **Important**
> 啟動執行個體之後，就不能對它啟用或停用休眠。

主題

- 為隨需執行個體啟用休眠功能
- 為 Spot 執行個體啟用休眠功能
- 檢視執行個體是否已啟用休眠

為隨需執行個體啟用休眠功能

使用下列任一種方法為隨需執行個體啟用休眠功能。

Console

為隨需執行個體啟用休眠功能

1. 遵循程序啟動執行個體，但在完成下列步驟以啟用休眠之前，請勿啟動執行個體。
2. 若要啟用休眠功能，請在啟動執行個體精靈中設定下列欄位：

   a. 在應用程式和作業系統映像 (Amazon 機器映像) 下，選取AMI支援休眠的。如需詳細資訊，請參閱AMIs。

   b. 在 Instance type (執行個體類型) 中，選取支援的執行個體類型。如需詳細資訊，請參閱例證族群。

   c. 在 Configure storage (設定儲存) 中，選擇 Advanced (進階) (在右側)，並指定根磁碟區的下列資訊：

      - 在大小 (GiB) 中，輸入EBS根磁碟區大小。體積必須足夠大，才能存儲RAM內容並適應您的預期使用情況。
      - 針對磁碟區類型，選取支援的EBS磁碟區類型：一般用途 SSD (gp2和gp3) 或已佈建 IOPS SSD (io1和io2)。
      - 針對 Encrypted (加密)，選擇 Yes (是)。如果您在此 AWS 區域中預設啟用加密，則會選取 [是]。
      - 對於KMS金鑰，請選取磁碟區的加密金鑰。如果您在此 AWS 區域中預設啟用加密，則會選取預設加密金鑰。

      如需根磁碟區必要條件的詳細資訊，請參閱Amazon EC2 執行個體休眠的先決條件。

   d. 展開 Advanced details (進階詳細資訊)，針對 Stop - Hibernate behavior (停止 - 休眠行為)，選擇 Enable (啟用)。
3. 在 Summary (摘要) 面板中，檢閱您的執行個體組態，然後選擇 Launch instance (啟動執行個體)。如需詳細資訊，請參閱使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體。

AWS CLI

為隨需執行個體啟用休眠功能

使用 run-instances 命令來啟動執行個體。使用參數指定EBS根磁碟區參`--block-device-mappings file://mapping.json`數，並使用參`--hibernation-options Configured=true`數啟用休眠。

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --instance-type m5.large \
    --block-device-mappings file://mapping.json \
    --hibernation-options Configured=true \
    --count 1 \
    --key-name MyKeyPair
```

在 `mapping.json` 中指定下列內容。

```
[
    {
        "DeviceName": "/dev/xvda",
        "Ebs": {
            "VolumeSize": 30,
            "VolumeType": "gp2",
            "Encrypted": true
        }
    }
]
```

> **Note**
>
> 的值`DeviceName`必須與相關聯的根裝置名稱相符AMI。若要尋找根設備名稱，請使用 describe-images 命令。
>
> ```
> aws ec2 describe-images --image-id ami-0abcdef1234567890
> ```
>
> 如果您在此 AWS 區域中預設啟用加密功能，則可以省略`"Encrypted": true`。

PowerShell

若要啟用隨需執行個體的休眠，請使用 AWS Tools for Windows PowerShell

使用指New-EC2Instance令啟動執行個體。首先定義區塊裝置對映，然後使用-BlockDeviceMappings參數將其新增至指令，以指定EBS根磁碟區。使用 -HibernationOptions_Configured $true 參數啟用休眠。

```
PS C:\> $ebs_encrypt = New-Object Amazon.EC2.Model.BlockDeviceMapping
PS C:\> $ebs_encrypt.DeviceName = "/dev/xvda"
PS C:\> $ebs_encrypt.Ebs = New-Object Amazon.EC2.Model.EbsBlockDevice
PS C:\> $ebs_encrypt.Ebs.VolumeSize = 30
PS C:\> $ebs_encrypt.Ebs.VolumeType = "gp2"
PS C:\> $ebs_encrypt.Ebs.Encrypted = $true

PS C:\> New-EC2Instance `
             -ImageId ami-0abcdef1234567890 `
             -InstanceType m5.large `
             -BlockDeviceMappings $ebs_encrypt `
             -HibernationOptions_Configured $true `
             -MinCount 1 `
             -MaxCount 1 `
             -KeyName MyKeyPair
```

> **Note**
>
> 的值DeviceName必須符合與相關聯的根裝置名稱AMI。若要尋找根裝置名稱，請使用Get-EC2Image指令。
>
> ```
> Get-EC2Image -ImageId ami-0abcdef1234567890
> ```
>
> 如果您在此 AWS 區域中預設啟用加密功能，則可以Encrypted = $true從區塊裝置對應中省略。

為 Spot 執行個體啟用休眠功能

使用下列任一種方法為 Spot 執行個體啟用休眠功能。如需中斷時讓 Spot 執行個體休眠的詳細資訊，請參閱 Spot 執行個體中斷。

Console

您可以使用 Amazon EC2 主控台中的啟動執行個體精靈，為 Spot 執行個體啟用休眠。

為 Spot 執行個體啟用休眠功能

1. 遵循程序使用啟動執行個體精靈要求 Spot 執行個體，但在完成下列步驟以啟用休眠之前，請勿啟動執行個體。
2. 若要啟用休眠功能，請在啟動執行個體精靈中設定下列欄位：
   a. 在應用程式和作業系統映像 (Amazon 機器映像) 下，選取AMI支援休眠的。如需詳細資訊，請參閱AMIs。
   b. 在 Instance type (執行個體類型) 中，選取支援的執行個體類型。如需詳細資訊，請參閱例證族群。
   c. 在 Configure storage (設定儲存) 中，選擇 Advanced (進階) (在右側)，並指定根磁碟區的下列資訊：
      - 在大小 (GiB) 中，輸入EBS根磁碟區大小。體積必須足夠大，才能存儲RAM內容並適應您的預期使用情況。
      - 針對磁碟區類型，選取支援的EBS磁碟區類型：一般用途 SSD (gp2和gp3) 或已佈建 IOPS SSD (io1和io2)。
      - 針對 Encrypted (加密)，選擇 Yes (是)。如果您在此 AWS 區域中預設啟用加密，則會選取 [是]。
      - 對於KMS金鑰，請選取磁碟區的加密金鑰。如果您在此 AWS 區域中預設啟用加密，則會選取預設加密金鑰。

      如需根磁碟區必要條件的詳細資訊，請參閱Amazon EC2 執行個體休眠的先決條件。
   d. 展開進階詳細資訊，除了用於設定 Spot 執行個體的欄位之外，請執行下列操作：
      i. 對於要求類型，請選擇持續。
      ii. 對於中斷行為，請選擇休眠。或者，對於停止 - 休眠行為，請選擇啟用。這兩個欄位都能在 Spot 執行個體上啟用休眠。您只須設定其中之一。
3. 在 Summary (摘要) 面板中，檢閱您的執行個體組態，然後選擇 Launch instance (啟動執行個體)。如需詳細資訊，請參閱使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體。

AWS CLI

您可以使用 run-instances AWS CLI 命令，為 Spot 執行個體啟用休眠功能。

使用 **hibernation-options** 參數為 Spot 執行個體啟用休眠功能

使用 run-instances 命令要求 Spot 執行個體。使用參數指定EBS根磁碟區參`--block-device-mappings file://mapping.json`數，並使用參`--hibernation-options Configured=true`數啟用休眠。Spot 要求類型 (`SpotInstanceType`) 必須是 `persistent`。

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --instance-type c4.xlarge \
    --block-device-mappings file://mapping.json \
    --hibernation-options Configured=true \
    --count 1 \
    --key-name MyKeyPair
    --instance-market-options
        {
           "MarketType":"spot",
           "SpotOptions":{
              "MaxPrice":"1",
              "SpotInstanceType":"persistent"
            }
        }
```

在中指定EBS根磁碟區參數`mapping.json`，如下所示。

```
[
    {
        "DeviceName": "/dev/xvda",
        "Ebs": {
            "VolumeSize": 30,
            "VolumeType": "gp2",
            "Encrypted": true
        }
    }
]
```

> **Note**
>
> 的值`DeviceName`必須與相關聯的根裝置名稱相符AMI。若要尋找根設備名稱，請使用 describe-images 命令。
>
> ```
> aws ec2 describe-images --image-id ami-0abcdef1234567890
> ```

如果您在此 AWS 區域中預設啟用加密功能，則可以省略`"Encrypted": true`。

PowerShell

若要啟用 Spot 執行個體的休眠，請使用 AWS Tools for Windows PowerShell

使用命New-EC2Instance令要求 Spot 執行個體。首先定義區塊裝置對映，然後使用`-BlockDeviceMappings`參數將其新增至指令，以指定EBS根磁碟區。使用 `-HibernationOptions_Configured $true` 參數啟用休眠。

```
PS C:\> $ebs_encrypt = New-Object Amazon.EC2.Model.BlockDeviceMapping
PS C:\> $ebs_encrypt.DeviceName = "/dev/xvda"
PS C:\> $ebs_encrypt.Ebs = New-Object Amazon.EC2.Model.EbsBlockDevice
PS C:\> $ebs_encrypt.Ebs.VolumeSize = 30
PS C:\> $ebs_encrypt.Ebs.VolumeType = "gp2"
PS C:\> $ebs_encrypt.Ebs.Encrypted = $true

PS C:\> New-EC2Instance `
             -ImageId ami-0abcdef1234567890 `
             -InstanceType m5.large `
             -BlockDeviceMappings $ebs_encrypt `
             -HibernationOptions_Configured $true `
             -MinCount 1 `
             -MaxCount 1 `
             -KeyName MyKeyPair `
             -InstanceMarketOption @(
                   MarketType = spot;
                   SpotOptions @{
                   MaxPrice = 1;
                   SpotInstanceType = persistent}
                )
```

Note

的值`DeviceName`必須符合與相關聯的根裝置名稱AMI。若要尋找根裝置名稱，請使用Get-EC2Image指令。

```
Get-EC2Image -ImageId ami-0abcdef1234567890
```

如果您在此 AWS 區域中預設啟用加密功能，則可以`Encrypted = $true`從區塊裝置對應中省略。

檢視執行個體是否已啟用休眠

透過下列說明來檢視執行個體是否已啟用休眠。

Console

檢視執行個體是否已啟用休眠

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後在 Instance details (執行個體詳細資訊) 區段的 Details (詳細資料) 標籤上，檢查 Stop-hibernate behavior (停止休眠行為)。Enabled (啟用) 表示執行個體已啟用休眠。

AWS CLI

檢視執行個體是否已啟用休眠

使用 describe-instances 命令，並指定 `--filters "Name=hibernation-options.configured,Values=true"` 參數來篩選已啟用休眠的執行個體。

```
aws ec2 describe-instances \
    --filters "Name=hibernation-options.configured,Values=true"
```

輸出中的下列欄位表示執行個體已啟用休眠。

```
"HibernationOptions": {
    "Configured": true
}
```

PowerShell

使用 AWS Tools for Windows PowerShell來檢視執行個體是否已啟用休眠

使用命Get-EC2Instance令並指定`-Filter @{ Name="hibernation-options.configured"; Value="true"}`參數來篩選啟用休眠的執行個體。

```
(Get-EC2Instance -Filter @{Name="hibernation-options.configured";
 Value="true"}).Instances
```

輸出會列出已啟用休眠的EC2執行個體。

在執行個體KASLR上停用 (僅限 Ubuntu)

要在使用 Ubuntu 16.04LTS（Xerus），Ubuntu 18.04（仿生海狸）發布的序列號 20190722.1 或更高版本，或 Ubuntu 20.04LTS（焦點弗薩）上運行休眠，序列號 20210820 或更高版本發布，我們建議禁用LTS（內核地址空間佈局隨機化）。KASLR在 Ubuntu 16.04LTS，Ubuntu 18.04 或 Ubuntu 20.04 LTSLTS，默認情況下處KASLR於啟用狀態。

KASLR是一項標準的 Linux 核心安全性功能，藉由隨機化核心的基本位址值，有助於減少尚未發現的記憶體存取弱點所遭受的風險和後果。KASLR啟用後，執行個體在休眠後可能無法繼續執行。

若要深入瞭解KASLR，請參閱 Ubuntu 功能。

若要KASLR在使用 Ubuntu 啟動的執行個體上停用

1. 使用 Connect 至您的執行個體SSH。如需詳細資訊，請參閱使用 Connect 到您的 Linux 執行個體SSH。
2. 在您選擇的編輯器中開啟 `/etc/default/grub.d/50-cloudimg-settings.cfg` 檔案。編輯 `GRUB_CMDLINE_LINUX_DEFAULT` 行，在行尾附加 `nokaslr` 選項，如下列範例所示。

   ```
   GRUB_CMDLINE_LINUX_DEFAULT="console=tty1 console=ttyS0
    nvme_core.io_timeout=4294967295 nokaslr"
   ```

3. 儲存檔案並結束您的編輯器。
4. 執行下列命令來重建 grub 組態。

   ```
   sudo update-grub
   ```

5. 重新啟動執行個體。

   ```
   sudo reboot
   ```

6. 執行以下命令來確認已新增 `nokaslr`。

   ```
   cat /proc/cmdline
   ```

命令的輸出應該會包含 `nokaslr` 選項。

休眠 Amazon EC2 實例

如果執行個體是EBS備份的執行個體、啟用休眠功能並符合休眠先決條件，您可以在隨需執行個體或競價型執行個體上啟動休眠。如果無法讓執行個體進入休眠，則會以一般方式關機。

Console

讓執行個體休眠

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後選取 Instance state (執行個體狀態)、Hibernate instance (休眠執行個體)。如果停用了 Hibernate instance (休眠執行個體)，則已進入休眠或已停止的執行個體將無法進入休眠。如需詳細資訊，請參閱Amazon EC2 執行個體休眠的先決條件。
4. 出現確認提示時，請選擇 Hibernate (休眠)。讓執行個體進入休眠可能需要幾分鐘。執行個體狀態會先變更為 Stopping (正在停止)，然後在執行個體已進入休眠時變更為 Stopped (已停止)。

AWS CLI

若要讓備份的執行個EBS體休眠

使用 stop-instances 命令，並指定 `--hibernate` 參數。

```
aws ec2 stop-instances \
    --instance-ids i-1234567890abcdef0 \
    --hibernate
```

PowerShell

若要使執行個體休眠，請使用 AWS Tools for Windows PowerShell

使用指Stop-EC2Instance令並指定`-Hibernate $true`參數。

```
Stop-EC2Instance `
```

```
    -InstanceId i-1234567890abcdef0 `
    -Hibernate $true
```

Console

檢視執行個體是否已啟動休眠

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後在詳細資訊索引標籤的執行個體詳細資訊區段中，查看狀態轉換訊息的值。

   用戶端。 UserInitiatedHibernate：使用者啟動的休眠表示您已在隨需執行個體或 Spot 執行個體上啟動休眠。

AWS CLI

檢視執行個體是否已啟動休眠

使用 describe-instances 命令並指定 `state-reason-code` 篩選條件，以查看已初始化休眠的執行個體。

```
aws ec2 describe-instances \
    --filters "Name=state-reason-code,Values=Client.UserInitiatedHibernate"
```

輸出中的下列欄位表示隨需執行個體或 Spot 執行個體已啟動休眠。

```
"StateReason": {
    "Code": "Client.UserInitiatedHibernate"
}
```

PowerShell

使用 AWS Tools for Windows PowerShell來檢視執行個體是否已初始化休眠

使用命Get-EC2Instance令並指定`state-reason-code`篩選器，以查看起始休眠的執行個體。

```
Get-EC2Instance `
    -Filter @{Name="state-reason-code";Value="Client.UserInitiatedHibernate"}
```

輸出會列出啟動休眠的EC2執行個體。

啟動休眠狀態的 Amazon 執行個體 EC2

透過與啟動已停止執行個體的相同方式，來啟動休眠的執行個體。

> **Note**
>
> 對於競價型執行個體，如果 Amazon 將執行個EC2體休眠，則只有 Amazon 才EC2能恢復執行個體。只有是「您」讓 Spot 執行個體休眠時，您才能恢復該執行個體。只有在容量可用，且 Spot 價格低於或等於您指定的最高價格時，才能恢復 Spot 執行個體。

Console

啟動休眠的執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取休眠的執行個體，並選取 Instance state (執行個體狀態)、Start instance (啟動執行個體)。執行個體進入 `running` 狀態可能需要數分鐘。在此期間內，執行個體的狀態檢查會顯示執行個體處於失敗狀態，直到執行個體啟動。

AWS CLI

啟動休眠的執行個體

使用 start-instances 命令。

```
aws ec2 start-instances \
    --instance-ids i-1234567890abcdef0
```

PowerShell

使用啟動休眠的執行個體 AWS Tools for Windows PowerShell

使用指Start-EC2Instance令。

```
Start-EC2Instance `
    -InstanceId i-1234567890abcdef0
```

疑難排解 Amazon EC2 執行個體休眠

使用此資訊來協助您診斷並修正讓執行個體進入休眠時可能遇到的問題。

休眠問題

- 無法在啟動後立即進入休眠
- 從 stopping 轉換為 stopped 花費太長時間，且記憶體狀態未在啟動後復原
- 執行個體「卡」在stopping 狀態。
- 休眠後無法立即啟動 Spot 執行個體
- 繼續執行 Spot 執行個體失敗

無法在啟動後立即進入休眠

如果您在啟動執行個體後太快嘗試讓執行個體進入休眠，則會出現錯誤。

Linux 執行個體必須等待大約兩分鐘，啟動後 Windows 執行個體必須等待約五分鐘，然後才能休眠。

從 stopping 轉換為 stopped 花費太長時間，且記憶體狀態未在啟動後復原

如果休眠的執行個體花費長時間才能從 `stopping` 狀態轉換為 `stopped`，且如果記憶體狀態未在啟動後復原，則可能表示休眠未正確設定。

Linux 執行個體

請檢查執行個體系統日誌，尋找與休眠相關的訊息。若要存取系統記錄檔，請連線至執行個體或使用get-console-output指令。從 `hibinit-agent` 尋找日誌行。如果日誌行指出失敗或日誌行不存在，則很可能在啟動時有錯誤的休眠設定。

例如，下列訊息表示執行個體根磁碟區不夠大：`hibinit-agent: Insufficient disk space. Cannot create setup for hibernation. Please allocate a larger root device.`

如果 `hibinit-agent` 最後的日誌行是 `hibinit-agent: Running: swapoff /swap`，則表示休眠已成功設定。

如果您沒有看到這些程序的任何記錄，表示您AMI可能不支援休眠。如需有關支援的 AMIs 的資訊，請參閱Amazon EC2 執行個體休眠的先決條件。如果您使用自己的 LinuxAMI，請確定您遵循的指示AMI將 Linux 設定為支援休眠。

Windows Server 2016 及更新版本

檢查EC2啟動日誌並查找與休眠相關的消息。若要存取EC2啟動記錄，請連線至執行個體，然後在文字編輯器中開啟`C:\ProgramData\Amazon\EC2-Windows\Launch\Log\Ec2Launch.log`檔案。如果您使用 EC2Launch v2，請開啟`C:\ProgramData\Amazon\EC2Launch\log\agent.log`。

> **Note**
>
> 根據預設，Windows 會將檔案和資料夾隱藏在 `C:\ProgramData` 下。若要檢視EC2啟動目錄和檔案，請在 Windows 檔案總管中輸入路徑，或變更資料夾內容以顯示隱藏的檔案和資料夾。

尋找休眠的日誌行。如果日誌行指出失敗或日誌行不存在，則很可能在啟動時有錯誤的休眠設定。

例如，下列訊息表示無法設定休眠：`Message: Failed to enable hibernation.`如果錯誤訊息包含十進位ASCII值，您可以將ASCII值轉換為純文字，以便讀取完整的錯誤訊息。

如果日誌行包含 `HibernationEnabled: true`，則表示休眠已成功設定。

Windows Server 2012 R2 和舊版

檢查EC2配置日誌並查找與休眠相關的消息。若要存取EC2設定記錄，請連線至執行個體，然後在文字編輯器中開啟`C:\Program Files\Amazon\Ec2ConfigService\Logs\Ec2ConfigLog.txt`檔案。尋找 `SetHibernateOnSleep` 的日誌行。如果日誌行指出失敗或日誌行不存在，則很可能在啟動時有錯誤的休眠設定。

例如，下列訊息表示執行個體根磁碟區不夠大：`SetHibernateOnSleep: Failed to enable hibernation: Hibernation failed with the following error: There is not enough space on the disk.`

如果日誌行是 `SetHibernateOnSleep: HibernationEnabled: true`，則表示休眠已成功設定。

視窗實體大小

如果您使用的是小於 1 GB 的 T3 或 T3A Windows 執行個體RAM，請嘗試將執行個體的大小增加為至少有 1 GB 的執行個體。RAM

執行個體「卡」在stopping 狀態。

如果您讓執行個體進入休眠，但該執行個體「卡」在stopping 狀態，您可以強制將其停止。如需詳細資訊，請參閱解決 Amazon EC2 執行個體停止問題。

休眠後無法立即啟動 Spot 執行個體

如果您嘗試在休眠後兩分鐘內啟動 Spot 執行個體，可能會出現下列錯誤：

```
You failed to start the Spot Instance because the associated Spot Instance
request is not in an appropriate state to support start.
```

請等待 Linux 執行個體大約兩分鐘，Windows 執行個體約 5 分鐘，然後重試啟動執行個體。

繼續執行 Spot 執行個體失敗

如果您的 Spot 執行個體已成功休眠，但是無法繼續執行，而是會重新啟動 (未保留休眠狀態而直接重新啟動)，可能是因為使用者資料包含下列指令碼：

```
/usr/bin/enable-ec2-spot-hibernation
```

從啟動範本的使用者資料欄位中移除此指令碼，然後請求新的 Spot 執行個體。

請注意，即使執行個體無法繼續執行，在不保留休眠狀態的狀態下，仍可使用與從 stopped 狀態啟動的相同方式啟動執行個體。

重新啟動您的執行個體

重新啟動執行個體等同於重新啟動作業系統。在大多數情況下，將執行個體重新開機只需要幾分鐘的時間。

重新啟動執行個體時，其會保留下列項目：

- 公共DNS名稱（IPv4）
- 私人IPv4地址
- 公共IPv4地址
- IPv6地址 (如適用)
- 其執行個體儲存體磁碟區上的任何資料

有別於停用和啟動執行個體，將執行個體重新開機不會啟動新的執行個體計費期間 (最少收費一分鐘)。

我們可能會因必要的維護而排程重新開機執行個體 (例如，套用更新後需要重新開機)。您不需要採取任何動作；建議您於排程的時間範圍內等待其重新開機。如需詳細資訊，請參閱 Amazon EC2 執行個體的排程事件。

建議您使用 Amazon EC2 主控台、命令列工具或 Amazon EC2 API 重新啟動執行個體，而不是從執行個體執行作業系統重新開機命令。如果您使用 Amazon EC2 主控台、命令列工具或 Amazon EC2 API 重新啟動執行個體，如果執行個體未在幾分鐘內完全關閉，我們會執行硬重新開機。如果您使用 AWS CloudTrail，則使用 Amazon EC2 重新啟動執行個體也會建立執行個體重新開機的API記錄。

Windows 執行個體

如果 Windows 正在執行個體上安裝更新，建議您在安裝所有更新之前，不要使用 Amazon EC2 主控台或命令列重新啟動或關閉執行個體。當您使用 Amazon EC2 主控台或命令列重新啟動或關閉執行個體時，可能會導致執行個體重新啟動。正在安裝更新時的硬式重新開機可能會讓執行個體進入不穩定狀態。

Console

使用主控台重新開機執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇執行個體。
3. 選取執行個體，然後選取 Instance state (執行個體狀態)、Reboot instance (將執行個體重新開機)。

   或者，選取執行個體，然後選取 Actions (動作)、Manage instance state (管理執行個體狀態)。在開啟的畫面中，選擇 Reboot (重新開機)，然後選擇 Change state (變更狀態)。
4. 出現確認提示時，選擇重新開機。

   執行個體會保持在 `running` 狀態。

Command line

重新啟動執行個體

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱訪問 Amazon EC2。

- reboot-instances (AWS CLI)
- Restart-EC2Instance (AWS Tools for Windows PowerShell)

若要執行受控的錯誤注入實驗

您可以用 AWS Fault Injection Service 來測試執行個體重新啟動時應用程式的回應方式。如需詳細資訊，請參閱《AWS Fault Injection Service 使用者指南》。

終止 Amazon EC2 實例

當您不再需要執行個體時，可以將它刪除。這就是終止您的執行個體。執行個體的狀態變更為 `shutting-down` 或 `terminated` 時刻起，該執行個體便停止收取費用。

執行個體終止後，您將無法連接或啟動該執行個體。但是，您可以使用相同的方法啟動其他執行個體AMI。如果您想要停止或休眠執行個體，請參閱停止並啟動 Amazon EC2 執行個體或休眠您的 Amazon EC2 實例。如需詳細資訊，請參閱實例狀態之間的差異。

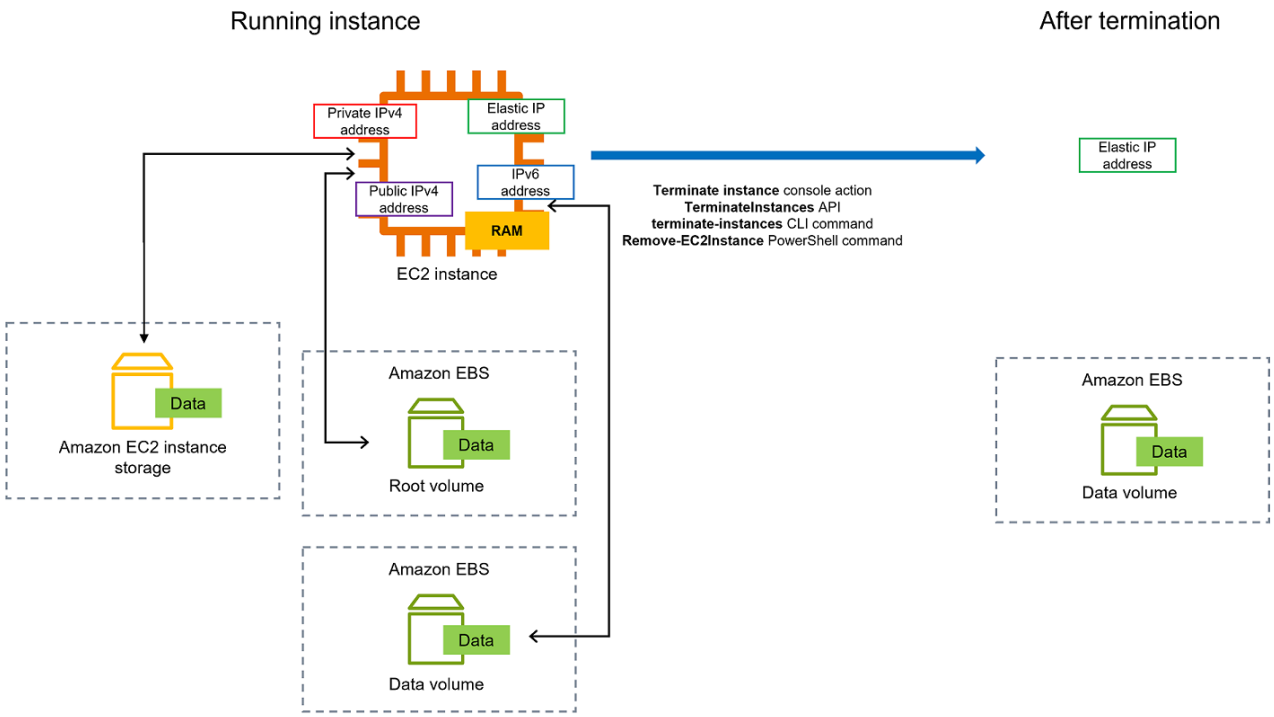
目錄

- 實例終止的工作原理
- 終止執行個體
- 對執行個體終止進行故障診斷
- 啟用終止保護
- 變更執行個體啟動的關機行為
- 在執行個體終止時保留資料

實例終止的工作原理

當您終止執行個體時，變更會在執行個體的作業系統層級註冊，部分資源會遺失，而且部分資源會持續存在。

下圖顯示遺失的內容，以及 Amazon EC2 執行個體終止時會持續存在的內容。執行個體終止時，會清除任何執行個體儲存磁碟區上的資料以及儲存該執行RAM個體的資料。與執行個體相關聯的任何彈性 IP 位址都會中斷連線。對於 Amazon EBS 磁碟區和這些磁碟區上的資料，結果取決於磁碟區的終止時刪除設定。根據預設，會刪除根磁碟區並保留資料磁碟區。



考量事項

- 執行個體終止時，與該執行個體建立關聯的執行個體存放區磁碟區上之資料，將被刪除。
- 依預設，Amazon EBS 根裝置磁碟區會在執行個體終止時自動刪除。但是，即使執行個體終止後，您在啟動時連接的任何其他EBSEBS磁碟區，或連接至現有執行個體的任何磁碟區仍會保留。如需詳細資訊，請參閱在執行個體終止時保留資料。

> **Note**
>
> 執行個體終止時未刪除的任何磁碟區將繼續產生費用。

- 若要防止執行個體遭人意外終止，請啟用終止保護。
- 若要控制執行個體在執行個體啟動關機時是否停止或終止，請變更執行個體起始的關機行為。
- 如果您在執行個體終止時執行指令碼，則執行個體可能會發生異常終止，因為我們無法確保關閉指令碼執行。Amazon EC2 嘗試徹底關閉執行個體並執行任何系統關機指令碼；但是，某些事件 (例如硬體故障) 可能會導致這些系統關閉指令碼無法執行。
- x86 裸機執行個體不支援協同關機。

若您終止執行個體會如何

在作業系統層級註冊的變更

- 要API求會傳送按鈕按下事件給客座端。
- 由於按鈕事件而停止各種系統服務。系統的正常關機由systemd（Linux）或系統進程（視窗）提供。正常關機是由 Hypervisor 的ACPI關機按鈕按下事件觸發。
- ACPI啟動關閉。
- 執行個體會在正常關機程序結束後關閉。OS 關機時間無法設定。執行個體仍會短暫顯示於主控台，之後該項目會自動刪除。

資源丟失

- 存放在執行個體儲存體磁碟區的資料。
- 如果`DeleteOnTermination`屬性設定為 true，則存放在 Amazon EBS 根裝置磁碟區上的資料。

持續存在的資源

- 儲存在啟動時或啟動執行個體後連接的其他 Amazon EBS 磁碟區上的資料。

測試應用程式對執行個體終止的回應

您可以用 AWS Fault Injection Service 來測試執行個體終止時應用程式的回應方式。如需詳細資訊，請參閱《AWS Fault Injection Service 使用者指南》https://docs.aws.amazon.com/fis/latest/userguide。

終止執行個體

您可以隨時終止執行個體。

Console

欲使用主控台來終止執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，並選取 Instance state (執行個體狀態)、Terminate instance (終止執行個體)。
4. 出現確認提示時，請選擇 Terminate (終止)。

5. 終止執行個體之後，它會在短時間內保持可見狀態，狀態為terminated。

   如果終止失敗或終止的執行個體可見超過數小時，請參閱仍顯示已終止的執行個體。

Command line

欲使用命令列來終止執行個體

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 訪問 Amazon EC2。

- terminate-instances (AWS CLI)
- Remove-EC2Instance (AWS Tools for Windows PowerShell)

對執行個體終止進行故障診斷

請求者必須具有撥打電話ec2:TerminateInstances的權限。如需詳細資訊，請參閱使用執行個體的範例原則。

如果您終止執行個體並啟動另一個執行個體，很可能您已透過EC2叢集或 Amazon EC2 Auto Scaling 等功能設定了自動擴展。如需詳細資訊，請參閱執行個體自動啟動或終止。

如果終止保護已開啟，則無法終止執行個體。如需詳細資訊，請參閱終止保護。

如果您的執行個體處於shutting-down狀態的時間比平常更長，則應透過 Amazon EC2 服務中的自動程序清理 (終止) 執行個體。如需詳細資訊，請參閱延遲的執行個體終止。

啟用終止保護

若要避免意外終止執行個體，您可啟用執行個體的終止保護。DisableApiTermination屬性控制是否可以使用 AWS Management Console、 AWS Command Line Interface (AWS CLI) 或終止實例 API。根據預設，您的執行個體會停用終止保護，這表示您的執行個體可以使用 AWS Management Console AWS CLI、或來終止您的執行個體API。您可以在啟動執行個體、執行個體執行時或執行個體停止時 (針對 Amazon EBS 支援的執行個體) 設定此屬性的值。

若已設定 InstanceInitiatedShutdownBehavior 屬性，則 DisableApiTermination 屬性不會讓您無法從執行個體啟動關機 (使用作業系統命令進行系統關機) 來終止該執行個體。如需詳細資訊，請參閱變更執行個體啟動的關機行為。

考量事項

- 啟用終止保護並不會防 AWS 止在有排程事件終止執行個體時終止執行個體。

- 啟用終止保護不會防止 Amazon EC2 Auto Scaling 在執行個體運作狀態不良或擴充事件期間終止執行個體。您可以使用執行個體停止保護功能，控制 Auto Scaling 群組是否能在縮減時終止特定的執行個體。您可以暫停 ReplaceUnhealthy 縮減程序，以控制 Auto Scaling 群組是否能終止運作狀態不佳的執行個體。
- 您無法為 競價型執行個體啟用終止保護。

欲在執行個體啟動時啟用終止保護

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在儀表板中，選擇 Launch Instance (啟動執行個體)，然後依據精靈的說明執行。
3. 在 Configure Instance Details (設定執行個體詳細資訊) 頁面選取 Enable termination protection (啟用終止保護) 核取方塊。

欲在執行個體運作中或停止時啟用終止保護

1. 選取執行個體，然後選取 Actions (動作)、Instance Settings (執行個體設定)、Change Termination Protection (變更終止保護)。
2. 選擇 Yes, Enable (是，啟用)。

欲在執行個體運作中或停止時停用終止保護

1. 選取執行個體，然後選取 Actions (動作)、Instance Settings (執行個體設定)、Change Termination Protection (變更終止保護)。
2. 選擇 Yes, Disable (是，停用)。

欲使用命令列啟用或停用終止保護

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱訪問 Amazon EC2。

- modify-instance-attribute (AWS CLI)
- Edit-EC2InstanceAttribute (AWS Tools for Windows PowerShell)

終止具有終止保護的多個實例

如果您在同一要求中終止跨多個可用區域的多個執行個體，並且啟用了終止保護的一或多個指定執行個體，則要求會失敗，並產生下列結果：

- 不會終止與受保護執行個體位於相同可用區域中的指定執行個體。
- 位於不同可用區域中的指定執行個體 (其中沒有其他指定執行個體受到保護) 會成功終止。

範例

假設您在兩個可用區域中有以下四個執行個體。

執行個體	可用區域	終止保護
執行個體 1	AZ 一個	Disabled
執行個體 2		Disabled
執行個體 3	AZ B	Enabled
執行個體 4		Disabled

如果您嘗試在同一個請求中終止所有這些執行個體，請求會報告失敗，結果如下：

- 執行個體 1 和執行個體 2 已成功終止，因為兩個執行個體均未啟用終止保護。
- 執行個體 3 和執行個體 4 無法終止，因為執行個體 3 已啟用終止保護。

變更執行個體啟動的關機行為

根據預設，當您從 Amazon EBS 支援的執行個體啟動關機時 (使用shutdown或等命令poweroff)，執行個體會停止。您可變更此行為，來讓執行個體終止，而無需變更執行個體的 `InstanceInitiatedShutdownBehavior` 屬性。您可在執行個體運作中或停止時，變更此屬性。

halt 命令不會啟動關機。如果使用，則執行個體不會終止；相反地，它會放CPU入執行個體HLT，執行個體會繼續執行。

> **Note**
>
> `InstanceInitiatedShutdownBehavior` 屬性僅適用從執行個體的作業系統執行關機，當您使用`StopInstances`API或 Amazon EC2 主控台停止執行個體時，它不適用。

您可以使用 Amazon EC2 主控台或命令列變更`InstanceInitiatedShutdownBehavior`屬性。

Console

變更執行個體啟動的關機行為

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取實例。
4. 依序選擇 Actions (動作)、Instance settings (執行個體設定)、Change shutdown behavior (變更關機行為)。

   關機行為會顯示目前的行為。
5. 若要變更此行為，對於關機行為，選擇停止或終止。
6. 選擇 Save (儲存)。

Command line

變更執行個體啟動的關機行為

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱訪問 Amazon EC2。

- modify-instance-attribute (AWS CLI)
- Edit-EC2InstanceAttribute (AWS Tools for Windows PowerShell)

在執行個體終止時保留資料

根據您的使用案例，您可能希望在 Amazon 執行個體終止時保留執行個體存放EBS區磁碟區或 Amazon EC2 磁碟區上的資料。在終止執行個體時，執行個體儲存體磁碟區上的資料就不會再持續存在。如果您需要保留執行個體存放區磁碟區上存放的資料超過執行個體的生命週期，則必須手動將該資料複製到更持久的儲存，例如 Amazon EBS 磁碟區、Amazon S3 儲存貯體或 Amazon EFS 檔案系統。如需詳細資訊，請參閱適用於 Amazon EC2 執行個體的儲存選項。

對於 Amazon EBS 卷上的資料，Amazon EC2 會針對每個連接的 Amazon 磁EBS碟區使用`DeleteOnTermination`屬性值來確定是保留還是刪除磁碟區。

`DeleteOnTermination` 屬性的預設值會根據磁碟區是執行個體的根磁碟區，還是連接至執行個體的非根磁碟區而有所不同。

根磁碟區

根據預設，當您啟動執行個體時，執行個體根磁碟區的`DeleteOnTermination`屬性會設定為`true`。因此，執行個體終止時，預設會刪除執行個體的根磁碟區。

非根磁碟區

依預設，當您將非根EBS磁碟區附加至執行個體時，其`DeleteOnTermination`屬性會設定為`false`。因此，預設會保留這些磁碟區。

> **Note**
>
> 執行個體終止後，您可拍攝已保留磁碟區的快照，並將其連接到另一個執行個體。您必須刪除磁碟區以免日後產生費用。

`DeleteOnTermination`屬性可由建立者以AMI及啟動執行個體的人員設定。當屬性由實例的建立者AMI或啟動實例的人員變更時，新設定會覆寫原始AMI預設設定。建議您在使用. 啟動執行個體之後，驗證`DeleteOnTermination`屬性的預設設定AMI。

若要確認是否在執行個體終止時刪除 Amazon EBS 磁碟區，請在執行個體的詳細資料窗格上檢視磁碟區的詳細資訊。在 Storage (儲存) 索引標籤上的 Block devices (區塊型儲存設備) 之下，向右捲動以檢視磁碟區的 Delete on termination (在終止時刪除) 設定。

- 如果為是，磁碟區會在執行個體終止時遭到刪除。
- 如果為否，磁碟區不會在執行個體終止時遭到刪除。執行個體終止時未刪除的任何磁碟區將繼續產生費用。

將根磁碟區變更為在啟動時保留

啟動執行個體時，您可以使用主控台變更 `DeleteOnTermination` 屬性。若要變更運作中執行個體的這項屬性，您必須使用命令列。

使用下列其中一種方法，將根磁碟區變更為在啟動時予以保留。

Console

欲使用主控台在啟動時將執行個體的根磁碟區變更為保留

1. 遵循程序啟動執行個體，但在完成下列步驟以將根磁碟區變更為保留之前，請勿啟動執行個體。

2. 在儲存 (磁碟區) 下方，展開根磁碟區下的資訊。
3. 針對在終止時刪除，選擇否
4. 在 Summary (摘要) 面板中，檢閱您的執行個體組態，然後選擇 Launch instance (啟動執行個體)。如需詳細資訊，請參閱使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體。

Command line

欲使用命令列在啟動時將執行個體的根磁碟區變更為保留

當您啟動 EBS-backed 執行個體時，您可以使用下列其中一個指令來變更根裝置磁碟區以持續存在。如需有關這些命令列介面的詳細資訊，請參閱訪問 Amazon EC2。

- run-instances (AWS CLI)
- New-EC2Instance (AWS Tools for Windows PowerShell)

在您要保留的磁碟區的區塊型裝置映射中，包括 `--DeleteOnTermination`，並指定 `false`。

例如，若要保留磁碟區，將下列選項新增至 `run-instances` 命令：

```
--block-device-mappings file://mapping.json
```

在 `mapping.json` 中，指定裝置名稱，例如 `/dev/sda1` 或者 `/dev/xvda`，針對 `--DeleteOnTermination`，指定 `false`。

```
[
  {
    "DeviceName": "device_name",
    "Ebs": {
      "DeleteOnTermination": false
    }
  }
]
```

將執行中執行個體的根磁碟區變更為持續存在

您可以使用下列其中一個命令，將執行中 EBS-backed 執行個體的根裝置磁碟區變更為持續存在。如需有關這些命令列介面的詳細資訊，請參閱訪問 Amazon EC2。

- modify-instance-attribute (AWS CLI)
- Edit-EC2InstanceAttribute (AWS Tools for Windows PowerShell)

例如，使用下列命令：

```
aws ec2 modify-instance-attribute --instance-id i-1234567890abcdef0 --block-device-
mappings file://mapping.json
```

在 `mapping.json` 中，指定裝置名稱，例如 `/dev/sda1` 或者 `/dev/xvda`，針對 `--DeleteOnTermination`，指定 `false`。

```
[
  {
    "DeviceName": "device_name",
    "Ebs": {
      "DeleteOnTermination": false
    }
  }
]
```

執行個體淘汰

當 AWS 偵測到託管執行個體的基礎硬體無法修復的故障時，執行個體就會排定為淘汰。執行個體根裝置會決定執行個體淘汰的行為：

- 如果您的執行個體根裝置是 Amazon EBS 磁碟區，則執行個體會停止，您可以隨時重新啟動該執行個體。啟動停止的執行個體會將其遷移至新硬體。
- 如果您的執行個體根裝置是執行個體存放區磁碟區，則執行個體會終止，且無法再次使用。

如需執行個體事件類型的詳細資訊，請參閱Amazon EC2 執行個體的排程事件。

內容

- 識別排程淘汰的執行個體
- 針對已排程處分的EBS支援執行個體採取的動作
- 針對排程淘汰的執行個體存放區支援執行個體採取的動作

識別排程淘汰的執行個體

如果已排程淘汰您的執行個體，則您會在事件執行之前收到一封內含執行個體 ID 及淘汰日期的電子郵件。您也可以使用 Amazon EC2 主控台或命令列檢查是否有排定要淘汰的執行個體。

> **Important**
>
> 如果執行個體已排程停用，建議您儘快採取動作，因為執行個體可能無法連線。(您收到的電子郵件通知說明如下：「由於此降級，您的執行個體可能已經無法連線。」) 如需您應採取之建議動作的詳細資訊，請參閱Check if your instance is reachable。

識別排程淘汰的執行個體的方法

- 電子郵件通知
- 主控台識別

電子郵件通知

如果已排程淘汰您的執行個體，則您會在事件執行之前收到一封內含執行個體 ID 及淘汰日期的電子郵件。

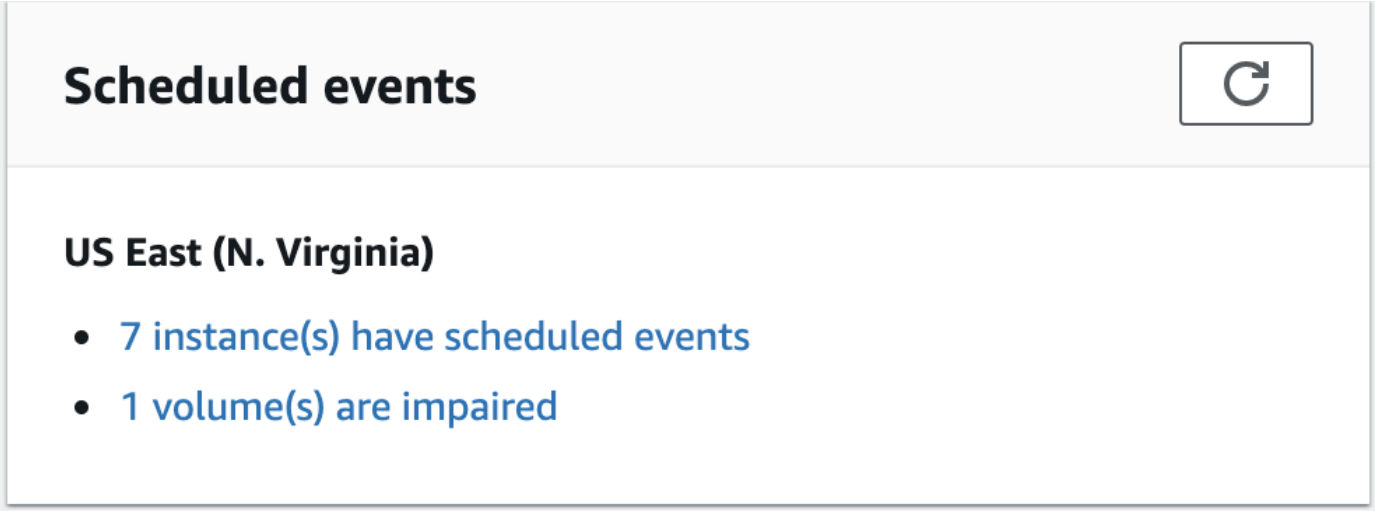
電子郵件會傳送給主要帳戶持有人和營運聯絡人。如需詳細資訊，請參閱 AWS Billing 使用者指南中的新增、變更或移除替代聯絡人。

主控台識別

如果您使用的電子郵件帳戶並未定期檢查執行個體淘汰通知，則可以使用 Amazon EC2 主控台或命令列來判斷是否有任何執行個體已排程停用。

使用主控台來識別排定淘汰的執行個體

1. 打開 Amazon EC2 控制台。
2. 在導覽窗格中，選擇 [EC2儀表板]。在排程事件下，您可以查看與 Amazon EC2 執行個體和磁碟區相關聯的事件，並依區域組織。



3. 如果您有一個已列出排程事件的執行個體，請在區域名稱下方選取其連結以前往 Events (事件) 頁面。
4. 事件頁面列出所有資源，以及與其相關聯的事件。若要檢視已排定淘汰的執行個體，請從第一個篩選條件清單中選取 Instance resources (執行個體資源)，然後從第二個篩選條件清單中選取 Instance stop or retirement (停止或淘汰執行個體)。
5. 如果篩選條件結果顯示執行個體已排定淘汰，請將其選取，然後記下詳細資訊窗格中 Start time (開始時間) 欄位中的日期和時間。這是您的執行個體淘汰日期。

使用命令列來識別排定淘汰的執行個體

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱訪問 Amazon EC2。

- describe-instance-status (AWS CLI)
- Get-EC2InstanceStatus (AWS Tools for Windows PowerShell)

針對已排程處分的EBS支援執行個體採取的動作

若要保留已淘汰執行個體上的資料，您可以執行下列其中一個動作。在執行個體淘汰日期之前採取此動作非常重要，以防止不可預期的停機時間和資料遺失。

對於 Linux 執行個體，如果您不確定執行個體是由執行個體EBS或執行個體存放區支援，請參閱Amazon EC2 執行個體的根磁碟區。

檢查您的執行個體是否可存取

當您收到執行個體已排程停用的通知時，我們建議您儘快採取下列動作：

- 透過連線或 ping 您的執行個體來檢查您的執行個體是否可以存取。
- 如果可取得您的執行個體，您應該計劃在排程的停止日期之前的適當時間停止/啟動執行個體，這影響最小。如需停止和啟動執行個體的詳細資訊，以及執行個體停止時會發生何事，例如與該執行個體相關聯的公有、私有和彈性 IP 地址的影響，請參閱停止並啟動 Amazon EC2 執行個體。請注意，當您停止並啟動執行個體時，執行個體存放區磁碟區上的資料會遺失。
- 如果您的執行個體無法連線，您應立即採取動作並執行停止/啟動來恢復您的執行個體。
- 或者，如果您想要終止執行個體，請計劃盡快執行個體，以便停止對執行個體產生費用。

為您的執行個體建立備份

AMI從您的執行個體建立 EBS-backup，以便備份。若要確保資料完整性，請在建立執行個體之前停止執行個體AMI。您可等候預定的淘汰日期 – 當執行個體停止時 – 或在淘汰日期之前自行停止執行個體。您可以隨時再次啟動該執行個體。如需詳細資訊，請參閱創建一個 Amazon 支EBS持 AMI。

啟動替代執行個體

AMI從執行個體建立之後，您可以使用啟動AMI取代執行個體。從 Amazon 主EC2控台選取新的執行個體，AMI然後從中選擇啟動執行個體AMI。設定執行個體的參數，然後選擇 Launch 執行個體。如需每個欄位的詳細資訊，請參閱使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體。

針對排程淘汰的執行個體存放區支援執行個體採取的動作

若要保留已淘汰執行個體上的資料，您可以執行下列其中一個動作。在執行個體淘汰日期之前採取此動作非常重要，以防止不可預期的停機時間和資料遺失。

> **Warning**
>
> 如果您的執行個體後端執行個體超其淘汰日期，則將會終止，且您無法復原該執行個體或儲存在其中的任何資料。無論執行個體的根裝置為何，執行個體儲存磁碟區上的資料都會在執行個體停用時遺失，即使磁碟區已連接至已EBS備份的執行個體也一樣。

檢查您的執行個體是否可存取

當您收到執行個體已排程停用的通知時，我們建議您儘快採取下列動作：

- 透過連線或 ping 您的執行個體來檢查您的執行個體是否可以存取。

- 如果您的執行個體無法連線，則復原執行個體的可能性很少。如需詳細資訊，請參閱疑難排解無法存取的 Amazon EC2 執行個。 AWS 將在排定的淘汰日期終止您的執行個體，因此，對於無法連線的執行個體，您可以立即自行終止執行個體。

啟動替代執行個體

如中所述，使用這些AMI工具建立AMI從執行個體支援的執行個體存放區。建立執行個體存放區支援AMI從 Amazon 主EC2控台選取新的執行個體，AMI然後從中選擇啟動執行個體AMI。設定執行個體的參數，然後選擇 Launch 執行個體。如需每個欄位的詳細資訊，請參閱使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體。

將您的執行個體轉換為EBS支援的執行個體

將資料傳輸到EBS磁碟區、擷取磁碟區的快照，然後AMI從快照建立。您可以從新的執行個體啟動取代執行個體AMI。如需詳細資訊，請參閱將您的執行個體存放區轉換AMI為支援 EBS AMI。

執行個體恢復

> **Important**
>
> 下列資訊適用於在狀態良好的執行個體上設定復原相關功能。如果您目前無法存取執行個體，請參閱疑難排解EC2執行個體。

如果 AWS 判斷執行個體因為基礎硬體問題而無法使用，您可以為執行個體彈性設定兩種機制，以恢復可用性：簡化的自動復原和以 Amazon CloudWatch 動作為基礎的復原。這個程序稱為執行個體復原。

至少必須事先設定或啟用一個機制，並使用支援的資源，才能執行執行個體復原程序。根據預設，啟動支援的執行個體時，會啟用簡化的自動復原功能。

主題

- 執行個體復原概觀
- 執行個體復原選
- 設定 CloudWatch 動作型復原
- 設定簡化的自動復原

執行個體復原概觀

以下是可能需要執行個體復原的基礎硬體問題範例：

- 網路連線中斷
- 系統電力中斷
- 實體主機的軟體問題
- 實體主機上會影響網路連線的硬體問題

復原的執行個體與原始執行個體相同，包括：

- 執行個體 ID
- 公用、私有和彈性 IP 位址
- 執行個體中繼資料
- 置放群組
- 附加EBS磁碟區
- 可用區域

執行個體復原成功將會顯示為非計劃的重新啟動。換句話說，存儲在易失性存儲器中的內容將丟失，實例存儲數據將被刪除，並且操作系統的正常運行時間將從零開始。

為協助防止資料遺失，我們建議您定期建立有價值資料的備份。如需 Amazon EC2 執行個體備份和復原最佳實務的詳細資訊，請參閱 Amazon 的最佳實務EC2。

執行個體復原選

下列執行個體復原的替代方案符合執行個體的使用案例時，可以考慮這些替代方法。

Auto Scaling 群組

> 您可以使用 Auto Scaling 群組，將執行個體集合分組在一起，以達到擴展和可用性的目的。如果 Auto Scaling 群組中的某個執行個體無法使用，Auto Scaling 群組將自動取代 (不會復原) 該執行個體。如需詳細資訊，請參閱什麼是 Amazon EC2 Auto Scaling？ 在 Amazon EC2 Auto Scaling 用戶指南中。

Amazon EBS 多重連接

您可以為執行個體設定 Amazon M EBS ulti-Attach，以允許多個執行個體連接到相同的EBS磁碟區。與適當的軟體結合使用時，可啟用高可用性叢集。如需 Linux 執行個體的設定範例，請參閱簡化的叢集儲存：儲存部落格GFS2上已啟用 Amazon EBS 多重連接的 AWS 磁碟區。

設定 CloudWatch 動作型復原

> **Important**
>
> - 下列資訊適用於在狀態良好的執行個體上設定復原相關功能。如果您目前無法存取執行個體，請參閱疑難排解EC2執行個體。
> - 為了讓您的工作負載在執行個體復原成功後正常運作，您的執行個體必須啟動並接受流量，而不需要手動介入。

您可以設定 Amazon CloudWatch 動作型復原，以將復原動作新增至 Amazon CloudWatch 警示。CloudWatch 以動作為基礎的復原可搭配`StatusCheckFailed_System`量度使用。 CloudWatch 以動作為基礎的 to-the-minute 復原可提供復原回應時間粒度，以及 Amazon 簡單通知服務 (AmazonSNS) 復原動作和結果的通知。與簡化的自動復原相較之下，這些組態選項可以更精細地控制系統狀態檢查失敗事件回應，進而加快復原嘗試的速度。如需有關可用 CloudWatch 選項的詳細資訊，請參閱執行個體的狀態檢查。

Amazon CloudWatch 動作型復原不會在 AWS Health Dashboard. 如需詳細資訊，請參閱the section called “疑難排解 CloudWatch動作型復原失敗”。

主題

- 以 CloudWatch 動作為基礎的復原需求與限制
- 設定 CloudWatch動作型復原
- 疑難排解 CloudWatch動作型復原失敗

以 CloudWatch 動作為基礎的復原需求與限制

CloudWatch 以動作為基礎的復原可在下列情況下嘗試復原執行個體：

- 處於該`running`狀態。如需詳細資訊，請參閱the section called “實例狀態變更”。

- 使用 `default` (隨選) 或`dedicated`執行個體租用。如需詳細資訊，請參閱the section called “帳單和購買選項”。
- 屬於 Amazon EC2 具有可用容量的執行個體類型。在某些情況下，例如嚴重中斷，可用的容量不足，並且某些復原嘗試可能會失敗。
- 不使用`dedicated`執行個體租用。對於 Amazon EC2 專用主機，您可以使用專用主機自動復原來自動復原運作狀態不良的執行個體。
- 不使用彈性織物適配器。
- 不是「Auto Scaling」群組的成員。
- 目前尚未進行排定的維護事件。
- 使用下列其中一種執行個體類型：
  - 一般用途： A1 | M3 | M4 | M5 | M5a | M5n | M5zn | M6a | M6g | M6i | M6in | M7a | M7g | M7i | M7i-flex | T1 | T2 | T3 | T3a | T4g
  - 計算優化:C3 | C4 | C5 | C5 安培 | C5 安 | C6 g | C6 g | 六分之一 | C6 英寸 | C7 安培 | C7 克 | 七分之二氧化氮 | C7 我-柔性
  - 內存優化:R3 | R4 | R5 | R5A | R5B | R5 毫升 | R6 克 | 六分之一 | 六英寸 | 六英寸 | R7 克 | R7 克 | R7 | R7I | | 烏 7 在十六 TB | 二十四 TB | 三十二 TB | X1
  - 加速運算：G3 | G3 | G5G | 第一階段 | P3 | VT1
  - 高效能運算：HPC6a | HPC7a
  - 金屬例證：上述任何具有金屬例證大小的類型。
- 具有執行個體儲存磁碟區，並使用下列其中一種執行個體類型：M3 | C3 | X1 | X1

> **Warning**
>
> - 如果執行個體停止，執行個體儲存磁碟區上的資料將會遺失。如需停止執行個體的詳細資訊，請參閱the section called “停止的執行”。
> - 如果系統狀態檢查失敗，執行個體儲存區和區塊裝置對應的資料可能會遺失。對於這些執行個體類型，您可以考慮使用the section called “啟用終止保護”.
>
> 我們建議您定期建立有價值資料的備份。如需 Amazon 備份和復原最佳實務的相關資訊EC2，請參閱 Amazon 的最佳實務EC2。

您也可以使用 AWS Management Console 或 AWS CLI 來檢視支援 CloudWatch動作型復原的執行個體類型。

Console

檢視支援 Amazon CloudWatch 動作型復原的執行個體類型

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側導覽窗格中選擇 Instance Types (執行個體類型)。
3. 在篩選條件列中，輸入 Auto Recovery support: true (支援自動復原程序：是)。或者，當您輸入字元且篩選條件名稱顯示時，您可以選取它。

   「執行個體類型」表格會顯示支援 Amazon CloudWatch 動作型復原的所有執行個體類型。

AWS CLI

檢視支援 Amazon CloudWatch 動作型復原的執行個體類型

使用指describe-instance-types令。

```
aws ec2 describe-instance-types --filters Name=auto-recovery-supported,Values=true
--query "InstanceTypes[*].[InstanceType]" --output text | sort
```

設定 CloudWatch動作型復原

CloudWatch 以動作為基礎的復原可搭配`StatusCheckFailed_System`量度使用。 CloudWatch 透過 CloudWatch 主控台設定以動作為基礎的復原。若要設定以 CloudWatch 動作為基礎的復原，請參閱 Amazon CloudWatch 使用者指南中的將復原動作新增至 CloudWatch 警示。

疑難排解 CloudWatch動作型復原失敗

下列問題可能會造成透過動作型復原的執 CloudWatch 行個體復原失敗：

- CloudWatch 在中的服務事件期間，以動作為基礎的復原不會運作 AWS Health Dashboard。您可能不會收到此類事件的復原故障通知。如需最新的服務使用狀態資訊，請參閱服務健全狀況狀態頁面。
- 替換硬體的容量暫時不足。
- 執行個體已達到復原嘗試的每日限額上限。如果自動復原失敗，而原始系統狀態檢查未通過的原因判定為硬體劣化，則執行個體可能隨後會遭到淘汰。

如果即使嘗試多次復原，執行個體的系統狀態檢查失敗仍然存在，請參閱使用失敗狀態檢查進行執行個體疑難排解，

設定簡化的自動復原

> **Important**
>
> - 下列資訊適用於在狀態良好的執行個體上設定復原相關功能。如果您目前無法存取執行個體，請參閱疑難排解EC2執行個體。
> - 為了讓您的工作負載在執行個體復原成功後正常運作，您的執行個體必須啟動並接受流量，而不需要手動介入。

依預設，簡化的自動復原會監控所有支援的執行個體。在偵測到系統狀態檢查失敗的情況下，簡化了自動復原嘗試將執行個體修復為健康狀態。簡化的自動復原在中的服務事件期間無法運作 AWS Health Dashboard。如需詳細資訊，請參閱the section called “簡化的自動復原失敗疑難”。

當簡化的自動恢復事件發生時，您將收到一個 AWS Health Dashboard 事件。若要設定這些事件的通知，請參閱《使AWS 使用者通知 用指南》 AWS 使用者通知中的《入門使用》。您也可以使用 Amazon EventBridge 規則，使用下列事件代碼監控簡化的自動復原事件：

- `AWS_EC2_SIMPLIFIED_AUTO_RECOVERY_SUCCESS` - 成功的事件
- `AWS_EC2_SIMPLIFIED_AUTO_RECOVERY_FAILURE` - 失敗的事件

如需詳細資訊，請參閱 Amazon EventBridge 規則。

主題

- 簡化自動復原的需求與限制
- 設定簡化的自動復原
- 簡化的自動復原失敗疑難

簡化自動復原的需求與限制

在下列情況下，簡化的自動復原將嘗試復原執行個體：

- 處於該`running`狀態。如需詳細資訊，請參閱the section called “實例狀態變更”。

- 使用`default`（按需）或`dedicated`租賃。如需詳細資訊，請參閱the section called “帳單和購買選項”。
- 屬於 Amazon EC2 具有可用容量的執行個體類型。在某些情況下，例如嚴重中斷，可用的容量不足，並且某些復原嘗試可能會失敗。
- 不使用`host`租賃。對於 Amazon EC2 專用主機，您可以使用專用主機自動復原來自動復原運作狀態不良的執行個體。
- 不使用彈性織物適配器。
- 不是`metal`執行個體大小。
- 不是「Auto Scaling」群組的成員。
- 目前尚未進行排定的維護事件。
- 沒有實例存儲卷。
- 使用下列其中一種執行個體類型：
  - 一般用途： A1 | M3 | M4 | M5 | M5a | M5n | M5zn | M6a | M6g | M6i | M6in | M7a | M7g | M7i | M7i-flex | T1 | T2 | T3 | T3a | T4g
  - 計算優化:C3 | C4 | C5 | C5 安培 | C5 安 | C6 g | C6 g | 六分之一 | C6 英寸 | C7 安培 | C7 克 | 七分之二氧化氮 | C7 我-柔性
  - 內存優化:R3 | R4 | R5 | R5A | R5B | R5 毫升 | R6 克 | 六分之一 | 六英寸 | 六英寸 | R7 克 | R7 克 | R7 | R7I | | 烏 7 在十六 TB | 二十四 TB | 三十二 TB | X1
  - 加速運算：G3 | G3 | G5G | 第一階段 | P3 | VT1
  - 高效能運算：HPC6a | HPC7a

> **Warning**
>
> - 如果執行個體停止，執行個體儲存磁碟區上的資料將會遺失。如需停止執行個體的詳細資訊，請參閱the section called “停止的執行”。
> - 如果系統狀態檢查失敗，執行個體儲存區和區塊裝置對應的資料可能會遺失。對於這些執行個體類型，您可以考慮使用the section called “啟用終止保護”.
>
> 我們建議您定期建立有價值資料的備份。如需 Amazon 備份和復原最佳實務的相關資訊EC2，請參閱 Amazon 的最佳實務EC2。

設定簡化的自動復原

當您啟動支援的執行個體時，預設會啟用簡化的自動復原。您可以將自動復原行為設定為啟動執行個體`disabled`期間或之後。此`default`設定不會針對不支援的執行個體類型啟用簡化的自動復原功能。

Console

啟動執行個體時停用簡化的自動復原程序

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)，然後選擇 Launch Instance (啟動執行個體)。
3. 在 Advanced details (進階詳細資訊) 區段，針對 Instance auto-recovery (執行個體自動復原)，選取 Disabled (已停用)。
4. 根據需要進行剩餘的執行個體啟動設定，然後啟動執行個體。

針對執作中或已停止的執行個體停用簡化的自動復原程序

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後選取 Actions (動作)、Instance Settings (執行個體設定)、Change Shutdown Behavior (變更關機行為)。
4. 選擇 Off (關閉)，然後選擇 Save (儲存)。

針對執作中或已停用的執行個體將自動復原行為設定為 **`default`**

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後選取 Actions (動作)、Instance Settings (執行個體設定)、Change Shutdown Behavior (變更關機行為)。
4. 選擇預設，然後選擇儲存。

AWS CLI

啟動時停用簡化的自動復原程序

使用 run-instances 命令。

```
aws ec2 run-instances \
--image-id ami-1a2b3c4d \
--instance-type t2.micro \
--key-name MyKeyPair \
--maintenance-options AutoRecovery=Disabled \
  [...]
```

針對執作中或已停止的執行個體停用簡化的自動復原程序

使用指modify-instance-maintenance-options令。

```
aws ec2 modify-instance-maintenance-options \
--instance-id i-0abcdef1234567890 \
--auto-recovery disabled
```

針對執作中或已停用的執行個體將自動復原行為設定為 **default**

使用指modify-instance-maintenance-options令。

```
aws ec2 modify-instance-maintenance-options \
--instance-id i-0abcdef1234567890 \
--auto-recovery default
```

簡化的自動復原失敗疑難

下列問題可能會導致透過簡化的自動復原執行個體復原失敗：

- 在中的服務事件期間，簡化的自動復原無法運作 AWS Health Dashboard。您可能不會收到此類事件的復原故障通知。如需最新的服務使用狀態資訊，請參閱服務健全狀況狀態頁面。
- 替換硬體的容量暫時不足。
- 執行個體已達到復原嘗試的每日限額上限。如果自動復原失敗，而原始系統狀態檢查未通過的原因判定為硬體劣化，則執行個體可能隨後會遭到淘汰。

如果即使嘗試多次復原，執行個體的系統狀態檢查失敗仍然存在，請參閱使用失敗狀態檢查進行執行個體疑難排解，

使用執行個體中繼資料管理EC2執行個

執行個體中繼資料 是關於您執行個體的資料，您可以用來設定或管理執行中的執行個體。執行個體中繼資料包括：

實例元數據屬性

執行個體中繼資料屬性分成幾種類別，例如主機名稱、事件和安全性群組。

動態資料

動態資料是執行個體啟動時產生的中繼資料，例如執行個體身分識別文件。如需詳細資訊，請參閱動態資料分類。

使用者資料

您也可以使用執行個體中繼資料存取啟動執行個體時指定的使用者資料。例如，您可以指定設定您執行個體的參數，或是包含簡單的指令碼。您也可以建立泛型，AMIs並使用使用者資料來修改啟動時提供的組態檔案。例如，如果您為各種小型企業執行 Web 伺服器，它們都可以使用相同的通用型，AMI並從您在啟動時在使用者資料中指定的 Amazon S3 儲存貯體擷取其內容。若要隨時新增客戶，請為客戶建立值區、新增其內容，然後使用在使用者資料中為您AMI的程式碼提供的唯一值區名稱啟動您的儲存貯體。如果您使用相同`RunInstances`呼叫啟動多個執行個體，則該保留項目中的所有執行個體都可以使用該使用者資料。屬於相同保留項目的每個執行個體都有唯一的編`ami-launch-index`號，因此您可以撰寫程式碼來控制執行個體的功能。例如，第一個主機可能會選取本身做為叢集中的原始節點。如需詳細的AMI啟動範例，請參閱識別在單一要求中啟動的每個執行個體。

> **Important**
>
> 雖然您只能在執行個體內部存取執行個體中繼資料和使用者資料，資料並未受到驗證或密碼編譯法保護。可直接存取執行個體的任何人，以及可能在該執行個體上執行的任何軟體，都能檢視其中繼資料。因此，您不應該將敏感性資料 (例如密碼或長期加密金鑰) 儲存為使用者資料。

目錄

- 執行個體中繼資料分類
- 動態資料分類
- 訪問實例的實EC2例元數據

- 設定執行處理中繼資料服務選
- 使用使用者資料輸入啟動執行個EC2體時執行指令
- 識別在單一要求中啟動的每個執行個體

執行個體中繼資料分類

執行個體中繼資料屬性分為幾種類別。若要擷取執行個體中繼資料屬性，請在要求中指定類別，然後在回應中傳回中繼資料。

發佈新類別時，將會使用新版本編號建立新的執行個體中繼資料建置。在下表中，Version when category was released (類別發佈時的版本) 資料欄指定了發佈執行個體中繼資料類別時的建置版本。為避免每次 Amazon EC2 發行新執行個體中繼資料組建時都必須更新程式碼，請在中繼資料請求中使用`latest`而非版本號碼。如需詳細資訊，請參閱取得執行個體中繼資料的可用版本。

Amazon EC2 發行新的執行個體中繼資料類別時，新類別的執行個體中繼資料可能無法用於現有執行個體。使用建置在 Nitro 系統中的執行個體，您只能擷取啟動時可用類別的執行個體中繼資料。對於使用 Xen hypervisor 的執行個體，您可以停止然後啟動執行個體，以更新執行個體可用的類別。

下表列出執行個體中繼資料的分類。下表中的某些類別名稱包含對您的執行個體而言唯一的資料預留位置。例如 *`mac`* 代表網路介面的MAC位址。擷取執行個體中繼資料時，必須使用實際的值取代預留位置。

類別	描述	類別發佈時的版本
`ami-id`	用AMI來啟動執行個體的 ID。	1.0
`ami-launch-index`	如果您使用相同的`RunInstances` 呼叫啟動多個執行個體，此值會指出每個執行個體的啟動順序。第一個啟動的執行個體值為 0。如果您使用 Auto Scaling 或EC2叢集啟動執行個體，則此值永遠為 0。	1.0
`ami-manifest-path`	Amazon S3 中AMI資訊清單檔案的路徑。如果您使用 Amazon EBS 支援啟動執AMI行個體，則傳回的結果為unknown。	1.0

類別	描述	類別發佈時的版本
`ancestor-ami-ids`	任何重新捆綁以創建此AMI實例AMI IDs的實例。只有當AMI資訊清單檔案包含`ancestor-amis` 金鑰時，此值才會存在。	2007-10-10
`autoscaling/target-lifecycle-state`	顯示 Auto Scaling 執行個體正在轉換到的目標 Auto Scaling 生命週期狀態的值。當執行個體在 2022 年 3 月 10 日之後轉換到目標生命週期狀態之一時存在。可能的值如下：`Detached` | `InService` | `Standby` | `Terminated` | `Warmed:Hibernated` | `Warmed:Running` | `Warmed:Stopped` | `Warmed:Terminated` 。請參閱 Amazon EC2 Auto Scaling 使用指南中的透過執行個體中繼資料擷取目標生命週期狀態	2021-07-15
`block-device-mapping/ami`	包含根/開機檔案系統的虛擬裝置。	2007-12-15
`block-device-mapping/ebs` N	與任何 Amazon EBS 磁碟區相關聯的虛擬裝置。Amazon EBS 磁碟區僅在中繼資料中存在於啟動時或執行個體上次啟動時，才能在中繼資料中使用。N 表示 Amazon EBS 卷的索引（如`ebs1`或`ebs2`）。	2007-12-15

類別	描述	類別發佈時的版本
`block-device-mapping/ephemeral N`	任何非NVMe執行個體儲存磁碟區的虛擬裝置。N 表示每個磁碟區的索引。區塊型設備映射內的執行個體存放區磁碟區數量，可能與執行個體的執行個體存放區磁碟區實際數量不同。執行個體類型決定了執行個體存放區磁碟區可用執行個體的數量。如果區塊型設備映射內執行個體存放區磁碟區的數量，超過可用執行個體，請忽略額外的執行個體存放區磁碟區。	2007-12-15
`block-device-mapping/root`	與根設備相關聯的虛擬裝置或分割區，或虛擬裝置上的分割區，其中根 (/ 或 C:) 檔案系統與指定的執行個體相關聯。	2007-12-15
`block-device-mapping/swap`	與 `swap` 相關聯的虛擬裝置。並非總是存在。	2007-12-15
`elastic-gpus/associations/` *`elastic-gpu-id`*	如果執行個體有 Elastic GPU 連接，則會包含一個JSON字串，其中包含 Elastic 的相關資訊GPU，包括其 ID 和連線資訊。	2016-11-30
`elastic-inference/associations/` *`eia-id`*	如果例證有附加「Elastic Inference」加速器，則會包含一個JSON字串，其中包含有關「Elastic Inference」加速器的資訊，包括其 ID 和類型。	2018-11-29
`events/maintenance/history`	如果執行個體有已完成或取消的維護事件，則會包含含有關事件資訊的JSON字串。	2018-08-17

類別	描述	類別發佈時的版本
`events/maintenance/scheduled`	如果執行個體有使用中的維護事件，則會包含包含事件相關資訊的JSON字串。如需詳細資訊，請參閱檢視影響 Amazon EC2 執行個體的排程事件。	2018-08-17
`events/recommendations/rebalance`	針對執行處理發出EC2執行處理重新平衡建議通知時的大約時間 (以表示)。UTC以下是此類別中繼資料的範例：`{"noticeTime": "2020-11-05T08:22:00Z"}` 。此類別只有在發出通知後才可用。如需詳細資訊，請參閱EC2例項重新平衡建議。	2020-10-27
`hostname`	如果EC2執行個體使用以 IP 為基礎的命名 (IPBN)，這就是執行個體的私人IPv4DNS主機名稱。如果EC2執行個體使用以資源為基礎的命名 (RBN)，這就是. RBN 若有多個網路介面，則其代表 eth0 裝置 (裝置編號為 0 的裝置)。若要取得有關IPBN和的更多資訊RBN，請參閱Amazon EC2 執行個體主機名稱。	1.0
`iam/info`	如果有與執行個體相關聯的IAM角色，則會包含上次更新執行個體設定檔的相關資訊，包括執行個體的 LastUpdated 日期 InstanceProfileArn、和InstanceProfileId。否則不會存在。	2012-01-12

類別	描述	類別發佈時的版本
`iam/security-credentials/role-name`	如果有與執行個體相關聯的IAM角色，*`role-name`* 是角色的名稱，以及 *`role-name`* 包含與角色相關聯的臨時安全登入資料 (如需詳細資訊，請參閱從執行個體中繼資料中擷取安全登入資料)。否則不會存在。	2012-01-12
`identity-credentials/ec2/info`	`identity-credentials/ec2/security-credentials/ec2-instance` 中有關憑證的資訊。	2018-05-23
`identity-credentials/ec2/security-credentials/ec2-instance`	執行處理識別角色的證明資料，可讓執行處理內建軟體識別本身 AWS 以支援EC2執行處理 Connect 和 AWS Systems Manager 預設主機管理組態等功能。這些認證沒有附加任何原則，因此除了識別 AWS 功能的執行個體之外，他們沒有其他 AWS API權限。如需詳細資訊，請參閱Amazon EC2 執行個體的執行個體身分角。	2018-05-23
`instance-action`	通知執行個體應重新開機以準備綁定。有效值：`none` | `shutdown` | `bundle-pending` 。	2008-09-01
`instance-id`	此執行個體的 ID。	1.0
`instance-life-cycle`	此執行個體的購買選項。如需詳細資訊，請參閱Amazon EC2 帳單和購買選項。	2019-10-01
`instance-type`	執行個體的類型。如需詳細資訊，請參閱Amazon EC2 實例類型。	2007-08-29

類別	描述	類別發佈時的版本
ipv6	執行個體的IPv6位址。在存在多個網路介面的情況下，這是指 eth0 裝置 (裝置編號為 0 的裝置) 網路介面和指定的第一個IPv6位址。如果網路介面 [0] 上沒有IPv6位址，則不會設定此項目並產生 HTTP 404 回應。	2021-01-03
kernel-id	使用此執行個體啟動的核心 ID (若適用的話)。	2008-02-01
local-hostname	若有多個網路介面，則其代表 eth0 裝置 (裝置編號為 0 的裝置)。如果EC2執行個體使用以 IP 為基礎的命名 (IPBN)，這就是執行個體的私人IPv4DNS主機名稱。如果EC2執行個體使用以資源為基礎的命名 (RBN)，這就是. RBN 如需IPBN、RBN和EC2執行個體命名的詳細資訊，請參閱Amazon EC2 執行個體主機名稱。	2007-01-19
local-ipv4	執行個體的私人IPv4位址。若有多個網路介面，則其代表 eth0 裝置 (裝置編號為 0 的裝置)。如果這是IPv6唯一的實例，則不會設置此項目並導致 HTTP 404 響應。	1.0
mac	執行個體的媒體存取控制 (MAC) 位址。若有多個網路介面，則其代表 eth0 裝置 (裝置編號為 0 的裝置)。	2011-01-01
metrics/vhostmd	不再提供使用。	2011-05-01

類別	描述	類別發佈時的版本
`network/interfaces/macs/mac/device-number`	與該執行個體相關聯的唯一裝置編號。裝置編號對應至裝置名稱；例如，`device-number` 為 2 即為 eth2 裝置。此類別對應於 Amazon 所使用的`DeviceIndex` 和`device-index` 欄位，以 EC2API及的EC2命令 AWS CLI。	2011-01-01
`network/interfaces/macs/mac/interface-id`	網路介面的 ID。	2011-01-01
`network/interfaces/macs/mac/ipv4-associations/public-ip`	與每個公用 IP 位址相關聯並指派給該介面的私IPv4人位址。	2011-01-01
`network/interfaces/macs/mac/ipv6s`	指派給介面的IPv6位址。	2016-06-30
`network/interfaces/macs/mac/ipv6-prefix`	指定給網路介面的IPv6前置詞。	
`network/interfaces/macs/mac/local-hostname`	執行個體的私人IPv4DNS主機名稱。若有多個網路介面，則其代表 eth0 裝置 (裝置編號為 0 的裝置)。如果這是IPv6唯一的執行個體，這就是以資源為基礎的名稱。若要取得有關IPBN和的更多資訊RBN，請參閱Amazon EC2 執行個體主機名稱。	2007-01-19
`network/interfaces/macs/mac/local-ipv4s`	與介面相關聯的私人IPv4位址。如果這是IPv6唯一的網路介面，則不會設定此項目，並會產生 HTTP 404 回應。	2011-01-01

類別	描述	類別發佈時的版本
`network/interfaces/macs/mac/mac`	執行個體的MAC位址。	2011-01-01
`network/interfaces/macs/mac/network-card`	網路卡的索引。某些執行個體類型支援多個網路卡。	2020-11-01
`network/interfaces/macs/mac/owner-id`	網路介面擁有者的 ID。在多介面環境中，介面僅能由第三方 (例如 Elastic Load Balancing) 連接。介面上的流量一律由介面的擁有者付費。	2011-01-01
`network/interfaces/macs/mac/public-hostname`	接口的公共DNS（IPv4）。只在 `enableDnsHostnames` 屬性設為 `true` 時才會傳回此類別。如需詳細資訊，請參閱 Amazon VPC 使用者指南VPC中的DNS屬性。如果執行個體只有公開IPv6位址且沒有公開IPv4位址，則不會設定此項目，並會產生 HTTP 404 回應。	2011-01-01
`network/interfaces/macs/mac/public-ipv4s`	與界面相關聯的公有 IP 地址與彈性 IP 地址。執行個體上可能有多個IPv4位址。	2011-01-01
`network/interfaces/macs/mac/security-groups`	網路介面所屬的安全群組。	2011-01-01
`network/interfaces/macs/mac/security-group-ids`	網路介面所屬IDs的安全性群組。	2011-01-01
`network/interfaces/macs/mac/subnet-id`	介面所在的子網 ID。	2011-01-01

類別	描述	類別發佈時的版本
network/interfaces/macs/mac/subnet-ipv4-cidr-block	介面所在的子網路IPv4CIDR區塊。	2011-01-01
network/interfaces/macs/mac/subnet-ipv6-cidr-blocks	介面所在的子網路IPv6CIDR區塊。	2016-06-30
network/interfaces/macs/mac/vpc-id	介面所VPC在的識別碼。	2011-01-01
network/interfaces/macs/mac/vpc-ipv4-cidr-block	的主要IPv4CIDR區塊VPC。	2011-01-01
network/interfaces/macs/mac/vpc-ipv4-cidr-blocks	的IPv4CIDR區塊VPC。	2016-06-30
network/interfaces/macs/mac/vpc-ipv6-cidr-blocks	介面所VPC在的IPv6CIDR區塊。	2016-06-30
placement/availability-zone	啟動執行個體的可用區域。	2008-02-01
placement/availability-zone-id	啟動的執行個體所在的靜態可用區域 ID。可用區域 ID 會在各帳戶間保持一致。但是，它可能與可用區域不同，視帳戶而異。	2019-10-01
placement/group-name	在啟動執行個體的位置放置群組名稱。	2020-08-24
placement/host-id	啟動執行個體位置的主機 ID。僅適用於 專用執行個體。	2020-08-24

類別	描述	類別發佈時的版本
`placement/partition-number`	啟動執行個體位置的分割區編號。	2020-08-24
`placement/region`	啟動執行處理所在的 AWS 區域。	2020-08-24
`product-codes`	AWS Marketplace 與例證相關聯的產品代碼 (如果有的話)。	2007-03-01
`public-hostname`	實例的公共DNS（IPv4）。只在 `enableDnsHostnames` 屬性設為 `true` 時才會傳回此類別。如需詳細資訊，請參閱 Amazon VPC 使用者指南VPC中的DNS屬性。如果執行個體只有公開IPv6位址且沒有公開IPv4位址，則不會設定此項目，並會產生 HTTP 404 回應。	2007-01-19
`public-ipv4`	公共IPv4地址。若彈性 IP 地址與執行個體相關聯，傳回的值即為彈性 IP 地址。	2007-01-19
`public-keys/0/openssh-key`	公有金鑰。只有在執行個體啟動階段提供時才可用。	1.0
`ramdisk-id`	啟動時指定的RAM磁碟識別碼 (如果適用)。	2007-10-10
`reservation-id`	保留 ID。	1.0
`security-groups`	套用至執行個體的安全群組名稱。 在啟動之後，您可以變更執行個體的安全群組。這些更改反映在這裡以及網絡/接口/Mac/*mac*/安全性群組。	1.0
`services/domain`	區域資 AWS 源的網域。	2014-02-25

類別	描述	類別發佈時的版本
`services/partition`	資源所在的分割區。對於標準 AWS 區域，分割區為`aws`。如果您有其他分割區的資源，則該分割區為 `aws-`*`partitionname`* 。例如，中國 (北京) 區域的資源分割區，即為 `aws-cn`。	2015-10-20
`spot/instance-action`	動作 (休眠, 停止, 或終止) 和大約時間, 在UTC, 當動作將發生. 只有在 Spot 執行個體已標記為休眠、停止或終止時，才會顯示此項目。如需詳細資訊，請參閱instance-action。	2016-11-15
`spot/termination-time`	Spot 執行個體的UTC作業系統接收關機訊號的大約時間 (以單位為單位)。僅當競價型執行個體已由 Amazon EC2 標記為終止時，此項目才會存在且包含時間值 (例如 2015-01-05T18:02:00 Z)。若您自行終止 Spot 執行個體，則 termination-time 項目便不會設為時間。如需詳細資訊，請參閱termination-time。	2014-11-05
`tags/instance`	與執行個體相關聯的執行個體標籤。僅當您明確允許存取執行個體中繼資料中的標籤時才可用。如需詳細資訊，請參閱 允許存取執行個體中繼資料中的標籤。	2021-03-23

動態資料分類

下表列出動態資料的分類。

類別	描述	類別發佈時的版本
`fws/instance-monitoring`	顯示客戶是否已在中啟用詳細一分鐘監控的 CloudWatch 值。有效值：`enabled` | `disabled`	2009-04-04
`instance-identity/document`	JSON包含實例屬性，例如實例 ID，私有 IP 地址等 請參閱 Amazon EC2 執行個體的執行個體身分證。	2009-04-04
`instance-identity/pkcs7`	用來使用簽章確認文件的真確性和其內容。請參閱 Amazon EC2 執行個體的執行個體身分證。	2009-04-04
`instance-identity/signature`	可由其他廠商用來確認其來源和真確性的資料。請參閱 Amazon EC2 執行個體的執行個體身分證。	2009-04-04

訪問實例的實EC2例元數據

您可以從EC2執行個體本身內部存取執行個體中繼資料，或從EC2主控台APISDKs、、或 AWS CLI. 若要從主控台或命令列取得執行個體目前的執行個體中繼資料設定，請參閱查詢現有執行個體的執行個體中繼資料選項。

您也可以修改具有EBS根磁碟區之執行個體的使用者資料。執行個體必須處於停止狀態。如需主控台操作指示，請參閱更新執行個體使用者資料。如需使用的 Linux 範例 AWS CLI，請參閱modify-instance-attribute。如需使用視窗工具的視窗範例 PowerShell，請參閱the section called “使用者資料和視窗工具 PowerShell”。

> **Note**
>
> 用於擷取執行個體中繼資料和使用者資料的HTTP要求不會向您收費。

執行個體元資料存取

若要避免執行個體中繼資料擷取問題，請考慮下列事項。

命令格式

根據您使用的是執行個體中繼資料服務版本 1 (IMDSv1) 還是執行個體中繼資料服務第 2 版，命令格式會有所不同IMDSv2。根據預設，您可以使用兩個版本的「執行個體中繼資料服務」。若要要求使用IMDSv2，請參閱使用執行個體中繼資料服務來存取執行個體。

（IMDSv2）如果IMDSv2需要，IMDSv1不起作用

若要檢查IMDSv2是否需要，請選取執行處理以檢視其詳細資訊。的值IMDSv2為 [必要] (您必須使用IMDSv2) 或 [選用] (您可以使用IMDSv2或IMDSv1)。

（IMDSv2）用/latest/api/token於檢索令牌

例如`/2021-03-23/api/token`，向任何版本特定路徑發出PUT要求，會導致中繼資料服務傳回 403 禁止錯誤。這是預期行為。

IPv6支持

若要使用IPv6位址擷取執行個體中繼資料，請務必啟用並使用，`[fd00:ec2::254]`而非IPv4位址。執行個體必須建置在 AWS Nitro 系統上，並在支援IPv6的子網路中啟動。

(視窗) AMIs 使用視窗系統建立自訂

若要確保在您從自訂視窗啟動執行個體時IMDS正常運作AMI，AMI必須是使用 Windows Sysprep 建立的標準化映像檔。否則，IMDS將無法正常工作。表單詳細資訊，請參閱EC2AMI使用視窗系統創建一個 Amazon。

在容器環境中，將躍點限制設定為 2

默認情況下， AWS SDKs使用IMDSv2調用。如果IMDSv2通話未收到任何回應，則會SDK重試通話，如果仍不成功，則會使用IMDSv1。這可能會導致延遲，特別是在容器環境中。在容器環境中，如果躍點限制為 1，則回IMDSv2應不會傳回，因為移至容器會被視為額外的網路躍點。為了避免退回的過程IMDSv1和產生的延遲，我們建議您在容器環境中將躍點限制設置為 2。如需詳細資訊，請參閱設定執行處理中繼資料服務選。

元數據版本

為避免每次 Amazon EC2 發布新的執行個體中繼資料組建時都必須更新程式碼，建議您`latest`在路徑中使用，而不是版本號碼。

使用者資料存取的其他考量

- 使用者資料會被視為不透明資料：您指定的是擷取時取回的資料。它是由實例來解釋和對用戶數據採取行動。

- 使用者資料必須為 base64 編碼形式。根據您使用的工具或您SDK使用的工具，可能會為您執行 base64 編碼。例如：
  - Amazon 主EC2控台可以為您執行 base64 編碼，或接受以 base64 編碼的輸入。
  - AWS CLI 版本 2 默認情況下為您執行二進制參數的 base64 編碼。 AWS CLI 版本 1 為您執行`--user-data`參數的 base64 編碼。
  - 會為您 AWS SDK for Python (Boto3) 執行`UserData`參數的 base64 編碼。
- 使用者資料在 base64 編碼之前限制為 16 KB (原始形式)。長度為 n 的字串在 base64 編碼之後的大小為 ceil(n/3)*4。
- 使用者資料在擷取時必須以 base64 解碼。如果您使用執行個體中繼資料或主控台擷取資料，將會自動為您解碼資料。
- 若您停止執行個體，請修改其使用者資料，然後啟動執行個體，則更新後的使用者資料便不會在您啟動執行個體時自動執行。使用 Windows 執行個體時，您可以設定設定，以便在啟動執行個體時執行一次更新的使用者資料指令碼，或是每次重新開機或啟動執行個體時執行。
- 使用者資料是執行個體屬性。如果您AMI從執行個體建立，則執行個體使用者資料不會包含在中AMI。

從執行個體內存取執EC2行個體中繼資

由於您的執行個體中繼資料可從執行中的執行個體取得，因此您不需要使用 Amazon EC2 主控台或 AWS CLI. 若您正在撰寫要從您的執行個體執行的指令碼，這將會很有幫助。例如，您可以存取從執行個體中繼資料存取您執行個體的本機 IP 地址，管理與外部應用程式的連線。

以下所有內容都被視為執行個體中繼資料，但它們的存取方式不同。選取代表您要存取之執行個體中繼資料類型的索引標籤，以查看詳細資訊。

Metadata

執行個體中繼資料屬性分為幾種類別。對於每個執行個體中繼資料類別的說明，請參閱 執行個體中繼資料分類。

若要從執行中的執行個體中存取執行個體中繼資料屬性，請從下列IPv4或取得資料IPv6URIs。這些 IP 位址是連結本機位址，且僅在執行個體中有效。如需詳細資訊，請參閱Link-local 地址。

IPv4

```
http://169.254.169.254/latest/meta-data/
```

IPv6

```
http://[fd00:ec2::254]/latest/meta-data/
```

Dynamic data

若要從執行中的執行個體中擷取動態資料，請使用下列其中一項URIs。

IPv4

```
http://169.254.169.254/latest/dynamic/
```

IPv6

```
http://[fd00:ec2::254]/latest/dynamic/
```

示例：使用 c 訪問 URL

下列範例會用cURL來擷取高階執行個體識別類別。

IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-
aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/dynamic/
instance-identity/
rsa2048
pkcs7
document
signature
dsa2048
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/dynamic/instance-identity/
rsa2048
pkcs7
document
signature
dsa2048
```

範例：存取 PowerShell

下列範例會用 PowerShell 來擷取高階執行個體識別類別。

IMDSv2

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/dynamic/instance-identity/
document
rsa2048
pkcs7
signature
```

IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/dynamic/instance-identity/
document
rsa2048
pkcs7
signature
```

如需動態資料的詳細資訊及如何擷取它的範例，請參閱Amazon EC2 執行個體的執行個體身分證。

User data

若要從執行個體擷取使用者資料，請使用下列其中一項URIs。若要使用該IPv6位址擷取使用者資料，您必須啟用該位址，且執行個體必須是在支援IPv6的子網路中建置於 AWS Nitro System 上的執行個體。

IPv4

```
http://169.254.169.254/latest/user-data
```

IPv6

```
http://[fd00:ec2::254]/latest/user-data
```

使用者資料的請求會直接傳回資料 (內容類型 `application/octet-stream`)。如果執行個體沒有任何使用者資料，則請求會傳回 `404 - Not Found`。

實例：用 c URL 訪問以檢索逗號分隔的文本

下列範例會用cURL來擷取指定為逗號分隔文字的使用者資料。

IMDSv2

```
TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/user-data
1234,john,reboot,true | 4512,richard, | 173,,,
```

IMDSv1

```
curl http://169.254.169.254/latest/user-data
1234,john,reboot,true | 4512,richard, | 173,,,
```

示例：訪問以檢索逗號分隔的 PowerShell 文本

下列範例會用 PowerShell 來擷取指定為逗號分隔文字的使用者資料。

IMDSv2

```
[string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds"
 = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri
 http://169.254.169.254/latest/user-data
1234,john,reboot,true | 4512,richard, | 173,,,
```

IMDSv1

```
Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = Invoke-RestMethod -Headers
 @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} `
-Method PUT -Uri http://169.254.169.254/latest/api/token} -Method GET -uri
 http://169.254.169.254/latest/user-data
1234,john,reboot,true | 4512,richard, | 173,,,
```

示例：使用 c 訪問URL以檢索腳本

下列範例會用cURL來擷取指定為指令碼的使用者資料。

IMDSv2

```
TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/user-data
#!/bin/bash
yum update -y
service httpd start
chkconfig httpd on
```

IMDSv1

```
curl http://169.254.169.254/latest/user-data
#!/bin/bash
yum update -y
service httpd start
chkconfig httpd on
```

範例：存取 PowerShell 以擷取指令碼

下列範例會用 PowerShell 來擷取指定為指令碼的使用者資料。

IMDSv2

```
[string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/user-data
<powershell>
$file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
New-Item $file -ItemType file
</powershell>
<persist>true</persist>
```

IMDSv1

```
Invoke-RestMethod -uri http://169.254.169.254/latest/user-data
<powershell>
$file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
New-Item $file -ItemType file
</powershell>
<persist>true</persist>
```

查詢現有執行個體的執行個體中繼資料選項

您可以使用下列其中一種方法，查詢現有執行個體的執行個體中繼資料選項。

Console

若要查詢現有執行個體的執行個體中繼資料選項，請使用主控台

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體。
4. 選擇動作、執行個體設定、修改執行個體中繼資料選項。
5. 在「修改執行個體中繼資料選項」對話方塊中，檢閱目前執行個體中繼資料選項。

AWS CLI

若要查詢現有執行個體的執行個體中繼資料選項，請使用 AWS CLI

使用描述實例命令CLI。

```
aws ec2 describe-instances \
    --instance-id i-1234567898abcdef0 \
    --query 'Reservations[].Instances[].MetadataOptions'
```

PowerShell

若要使用工具查詢現有執行個體的執行個體中繼資料選項 PowerShell

使用指Get-EC2Instance令程式。

```
(Get-EC2Instance `
```

```
    -InstanceId i-1234567898abcdef0).Instances.MetadataOptions
```

回應及錯誤訊息

所有執行個體中繼資料都會傳回為文字 (HTTP內容類型`text/plain`)。

特定中繼資料資源的要求會傳回適當的值，如果資源不可用，則會傳回`404 - Not Found`HTTP錯誤碼。

一般中繼資料資源 (以/結URI尾) 的要求會傳回可用資源清單，如果沒有此類資源，則傳回`404 - Not Found`HTTP錯誤碼。列表項在單獨的行上，由換行符（ASCII10）終止。

針對使用執行個體中繼資料服務版本 2 提出的要求，可傳回下列HTTP錯誤碼：

- `400 - Missing or Invalid Parameters` –PUT 請求無效。
- `401 - Unauthorized` – GET 請求使用的字符無效。建議動作會產生新字符。
- `403 - Forbidden`— 不允許要求或IMDS已關閉。
- `503`— 請求無法完成。重試 請求。

使用執行個體中繼資料服務來存取執行個體

您可以使用以下其中一個方法，從執行中的執行個體存取執行個體中繼資料：

- 執行個體中繼資料服務版本 2 (IMDSv2)-工作階段導向方法

  如需範例，請參閱 範例 IMDSv2。
- 執行個體中繼資料服務版本 1 (IMDSv1) — 請求/回應方法

  如需範例，請參閱 範例 IMDSv1。

依預設，您可以使用IMDSv1或IMDSv2，或兩者。

您可以在每個執行個體上設定執行個體中繼資料服務 (IMDS)，以便本機程式碼或使用者必須使用IMDSv2。當您指定IMDSv2必須使用時，IMDSv1不再有效。如需如何設定要使用的執行個體的詳細資訊IMDSv2，請參閱設定執行處理中繼資料服務選。

PUT或標`GET`頭是唯一的IMDSv2。如果這些標頭存在於請求中，則該請求的目的是IMDSv2。如果沒有標頭存在，則假定該請求的目的IMDSv1。

如需詳細檢閱IMDSv2，請參閱透過EC2執行個體中繼資料服務的增強功能，為開放式防火牆、反向Proxy 和SSRF弱點新增深入防禦。

主題

- 執行個體中繼資料服務第 2 版 的運作方式
- 轉換為使用 執行個體中繼資料服務第 2 版
- 使用支持 AWS SDK
- 範例 IMDSv2
- 範例 IMDSv1

執行個體中繼資料服務第 2 版 的運作方式

IMDSv2 會使用工作階段導向請求。使用工作階段導向請求，您就能建立定義工作階段持續期間的工作階段字符，此期間最短 1 秒，最長可達 6 小時。在指定持續期間，您可以將相同的工作階段字符用於後續請求。在指定持續期間到期之後，您必須建立新的工作階段字符，才能使用未來請求。

> **Note**
>
> 本節中的範例使用「執行處理中繼資料服務」(IMDS) 的IPv4位址：`169.254.169.254`。如果您要透過該IPv6位址擷取EC2執行個體的執行個體中繼資料，請務必啟用並使用IPv6位址：`[fd00:ec2::254]`。的IPv6位址與IMDSv2指令相容。IMDS該IPv6地址只能在 AWS Nitro 系統上構建的實例和IPv6支持的子網（雙堆棧或IPv6僅）中訪問。

下列範例使用 shell 指令碼，並擷IMDSv2取頂層執行個體中繼資料項目。每個例子：

- 使用 PUT 請求，建立持續 6 小時 (21,600 秒) 的工作階段字符
- 將工作階段權杖標頭儲存在名為 `TOKEN` (Linux 執行個體) 或 `token` (Windows 執行個體) 的變數中
- 使用字符請求上層中繼資料項目

Linux 範例

您可以執行兩個單獨的命令，或將它們合併。

單獨命令

首先，使用以下命令產生字符。

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-
ec2-metadata-token-ttl-seconds: 21600"`
```

然後，使用字符產生使用下列命令的頂層中繼資料項目。

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/
meta-data/
```

組合命令

您可以存放字符並組合命令。下面的例子結合了上述兩個命令，並將會話令牌頭存儲在一個名為的變量TOKEN。

> **Note**
>
> 如果在建立字符時發生錯誤，變數中會存放錯誤消息而非有效字符，命令也不會發揮作用。

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-
ec2-metadata-token-ttl-seconds: 21600"` \
 && curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/
```

建立字符之後，您可以重複使用直到到期為止。在下列範例命令 (取得AMI用來啟動執行個體的 ID) 中，會重複使用儲存$TOKEN在上一個範例中的權杖。

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/
meta-data/ami-id
```

Windows 範例

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-
seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -
Uri http://169.254.169.254/latest/meta-data/
```

建立字符之後，您可以重複使用直到到期為止。在下列範例命令 (取得AMI用來啟動執行個體的 ID) 中，會重複使用儲存$token在上一個範例中的權杖。

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} `
 -Method GET -uri http://169.254.169.254/latest/meta-data/ami-id
```

當您使用IMDSv2要求執行個體中繼資料時，要求必須包含下列項目：

1. 使用 PUT 請求，在執行個體中繼資料服務中起始工作階段。PUT 請求會傳回一個字符，其必須包含在執行個體中繼資料服務的後續 GET 請求中。使用訪問元數據需要令牌IMDSv2。
2. 在所有GET要求中包含權杖IMDS。當 Token 用法設為時`required`，沒有有效令牌或具有過期令牌的請求會收到`401 - Unauthorized`HTTP錯誤代碼。
   - 字符是執行個體特定金鑰。該令牌在其他實EC2例上無效，如果您嘗試在生成它的實例之外使用它，則將被拒絕。
   - PUT請求必須包含一個標題，用於指定令牌的活時間（TTL），以秒為單位，最多六個小時（21,600 秒）。字符會代表邏輯工作階段。會TTL指定權杖有效的時間長度，因此指定工作階段的持續時間。
   - 在字符到期之後，若要繼續存取執行個體中繼資料，您必須使用另一個 PUT 建立新的工作階段。
   - 您可以選擇重複使用字符或使用每個請求來建立新字符。對於少數請求，每次需要訪問時都可能更容易生成並立即使用令牌IMDS。但為了提升效率，您可以為該字符指定時間更長的持續期間，然後再重複使用該字符，而不需要在每次要請求執行個體中繼資料時寫入 PUT。並行字符數量沒有實際限額，每個都代表它自己的工作階段。IMDSv2但是，仍然受到正常IMDS連接和節流限制的限制。如需詳細資訊，請參閱查詢調節。

HTTPGET和HEAD方法是允許在IMDSv2實例元數據請求。 PUT如果要求包含 X 轉寄用於標頭，則會遭到拒絕。

依預設，PUT 請求在 IP 通訊協定等級的 1 回應跳轉限額 (存留時間)。如果您需要更大的躍點限制，可以使用modify-instance-metadata-options AWS CLI 指令對其進行調整。例如，您可以需要更大的跳轉限制，才能與執行個體上執行之容器服務回溯相容。如需詳細資訊，請參閱修改現有執行個體的執行個體中繼資料選項。

轉換為使用 執行個體中繼資料服務第 2 版

移轉至時IMDSv2，建議您使用下列工具和轉移路徑。

主題

- 幫助過渡到的工具 IMDSv2
- 推薦的需求路徑 IMDSv2

幫助過渡到的工具 IMDSv2

如果您的軟體使用IMDSv1，請使用下列工具來協助重新設定要使用IMDSv2的軟體。

AWS 軟體

最新版本的 AWS CLI 和 AWS SDKs支援IMDSv2。若要使用IMDSv2，請確定您的EC2執行個體具有CLI和的最新版本SDKs。若要取得有關更新的資訊CLI，請參閱《使用指南》 AWS CLI中的〈安裝、更新和解除安裝AWS Command Line Interface 〉。

所有 Amazon Linux 2 和 Amazon 2023 軟體套件都支援IMDSv2。在 Amazon 2023 中，默認情況下IMDSv1處於禁用狀態。

如需支援的最低 AWS SDK版本IMDSv2，請參閱使用支持 AWS SDK。

IMDS封包分析器

IMDS封包分析器是一種開放原始碼工具，可識別並記錄執行個體開機階段的IMDSv1呼叫。這有助於識別在EC2執行個體上IMDSv1撥打呼叫的軟體，讓您準確找出需要更新的項目，讓您的執行個體IMDSv2只能使用。您可以從命令列執行IMDS封包分析器，或將其安裝為服務。如需詳細資訊，請參閱上的IMDS封包分析器GitHub。

CloudWatch

IMDSv2使用令牌支持的會話，而IMDSv1不使用。此`MetadataNoToken` CloudWatch 測量結果會追蹤正在使用之「執行處理中繼資料服務」(IMDS) 的呼叫次數IMDSv1。透過將此指標追蹤為零，您可以判斷是否已升級所有軟體以及何時升級為使用IMDSv2。

停用後IMDSv1，您可以使用`MetadataNoTokenRejected` CloudWatch 指標追蹤嘗試IMDSv1通話和拒絕的次數。通過跟踪此指標，您可以確定是否需要更新軟件才能使IMDSv2用。

如需詳細資訊，請參閱執行個體指標。

EC2APIs和的更新 CLIs

對於新執行個體，您可以使RunInstancesAPI用啟動需要使用的新執行個體IMDSv2。如需詳細資訊，請參閱設定新執行個體的執行個體中繼資料選項。

對於現有執行個體，您可ModifyInstanceMetadataOptionsAPI以使用來要求使用IMDSv2。如需詳細資訊，請參閱修改現有執行個體的執行個體中繼資料選項。

若要要求IMDSv2在 Auto Scaling 群組啟動的所有新執行個體上使用，Auto Scaling 群組可以使用啟動範本或啟動設定。當您建立啟動範本或建立啟動組態時，您必須設定要求使用

的MetadataOptions參數IMDSv2。Auto Scaling 群組會使用新啟動範本或啟動組態來啟動新執行個體，但現有的執行個體不受影響。對於 Auto Scaling 群組中的現有執行個體，您可以使用ModifyInstanceMetadataOptionsAPI來要求對現有執行個體使用，或終止執行個體，Auto Scaling 群組會使用新啟動範本或啟動設定中定義的執行個體中繼資料選項設定來啟動新的取代執行個體。IMDSv2

使用默AMI認配置 IMDSv2

當您啟動執行個體時，您可以透IMDSv2過將HttpTokens參數設定為設定為的啟動執行個體來啟動執行個體，以自動將其設AMI定為預設使用 (ImdsSupport參數設定為required) v2.0。您可以在AMI使用寄存器影像CLI指令註冊v2.0時將ImdsSupport參數設定為，也可以使用指令修改現有AMI參數modify-image-attributeCLI。如需詳細資訊，請參閱配置 AMI。

IAM政策和 SCPs

您可以使用IAM策略或 AWS Organizations 服務控制策略 (SCP) 來控制使用者，如下所示：

- 除非執行個體設定為使用，RunInstancesAPI否則無法使用啟動執行個體IMDSv2。
- 無法使用重新啟IMDSv1用ModifyInstanceMetadataOptionsAPI來修改執行中的執行個體。

IAM策略或SCP必須包含下列IAM條件金鑰：

- ec2:MetadataHttpEndpoint
- ec2:MetadataHttpPutResponseHopLimit
- ec2:MetadataHttpTokens

如果API或CLI呼叫中的參數與包含條件索引鍵的原則中指定的狀態不符，API或CLI呼叫會因UnauthorizedOperation回應而失敗。

此外，您可以選擇額外的保護層來強制執行從IMDSv1到的變更IMDSv2。在存取管理層相對於透過EC2角色認證APIs呼叫，您可以在IAM原則或 AWS Organizations 服務控制原則 (SCPs) 中使用新的條件金鑰。具體來說，使用IAM原則中值為ec2:RoleDelivery的條件索引鍵，使用從2.0中取得的EC2角色認證進行的API呼叫IMDSv1將會收到UnauthorizedOperation回應。同樣的事情可以更廣泛地實現與所要求的條件. SCP 這可確保透過傳送的認證實際上IMDSv1無法用於呼叫，APIs因為任何API不符合指定條件的呼叫都會收到錯UnauthorizedOperation誤訊息。

如需IAM原則範例，請參閱使用執行個體中繼資料。如需有關的詳細資訊SCPs，請參閱AWS Organizations 使用指南中的服務控制策略。

推薦的需求路徑 IMDSv2

使用上述工具，我們建議您遵循此路徑以轉換至IMDSv2。

步驟 1：開始時

將EC2執行SDKs個體上使用 Role 認證的CLIs、和您的軟體更新為相容的版本IMDSv2。若要取得有關更新的資訊CLI，請參閱《使用指南》 AWS CLI中的〈升級到最新版本AWS Command Line Interface〉。

然後，使用IMDSv2請求更改直接訪問實例元數據的軟件（換句話說，不使用 aSDK）。您可以使用IMDS封包分析器來識別您需要變更才能使用要IMDSv2求的軟體。

步驟 2：追蹤轉換進度

使用指標追蹤您的轉換進 CloudWatch 度`MetadataNoToken`。此量度顯示執行個體IMDS上的IMDSv1呼叫次數。如需詳細資訊，請參閱執行個體指標。

步驟 3：當IMDSv1使用率為零時

當 CloudWatch 指標`MetadataNoToken`記錄零IMDSv1使用量時，您的執行個體就可以完全轉換為使用IMDSv2。在這個階段，您可以執行下列操作：

- 帳戶預設

  您可以將設IMDSv2定為必要作為帳戶預設值。啟動執行個體時，執行個體組態會自動設定為帳戶預設值。

  若要設定帳戶預設值，請執行下列動作：

  - Amazon EC2 主控台：在EC2儀表板上的帳戶屬性、資料保護和安全性下，針對IMDS預設值，將執行個體中繼資料服務設定為 [啟用]，將中繼資料版本設定為 [僅限 V2] (需要權杖) 如需詳細資訊，請參閱設定IMDSv2為帳戶的預設值。
  - AWS CLI：使用命modify-instance-metadata-defaultsCLI令並指定`--http-tokens required`和`--http-put-response-hop-limit 2`。
- 新執行個體

  啟動新執行個體時，您可以執行下列操作：

  - Amazon EC2 主控台：在啟動執行個體精靈中，將可存取的中繼資料設定為已啟用，將中繼資料版本設定為僅 V2 (需要權杖) 如需詳細資訊，請參閱啟動時設定執行個體。
  - AWS CLI：使用運行實例CLI命令並指定必要的命IMDSv2令。
- 現有執行個體

  針對現有執行個體，可以執行以下操作：

- Amazon EC2 主控台：在 [執行個體] 頁面上，選取執行個體，選擇 [動作]、[執行個體設定]、[修改執行個體中繼資料選項] IMDSv2，然後選擇 [如需詳細資訊，請參閱需要使用 IMDSv2。
- AWS CLI：使用指modify-instance-metadata-optionsCLI令來指定IMDSv2只要使用。

您可以在執行中的執行個體上修改執行個體中繼資料選項，不需要在修改執行個體中繼資料選項後重新啟動執行個體。

步驟 4：檢查您的執行個體是否已轉換為 IMDSv2

您可以檢查是否有任何實例尚未配置為要求使用IMDSv2，換句話說，IMDSv2仍將其配置為`optional`。如果仍將任何執行個體設定為`optional`，您可以重複上述步驟 3，修改要建立IMDSv2`required`的執行個體中繼資料選項。

篩選執行個體：

- Amazon EC2 主控台：在「執行個體」頁面上，使用 IMDSv2= 選用篩選器篩選執行個體。如需有關篩選的詳細資訊，請參閱 使用主控台篩選資源。您也可以檢視每個實例的必要或選用性：在「偏好設定」視窗中，開啟以將IMDSv2欄新增IMDSv2至「實例」表格。IMDSv2
- AWS CLI：使用描述實例CLI命令並篩選依據`metadata-options.http-tokens = optional`，如下所示：

```
aws ec2 describe-instances --filters "Name=metadata-options.http-
tokens,Values=optional" --query "Reservations[*].Instances[*].[InstanceId]" --output
 text
```

步驟 5：將所有執行個體轉換為時 IMDSv2

`ec2:MetadataHttpTokensec2:MetadataHttpPutResponseHopLimit`、與`ec2:MetadataHttpEndpointIAM`條件鍵可用來控制RunInstances與與對應的使用CLIs。ModifyInstanceMetadataOptionsAPIs如果建立原則，且API呼叫中的參數與使用條件索引鍵的原則中指定的狀態不符，API或CLI呼叫會`UnauthorizedOperation`因回應而失敗。如需IAM原則範例，請參閱使用執行個體中繼資料。

此外，停用後IMDSv1，您可以使用`MetadataNoTokenRejected` CloudWatch 指標來追蹤IMDSv1呼叫嘗試和拒絕的次數。如果禁用後IMDSv1，您的軟件無法正常工作並且度`MetadataNoTokenRejected`量記錄IMDSv1調用，則可能需要更新此軟件才能使用IMDSv2。

使用支持 AWS SDK

若要使用IMDSv2，您的EC2執行個 AWS SDK體必須使用支援使用IMDSv2. 所有 AWS SDKs支持的最新版本使用IMDSv2.

> **Important**
>
> 我們建議您隨時掌握最新SDK版本，以隨時掌握最新功能、安全性更新和基礎相依性。我們不建議您繼續使用不受支援的SDK版本，並由您自行決定。如需詳細資訊，請參閱AWS SDKs和工具參考指南中的AWS SDKs和工具維護原則。

以下是支援使用的最低版本IMDSv2：

- AWS CLI— 1.16.289
- AWS Tools for Windows PowerShell – 4.0.1.0
- AWS SDK for .NET – 3.3.634.1
- AWS SDK for C++ – 1.7.229
- AWS SDK for Go – 1.25.38
- AWS SDK對於轉到第 2 版 — 0.19.0
- AWS SDK for Java – 1.11.678
- AWS SDK for Java 2.x – 2.10.21
- AWS SDK對 JavaScript 於在 Node.js 中
- AWS SDK for PHP – 3.147.7
- AWS SDK對於 Python（肉毒核）— 1.13.25
- AWS SDK for Python (Boto3) – 1.12.6
- AWS SDK for Ruby – 3.79.0

範例 IMDSv2

在 Amazon 執行EC2個體上執行下列範例，以擷取的執行個體中繼資料IMDSv2。

在視窗執行個體上，您可以使用視窗 PowerShell ，也可以安裝 c URL 或 wget。如果您在 Windows 執行個體上安裝協力廠商工具，請務必仔細閱讀隨附的文件，因為呼叫和輸出可能與此處所述不同。

範例

- 取得執行個體中繼資料的可用版本
- 取得上層中繼資料項目
- 取得中繼資料項目的值
- 取得可用公有金鑰清單
- 顯示可使用公有金鑰 0 的格式
- 取得公開金鑰 0 (以開放SSH金鑰格式)
- 取得執行個體的子網 ID
- 取得執行個體的執行個體標籤

取得執行個體中繼資料的可用版本

此範例會取得執行個體中繼資料的可用版本。每個版本會參照在發佈新執行個體中繼資料類別時的執行個體中繼資料建置。執行個體中繼資料建置版本與 Amazon EC2 API 版本不相關。若您有依存於先前版本中結構和資訊的指令碼，您也可以取得先前版本。

cURL

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-
aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/
1.0
2007-01-19
2007-03-01
2007-08-29
2007-10-10
2007-12-15
2008-02-01
2008-09-01
2009-04-04
2011-01-01
2011-05-01
2012-01-12
2014-02-25
2014-11-05
2015-10-20
2016-04-19
...
latest
```

PowerShell

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-
seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
 GET -Uri http://169.254.169.254/
1.0
2007-01-19
2007-03-01
2007-08-29
2007-10-10
2007-12-15
2008-02-01
2008-09-01
2009-04-04
2011-01-01
2011-05-01
2012-01-12
2014-02-25
2014-11-05
2015-10-20
2016-04-19
...
latest
```

取得上層中繼資料項目

此範例會取得最上層的中繼資料項目。有關響應中物件的更多內容，敬請參閱執行個體中繼資料分類。

請注意，只有在您允許存取的情況下，標籤才會包含在此輸出中。如需詳細資訊，請參閱the section called “允許存取執行個體中繼資料中的標籤”。

cURL

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-
aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-
data/
ami-id
ami-launch-index
```

```
ami-manifest-path
block-device-mapping/
events/
hostname
iam/
instance-action
instance-id
instance-life-cycle
instance-type
local-hostname
local-ipv4
mac
metrics/
network/
placement/
profile
public-hostname
public-ipv4
public-keys/
reservation-id
security-groups
services/
tags/
```

PowerShell

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-
seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
 GET -Uri http://169.254.169.254/latest/meta-data/
ami-id
ami-launch-index
ami-manifest-path
block-device-mapping/
hostname
iam/
instance-action
instance-id
instance-life-cycle
instance-type
local-hostname
local-ipv4
```

```
mac
metrics/
network/
placement/
profile
public-hostname
public-ipv4
public-keys/
reservation-id
security-groups
services/
tags/
```

取得中繼資料項目的值

這些範例會取得上述範例中取得的某些頂層中繼資料項目的值。這些要求會使用前一個範例中使用命令建立的已儲存 Token。令牌不得過期。

cURL

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/
latest/meta-data/ami-id
ami-0abcdef1234567890
```

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/
latest/meta-data/reservation-id
r-0efghijk987654321
```

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/
latest/meta-data/local-hostname
ip-10-251-50-12.ec2.internal
```

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/
latest/meta-data/public-hostname
ec2-203-0-113-25.compute-1.amazonaws.com
```

PowerShell

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
 GET -Uri http://169.254.169.254/latest/meta-data/ami-id
```

```
ami-0abcdef1234567890
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
 GET -Uri http://169.254.169.254/latest/meta-data/reservation-id
r-0efghijk987654321
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
 GET -Uri http://169.254.169.254/latest/meta-data/local-hostname
ip-10-251-50-12.ec2.internal
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
 GET -Uri http://169.254.169.254/latest/meta-data/public-hostname
ec2-203-0-113-25.compute-1.amazonaws.com
```

取得可用公有金鑰清單

此範例會取得可用公有金鑰的清單。

cURL

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-
aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-
data/public-keys/
0=my-public-key
```

PowerShell

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-
seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
 GET -Uri http://169.254.169.254/latest/meta-data/public-keys/
0=my-public-key
```

顯示可使用公有金鑰 0 的格式

此範例會顯示可使用公有金鑰 0 的格式。

cURL

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-
aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-
data/public-keys/0/
openssh-key
```

PowerShell

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-
seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
 GET -Uri http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key
openssh-key
```

取得公開金鑰 0 (以開放SSH金鑰格式)

此範例會取得公開金鑰 0 (採用 Open SSH 金鑰格式)。

cURL

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-
aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-
data/public-keys/0/openssh-key
ssh-rsa MIICiTCCAfICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAsTC0lBTSBDb25zb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
VQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25z
b2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFt
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE my-public-key
```

PowerShell

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-
seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
 GET -Uri http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key
ssh-rsa MIICiTCCAfICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAsTC0lBTSBDb25zb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
VQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25z
b2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFt
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE my-public-key
```

取得執行個體的子網 ID

此範例會取得執行個體的子網 ID。

cURL

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-
aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-
data/network/interfaces/macs/02:29:96:8f:6a:2d/subnet-id
subnet-be9b61d7
```

PowerShell

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-
seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -
Method GET -Uri http://169.254.169.254/latest/meta-data/network/interfaces/
macs/02:29:96:8f:6a:2d/subnet-id
subnet-be9b61d7
```

取得執行個體的執行個體標籤

如果已開啟執行個體中繼資料中執行個體標籤的存取權，您可以從執行個體中繼資料取得執行個體的標籤。如需詳細資訊，請參閱從執行個體中繼資料擷取標籤。

範例 IMDSv1

在 Amazon 執行EC2個體上執行下列範例，以擷取的執行個體中繼資料IMDSv1。

在視窗執行個體上，您可以使用視窗 PowerShell ，也可以安裝 c URL 或 wget。如果您在 Windows 執行個體上安裝協力廠商工具，請務必仔細閱讀隨附的文件，因為呼叫和輸出可能與此處所述不同。

範例

- 取得執行個體中繼資料的可用版本
- 取得上層中繼資料項目
- 取得中繼資料項目的值
- 取得可用公有金鑰清單
- 顯示可使用公有金鑰 0 的格式
- 取得公開金鑰 0 (以開放SSH金鑰格式)
- 取得執行個體的子網 ID
- 取得執行個體的執行個體標籤

取得執行個體中繼資料的可用版本

此範例會取得執行個體中繼資料的可用版本。每個版本會參照在發佈新執行個體中繼資料類別時的執行個體中繼資料建置。執行個體中繼資料建置版本與 Amazon EC2 API 版本不相關。若您有依存於先前版本中結構和資訊的指令碼，您也可以取得先前版本。

cURL

```
[ec2-user ~]$ curl http://169.254.169.254/
1.0
2007-01-19
```

```
2007-03-01
2007-08-29
2007-10-10
2007-12-15
2008-02-01
2008-09-01
2009-04-04
2011-01-01
2011-05-01
2012-01-12
2014-02-25
2014-11-05
2015-10-20
2016-04-19
...
latest
```

PowerShell

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/
1.0
2007-01-19
2007-03-01
2007-08-29
2007-10-10
2007-12-15
2008-02-01
2008-09-01
2009-04-04
2011-01-01
2011-05-01
2012-01-12
2014-02-25
2014-11-05
2015-10-20
2016-04-19
...
latest
```

取得上層中繼資料項目

此範例會取得最上層的中繼資料項目。有關響應中物件的更多內容，敬請參閱執行個體中繼資料分類。

請注意，只有在您允許存取的情況下，標籤才會包含在此輸出中。如需詳細資訊，請參閱the section called “允許存取執行個體中繼資料中的標籤”。

cURL

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/
ami-id
ami-launch-index
ami-manifest-path
block-device-mapping/
events/
hostname
iam/
instance-action
instance-id
instance-type
local-hostname
local-ipv4
mac
metrics/
network/
placement/
profile
public-hostname
public-ipv4
public-keys/
reservation-id
security-groups
services/
tags/
```

PowerShell

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/
ami-id
ami-launch-index
ami-manifest-path
block-device-mapping/
hostname
iam/
instance-action
instance-id
instance-type
```

```
local-hostname
local-ipv4
mac
metrics/
network/
placement/
profile
public-hostname
public-ipv4
public-keys/
reservation-id
security-groups
services/
tags/
```

取得中繼資料項目的值

這些範例會取得上一個範例中取得的某些頂層中繼資料項目的值。

cURL

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/ami-id
ami-0abcdef1234567890
```

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/reservation-id
r-0efghijk987654321
```

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/local-hostname
ip-10-251-50-12.ec2.internal
```

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/public-hostname
ec2-203-0-113-25.compute-1.amazonaws.com
```

PowerShell

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/ami-id
ami-0abcdef1234567890
```

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/reservation-
id
```

```
r-0efghijk987654321
```

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/local-
hostname
ip-10-251-50-12.ec2.internal
```

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/public-
hostname
ec2-203-0-113-25.compute-1.amazonaws.com
```

取得可用公有金鑰清單

此範例會取得可用公有金鑰的清單。

cURL

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/public-keys/
0=my-public-key
```

PowerShell

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/public-
keys/ 0=my-public-key
```

顯示可使用公有金鑰 0 的格式

此範例會顯示可使用公有金鑰 0 的格式。

cURL

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/public-keys/0/
openssh-key
```

PowerShell

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/public-
keys/0/openssh-key
openssh-key
```

取得公開金鑰 0 (以開放SSH金鑰格式)

此範例會取得公開金鑰 0 (採用 Open SSH 金鑰格式)。

cURL

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key
ssh-rsa MIICiTCCAfICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAsTC0lBTSBDb25zb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
VQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25z
b2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFt
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE my-public-key
```

PowerShell

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/public-
keys/0/openssh-key
ssh-rsa MIICiTCCAfICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAsTC0lBTSBDb25zb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
VQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25z
b2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFt
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE my-public-key
```

取得執行個體的子網 ID

此範例會取得執行個體的子網 ID。

cURL

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/network/interfaces/
macs/02:29:96:8f:6a:2d/subnet-id
subnet-be9b61d7
```

PowerShell

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/network/
interfaces/macs/02:29:96:8f:6a:2d/subnet-id
subnet-be9b61d7
```

取得執行個體的執行個體標籤

如果已開啟執行個體中繼資料中執行個體標籤的存取權，您可以從執行個體中繼資料取得執行個體的標籤。如需詳細資訊，請參閱從執行個體中繼資料擷取標籤。

查詢調節

我們將查詢限制為每個執行個體的基礎，並限制執行個體與. IMDS IMDS

如果您使用擷取 AWS 安全認證，請避免在每次交易期間查詢認證，或從大量執行緒或處理序同時查詢認證，因為這可能會導致限制。IMDS相反的，我們建議您快取登入資料，直到其接近到期時間為止。如需與角色相關聯之IAM角色和安全性認證的詳細資訊，請參閱從執行個體中繼資料中擷取安全登入資料。

如果您在存取時受到限制IMDS，請使用指數輪詢策略重試查詢。

限制執行個體中繼資料服務的存取

您可以考慮使用本機防火牆規則來停用部分或所有處理序存取執行個體中繼資料服務 (IMDS)。

對於 AWS Nitro System 上建置的執行個體，當您內部的網路設備 (例如虛擬路由器) 將封包轉送至IMDS位址VPC，並停用執行個體的預設來源/目標檢查時，就IMDS可以從您自己的網路連線。若要防止來源不在您VPC觸及IMDS，建議您修改網路應用裝置的組態，以丟棄具有目標IPv4位址的封包IMDS`169.254.169.254`，如果您啟用了IPv6端點，則會丟棄 IMDS`[fd00:ec2::254]`. IPv6

限制 Linux 執行個體的IMDS存取

使用 iptables 限制存取

下列範例會使用 Linux iptables 及其 `owner` 模組以防止 Apache Web 伺服器 (依據 `apache` 的預設安裝使用者 ID) 存取 169.254.169.254。它會使用拒絕規則來拒絕以該使用者身分執行的任何程序的所有執行個體中繼資料要求 (無論是否 IMDSv1IMDSv2)。

```
$ sudo iptables --append OUTPUT --proto tcp --destination 169.254.169.254 --match owner
 --uid-owner apache --jump REJECT
```

或者，您可以使用允許規則，考慮只允許存取特定使用者或群組。就安全觀點而言，允許規則可能更加簡單，因為它們會要求您做出關於那些軟體需要存取執行個體中繼資料的決策。如果您使用允許規則，即使您稍後變更執行個體上的軟體或組態，您也不太可能意外地允許軟體存取中繼資料服務 (亦即您並非有意存取)。您也可以使用允許規則來組合群組使用方式，如此您就能新增並從允許的群組移除使用者，而不需要變更防火牆規則。

下列範例會防止所有處理序存取，但在使用者帳戶中執行的處理程序除外`trustworthy-user`。IMDS

```
$ sudo iptables --append OUTPUT --proto tcp --destination 169.254.169.254 --match
 owner ! --uid-owner trustworthy-user --jump REJECT
```

> **Note**
>
> - 若要使用本機防火牆規則，您需要採用之前的範例命令，才能滿足您的需求。
> - 依預設，iptables 規則在系統重新啟動時不會持續存在。但可使用 OS 功能將之設定為持續存在，但此處並不會敘述此內容。
> - 如果此群組是指定本機使用者的主要群組，iptables `owner` 模組僅會比對群組成員資格。而不會比對其他群組。

使用 PF 或限IPFW制訪問

如果您使用的是免費BSD或開放BSD，您也可以考慮使用 PF 或IPFW. 下列範例會限制只有 root 使用者的存取權。IMDS

PF

```
$ block out inet proto tcp from any to 169.254.169.254
```

```
$ pass out inet proto tcp from any to 169.254.169.254 user root
```

IPFW

```
$ allow tcp from any to 169.254.169.254 uid root
```

```
$ deny tcp from any to 169.254.169.254
```

> **Note**
>
> PF 和IPFW命令的順序很重要。PF 預設為上次比對規則，並IPFW預設為第一個相符規則。

限制視窗執行個體的IMDS存取

使用 Windows 防火牆以限制存取

下列 PowerShell 範例會使用內建的 Windows 防火牆來防止網際網路資訊伺服器網路伺服器 (根據其預設安裝使用者識別碼NT AUTHORITY\IUSR) 存取 169.254.169.254。它會使用拒絕規則來拒絕以該使用者身分執行的任何程序的所有執行個體中繼資料要求 (無論是否 IMDSv1IMDSv2)。

```
PS C:\> $blockPrincipal = New-Object -TypeName System.Security.Principal.NTAccount ("NT
 AUTHORITY\IUSR")
PS C:\> $BlockPrincipalSID =
 $blockPrincipal.Translate([System.Security.Principal.SecurityIdentifier]).Value
PS C:\> $BlockPrincipalSDDL = "D:(A;;CC;;;$BlockPrincipalSID)"
PS C:\> New-NetFirewallRule -DisplayName "Block metadata service from IIS" -Action
 block -Direction out `
-Protocol TCP -RemoteAddress 169.254.169.254 -LocalUser $BlockPrincipalSDDL
```

或者，您可以使用允許規則，考慮只允許存取特定使用者或群組。就安全觀點而言，允許規則可能更加簡單，因為它們會要求您做出關於那些軟體需要存取執行個體中繼資料的決策。如果您使用允許規則，即使您稍後變更執行個體上的軟體或組態，您也不太可能意外地允許軟體存取中繼資料服務 (亦即您並非有意存取)。您也可以使用允許規則來組合群組使用方式，如此您就能新增並從允許的群組移除使用者，而不需要變更防火牆規則。

下列範例會防止以變數 `blockPrincipal` 中指定 OS 群組身分執行的所有程序存取執行個體中繼資料 (在此範例中，Windows 群組 `Everyone`)，但 `exceptionPrincipal` 中指定的程序除外 (在此範例中，是名為 `trustworthy-users` 的群組)。您必須指定拒絕和允許原則，因為 Windows 防火牆 (與 Linux iptables 中的 ！ `--uid-owner trustworthy-user` 規則不同) 不會提供快速鍵機制，以透過拒絕所有其他規則而僅允許特定原則 (使用者或群組)。

```
PS C:\> $blockPrincipal = New-Object -TypeName System.Security.Principal.NTAccount
 ("Everyone")
PS C:\> $BlockPrincipalSID =
 $blockPrincipal.Translate([System.Security.Principal.SecurityIdentifier]).Value
PS C:\> $exceptionPrincipal = New-Object -TypeName System.Security.Principal.NTAccount
 ("trustworthy-users")
PS C:\> $ExceptionPrincipalSID =
 $exceptionPrincipal.Translate([System.Security.Principal.SecurityIdentifier]).Value
PS C:\> $PrincipalSDDL = "O:LSD:(D;;CC;;;$ExceptionPrincipalSID)(A;;CC;;;
$BlockPrincipalSID)"
PS C:\> New-NetFirewallRule -DisplayName "Block metadata service for
 $($blockPrincipal.Value), exception: $($exceptionPrincipal.Value)" -Action block -
Direction out `
-Protocol TCP -RemoteAddress 169.254.169.254 -LocalUser $PrincipalSDDL
```

> **Note**
>
> 若要使用本機防火牆規則，您需要採用之前的範例命令，才能滿足您的需求。

使用 netsh 規則限制存取

您可以使用 `netsh` 而考慮封鎖所有軟體，但那些規則較不彈性。

```
C:\> netsh advfirewall firewall add rule name="Block metadata service altogether"
 dir=out protocol=TCP remoteip=169.254.169.254 action=block
```

> **Note**
>
> - 若要使用本機防火牆規則，您需要採用之前的範例命令，才能滿足您的需求。
> - 必須從提升權限的命令提示設定 `netsh`，但不可設定為拒絕或允許特定原則。

設定執行處理中繼資料服務選

執行個體中繼資料服務 (IMDS) 會在每個執行個體本機EC2執行。執行個體中繼資料選項是指控制執行個體上的協助工具和行為IMDS的一組組EC2態。

您可以為每個執行個體設定下列執行個體中繼資料選項：

執行個體中繼資料服務 (IMDS): `enabled` | `disabled`

> 您可以在執行個體IMDS上啟用或停用。停用時，您或任何程式碼都無法存取執行個體上的執行個體中繼資料。
>
> 執行個體上IMDS有兩個端點：IPv4(`169.254.169.254`) 和 IPv6 (`[fd00:ec2::254]`)。當您啟用時IMDS，IPv4端點會自動啟用。如果要啟用IPv6端點，則需要明確執行此操作。

IMDSIPv6端點:`enabled`| `disabled`

> 您可以在執行個體上明確啟用IPv6IMDS端點。啟用IPv6端點後，IPv4端點會保持啟用狀態。只有在AWS Nitro System 上建置的執行個體和支援的子網路 (雙堆疊或IPv6僅限) 中的執行個體上才IPv6支援IPv6端點。

中繼資料版本:`IMDSv1 or IMDSv2 (token optional)`| `IMDSv2 only (token required)`

> 要求執行個體中繼資料時，IMDSv2呼叫需要 Token。IMDSv1調用不需要令牌。您可以將執行個體設定為允許IMDSv1或IMDSv2呼叫 (其中 Token 為選用項目)，或僅允許IMDSv2呼叫 (需要 Token)。

中繼資料回應躍點限制：1— 64

> 躍點限制是允許PUT回應建立的網路躍點數目。您可以將躍點限制設定為最小值1和最大值64。在容器環境中，建議將躍點限制設定為2。如需詳細資訊，請參閱執行個體元資料存取。

存取執行個體中繼資料中的標籤：`enabled`| `disabled`

> 您可以從執行個體的中繼資料啟用或停用執行個體標籤的存取權。如需詳細資訊，請參閱使用執行個體中繼資料檢視EC2執行個體的。

執行個體中繼資料選項的設定

執行個體中繼資料選項可以在不同層級設定，如下所示：

- 帳戶 — 您可以在帳戶層級為每個執行個體中繼資料選項設定預設值 AWS 區域。啟動執行個體時，執行個體中繼資料選項會自動設定為帳戶層級值。您可以在啟動時變更這些值。帳戶層級的預設值不會影響現有例項。

- AMI— 您可以在註冊或修改`imds-support`參數`v2.0`時將參數設定為AMI。使用此功能啟動執行個體時AMI，執行個體中繼資料版本會自動設定為，IMDSv2且躍點限制設定為 2。
- 執行個體 — 您可以在啟動時變更執行個體的所有執行個體中繼資料選項，覆寫預設設定。您也可以在執行中或已停止的執行個體上啟動後，變更執行個體中繼資料選項。請注意，變更可能受到IAM或SCP原則的限制。

如需詳細資訊，請參閱 設定新執行個體的執行個體中繼資料選項 和 修改現有執行個體的執行個體中繼資料選項。

執行個體中繼資料選項的優先順序

每個執行個體中繼資料選項的值都會在執行個體啟動時決定，並遵循階層優先順序。最上方優先順序最高的階層如下所示：

- 優先順序 1：啟動時的執行個體組態 — 您可以在啟動範本或執行個體組態中指定值。此處指定的任何值都會覆寫在帳戶層級或中指定的值AMI。
- 優先順序 2：帳戶設定 — 如果在執行個體啟動時未指定值，則由帳戶層級設定決定 (針對每個 AWS 區域設定)。帳戶層級設定可以包含每個中繼資料選項的值，或完全不指定任何偏好設定。
- 優先順序 3：AMI組態 — 如果未在執行個體啟動時或帳戶層級指定值，則由AMI組態決定。這僅會套用到 `HttpTokens` 和 `HttpPutResponseHopLimit`。

每個中繼資料選項都會分別評估。執行個體可以混合使用直接執行個體組態、帳戶層級預設值，以及. AMI

您可以在執行中或已停止的執行個體上啟動後變更任何中繼資料選項的值，除非變更受IAM或SCP政策的限制。

決定中繼資料選項的值-範例 1

在此範例中，EC2執行個體會啟動至「`HttpPutResponseHopLimit`區域」，其中的帳戶層級設定為1。指定的AMI已`ImdsSupport`設定為`v2.0`。啟動時不會直接在執行個體上指定中繼資料選項。執行個體會使用下列中繼資料選項啟動：

```
"MetadataOptions": {
    ...
    "HttpTokens": "required",
    "HttpPutResponseHopLimit": 1,
    ...
```

這些值的確定方式如下：

- 啟動時未指定中繼資料選項：在執行個體啟動期間，執行個體啟動參數或啟動範本中未提供中繼資料選項的特定值。
- 科目設定採用下一個優先順序：如果在啟動時沒有指定的特定值，則會優先考慮「區域」內帳戶層次的設定。這表示會套用在帳戶層級設定的預設值。在此情況下，會將設定HttpPutResponseHopLimit為1。
- AMI設定優先順序為最後：如果在啟動時或帳戶層級 HttpTokens (執行個體中繼資料版本) 沒有指定的特定值，則會套用此AMI設定。在這種情況下，AMI設置ImdsSupport: v2.0確定的設置HttpTokens被設置為required。請注意，雖然AMI設ImdsSupport: v2.0定已設定為設定HttpPutResponseHopLimit: 2，但帳戶層級設定已覆寫HttpPutResponseHopLimit: 1，該設定具有較高優先順序。

決定中繼資料選項的值-範例 2

在此範例中，EC2執行個體的啟動設定與上一個範例 1 相同，但在啟動時將HttpTokens設定為optional直接在執行個體上。執行個體會使用下列中繼資料選項啟動：

```
"MetadataOptions": {
    ...
    "HttpTokens": "optional",
    "HttpPutResponseHopLimit": 1,
    ...
```

的值HttpPutResponseHopLimit是以與範例 1 中相同的方式來決定。不過，的值決定如下：啟動時在執行個體上設定的中繼資料選項優先。HttpTokens即使已使用 ImdsSupport: v2.0 (換句話說，HttpTokens設定AMI為required) 進行設定，但在啟動時 (HttpTokens設為optional) 在執行個體上指定的值優先。

設定執行個體元資料版本

啟動執行個體時，執行個體中繼資料版本的值為IMDSv1 or IMDSv2 (token optional)或IMDSv2 only (token required)。

執行個體啟動時，您可以手動指定中繼資料版本的值，或使用預設值。如果您手動指定值，它會取代任何預設值。如果您選擇不手動指定值，它將由預設設定組合決定，如下表所述。

此表格顯示啟動時執行處理的中繼資料版本 (由欄 4 中的「結果執行處理組態」指示) 如何由不同組態層級的設定決定。優先順序是從左到右，其中第一欄的優先順序最高，如下所示：

- 欄 1：啟動參數 — 代表您在啟動時手動指定之執行處理的設定。
- 欄 2：帳戶層級預設值 — 代表帳戶的設定。
- 欄 3：AMI預設 — 代表上的設定AMI。

啟動參數	帳戶層級預設值	AMI默認	產生的實例配置
僅 V2（需要令牌）	無偏好	僅限 V2	僅限 V2
僅 V2（需要令牌）	僅限 V2	僅限 V2	僅限 V2
僅 V2（需要令牌）	V1 或 V2	僅限 V2	僅限 V2
V1 或 V2（令牌可選）	無偏好	僅限 V2	V1 或 V2
V1 或 V2（令牌可選）	僅限 V2	僅限 V2	V1 或 V2
V1 或 V2（令牌可選）	V1 或 V2	僅限 V2	V1 或 V2
未設定	無偏好	僅限 V2	僅限 V2
未設定	僅限 V2	僅限 V2	僅限 V2
未設定	V1 或 V2	僅限 V2	V1 或 V2
僅 V2（需要令牌）	無偏好	null	僅限 V2
僅 V2（需要令牌）	僅限 V2	null	僅限 V2
僅 V2（需要令牌）	V1 或 V2	null	僅限 V2
V1 或 V2（令牌可選）	無偏好	null	V1 或 V2
V1 或 V2（令牌可選）	僅限 V2	null	V1 或 V2

啟動參數	帳戶層級預設值	AMI默認	產生的實例配置
V1 或 V2（令牌可選）	V1 或 V2	null	V1 或 V2
未設定	無偏好	null	V1 或 V2
未設定	僅限 V2	null	僅限 V2
未設定	V1 或 V2	null	V1 或 V2

使用IAM條件鍵限制執行個體中繼資料選項

您可以在IAM策略中使用IAM條件金鑰，或SCP如下所示：

- 僅當執行個體設定為需要使用時，才允許啟動 IMDSv2
- 限制允許的轉跳數目
- 關閉對執行個體中繼資料的存取

任務

- 設定新執行個體的執行個體中繼資料選項
- 修改現有執行個體的執行個體中繼資料選項

> **Note**
>
> 您應該謹慎繼續執行和仔細進行測試，然後再進行任何變更。請記得下列事項：
>
> - 如果您強制使用IMDSv2，用於執行個體中繼資料存取的應IMDSv1用程式或代理程式將會中斷。
> - 如果您關閉對執行個體中繼資料的所有存取，則需要存取執行個體中繼資料才能正常運作的應用程式或代理程式便會中斷。
> - 對於IMDSv2，您必須在擷取權杖`/latest/api/token`時使用。
> - (僅適用於視窗) 如果您的 PowerShell 版本早於 4.0，您必須更新至視窗管理架構 4.0，才能要求使用IMDSv2.

設定新執行個體的執行個體中繼資料選項

您可以為新執行個體設定下列執行個體中繼資料選項。

選項

- 需要使用 IMDSv2
- 啟用IMDSIPv4和IPv6端點
- 關閉對執行個體中繼資料的存取

需要使用 IMDSv2

您可以使用下列方法來要求在新執行個體IMDSv2上使用。

要求 IMDSv2

- 設定IMDSv2為帳戶的預設值
- 啟動時設定執行個體
- 配置 AMI
- 使用IAM策略

設定IMDSv2為帳戶的預設值

您可以在帳戶層級為每個執行個體中繼資料服務 (IMDS) 設定預設版本 AWS 區域。這表示當您啟動新執行個體時，執行個體中繼資料版本會自動設定為帳戶層級的預設值。不過，您可以在啟動時或啟動後手動覆寫值。如需帳戶層級設定和手動覆寫如何影響執行個體的詳細資訊，請參閱。執行個體中繼資料選項的優先順序

> **Note**
>
> 設定帳戶層級預設值並不會重設現有執行個體。例如，如果您將帳戶層級預設值設定為IMDSv2，則設定為的任何現有執行個體IMDSv1都不會受到影響。如果要變更現有例證上的值，您必須手動變更例證本身的值。

您可以將執行個體中繼資料版本的帳戶預設值設定為，以IMDSv2便帳戶中的所有新執行個體都以IMDSv2必要的方式啟動，而且IMDSv1會停用。使用此帳戶的預設值，當您啟動執行個體時，執行個體的預設值如下：

- 主控台：中繼資料版本僅設為 V2 (需要 Token)，而中繼資料回應躍點限制設定為 2。
- AWS CLI: `HttpTokens` 設定為`required`且`HttpPutResponseHopLimit`設定為2。

> **Note**
>
> 將帳戶預設值設定為之前IMDSv2，請確定您的執行個體不依賴IMDSv1。如需詳細資訊，請參閱推薦的需求路徑 IMDSv2。

Console

若要設定IMDSv2為指定區域帳戶的預設值

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 若要變更 AWS 區域，請使用頁面右上角的「地區」選取器。
3. 在導覽窗格中，選擇 [EC2儀表板]。
4. 在帳戶屬性下，選擇資料保護和安全性。
5. 選擇IMDS預設值旁邊的 [管理]。
6. 在 [管理IMDS預設值] 頁面上，執行下列動作：
   a. 針對執行個體中繼資料服務，選擇 [啟
   b. 針對 Metadata version (中繼資料版本)，選擇 V2 only (token required) (僅限 V2 (需要權杖))。
   c. 對於中繼資料回應躍點限制，如果執行個體將託管容器，請指定 2。否則，請選取無偏好設定。如果未指定偏好設定，則在啟動時，如果AMI需要，值預設為 2IMDSv2；否則預設值為 1。
   d. 選擇更新。

AWS CLI

若要設定IMDSv2為指定區域帳戶的預設值

使用指modify-instance-metadata-defaults令並指定要在其中修改IMDS帳戶層級設定的「區域」。2如果您的執行個體將託管容器，則 `--http-put-response-hop-limit`「包含`--http-tokens`」設定為`required`和設定為。否則，請指定`-1`以指示沒有偏好。如果 `-1` (未指定偏好設定)，則在啟動時，2如果AMI需要，值預設為IMDSv2；否則預設為1。

```
aws ec2 modify-instance-metadata-defaults \
    --region us-east-1 \
    --http-tokens required \
    --http-put-response-hop-limit 2
```

預期的輸出結果

```
{
    "Return": true
}
```

若要檢視指定區域的執行個體中繼資料選項的預設帳戶設定

使用指get-instance-metadata-defaults令並指定「區域」。

```
aws ec2 get-instance-metadata-defaults --region us-east-1
```

範例輸出

```
{
    "AccountLevel": {
        "HttpTokens": "required",
        "HttpPutResponseHopLimit": 2
    }
}
```

若要設定IMDSv2為所有區域帳戶的預設值

使用指modify-instance-metadata-defaults令修改所有區域的IMDS帳戶層級設定。2如果您的執行個體將託管容器，則 `--http-put-response-hop-limit`「包含`--http-tokens`」設定為`required`和設定為。否則，請指定`-1`以指示沒有偏好。如果 `-1` (未指定偏好設定)，則在啟動時，2如果AMI需要，值預設為IMDSv2；否則預設為1。

```
echo -e "Region          \t Modified" ; \
echo -e "--------------  \t ---------" ; \
for region in $(
    aws ec2 describe-regions \
        --region us-east-1 \
        --query "Regions[*].[RegionName]" \
```

```
        --output text
    );
    do (output=$(
        aws ec2 modify-instance-metadata-defaults \
            --region $region \
            --http-tokens required \
            --http-put-response-hop-limit 2 \
            --output text)
        echo -e "$region        \t $output"
    );
done
```

預期的輸出結果

```
Region                  Modified
--------------          ---------
ap-south-1              True
eu-north-1              True
eu-west-3               True
...
```

若要檢視所有區域的執行個體中繼資料選項的預設帳戶設定

使用指get-instance-metadata-defaults令。

```
echo -e "Region   \t Level          Hops    HttpTokens" ; \
echo -e "-------------- \t ------------   ----    ----------" ; \
for region in $(
    aws ec2 describe-regions \
        --region us-east-1 \
        --query "Regions[*].[RegionName]" \
        --output text
    );
    do (output=$(
        aws ec2 get-instance-metadata-defaults \
            --region $region \
            --output text)
        echo -e "$region \t $output"
    );
done
```

預期的輸出結果

```
Region           Level          Hops    HttpTokens
--------------   ------------   ----    ----------
ap-south-1       ACCOUNTLEVEL   2       required
eu-north-1       ACCOUNTLEVEL   2       required
eu-west-3        ACCOUNTLEVEL   2       required
...
```

PowerShell

若要設定IMDSv2為指定區域帳戶的預設值

使用指Edit-EC2InstanceMetadataDefault令並指定要在其中修改IMDS帳戶層級設定的「區域」。2如果您的執行個體將託管容器，則 `-HttpPutResponseHopLimit`「包含`-HttpToken`」設定為`required`和設定為。否則，請指定`-1`以指示沒有偏好。如果 `-1` (未指定偏好設定)，則在啟動時，2如果AMI需要，值預設為IMDSv2；否則預設為1。

```
Edit-EC2InstanceMetadataDefault `
    -Region us-east-1 `
    -HttpToken required `
    -HttpPutResponseHopLimit 2
```

預期的輸出結果

```
True
```

若要檢視指定區域的執行個體中繼資料選項的預設帳戶設定

使用指Get-EC2InstanceMetadataDefault令並指定「區域」。

```
Get-EC2InstanceMetadataDefault -Region us-east-1 | Format-List
```

範例輸出

```
HttpEndpoint            :
HttpPutResponseHopLimit : 2
HttpTokens              : required
InstanceMetadataTags    :
```

若要設定IMDSv2為所有區域帳戶的預設值

使用指Edit-EC2InstanceMetadataDefault令程式來修改所有區域的IMDS帳戶層級設定。2如果您的執行個體將託管容器，則 -HttpPutResponseHopLimit「包含-HttpToken」設定為required和設定為。否則，請指定-1以指示沒有偏好。如果 -1 (未指定偏好設定)，則在啟動時，2如果AMI需要，值預設為IMDSv2；否則預設為1。

```
(Get-EC2Region).RegionName | `
    ForEach-Object {
    [PSCustomObject]@{
        Region   = $_
        Modified = (Edit-EC2InstanceMetadataDefault `
                -Region $_ `
                -HttpToken required `
                -HttpPutResponseHopLimit 2)
    }
} | `
Format-Table Region, Modified -AutoSize
```

預期的輸出結果

```
Region         Modified
------         --------
ap-south-1         True
eu-north-1         True
eu-west-3          True
...
```

若要檢視所有區域的執行個體中繼資料選項的預設帳戶設定

使用指Get-EC2InstanceMetadataDefault令程式。

```
(Get-EC2Region).RegionName | `
    ForEach-Object {
    [PSCustomObject]@{
        Region = $_
        HttpPutResponseHopLimit = (Get-EC2InstanceMetadataDefault -Region
 $_).HttpPutResponseHopLimit
        HttpTokens              = (Get-EC2InstanceMetadataDefault -Region
 $_).HttpTokens
    }
} | `
Format-Table -AutoSize
```

範例輸出

```
Region         HttpPutResponseHopLimit HttpTokens
------         ----------------------- ----------
ap-south-1                           2 required
eu-north-1                           2 required
eu-west-3                            2 required
...
```

啟動時設定執行個體

啟動執行個體時，您可以設定下列欄位，將執行個體IMDSv2設定為需要使用：

- Amazon EC2 主控台：將中繼資料版本設定為僅 V2（需要令牌）。
- AWS CLI：將 `HttpTokens` 設為 `required`。

當您指定必要項目時，也必須將「中繼資料」設定為「已啟用」(主控台IMDS) 或`HttpEndpoint`設定為 ()，以啟用「執行個體中繼資料服務」`enabled` (AWS CLI) 端點。IMDSv2

在容器環境中，如果IMDSv2需要，建議將躍點限制設定為2。如需詳細資訊，請參閱執行個體元資料存取。

Console

若要求在新執行個體IMDSv2上使用

- 在 Amazon EC2 主控台中啟動新執行個體時，展開進階詳細資料，然後執行以下操作：
  - 針對 Metadata accessible (可存取中繼資料)，選擇 Enabled (已啟用)。
  - 針對 Metadata version (中繼資料版本)，選擇 V2 only (token required) (僅限 V2 (需要權杖))。
  - (容器環境) 對於中繼資料回應躍點限制，請選擇 2。

  如需詳細資訊，請參閱進階詳細資訊。

AWS CLI

若要求在新執行個體IMDSv2上使用

下列 run-instances 範例會啟動 `c6i.large` 設定為 `--metadata-options` 的`HttpTokens=required` 執行個體。當您為 `HttpTokens` 指定值時，您也必須將 `HttpEndpoint` 設定為 `enabled`。由`required`於中繼資料擷取要求的安全權杖標頭設定為，因此在要求執行個體中繼資料IMDSv2時必須使用執行個體。

在容器環境中，如果IMDSv2需要，我們建議將躍點限制設定為2與`HttpPutResponseHopLimit=2`。

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --instance-type c6i.large \
 ...
    --metadata-options
 "HttpEndpoint=enabled,HttpTokens=required,HttpPutResponseHopLimit=2"
```

PowerShell

若要求在新執行個體IMDSv2上使用

下列New-EC2Instance指令程式範例會啟動`MetadataOptions_HttpEndpoint`設定為`enabled`且`MetadataOptions_HttpTokens`參數為的`c6i.large`執行個體。`required`當您為 `HttpTokens` 指定值時，您也必須將 `HttpEndpoint` 設定為 `enabled`。由`required`於中繼資料擷取要求的安全權杖標頭設定為，因此在要求執行個體中繼資料IMDSv2時必須使用執行個體。

```
New-EC2Instance `
    -ImageId ami-0abcdef1234567890 `
    -InstanceType c6i.large `
    -MetadataOptions_HttpEndpoint enabled `
    -MetadataOptions_HttpTokens required
```

AWS CloudFormation

若要使用指定執行個體的中繼資料選項 AWS CloudFormation，請參閱《使用指南》中的AWSEC2 AWS CloudFormation ::: LaunchTemplate MetadataOptions 屬性。

配置 AMI

註冊新的AMI或修改現有的參數時AMI，您可以將`imds-support`參數設定為`v2.0`。從此啟動的執行個體AMI會將中繼資料版本設定為僅 V2 (需要 Token) (主控台) 或`HttpTokens`設定為 `required` (AWS CLI)。透過這些設定，執行個體需要在要求執行IMDSv2個體中繼資料時使用此設定。

請注意，當您設定`imds-support`為時`v2.0`，從此項啟動的執行個體也AMI會將中繼資料回應躍點限制 (主控台AWS CLI) 或 `http-put-response-hop-limit` () 設定為 2。

> **Important**
>
> 除非您的AMI軟體支援，否則請勿使用此參數IMDSv2。將值設為 `v2.0` 後，將無法復原。「重置」您的唯一方法AMI是AMI從基礎快照創建一個新的。

若要設定新AMI的 IMDSv2

使用下列其中一種方法來配置新AMI的IMDSv2。

AWS CLI

下列暫存器映像範例會AMI使用EBS根磁碟區的指定快照來註冊為裝置。`/dev/xvda`請`v2.0`為`imds-support`參數指定，以便從此AMI啟動的執行個體在要求執行個體中繼資料時需要使用該IMDSv2執行個體。

```
aws ec2 register-image \
    --name my-image \
    --root-device-name /dev/xvda \
    --block-device-mappings DeviceName=/dev/
xvda,Ebs={SnapshotId=snap-0123456789example} \
    --architecture x86_64 \
    --imds-support v2.0
```

PowerShell

下列 Register-EC2ImageCmdlet 範例會AMI使用指定的EBS根磁碟區快照註冊為裝置`/dev/xvda`。請`v2.0`為`ImdsSupport`參數指定，以便從此AMI啟動的執行個體在要求執行個體中繼資料時需要使用該IMDSv2執行個體。

```
Register-EC2Image `
    -Name 'my-image' `
    -RootDeviceName /dev/xvda `
    -BlockDeviceMapping  (
    New-Object `
        -TypeName Amazon.EC2.Model.BlockDeviceMapping `
        -Property @{
        DeviceName = '/dev/xvda';
```

```
        EBS        = (New-Object -TypeName Amazon.EC2.Model.EbsBlockDevice -Property
 @{
                SnapshotId = 'snap-0123456789example'
                VolumeType = 'gp3'
                } )
        }  ) `
    -Architecture X86_64 `
    -ImdsSupport v2.0
```

若要設定現有AMI的 IMDSv2

使用下列其中一種方法來配置現有AMI的IMDSv2。

AWS CLI

下列modify-image-attribute範例IMDSv2只會修改現有AMI的。請`v2.0`為`imds-support`參數指定，以便從此AMI啟動的執行個體在要求執行個體中繼資料時需要使用該IMDSv2執行個體。

```
aws ec2 modify-image-attribute \
    --image-id ami-0123456789example \
    --imds-support v2.0
```

PowerShell

下列Edit-EC2ImageAttribute指令程式範例IMDSv2只會修改現有AMI的。請`v2.0`為`imds-support`參數指定，以便從此AMI啟動的執行個體在要求執行個體中繼資料時需要使用該IMDSv2執行個體。

```
Edit-EC2ImageAttribute `
    -ImageId ami-0abcdef1234567890 `
    -ImdsSupport 'v2.0'
```

使用IAM策略

您可以建立IAM政策，防止使用者啟動新執行個體，除非他們需要新執IMDSv2行個體。

使用IAM政策強制IMDSv2在所有新執行個體上使用

若要確保使用者只能啟動要求執行個體中繼資料IMDSv2時需要使用的執行個體，您可以指定IMDSv2必須符合要求的條件，才能啟動執行個體。如需原IAM則範例，請參閱使用執行個體中繼資料。

啟用IMDSIPv4和IPv6端點

執行個體上IMDS有兩個端點：IPv4(`169.254.169.254`) 和 IPv6 (`[fd00:ec2::254]`)。當您啟用時IMDS，IPv4端點會自動啟用。即使您將執行個體啟動至IPv6唯一子網路，IPv6端點仍會保持停用狀態。要啟用IPv6端點，您需要明確執行此操作。當您啟用端點時，IPv6端點會IPv4保持啟用狀態。

您可以在執行個體啟動時或之後啟用IPv6端點。

啟用IPv6端點的需求

- 選取的執行個體類型建立在 AWS Nitro 系統上。
- 選取的子網路支援IPv6，其中子網路可能是雙堆疊或IPv6僅。

使用下列任一方法來啟動IMDSIPv6端點已啟用的執行個體。

New console

若要在執行個體啟動時啟用IMDSIPv6端點

- 在 Amazon EC2 主控台啟動執行個體，並在進階詳細資料下指定下列內容：
  - 針對中繼資料IPv6端點，選擇已啟用。

如需詳細資訊，請參閱進階詳細資訊。

AWS CLI

若要在執行個體啟動時啟用IMDSIPv6端點

下列執行執行個體範例會啟動一個已啟用IPv6端點的`c6i.large`執行個體。IMDS若要啟用IPv6端點，請為`--metadata-options`參數指定`HttpProtocolIpv6=enabled`。當您為 `HttpProtocolIpv6` 指定值時，您也必須將 `HttpEndpoint` 設定為 `enabled`。

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --instance-type c6i.large \
    ...
    --metadata-options "HttpEndpoint=enabled,HttpProtocolIpv6=enabled"
```

PowerShell

若要在執行個體啟動時啟用IMDSIPv6端點

下列New-EC2Instance指令程式範例會啟動一個已啟用的IPv6端點的c6i.large執行個體IMDS。若要啟用IPv6端點，請指定MetadataOptions_HttpProtocolIpv6為enabled。當您為 MetadataOptions_HttpProtocolIpv6 指定值時，您也必須將 MetadataOptions_HttpEndpoint 設定為 enabled。

```
New-EC2Instance `
    -ImageId ami-0abcdef1234567890 `
    -InstanceType c6i.large `
    -MetadataOptions_HttpEndpoint enabled `
    -MetadataOptions_HttpProtocolIpv6 enabled
```

關閉對執行個體中繼資料的存取

您可以在啟動執行個體IMDS時停用，以關閉執行個體中繼資料的存取權。您可以稍後重新啟用來開啟存取IMDS。如需詳細資訊，請參閱開啟對執行個體中繼資料的存取。

> **Important**
>
> 您可以選擇在啟動IMDS時或啟動後停用。如果您IMDS在啟動時停用，下列項目可能無法運作：
>
> - 您可能無法SSH存取您的執行個體。這是執行個體的公開SSH金鑰，將無法存取，因為金鑰通常會提供並從EC2執行個體中繼資料存取。public-keys/0/openssh-key
> - EC2使用者資料將無法使用，也不會在執行個體啟動時執行。EC2使用者資料託管在IMDS. 如果停用IMDS，您可以有效地關閉對使用者資料的存取。
>
> 若要存取此功能，您可以在啟動IMDS後重新啟用。

Console

啟動時關閉對執行個體中繼資料的存取

- 在 Amazon EC2 主控台啟動執行個體，並在進階詳細資料下指定下列內容：
  - 針對 Metadata accessible (可存取中繼資料)，選擇 Disabled (已停用)。

如需詳細資訊，請參閱進階詳細資訊。

AWS CLI

啟動時關閉對執行個體中繼資料的存取

將 `--metadata-options` 設為 `HttpEndpoint=disabled` 來啟動執行個體。

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --instance-type c6i.large \
    ...
    --metadata-options "HttpEndpoint=disabled"
```

PowerShell

啟動時關閉對執行個體中繼資料的存取

下列New-EC2Instance指令程式範例會啟動`MetadataOptions_HttpEndpoint`設定為`disabled`的執行個體。

```
New-EC2Instance `
    -ImageId ami-0abcdef1234567890 `
    -InstanceType c6i.large `
    -MetadataOptions_HttpEndpoint disabled
```

AWS CloudFormation

若要使用指定執行個體的中繼資料選項 AWS CloudFormation，請參閱《使用指南》中的AWSEC2 AWS CloudFormation ::: LaunchTemplate MetadataOptions 屬性。

修改現有執行個體的執行個體中繼資料選項

您可以修改現有執行個體的執行個體中繼資料選項。

您也可以建立IAM政策，防止使用者修改現有執行個體上的執行個體中繼資料選項。若要控制哪些使用者可以修改執行個體中繼資料選項，請指定策略，防止具有指定角色之使用者以外的所有使用者使用ModifyInstanceMetadataOptionsAPI。如需原IAM則範例，請參閱使用執行個體中繼資料。

需要使用 IMDSv2

請使用下列其中一種方法，修改現有執行個體上的執行個體中繼資料選項，以便在要求執行IMDSv2個體中繼資料時要求使用。如果IMDSv2需要，則IMDSv1無法使用。

> **Note**
>
> 在使用要IMDSv2求之前，請確保實例未進行IMDSv1調用。`MetadataNoTokenCloudWatch`量度會追蹤IMDSv1通話。當執行個體`MetadataNoToken`記錄零IMDSv1使用量時，該執行個體就可以使用IMDSv2。

Console

若要求在現有執行個體IMDSv2上使用

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體。
4. 選擇動作、執行個體設定、修改執行個體中繼資料選項。
5. 在修改執行個體中繼資料選項對話方塊中，執行下列動作：
   a. 針對執行個體中繼資料服務，請選取啟用。
   b. 對於 IMDSv2，選擇必要。
   c. 選擇 Save (儲存)。

AWS CLI

若要求在現有執行個體IMDSv2上使用

使用指modify-instance-metadata-optionsCLI令並將`http-tokens`參數設定為`required`。當您為 `http-tokens` 指定值時，您也必須將 `http-endpoint` 設定為 `enabled`。

```
aws ec2 modify-instance-metadata-options \
    --instance-id i-1234567898abcdef0 \
    --http-tokens required \
    --http-endpoint enabled
```

PowerShell

若要求在現有執行個體IMDSv2上使用

使用指Edit-EC2InstanceMetadataOption令程式並將`HttpTokens`參數設定為`required`。當您為 `HttpTokens` 指定值時，您也必須將 `HttpEndpoint` 設定為 `enabled`。

```
(Edit-EC2InstanceMetadataOption `
    -InstanceId i-1234567898abcdef0 `
    -HttpTokens required `
    -HttpEndpoint enabled).InstanceMetadataOptions
```

恢復使用 IMDSv1

如果需IMDSv2要，請求執行個體中繼資料時IMDSv1將無法運作。如果IMDSv2是可選的，那麼IMDSv2和都IMDSv1會起作用。因此，若要還原IMDSv1，請使用下列其中一種方法設為IMDSv2選擇性。

Console

若要還原執行個體IMDSv1的使用

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體。
4. 選擇動作、執行個體設定、修改執行個體中繼資料選項。
5. 在修改執行個體中繼資料選項對話方塊中，執行下列動作：
   a. 針對執行個體中繼資料服務，請確定已選取啟用。
   b. 對於 IMDSv2，選擇可選。
   c. 選擇 Save (儲存)。

AWS CLI

若要還原執行個體IMDSv1的使用

您可以使用 `http-tokens` set to 的modify-instance-metadata-optionsCLI命令來還原`optional`要求執行個體中繼資料IMDSv1時的使用。

```
aws ec2 modify-instance-metadata-options \
    --instance-id i-1234567898abcdef0 \
    --http-tokens optional \
    --http-endpoint enabled
```

PowerShell

若要還原執行個體IMDSv1的使用

您可以使用`HttpTokens`設定為的 Edit-EC2InstanceMetadataOptionCmdlet 來還原`optional`要求執行個體中繼資料IMDSv1時的使用狀況。

```
(Edit-EC2InstanceMetadataOption `
    -InstanceId i-1234567898abcdef0 `
    -HttpTokens optional `
    -HttpEndpoint enabled).InstanceMetadataOptions
```

變更回PUT應躍點限制

對於現有執行個體，您可以變更 PUT 回應跳轉限制的設定。

目前只 AWS SDKs支援 AWS CLI 和變更PUT回應躍點限制。

AWS CLI

若要變更回PUT應躍點限制

使用指modify-instance-metadata-optionsCLI令並將`http-put-response-hop-limit`參數設定為所需的躍點數目。在下列範例中，跳轉限制設定為 3。請注意，為 `http-put-response-hop-limit` 指定數值時，您還必須將 `http-endpoint` 設定為 `enabled`。

```
aws ec2 modify-instance-metadata-options \
    --instance-id i-1234567898abcdef0 \
    --http-put-response-hop-limit 3 \
    --http-endpoint enabled
```

PowerShell

若要變更回PUT應躍點限制

使用 C Edit-EC2InstanceMetadataOptionmdlet 並將`HttpPutResponseHopLimit`參數設定為所需的躍點數目。在下列範例中，跳轉限制設定為 3。請注意，為 `HttpPutResponseHopLimit` 指定數值時，您還必須將 `HttpEndpoint` 設定為 `enabled`。

```
(Edit-EC2InstanceMetadataOption `
```

```
    -InstanceId i-1234567898abcdef0 `
    -HttpPutResponseHopLimit 3 `
    -HttpEndpoint enabled).InstanceMetadataOptions
```

啟用IMDSIPv4和IPv6端點

執行個體上IMDS有兩個端點：IPv4(`169.254.169.254`) 和 IPv6 (`[fd00:ec2::254]`)。當您啟用時IMDS，IPv4端點會自動啟用。即使您將執行個體啟動至IPv6唯一子網路，IPv6端點仍會保持停用狀態。要啟用IPv6端點，您需要明確執行此操作。當您啟用端點時，IPv6端點會IPv4保持啟用狀態。

您可以在執行個體啟動時或之後啟用IPv6端點。

啟用IPv6端點的需求

- 選取的執行個體類型建立在 AWS Nitro 系統上。
- 選取的子網路支援IPv6，其中子網路可能是雙堆疊或IPv6僅。

目前只有 AWS CLI 和 AWS SDKs支援在執行個體啟動後啟用IMDSIPv6端點。

AWS CLI

啟用執行個體的IMDSIPv6端點

使用指modify-instance-metadata-optionsCLI令並將`http-protocol-ipv6`參數設定為`enabled`。請注意，為 `http-protocol-ipv6` 指定數值時，您還必須將 `http-endpoint` 設定為 `enabled`。

```
aws ec2 modify-instance-metadata-options \
 --instance-id i-1234567898abcdef0 \
 --http-protocol-ipv6 enabled \
 --http-endpoint enabled
```

PowerShell

啟用執行個體的IMDSIPv6端點

使用指Edit-EC2InstanceMetadataOption令程式並將`HttpProtocolIpv6`參數設定為`enabled`。請注意，為 `HttpProtocolIpv6` 指定數值時，您還必須將 `HttpEndpoint` 設定為 `enabled`。

```
(Edit-EC2InstanceMetadataOption `
```

```
    -InstanceId i-1234567898abcdef0 `
    -HttpProtocolIpv6 enabled `
    -HttpEndpoint enabled).InstanceMetadataOptions
```

開啟對執行個體中繼資料的存取

無論您使用的是哪個版本，都可以啟用執行個體IMDS上的HTTP端點來開啟執行個體中繼資料的存IMDS取權。您可以隨時停用HTTP端點來還原此變更。

使用下列其中一種方法來開啟對執行個體上執行個體中繼資料的存取。

Console

開啟對執行個體中繼資料的存取

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體。
4. 選擇動作、執行個體設定、修改執行個體中繼資料選項。
5. 在修改執行個體中繼資料選項對話方塊中，執行下列動作：
   a. 針對執行個體中繼資料服務，請選取啟用。
   b. 選擇 Save (儲存)。

AWS CLI

開啟對執行個體中繼資料的存取

使用指modify-instance-metadata-optionsCLI令並將`http-endpoint`參數設定為`enabled`。

```
aws ec2 modify-instance-metadata-options \
    --instance-id i-1234567898abcdef0 \
    --http-endpoint enabled
```

PowerShell

開啟對執行個體中繼資料的存取

使用指Edit-EC2InstanceMetadataOption令程式並將`HttpEndpoint`參數設定為`enabled`。

```
(Edit-EC2InstanceMetadataOption `
    -InstanceId i-1234567898abcdef0 `
    -HttpEndpoint enabled).InstanceMetadataOptions
```

關閉對執行個體中繼資料的存取

無論您使用的是哪個版本，都可以停用執行個體IMDS上的HTTP端點來關閉執行個體中繼資料的存IMDS取權。您可以透過啟用HTTP端點隨時還原此變更。

使用下列其中一種方法來關閉對執行個體上執行個體中繼資料的存取。

Console

關閉對執行個體中繼資料的存取

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體。
4. 選擇動作、執行個體設定、修改執行個體中繼資料選項。
5. 在修改執行個體中繼資料選項對話方塊中，執行下列動作：
   a. 針對執行個體中繼資料服務，清除啟用。
   b. 選擇 Save (儲存)。

AWS CLI

關閉對執行個體中繼資料的存取

使用指modify-instance-metadata-optionsCLI令並將`http-endpoint`參數設定為`disabled`。

```
aws ec2 modify-instance-metadata-options \
    --instance-id i-1234567898abcdef0 \
    --http-endpoint disabled
```

PowerShell

關閉對執行個體中繼資料的存取

使用指Edit-EC2InstanceMetadataOption令程式並將`HttpEndpoint`參數設定為`disabled`。

```
(Edit-EC2InstanceMetadataOption `
    -InstanceId i-1234567898abcdef0 `
    -HttpEndpoint disabled).InstanceMetadataOptions
```

使用使用者資料輸入啟動執行個EC2體時執行指令

啟動 Amazon EC2 執行個體時，您可以將使用者資料傳遞到用於執行自動化組態任務的執行個體，或在執行個體啟動後執行指令碼。

如果您對更複雜的自動化案例感興趣，可以考慮 AWS CloudFormation 或 AWS OpsWorks。如需詳細資訊，請參閱下列內容：

- 使用使用者指南中的EC2功能 AWS CloudFormation在 Amazon 上部署應AWS CloudFormation用程式。
- AWS OpsWorks 使用者指南。

在 Linux 執行個體上，您可以將兩種類型的使用者資料傳遞給 AmazonEC2：殼層指令碼和雲端初始化指令。您也可以將這些資料以純文字、檔案形式傳遞至啟動執行個體精靈 (這對於使用命令列工具啟動執行個體非常有用)，或以 base64 編碼的文字 (API呼叫) 傳遞至啟動執行個體精靈。

在 Windows 執行個體上，啟動代理程式會處理使用者資料指令碼。以下各節說明在每個作業系統上處理使用者資料的方式之間的差異。

中的使用者資料 AWS Management Console

您可以在啟動執行個體時指定執行個體使用者資料。如果執行個體的根磁碟區是EBS磁碟區，您也可以停止執行個體並更新其使用者資料。

使用 Launch Wizard 指定啟動時的執行處理使用者資料

您可以在EC2主控台中使用 Launch Wizard 啟動執行個體時指定使用者資料。若要在啟動時指定使用者資料，請遵循啟動執行個體的程序。此 User data (使用者資料) 欄位位於啟動執行個體精靈的 進階詳細資訊 區段中。在 [使用者資料] 欄位中輸入 PowerShell 指令碼，然後完成執行個體啟動程序。

在以下使用者資料欄位的螢幕擷取畫面中，範例指令碼會在 Windows 暫時資料夾中建立一個檔案，並在檔名中使用目前日期和時間。當您包括 `<persist>true</persist>` 時，指令碼會在您每次重新開機或啟動執行個體時執行。如果您將 [使用者資料已經被 base64 編碼] 核取方塊保留空白，Amazon EC2 主控台會為您執行 base64 編碼。

User data - *optional* Info

Enter user data in the field.

```
<powershell>
$file = $env:SystemRoot+"\Temp\"+(Get-Date).ToString("MM-dd-yy-hh-mm")
New-Item $file -ItemType file
</powershell>
<persist>true</persist>
```

☐ User data has already been base64 encoded

如需詳細資訊，請參閱使用 Launch Wizard 指定啟動時的執行處理使用者資料。如需使用的 Linux 範例 AWS CLI，請參閱the section called “使用者資料和 AWS CLI”。如需使用視窗工具的視窗範例 PowerShell，請參閱the section called “使用者資料和視窗工具 PowerShell”。

檢視及更新執行個體使用者資料

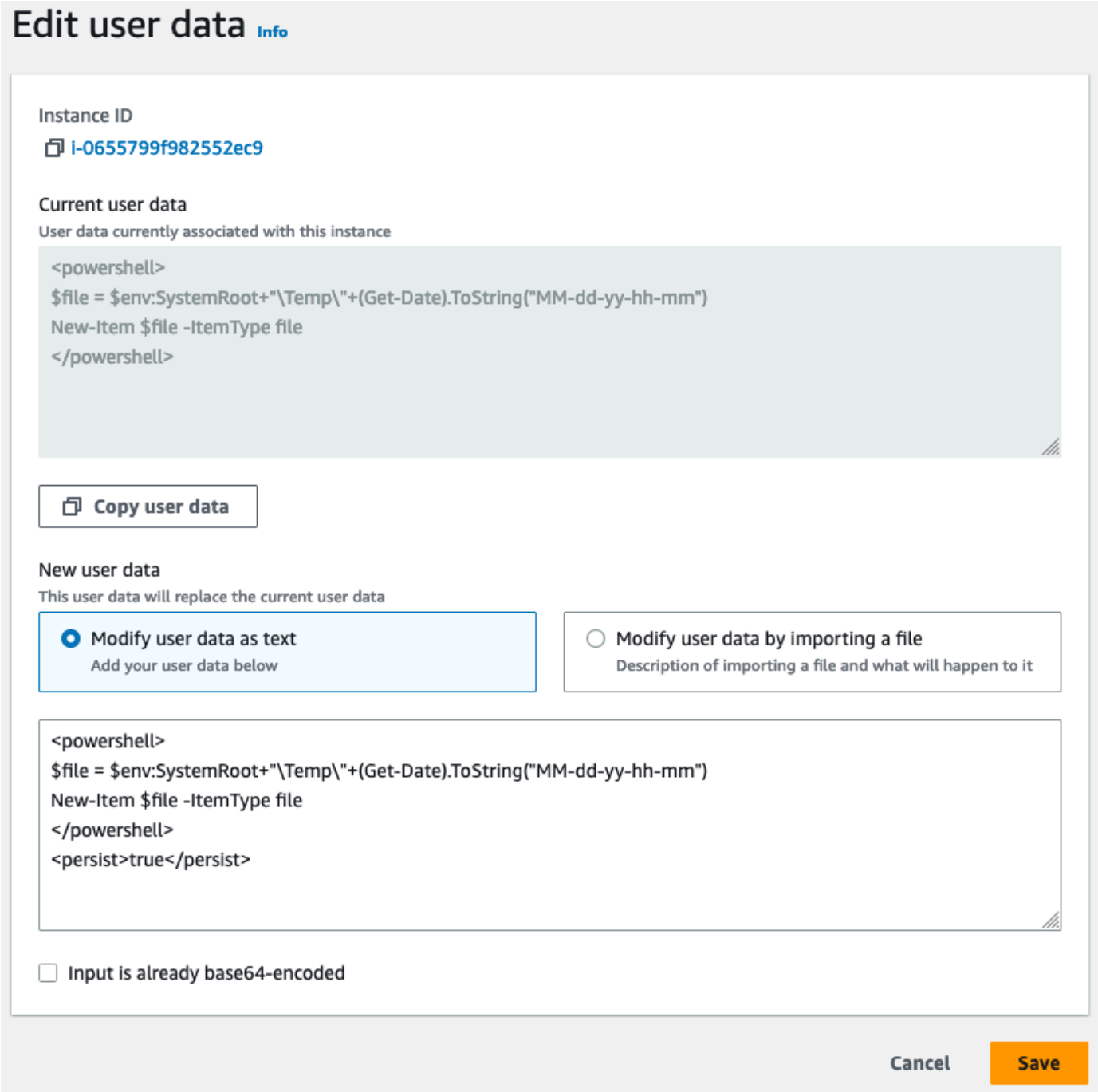
您可以檢視任何執行個體的執行個體使用者資料，並更新已終止之執行個體的執行個體使用者資料。

使用主控台更新執行個體的使用者資料

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，並選取 Actions (動作)、Instance state (執行個體狀態)、Stop instance (停止執行個體)。

> **Warning**
>
> 停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。若要保留執行個體存放區磁碟區的資料，請將資料備份至持久性儲存。

4. 出現確認提示時，請選擇 Stop (停止)。停止執行個體可能需要幾分鐘。
5. 在仍然選取執行個體的情況下，選取動作 > 執行個體設定 > 編輯使用者資料。您無法在執行個體仍在執行中時變更使用者資料，但您可以檢視它。
6. 在 Edit user data (編輯使用者資料) 對話方塊中，更新使用者資料，然後選擇 Save (儲存)。若要在每次重新開機或啟動執行個體時執行使用者資料指令碼，請新增 `<persist>true</persist>`，如下範例所示：



7. 啟動實例。若您已為後續的重新開機或啟動啟用使用者資料執行，即會在執行個體啟動程序期間執行更新的使用者資料指令碼。

Amazon 如何EC2處理 Linux 實例的用戶數據

在下列範例中，在 Amazon Linux 2 上安裝LAMP伺服器的命令會轉換為殼層指令碼和一組在執行個體啟動時執行的 Cloud init 指令。在每個範例中，以下任務都會由使用者資料執行：

- 更新分佈軟體套件。
- 安裝必要 Web 伺服器、`php` 和 `mariadb` 套件。
- 透過 systemctl 啟動並開啟 `httpd` 服務。
- `ec2-user` 即會新增至 apache 群組。
- 接著便會為 Web 目錄和其中包含的檔案設定適當的所有權和檔案許可。
- 創建一個簡單的網頁來測試 Web 服務器和PHP引擎。

目錄

- 必要條件
- 使用者資料與 Shell 指令碼
- 更新執行個體使用者資料
- 使用者資料與 cloud-init 指示詞
- 使用者資料和 AWS CLI
- 結合 shell 指令碼與 cloud-init 指令

必要條件

此主題中的範例假設如下內容：

- 您的執行個體具有可從網際網路存取的公開DNS名稱。
- 與執行個體相關聯的安全群組已設定為允許 SSH (連接埠 22) 流量，讓您可以連線至執行個體以檢視輸出記錄檔。
- 您的執行個體是透過 Amazon Linux 2 啟動的AMI。這些說明適用於與 Amazon Linux 2 搭配使用的情況，因此命令和指示詞可能無法在其他 Linux 發行版本上運作。如需其他分佈的詳細資訊，例如支援 cloud-init 的情形，請參閱其特定文件。

使用者資料與 Shell 指令碼

如果您熟悉 Shell 指令碼，這是在啟動時將指令傳送至執行個體的最簡單且最完整的方式。在開機階段新增這些任務也會增加開機執行個體所需要的時間長度。任務完成需要多花幾分鐘的時間，接著您便可以測試使用者指令碼是否已成功完成。

> **Important**
>
> 依預設，使用者資料指令碼和 cloud-init 指令只會在您第一次啟動執行個體時的開機週期過程中執行。您可以更新設定，以確保每次重新啟動執行個體時，您的使用者資料指令碼和 cloud-init 指令都會執行。如需詳細資訊，請參閱如何利用使用者資料在每次重新啟動 Amazon EC2 Linux 執行個體時自動執行指令碼？ 在 AWS 知識中心。

使用者資料 shell 指令碼必須以 #! 字元，以及您希望讀取指令碼 (通常是 /bin/bash)) 的解譯器路徑做為開頭。有關 shell 腳本的介紹，請參閱GNU操作系統網站上的 Bash 參考手冊。

做為使用者資料輸入的指令碼會以根使用者的身分執行，因此請不要在指令碼中使用 sudo 命令。請記得，您建立的任何檔案都會由根使用者擁有；若您需要讓非超級使用者擁有檔案存取權，建議您根據需求在指令碼中修改許可。此外，因為指令碼不會以互動方式執行，您無法包含需要使用者意見回饋的命令 (例如不帶有 `-y` 標記的 yum update)。

如果您在使用 AWS API者資料指令碼中使用 (包括)，則在啟動執行個體時必須使用執行個體設定檔。AWS CLI執行個體設定檔會提供使用者資料指令碼發出API呼叫所需的適當 AWS 憑證。如需詳細資訊，請參閱使用指南中的IAM使用執行個體設定檔。您指派給IAM角色的權限取決於您使用呼叫的服務API。如需詳細資訊，請參閱IAMAmazon 的角色 EC2。

cloud-init 輸出日誌檔案會擷取主控台輸出，因此，若執行個體在啟動後未以您想要的方式運作，則可輕鬆地對您的指令碼進行除錯。若要檢視日誌檔案，連線到執行個體並開啟 `/var/log/cloud-init-output.log`。

處理使用者資料指令碼時，會將其複製到 `/var/lib/cloud/instances/`*`instance-id`*`/` 並從中執行。在執行指令碼之後，不會將其刪除。在從執行個AMI體建立`/var/lib/cloud/instances/`*`instance-id`*`/`之前，請務必先刪除使用者資料指令碼。否則，指令碼將會存在於從啟動的任何執行個體上的此目錄中AMI。

更新執行個體使用者資料

若要更新執行個體使用者資料，您必須先停止執行個體。如果執行個體正在執行，您可以檢視使用者資料，但無法進行修改。

> **Warning**
>
> 停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。若要保留執行個體存放區磁碟區的資料，請將資料備份至持久性儲存。

修改執行個體使用者資料

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，並選取 Instance state (執行個體狀態)、Stop instance (停止執行個體)。如果此選項已停用，則執行個體已停止或其根設備是執行個體存放磁碟區。
4. 出現確認提示時，請選擇 Stop (停止)。停止執行個體可能需要幾分鐘。
5. 在仍然選取執行個體的情況下，選取動作 > 執行個體設定 > 編輯使用者資料。
6. 視需要修改使用者資料，然後選擇儲存。
7. 啟動實例。新的使用者資料便會在執行個體啟動之後出現在執行個體上；但不會執行使用者資料指令碼。

使用者資料與 cloud-init 指示詞

cloud-init 套件會在啟動時設定新 Amazon Linux 執行個體的特定部分；最明顯的是，它會設定 ec2-user 的 `.ssh/authorized_keys` 檔案，讓您可以使用自己的私有金鑰登入。如需雲端初始化套件針對 Amazon Linux 執行個體執行之組態任務的詳細資訊，請參閱 Amazon Linux 2 使用者指南中的在 Amazon Linux 2 上使用雲端初始化。

cloud-init 使用者指示詞可在啟動時，以傳遞指令碼相同的方式傳遞到執行個體，雖然語法不同。如需有關 cloud-init 的詳細資訊，請參閱 http://cloudinit.readthedocs.org/en/latest/index.html。

> **Important**
>
> 依預設，使用者資料指令碼和 cloud-init 指令只會在您第一次啟動執行個體時的開機週期過程中執行。您可以更新設定，以確保每次重新啟動執行個體時，您的使用者資料指令碼和 cloud-init 指令都會執行。如需詳細資訊，請參閱如何利用使用者資料在每次重新啟動 Amazon EC2 Linux 執行個體時自動執行指令碼？ 在 AWS 知識中心。

在開機階段新增這些任務也會增加開機執行個體所需要的時間長度。任務完成需要多花幾分鐘的時間，接著您便可以測試您的使用者資料指示詞是否已完成。

使用使用者資料將 cloud-init 指示詞傳遞給執行個體

1. 請遵循啟動執行個體的程序。此 User data (使用者資料) 欄位位於啟動執行個體精靈的 進階詳細資訊 區段中。在 User data (使用者資料) 欄位中輸入您的 cloud-init 指示詞文字，然後完成執行個體啟動程序。

   在以下範例中，指示詞會在 Amazon Linux 2 上建立及設定 Web 伺服器。頂端的 `#cloud-config` 為將命令識別為 cloud-init 指示詞的必要項目。

```
#cloud-config
repo_update: true
repo_upgrade: all

packages:
- httpd
- mariadb-server

runcmd:
- [ sh, -c, "amazon-linux-extras install -y lamp-mariadb10.2-php7.2 php7.2" ]
- systemctl start httpd
- sudo systemctl enable httpd
- [ sh, -c, "usermod -a -G apache ec2-user" ]
- [ sh, -c, "chown -R ec2-user:apache /var/www" ]
- chmod 2775 /var/www
- [ find, /var/www, -type, d, -exec, chmod, 2775, {}, \; ]
- [ find, /var/www, -type, f, -exec, chmod, 0664, {}, \; ]
- [ sh, -c, 'echo "<?php phpinfo(); ?>" > /var/www/html/phpinfo.php' ]
```

2. 給予執行個體充分的時間啟動及執行您使用者資料中的指示詞，然後檢查您的指示詞是否已完成您預計執行的任務。

   在此範例中，在網頁瀏覽器中，輸入URL建立指令的PHP測試檔案。這URL是執行個體的公用DNS位址，後面接著正斜線和檔案名稱。

```
http://my.public.dns.amazonaws.com/phpinfo.php
```

   您應該會看到PHP資訊頁面。如果您無法看到PHP資訊頁，請檢查您使用的安全性群組是否包含允許 HTTP (通訊埠 80) 流量的規則。如需詳細資訊，請參閱設定安全群組規則。

3. (選用) 若您的指令並未完成您預期執行的任務，或您只希望確認您的指令已順利完成，而沒有發生任何錯誤，請連線到執行個體，檢查 cloud-init 輸出日誌檔案 (`/var/log/cloud-init-`

output.log)，並在輸出中查看是否有錯誤訊息。如需其他偵錯資訊，您可以為您的指示詞新增下行：

```
output : { all : '| tee -a /var/log/cloud-init-output.log' }
```

這個指示詞會將 runcmd 的輸出傳送到 /var/log/cloud-init-output.log。

使用者資料和 AWS CLI

您可以使用指 AWS CLI 定、修改和檢視執行個體的使用者資料。如需使用執行個體中繼資料檢視您執行個體中使用者資料的資訊，請參閱訪問實例的實EC2例元數據。

在視窗上，您可以使用， AWS Tools for Windows PowerShell 而不是使用 AWS CLI. 如需詳細資訊，請參閱 使用者資料和視窗工具 PowerShell。

範例：在啟動時指定使用者資料

若要在啟動您的執行個體時指定使用者資料，請搭配 --user-data 參數使用 run-instances 命令。使用時run-instances，會為您 AWS CLI 執行使用者資料的 base64 編碼。

以下範例顯示如何在命令列上將指令碼指定為字串：

```
aws ec2 run-instances --image-id ami-abcd1234 --count 1 --instance-type m3.medium \
    --key-name my-key-pair --subnet-id subnet-abcd1234 --security-group-ids sg-abcd1234
 \
    --user-data echo user data
```

以下範例顯示如何使用文字檔案指定指令碼。請確認使用 file:// 前綴來指定檔案。

```
aws ec2 run-instances --image-id ami-abcd1234 --count 1 --instance-type m3.medium \
    --key-name my-key-pair --subnet-id subnet-abcd1234 --security-group-ids sg-abcd1234
 \
    --user-data file://my_script.txt
```

以下是具有 shell 指令碼的範例文字檔案。

```
#!/bin/bash
yum update -y
service httpd start
chkconfig httpd on
```

範例：修改已停止執行個體的使用者資料

您可以使用modify-instance-attribute指令修改已停止執行個體的使用者資料。使用時modify-instance-attribute， AWS CLI 不會為您執行使用者資料的 base64 編碼。

- 在 Linux 電腦上，請使用 base64 命令來編碼使用者資料。

```
base64 my_script.txt >my_script_base64.txt
```

- 在 Windows 電腦上，請使用 certutil 命令編碼使用者資料。在您可以搭配使用此檔案之前 AWS CLI，您必須移除第一行 (BEGINCERTIFICATE) 和 last (ENDCERTIFICATE) 行。

```
certutil -encode my_script.txt my_script_base64.txt
notepad my_script_base64.txt
```

請使用 `--attribute` 和 `--value` 參數，使用已編碼的文字檔案來指定使用者資料。請確認使用 `file://` 前綴來指定檔案。

```
aws ec2 modify-instance-attribute --instance-id i-1234567890abcdef0 --attribute
 userData --value file://my_script_base64.txt
```

範例：清除已停止執行個體的使用者資料

若要刪除現有的使用者資料，請依照下列方式使用modify-instance-attribute指令：

```
aws ec2 modify-instance-attribute --instance-id i-1234567890abcdef0 --user-data Value=
```

範例：檢視使用者資料

若要擷取執行個體的使用者資料，請使用describe-instance-attribute指令。使用時describe-instance-attribute， AWS CLI 不會為您執行 base64 解碼使用者資料。

```
aws ec2 describe-instance-attribute --instance-id i-1234567890abcdef0 --attribute
 userData
```

下列為已使用 base64 編碼之使用者資料的範例輸出。

```
{
    "UserData": {
```

```
        "Value":
 "IyEvYmluL2Jhc2gKeXVtIHVwZGF0ZSAteQpzZXJ2aWNlIGh0dHBkIHN0YXJ0CmNoa2NvbmZpZyBodHRwZCBvbg=="
    },
    "InstanceId": "i-1234567890abcdef0"
}
```

- 在 Linux 上，請使用 `--query` 選項取得已編碼的使用者資料，並使用 base64 命令將其解碼。

```
aws ec2 describe-instance-attribute --instance-id i-1234567890abcdef0 --attribute
 userData --output text --query "UserData.Value" | base64 --decode
```

- 在 Windows 電腦上，使用 `--query` 選項以取得已編碼的使用者資料，並使用 certutil 命令將其解碼。請注意，編碼輸出會存放在檔案中，解碼輸出則會存放在另一個檔案中。

```
aws ec2 describe-instance-attribute --instance-id i-1234567890abcdef0 --attribute
 userData --output text --query "UserData.Value" >my_output.txt
certutil -decode my_output.txt my_output_decoded.txt
type my_output_decoded.txt
```

下列為範例輸出。

```
#!/bin/bash
yum update -y
service httpd start
chkconfig httpd on
```

結合 shell 指令碼與 cloud-init 指令

根據預設，您一次只能在使用者資料中包含一種內容類型。不過，您可以使用 MIME 多部分檔案中的 `text/cloud-config` 和 `text/x-shellscript` 內容類型，以在使用者資料中包含 shell 指令碼和 cloud-init 指令。

以下展示 MIME 多部分格式。

```
Content-Type: multipart/mixed; boundary="//"
MIME-Version: 1.0

--//
Content-Type: text/cloud-config; charset="us-ascii"
MIME-Version: 1.0
```

```
Content-Transfer-Encoding: 7bit
Content-Disposition: attachment; filename="cloud-config.txt"

#cloud-config
cloud-init directives

--//
Content-Type: text/x-shellscript; charset="us-ascii"
MIME-Version: 1.0
Content-Transfer-Encoding: 7bit
Content-Disposition: attachment; filename="userdata.txt"

#!/bin/bash
shell script commands
--//--
```

例如，下列使用者資料包含 cloud-init 指令和 bash shell 指令碼。cloud-init 指令會建立檔案 (`/test-cloudinit/cloud-init.txt`)，並將 `Created by cloud-init` 寫入到該檔案。bash shell 指令碼會建立檔案 (`/test-userscript/userscript.txt`) 並將 `Created by bash shell script` 寫入到該檔案。

```
Content-Type: multipart/mixed; boundary="//"
MIME-Version: 1.0

--//
Content-Type: text/cloud-config; charset="us-ascii"
MIME-Version: 1.0
Content-Transfer-Encoding: 7bit
Content-Disposition: attachment; filename="cloud-config.txt"

#cloud-config
runcmd:
- [ mkdir, /test-cloudinit ]
write_files:
- path: /test-cloudinit/cloud-init.txt
content: Created by cloud-init

--//
Content-Type: text/x-shellscript; charset="us-ascii"
MIME-Version: 1.0
Content-Transfer-Encoding: 7bit
Content-Disposition: attachment; filename="userdata.txt"
```

```
#!/bin/bash
mkdir test-userscript
touch /test-userscript/userscript.txt
echo "Created by bash shell script" >> /test-userscript/userscript.txt
--//--
```

Amazon 如何EC2處理 Windows 實例的用戶數據

在 Windows 執行個體上，啟動代理程式會執行與使用者資料相關的工作。如需詳細資訊，請參閱下列內容：

- EC2Launch v2
- EC2Launch
- EC2Config服務

如需範 AWS CloudFormation 本中`UserData`屬性組件的範例，請參閱 Base64 編碼內 UserData 容和 Base64 編碼 UserData 屬性與與 AccessKey 。 SecretKey

如需在可搭配生命週期勾點使用之 Auto Scaling 群組內的執行個體上執行命令的範例，請參閱 Amazon EC2 Auto Scaling 使用者指南中的教學：設定使用者資料以透過執行個體中繼資料擷取目標生命週期狀態。

目錄

- 使用者資料指令碼
- 使用者資料執行
- 使用者資料和視窗工具 PowerShell

使用者資料指令碼

`EC2Launch`若要執行`EC2Config`或執行指令碼，當您將指令碼新增至使用者資料時，您必須將指令碼包含在特殊標記內。您使用的標籤取決於命令是在命令提示字元視窗 (批次命令) 中執行，還是使用 Windows PowerShell。

如果您同時指定批次指令碼和 Windows 指 PowerShell 令碼，則會先執行批次指令碼，然後再執行 Windows 指 PowerShell 令碼，而不論它們在執行個體使用者資料中出現的順序為何。

如果您在使用 AWS API者資料指令碼中使用 (包括)，則在啟動執行個體時必須使用執行個體設定檔。AWS CLI執行個體設定檔會提供使用者資料指令碼進行API呼叫所需的適當 AWS 認證。如需詳細資

訊，請參閱執行個體描述檔。您指派給IAM角色的權限取決於您使用呼叫的服務API。如需詳細資訊，請參閱IAMAmazon 的角色 EC2。

指令碼類型

- 批次指令碼的語法
- 語法視窗 PowerShell指令碼
- YAML組態指令碼的語法
- Base64 編碼

批次指令碼的語法

使用 `script` 標籤指定批次指令碼。使用換行符號分隔命令，如下列範例所示。

```
<script>
    echo Current date and time >> %SystemRoot%\Temp\test.log
    echo %DATE% %TIME% >> %SystemRoot%\Temp\test.log
</script>
```

在預設情況下，使用者資料指令碼僅會在您啟動執行個體時執行一次。若要在每次重新開機或啟動執行個體時執行使用者資料指令碼，請將 `<persist>true</persist>` 新增至使用者資料。

```
<script>
    echo Current date and time >> %SystemRoot%\Temp\test.log
    echo %DATE% %TIME% >> %SystemRoot%\Temp\test.log
</script>
<persist>true</persist>
```

EC2Launchv2 代理程式

若要將XML使用者資料指令碼作為`UserData`階段中 EC2Launch v2 executeScript 工作的分離處理程序執行，`<detach>true</detach>`請新增至使用者資料。

> **Note**
>
> 先前的啟動代理程式不支援此detach標籤。

```
<script>
```

```
    echo Current date and time >> %SystemRoot%\Temp\test.log
    echo %DATE% %TIME% >> %SystemRoot%\Temp\test.log
</script>
<detach>true</detach>
```

語法視窗 PowerShell指令碼

AWS Windows AMIs 包含 AWS Tools for Windows PowerShell，因此您可以在使用者資料中指定這些指令程式。如果將IAM角色與執行個體建立關聯，則不需要指定 Cmdlet 的登入資料，因為在執行個體上執行的應用程式會使用角色的登入資料來存取 AWS 資源 (例如，Amazon S3 儲存貯體)。

使用此標記指定 Windows 指 PowerShell 令`<powershell>`碼。使用分行符號來區隔命令。`<powershell>` 標籤區分大小寫。

例如：

```
<powershell>
    $file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
    New-Item $file -ItemType file
</powershell>
```

根據預設，使用者資料指令碼會在您啟動執行個體時執行一次。若要在每次重新開機或啟動執行個體時執行使用者資料指令碼，請將 `<persist>true</persist>` 新增至使用者資料。

```
<powershell>
    $file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
    New-Item $file -ItemType file
</powershell>
<persist>true</persist>
```

您可以使用`<powershellArguments>`標籤指定一個或多個 PowerShell 引數。如果沒有傳遞參數，EC2Launch並且 EC2Launch v2 默認添加以下參數：`-ExecutionPolicy Unrestricted`。

範例：

```
<powershell>
    $file = $env:SystemRoot + "\Temp" + (Get-Date).ToString("MM-dd-yy-hh-mm")
    New-Item $file -ItemType file
</powershell>
<powershellArguments>-ExecutionPolicy Unrestricted -NoProfile -NonInteractive</
powershellArguments>
```

EC2Launchv2 代理程式

若要將XML使用者資料指令碼作為`UserData`階段中 EC2Launch v2 executeScript 工作的分離處理程序執行，`<detach>true</detach>`請新增至使用者資料。

> **Note**
>
> 先前的啟動代理程式不支援此detach標籤。

```
<powershell>
    $file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
    New-Item $file -ItemType file
</powershell>
<detach>true</detach>
```

YAML組態指令碼的語法

如果您使用 EC2Launch v2 執行指令碼，您可以使用YAML格式。若要檢視 EC2Launch v2 的組態工作、詳細資訊和範例，請參閱EC2Launchv2 任務配置。

指定含有`executeScript`工作的指YAML令碼。

執行 PowerShell 指令碼的YAML語法範例

```
version: 1.0
    tasks:
    - task: executeScript
      inputs:
      - frequency: always
        type: powershell
        runAs: localSystem
        content: |-
          $file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
          New-Item $file -ItemType file
```

執行批次指令碼的範例YAML語法

```
version: 1.1
    tasks:
```

```
    - task: executeScript
      inputs:
      - frequency: always
        type: batch
        runAs: localSystem
        content: |-
          echo Current date and time >> %SystemRoot%\Temp\test.log
          echo %DATE% %TIME% >> %SystemRoot%\Temp\test.log
```

Base64 編碼

如果您使用的是 Amazon EC2 API 或不執行使用者資料 base64 編碼的工具，則必須自行編碼使用者資料。若否，系統就會記錄一則錯誤，指出找不到要執行的 `script` 或 `powershell` 標籤。以下是使用 Windows PowerShell 進行編碼的範例。

```
$UserData =
 [System.Convert]::ToBase64String([System.Text.Encoding]::ASCII.GetBytes($Script))
```

以下是使 PowerShell用解碼的範例。

```
$Script =
 [System.Text.Encoding]::UTF8.GetString([System.Convert]::FromBase64String($UserData))
```

如需有關 Base64 編碼的詳細資訊，請參閱 https://www.ietf.org/rfc/rfc4648.txt。

使用者資料執行

根據預設，所AMIs有 AWS Windows 都會啟用初始啟動的使用者資料執行。您可以指定在下一次重新開機或重新啟動執行個體時執行使用者資料指令碼。或者，您可以指定每次重新開機或重新啟動執行個體時都執行使用者資料指令碼。

> **Note**
>
> 在預設情況下，初始啟動後不會執行使用者資料。若要在重新開機或啟動執行個體後執行使用者資料，請參閱 在後續重新開機或啟動期間執行指令碼。

產生隨機密碼時，會透過本機管理員帳戶來執行使用者資料指令碼。否則，會透過系統帳戶執行使用者資料指令碼。

執行個體啟動指

執行個體使用者資料中的指令碼僅會在執行個體初次啟動時執行。如果找到 `persist` 標籤，即會為後續的重新開機或啟動啟用使用者資料執行。EC2Launchv2、EC2Launch和的記錄檔EC2Config包含來自標準輸出和標準錯誤串流的輸出。

EC2Launch v2

EC2Launchv2 的日誌文件是`C:\ProgramData\Amazon\EC2Launch\log\agent.log`。

> **Note**
>
> `C:\ProgramData` 資料夾可能隱藏不見。若要檢視此資料夾，您必須顯示隱藏的檔案和資料夾。

以下資訊會在執行使用者資料時記錄：

- `Info: Converting user-data to yaml format`— 如果使用者資料是以XML格式提供
- `Info: Initialize user-data state` - 開始執行使用者資料時
- `Info: Frequency is: always` - 如果使用者資料工作在每次開機時執行
- `Info: Frequency is: once` - 如果使用者資料任務只執行一次
- `Stage: postReadyUserData execution completed` - 使用者資料執行結束時

EC2Launch

EC2Launch 的記錄檔是 `C:\ProgramData\Amazon\EC2-Windows\Launch\Log\UserdataExecution.log`。

`C:\ProgramData` 資料夾可能隱藏不見。若要檢視此資料夾，您必須顯示隱藏的檔案和資料夾。

以下資訊會在執行使用者資料時記錄：

- `Userdata execution begins` - 開始執行使用者資料時
- `<persist> tag was provided: true` - 如果找到 persist 標籤
- `Running userdata on every boot` - 如果找到 persist 標籤
- `<powershell> tag was provided.. running powershell content` - 如果找到 powershell 標籤

- `<script> tag was provided.. running script content` - 如果找到指令碼標籤
- `Message: The output from user scripts` - 如果執行使用者資料指令碼，則會記錄其輸出

EC2Config

EC2Config 的記錄檔是 `C:\Program Files\Amazon\Ec2ConfigService\Logs\Ec2Config.log`。以下資訊會在執行使用者資料時記錄：

- `Ec2HandleUserData: Message: Start running user scripts` - 開始執行使用者資料時
- `Ec2HandleUserData: Message: Re-enabled userdata execution` - 如果找到 persist 標籤
- `Ec2HandleUserData: Message: Could not find <persist> and </persist>` - 如果沒有找到 persist 標籤
- `Ec2HandleUserData: Message: The output from user scripts` - 如果執行使用者資料指令碼，則會記錄其輸出

在後續重新開機或啟動期間執行指令碼

當您更新執行個體使用者資料時，使用者資料指令碼並未在您重新開機或啟動執行個體時自動執行。不過，您可以啟用使用者資料執行，以讓使用者資料指令碼在您重新開機或啟動執行個體時執行一次，或每次重新開機或啟動執行個體時都執行。

如果您選擇 Shutdown with Sysprep (使用 Sysprep 關機) 選項，使用者資料指令碼會在下次執行個體啟動或重新開機時執行，即使您並沒有為後續的重新開機或啟動啟用使用者資料執行亦同。使用者資料指令碼將不會在後續的重新開機或啟動時執行。

若要使用 EC2Launch v2 啟用使用者資料執行

- 若要在第一次開機時在使用者資料中執行工作，請將 `frequency` 設定為 `once`。
- 若要在每次開機時在使用者資料中執行工作，請將 `frequency` 設定為 `always`。

若要啟用使用者資料執行 EC2Launch

1. 連接至 Windows 執行個體。
2. 開啟命 PowerShell 令視窗並執行下列命令：

```
C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 -Schedule
```

3. 中斷連線您的 Windows 執行個體。若要在下一次啟動執行個體時執行更新的指令碼，請停止執行個體並更新使用者資料。

若要啟用使用者資料執行 EC2Config

1. 連接至 Windows 執行個體。
2. Open `C:\Program Files\Amazon\Ec2ConfigService\Ec2ConfigServiceSetting.exe`.
3. 對於使用者資料，請選取啟用下次服務啟動時 UserData執行。
4. 中斷連線您的 Windows 執行個體。若要在下一次啟動執行個體時執行更新的指令碼，請停止執行個體並更新使用者資料。

使用者資料和視窗工具 PowerShell

您可以使用 Windows 專用的工具 PowerShell 來指定、修改和檢視執行個體的使用者資料。如需使用執行個體中繼資料檢視您執行個體中使用者資料的資訊，請參閱訪問實例的實EC2例元數據。若要取得有關使用者資料的資訊 AWS CLI，請參閱使用者資料和 AWS CLI。

範例：在啟動時指定執行個體使用者資料

使用執行個體使用者資料建立文字檔案。若要在每次重新開機或啟動執行個體時執行使用者資料指令碼，請新增 `<persist>true</persist>`，如下範例所示。

```
<powershell>
    $file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
    New-Item $file -ItemType file
</powershell>
<persist>true</persist>
```

若要在啟動執行個體時指定執行個體使用者資料，請使用指New-EC2Instance令。此命令不會為您執行使用者資料的 base64 編碼。使用以下命令，將使用者資料編碼至名為 `script.txt` 的文字檔案：

```
PS C:\> $Script = Get-Content -Raw script.txt
PS C:\> $UserData =
 [System.Convert]::ToBase64String([System.Text.Encoding]::ASCII.GetBytes($Script))
```

使用 `-UserData` 參數，將使用者資料傳遞到 New-EC2Instance 命令。

```
PS C:\> New-EC2Instance -ImageId ami-abcd1234 -MinCount 1 -MaxCount 1 -
InstanceType m3.medium \
    -KeyName my-key-pair -SubnetId subnet-12345678 -SecurityGroupIds sg-1a2b3c4d \
    -UserData $UserData
```

範例：更新已停止之執行個體的執行個體使用者資料

您可以使用Edit-EC2InstanceAttribute指令修改已停止執行個體的使用者資料。

使用新的指令碼建立文字檔案。使用以下命令，將使用者資料編碼至名為 `new-script.txt` 的文字檔案：

```
PS C:\> $NewScript = Get-Content -Raw new-script.txt
PS C:\> $NewUserData =
 [System.Convert]::ToBase64String([System.Text.Encoding]::ASCII.GetBytes($NewScript))
```

使用 `-UserData` 和 `-Value` 參數指定使用者資料。

```
PS C:\> Edit-EC2InstanceAttribute -InstanceId i-1234567890abcdef0 -Attribute userData -
Value $NewUserData
```

範例：檢視執行個體使用者資料

若要擷取執行個體的使用者資料，請使用Get-EC2InstanceAttribute指令。

```
PS C:\> (Get-EC2InstanceAttribute -InstanceId i-1234567890abcdef0 -Attribute
 userData).UserData
```

下列為範例輸出。請注意使用者資料已編碼。

```
PHBvd2Vyc2hlbGw
+DQpSZW5hbWUtQ29tcHV0ZXIgLU5ld05hbWUgdXNlci1kYXRhLXRlc3QNCjwvcG93ZXJzaGVsbD4=
```

使用以下命令，將已編碼的使用者資料存放至變數中，然後加以解碼。

```
PS C:\> $UserData_encoded = (Get-EC2InstanceAttribute -InstanceId i-1234567890abcdef0 -
Attribute userData).UserData
PS C:
\> [System.Text.Encoding]::UTF8.GetString([System.Convert]::FromBase64String($UserData_encoded)
```

下列為範例輸出。

```
<powershell>
    $file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
    New-Item $file -ItemType file
</powershell>
<persist>true</persist>
```

範例：重新命名執行個體以符合標籤值

您可以使用指Get-EC2Tag令讀取標籤值、在第一次開機時重新命名執行個體以符合標籤值，然後重新啟動。若要成功執行此命令，您必須擁有具有連接至執行個體的`ec2:DescribeTags`權限的角色，因為API呼叫會擷取標記資訊。如需使用IAM角色設定權限的詳細資訊，請參閱將IAM角色附加至執行個體。

IMDSv2

```
<powershell>
    [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-
seconds" = "21600"} -Method PUT -Uri 'http://169.254.169.254/latest/api/token' -
UseBasicParsing
    $instanceId = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" =
 $token} -Method GET -Uri 'http://169.254.169.254/latest/meta-data/instance-id' -
UseBasicParsing
 $nameValue = (Get-EC2Tag -Filter @{Name="resource-id";Value=
$instanceid},@{Name="key";Value="Name"}).Value
 $pattern = "^(?![0-9]{1,15}$)[a-zA-Z0-9-]{1,15}$"
 #Verify Name Value satisfies best practices for Windows hostnames
 If ($nameValue -match $pattern)
     {Try
         {Rename-Computer -NewName $nameValue -Restart -ErrorAction Stop}
     Catch
         {$ErrorMessage = $_.Exception.Message
         Write-Output "Rename failed: $ErrorMessage"}}
 Else
     {Throw "Provided name not a valid hostname. Please ensure Name value is between
 1 and 15 characters in length and contains only alphanumeric or hyphen characters"}
</powershell>
```

IMDSv1

```
<powershell>
```

```
 $instanceId = (Invoke-WebRequest http://169.254.169.254/latest/meta-data/instance-
id -UseBasicParsing).content
 $nameValue = (Get-EC2Tag -Filter @{Name="resource-id";Value=
$instanceid},@{Name="key";Value="Name"}).Value
 $pattern = "^(?![0-9]{1,15}$)[a-zA-Z0-9-]{1,15}$"
 #Verify Name Value satisfies best practices for Windows hostnames
 If ($nameValue -match $pattern)
     {Try
         {Rename-Computer -NewName $nameValue -Restart -ErrorAction Stop}
     Catch
         {$ErrorMessage = $_.Exception.Message
         Write-Output "Rename failed: $ErrorMessage"}}
 Else
     {Throw "Provided name not a valid hostname. Please ensure Name value is between
 1 and 15 characters in length and contains only alphanumeric or hyphen characters"}
</powershell>
```

如果您的執行個體設定為從執行個體中繼資料存取標籤，則還可以使用執行個體中繼資料中的標籤重新命名執行個體，。如需詳細資訊，請參閱使用執行個體中繼資料檢視EC2執行個體的。

IMDSv2

```
<powershell>
    [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-
seconds" = "21600"} -Method PUT -Uri 'http://169.254.169.254/latest/api/token' -
UseBasicParsing
 $nameValue = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token}
 -Method GET -Uri 'http://169.254.169.254/latest/meta-data/tags/instance/Name' -
UseBasicParsing
 $pattern = "^(?![0-9]{1,15}$)[a-zA-Z0-9-]{1,15}$"
 #Verify Name Value satisfies best practices for Windows hostnames
 If ($nameValue -match $pattern)
     {Try
         {Rename-Computer -NewName $nameValue -Restart -ErrorAction Stop}
     Catch
         {$ErrorMessage = $_.Exception.Message
         Write-Output "Rename failed: $ErrorMessage"}}
 Else
     {Throw "Provided name not a valid hostname. Please ensure Name value is between
 1 and 15 characters in length and contains only alphanumeric or hyphen characters"}
</powershell>
```

IMDSv1

```
<powershell>
 $nameValue = Get-EC2InstanceMetadata -Path /tags/instance/Name
 $pattern = "^(?![0-9]{1,15}$)[a-zA-Z0-9-]{1,15}$"
 #Verify Name Value satisfies best practices for Windows hostnames
 If ($nameValue -match $pattern)
     {Try
         {Rename-Computer -NewName $nameValue -Restart -ErrorAction Stop}
     Catch
         {$ErrorMessage = $_.Exception.Message
         Write-Output "Rename failed: $ErrorMessage"}}
 Else
     {Throw "Provided name not a valid hostname. Please ensure Name value is between
 1 and 15 characters in length and contains only alphanumeric or hyphen characters"}
</powershell>
```

識別在單一要求中啟動的每個執行個體

本範例示範如何使用使用者資料和執行個體中繼資料來設定 Amazon EC2 執行個體。

> **Note**
>
> 本節中的範例使用IMDS:的IPv4位址169.254.169.254。如果您要透過該IPv6位址擷取EC2執行個體的執行個體中繼資料，請務必啟用並使用該IPv6位址：[fd00:ec2::254]。的IPv6位址與IMDSv2指令相容。IMDS該IPv6地址只能在 AWS Nitro 系統上構建的實例和IPv6支持的子網（雙堆棧或IPv6僅）中訪問。

Alice 想要啟動她最愛的資料庫的四個執行個體AMI，第一個執行個體做為原始執行個體，剩下的三個執行個體做為複本。當她啟動這些執行個體時，她想要為每個複本新增關於複寫策略的使用者資料。她知道資料將會在四個執行個體上可用，因此她需要使用允許每個執行個體識別其適用部分的方式架構使用者資料。她可以使用 `ami-launch-index` 執行個體中繼資料值執行此作業，每一個執行個體的該值都是唯一的。如果您同時啟動一個以上的執行個體，`ami-launch-index` 表示執行個體的啟動順序。第一個啟動的執行個體的值為 `0`。

下列是 Alice 建構的使用者資料。

```
replicate-every=1min | replicate-every=5min | replicate-every=10min
```

`replicate-every=1min` 資料定義第一個複本的組態，`replicate-every=5min` 定義第二個複本的組態，以此類推。Alice 決定以ASCII字串形式提供此資料，其中包含管線符號 (|)，以分隔不同實體的資料。

Alice 使用 run-instances 命令啟動四個執行個體，指定使用者資料。

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --count 4 \
    --instance-type t2.micro \
    --user-data "replicate-every=1min | replicate-every=5min | replicate-every=10min"
```

在啟動它們之後，所有執行個體都會擁有使用者資料的複本，並且常見的中繼資料會顯示於此：

- AMI身份證明文件：阿密
- 保留 ID：r-1234567890abcabc0
- 公有金鑰：無
- 安全群組名稱：預設
- 執行個體類型：t2.micro

不過，每個執行個體都有唯一的中繼資料，如下表所示。

中繼資料	Value
instance-id	i-1234567890abcdef0
ami-launch-index	0
public-hostname	ec2-203-0-113-25.compute-1.amazonaws.com
public-ipv4	67.202.51.223
local-hostname	ip-10-251-50-12.ec2.internal
local-ipv4	10.251.50.35

中繼資料	Value
instance-id	i-0598c7d356eba48d7
ami-launch-index	1
public-hostname	ec2-67-202-51-224.compute-1.amazonaws.com
public-ipv4	67.202.51.224
local-hostname	ip-10-251-50-36.ec2.internal
local-ipv4	10.251.50.36

中繼資料	Value
instance-id	i-0ee992212549ce0e7
ami-launch-index	2
public-hostname	ec2-67-202-51-225.compute-1.amazonaws.com
public-ipv4	67.202.51.225
local-hostname	ip-10-251-50-37.ec2.internal
local-ipv4	10.251.50.37

中繼資料	Value
instance-id	i-1234567890abcdef0
ami-launch-index	3
public-hostname	ec2-67-202-51-226.compute-1.amazonaws.com
public-ipv4	67.202.51.226

中繼資料	Value
local-hostname	ip-10-251-50-38.ec2.internal
local-ipv4	10.251.50.38

Alice 可以使用 `ami-launch-index` 值判斷使用者資料中的哪一個部分適用於特定執行個體。

1. 她連線到其中一個執行個體，擷取該執行個體的 `ami-launch-index`，確保它是其中一個複本：

   IMDSv2

   ```
   [ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/meta-data/api/
   token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
   && curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-
   data/ami-launch-index
   2
   ```

   對於以下步驟，IMDSv2請求使用前面IMDSv2命令中的存儲令牌（假設令牌尚未過期）。

   IMDSv1

   ```
   [ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/ami-launch-index
   2
   ```

2. 她將 `ami-launch-index` 儲存為變數。

   IMDSv2

   ```
   [ec2-user ~]$ ami_launch_index=`curl -H "X-aws-ec2-metadata-token: $TOKEN"
    http://169.254.169.254/latest/meta-data/ami-launch-index`
   ```

   IMDSv1

   ```
   [ec2-user ~]$ ami_launch_index=`curl http://169.254.169.254/latest/meta-data/ami-
   launch-index`
   ```

3. 她將使用者資料儲存為變數。

IMDSv2

```
[ec2-user ~]$ user_data=`curl -H "X-aws-ec2-metadata-token: $TOKEN"
 http://169.254.169.254/latest/user-data`
```

IMDSv1

```
[ec2-user ~]$ user_data=`curl http://169.254.169.254/latest/user-data`
```

4. 最後，Alice 使用 cut 命令擷取適用於該執行個體的使用者資料部分。

IMDSv2

```
[ec2-user ~]$ echo $user_data | cut -d"|" -f"$ami_launch_index"
replicate-every=5min
```

IMDSv1

```
[ec2-user ~]$ echo $user_data | cut -d"|" -f"$ami_launch_index"
replicate-every=5min
```

偵測主機是否為EC2執行個體

您可能需要知道您的應用程式或網站是否在執行個EC2體上執行，尤其是當您有混合運算環境時。您可以使用下列其中一個選項來判斷應用程式或網站的主機是否為EC2執行個體。

選項

- 檢查 執行個體身分文件
- 檢查系統 UUID
- 檢查系統虛擬機器產生識別符

檢查 執行個體身分文件

每個執行個體都有簽署的執行個體身分識別文件，您可以透過密碼方式 您可以使用執行個體中繼資料服務 (IMDS) 來尋找這些文件。

如需詳細資訊，請參閱執行個體身分文件。

檢查系統 UUID

您可以獲取系統UUID並查看 in 的開頭八位字節EC2（在 Linux 中，這可能是小寫字母ec2）。UUID這種方法很快，但可能不準確，因為不是EC2實例的系統很有可能具有以這些字符開頭的可能性。UUID此外，某些版本的SMBIOS使用 little-endian 格式，它不包括EC2在. UUID 對於使用 SMBIOS 2.4 進行Windows 的EC2執行個體或具有自己實作的 Amazon Linux 以外的 Linux 發行版，可能就是這種情況SMBIOS。

Linux 示例：UUID從DMI（HVMAMIs僅）獲取

使用下列命令來取得UUID使用桌面管理介面 (DMI)：

```
[ec2-user ~]$ sudo dmidecode --string system-uuid
```

在下列範例輸出中，以 "EC2" UUID 開頭，表示系統可能是EC2執行個體。

```
EC2E1916-9099-7CAF-FD21-012345ABCDEF
```

在下面的例子輸出中UUID，以小端格式表示。

```
45E12AEC-DCD1-B213-94ED-012345ABCDEF
```

或者，對於建立在 Nitro 系統上的執行個體，您可以使用下列命令：

```
[ec2-user ~]$ cat /sys/devices/virtual/dmi/id/board_asset_tag
```

如果輸出為執行個體 ID，如下列範例輸出所示，則系統為EC2執行個體：

```
i-0af01c0123456789a
```

Linux 範例：UUID從虛擬機器管理程序取得 (AMIs僅限 PV)

使用以下命令從管理程序UUID中獲取：

```
[ec2-user ~]$ cat /sys/hypervisor/uuid
```

在下列範例輸出中，以「ec2" UUID 開頭，表示系統可能是EC2執行個體。

```
ec2e1916-9099-7caf-fd21-012345abcdef
```

視窗範例：取得使UUID用WMI或視窗 PowerShell

使用 Windows 管理檢測指令列 (WMIC)，如下所示：

```
wmic path win32_computersystemproduct get uuid
```

或者，如果您使用的是 Windows PowerShell，請依照下列方式使用Get-WmiObject指令程式：

```
PS C:\> Get-WmiObject -query "select uuid from Win32_ComputerSystemProduct" | Select
 UUID
```

在下列範例輸出中，以 "EC2" UUID 開頭，表示系統可能是EC2執行個體。

```
EC2AE145-D1DC-13B2-94ED-012345ABCDEF
```

對於使用 SMBIOS 2.4 的實例，UUID可能會以小端格式表示；例如：

```
45E12AEC-DCD1-B213-94ED-012345ABCDEF
```

檢查系統虛擬機器產生識別符

虛擬機器產生識別符由 128 位元的專屬緩衝區組成，該緩衝區解譯為加密隨機整數識別符。您可以擷取虛擬機器產生識別符以識別您的 Amazon Elastic Compute Cloud 執行個體。產生識別碼會透過ACPI表格項目在執行個體的客體作業系統中公開。若您的電腦被複製或匯入到 AWS(例如使用 VM Import/Export)，那麼值就會改變。

範例：從 Linux 擷取虛擬機器產生識別碼

您可使用以下命令從執行 Linux 的執行個體中擷取虛擬機器產生識別符。

Amazon Linux 2

1. 根據需要，使用以下命令更新現有軟體套件：

   ```
   sudo yum update
   ```

2. 如有必要，使用下列命令以 busybox 套件作為來源：

```
sudo curl https://www.rpmfind.net/linux/epel/next/8/Everything/x86_64/Packages/
b/busybox-1.35.0-2.el8.next.x86_64.rpm --output busybox.rpm
```

3. 如有必要，使用下列命令安裝必要條件套件：

```
sudo yum install busybox.rpm iasl -y
```

4. 執行下列`iasl`命令以產生ACPI資料表的輸出：

```
sudo iasl -p ./SSDT2 -d /sys/firmware/acpi/tables/SSDT2
```

5. 執行下列命令，以檢閱 `iasl` 命令的輸出結果：

```
cat SSDT2.dsl
```

輸出結果應產生擷取虛擬機器產生識別符所需的地址空間：

```
Intel ACPI Component Architecture
ASL+ Optimizing Compiler/Disassembler version 20190509
Copyright (c) 2000 - 2019 Intel Corporation

File appears to be binary: found 32 non-ASCII characters, disassembling
Binary file appears to be a valid ACPI table, disassembling
Input file /sys/firmware/acpi/tables/SSDT2, Length 0x7B (123) bytes
ACPI: SSDT 0x0000000000000000 00007B (v01 AMAZON AMZNSSDT 00000001 AMZN
 00000001)
Pass 1 parse of [SSDT]
Pass 2 parse of [SSDT]
Parsing Deferred Opcodes (Methods/Buffers/Packages/Regions)

Parsing completed
Disassembly completed
ASL Output:    ./SSDT2.dsl - 1065 bytes
$
/*
* Intel ACPI Component Architecture
* AML/ASL+ Disassembler version 20190509 (64-bit version)
* Copyright (c) 2000 - 2019 Intel Corporation
*
* Disassembling to symbolic ASL+ operators
```

```
*
* Disassembly of /sys/firmware/acpi/tables/SSDT2, Tue Mar 29 16:15:14 2022
*
* Original Table Header:
*     Signature        "SSDT"
*     Length           0x0000007B (123)
*     Revision         0x01
*     Checksum         0xB8
*     OEM ID           "AMAZON"
*     OEM Table ID     "AMZNSSDT"
*     OEM Revision     0x00000001 (1)
*     Compiler ID      "AMZN"
*     Compiler Version 0x00000001 (1)
*/
DefinitionBlock ("", "SSDT", 1, "AMAZON", "AMZNSSDT", 0x00000001)
{
Scope (\_SB)
{
    Device (VMGN)
    {
        Name (_CID, "VM_Gen_Counter")  // _CID: Compatible ID
        Name (_DDN, "VM_Gen_Counter")  // _DDN: DOS Device Name
        Name (_HID, "AMZN0000")  // _HID: Hardware ID
        Name (ADDR, Package (0x02)
        {
            0xFED01000,
            Zero
        })
    }
}
}
```

6. (選用) 使用下列命令提升其餘步驟的終端許可：

```
sudo -s
```

7. 使用下列命令存放先前收集的地址空間：

```
VMGN_ADDR=0xFED01000
```

8. 使用下列命令重複執行地址空間並建置虛擬機器產生識別符：

```
for offset in 0x0 0x4 0x8 0xc; do busybox devmem $(($VMGN_ADDR + $offset)) | sed
 's/0x//' | sed -z '$ s/\n$//' >> vmgenid; done
```

9. 使用下列命令從輸出檔案中擷取虛擬機器產生識別符：

```
cat vmgenid ; echo
```

您的輸出應類似以下內容：

```
EC2F335D979132C4165896753E72BD1C
```

Ubuntu

1. 根據需要，使用以下命令更新現有軟體套件：

```
sudo apt update
```

2. 如有必要，使用下列命令安裝必要條件套件：

```
sudo apt install busybox iasl -y
```

3. 執行下列`iasl`命令以產生ACPI資料表的輸出：

```
sudo iasl -p ./SSDT2 -d /sys/firmware/acpi/tables/SSDT2
```

4. 執行下列命令，以檢閱 `iasl` 命令的輸出結果：

```
cat SSDT2.dsl
```

輸出結果應產生擷取虛擬機器產生識別符所需的地址空間：

```
Intel ACPI Component Architecture
ASL+ Optimizing Compiler/Disassembler version 20190509
Copyright (c) 2000 - 2019 Intel Corporation

File appears to be binary: found 32 non-ASCII characters, disassembling
Binary file appears to be a valid ACPI table, disassembling
Input file /sys/firmware/acpi/tables/SSDT2, Length 0x7B (123) bytes
```

```
ACPI: SSDT 0x0000000000000000 00007B (v01 AMAZON AMZNSSDT 00000001 AMZN
 00000001)
Pass 1 parse of [SSDT]
Pass 2 parse of [SSDT]
Parsing Deferred Opcodes (Methods/Buffers/Packages/Regions)

Parsing completed
Disassembly completed
ASL Output:    ./SSDT2.dsl - 1065 bytes
$
/*
* Intel ACPI Component Architecture
* AML/ASL+ Disassembler version 20190509 (64-bit version)
* Copyright (c) 2000 - 2019 Intel Corporation
*
* Disassembling to symbolic ASL+ operators
*
* Disassembly of /sys/firmware/acpi/tables/SSDT2, Tue Mar 29 16:15:14 2022
*
* Original Table Header:
*     Signature        "SSDT"
*     Length           0x0000007B (123)
*     Revision         0x01
*     Checksum         0xB8
*     OEM ID           "AMAZON"
*     OEM Table ID     "AMZNSSDT"
*     OEM Revision     0x00000001 (1)
*     Compiler ID      "AMZN"
*     Compiler Version 0x00000001 (1)
*/
DefinitionBlock ("", "SSDT", 1, "AMAZON", "AMZNSSDT", 0x00000001)
{
Scope (\_SB)
{
    Device (VMGN)
    {
        Name (_CID, "VM_Gen_Counter")  // _CID: Compatible ID
        Name (_DDN, "VM_Gen_Counter")  // _DDN: DOS Device Name
        Name (_HID, "AMZN0000")  // _HID: Hardware ID
        Name (ADDR, Package (0x02)
        {
            0xFED01000,
            Zero
        })
```

```
    }
}
}
```

5. (選用) 使用下列命令提升其餘步驟的終端許可：

```
sudo -s
```

6. 使用下列命令存放先前收集的地址空間：

```
VMGN_ADDR=0xFED01000
```

7. 使用下列命令重複執行地址空間並建置虛擬機器產生識別符：

```
for offset in 0x0 0x4 0x8 0xc; do busybox devmem $(($VMGN_ADDR + $offset)) | sed
 's/0x//' | sed -z '$ s/\n$//' >> vmgenid; done
```

8. 使用下列命令從輸出檔案中擷取虛擬機器產生識別符：

```
cat vmgenid ; echo
```

您的輸出應類似以下內容：

```
EC2F335D979132C4165896753E72BD1C
```

範例：從 Windows 擷取虛擬機器產生識別碼

您可建立範例應用程式，從執行 Windows 的執行個體中擷取虛擬機器產生識別符。如需詳細資訊，請參閱 Microsoft 文件集中的《取得虛擬機器產生識別符》。

Amazon EC2 執行個體的執行個體身分證

您啟動的每個執行個體都有一個提供執行個體本身相關資訊的 執行個體身分文件。您可以使用 執行個體身分文件 來驗證執行個體的屬性。

執行個體停止、開始、重新開始或啟動時，會產生執行個體身分文件。您可以透過執行個體中繼資料服務 (IMDS) 存取執行個體的執行個體識別文件。如需指示，請參閱檢索實例身份證明文件。

執行個體識別文件使用純JSON文字格式。它包括以下信息。

資料	描述
accountId	啟動執行個體的 AWS 帳戶 ID。
architecture	AMI用來啟動執行個體的架構 (i386 | 64)。
availabilityZone	執行個體執行的可用區域。
billingProducts	執行個體的計費產品。
devpayProductCodes	已廢除。
imageId	AMI用來啟動執行個體的識別碼。
instanceId	執行個體的 ID。
instanceType	執行個體的執行個體類型。
kernelId	與執行個體相關聯的核心 ID (如適用)。
marketplaceProductCodes	AMI用來啟動執行個體的 AWS Marketplace 產品代碼。
pendingTime	執行個體啟動的日期和時間。
privateIp	執行個體的私人IPv4位址。
ramdiskId	與執行個體相關聯的RAM磁碟識別碼 (如果適用)。
region	執行個體執行時所在區域。
version	執行個體身分文件 格式的版本。

擷取 Amazon 執行個體的執行個EC2體身分文件

Amazon 執行個體的身分識別文EC2件使用純JSON文字格式。如需執行個體身分識別文件內容的說明，請參閱the section called “執行個體身分文件”。

執行個體身分識別文件會儲存在執行個體的執行個體中繼資料中，`instance-identity/document`動態資料類別中。您可以連線至執行個體並從執行個體中繼資料擷取，以存取執行個體的執行個體身分識別文件。

您可以使用IPv4位址169.254.169.254或位址存取執行個體中繼資料fd00:ec2::254。IPv6這些是鏈接本地地址，這意味著您只能從實例訪問它們。如需詳細資訊，請參閱Link-local 地址。此頁面上的範例使用IMDS:的IPv4位址169.254.169.254。若要IPv6改為擷取執EC2行個體的執行個體中繼資料，請fd00:ec2::254改用。

在擷取執行個體身份證明文件後驗證其真實性。如需詳細資訊，請參閱驗證執行個體身份證。

擷取執行個體身份證明文件

Connect 至執行個體並執行下列命令，從執行個體中繼資料存取執行個體身分識別文件。

cURL

IMDSv2

```
$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/dynamic/instance-identity/document
```

IMDSv1

```
$ curl http://169.254.169.254/latest/dynamic/instance-identity/document
```

PowerShell

IMDSv2

```
PS C:\> [string]$token = (Invoke-WebRequest -Method Put -Headers @{'X-aws-ec2-metadata-token-ttl-seconds' = '21600'} http://169.254.169.254/latest/api/token).Content
```

```
PS C:\> (Invoke-WebRequest -Headers @{'X-aws-ec2-metadata-token' = $Token}
 http://169.254.169.254/latest/dynamic/instance-identity/document).Content
```

IMDSv1

```
PS C:\> (Invoke-WebRequest http://169.254.169.254/latest/dynamic/instance-identity/
document).Content
```

下列為範例輸出。

```
{
    "devpayProductCodes" : null,
    "marketplaceProductCodes" : [ "1abc2defghijklm3nopqrs4tu" ],
    "availabilityZone" : "us-west-2b",
    "privateIp" : "10.158.112.84",
    "version" : "2017-09-30",
    "instanceId" : "i-1234567890abcdef0",
    "billingProducts" : null,
    "instanceType" : "t2.micro",
    "accountId" : "123456789012",
    "imageId" : "ami-5fb8c835",
    "pendingTime" : "2016-11-19T16:32:11Z",
    "architecture" : "x86_64",
    "kernelId" : null,
    "ramdiskId" : null,
    "region" : "us-west-2"
}
```

驗證 Amazon 執行個體的執行個EC2體身分文件

如果您打算將 執行個體身分文件 的內容用於重要目的，您應該在使用它之前驗證其內容與真實性。

純文字 執行個體身分文件 會伴隨著三個雜湊和加密的簽章。您可以使用這些簽章來驗證 執行個體身分文件 的來源和真實性及其包含的資訊。提供的簽章如下：

- Base64 編碼簽名 — 這是使用 key pair 加密的執行個體身分識別文件的 base64 編碼SHA256雜湊。RSA
- PKCS7簽章-這是使用 key pair 加SHA1密的執行個體身分識別文件的雜湊值。DSA
- RSA-2048 簽名 — 這是使用 -2048 key pair 加密的執行個體身分識別文件的SHA256雜湊值。RSA

每個簽章都可以在執行個體中繼資料的不同端點上使用。您可以根據您的雜湊和加密需求使用任何一個簽章。若要驗證簽章，您必須使用對應的 AWS 公用憑證。

選項

- 選項 1：使用PKCS7簽名驗證實例身份證明文件
- 選項 2：使用 base64 編碼的簽名驗證執行個體身分文件
- 選項 3：使用 RSA -2048 簽名驗證實例身份證明文件

選項 1：使用PKCS7簽名驗證實例身份證明文件

本主題說明如何使用PKCS7簽名和 AWS DSA公用憑證來驗證執行個體身分識別文件。

Linux 執行個體

使用PKCS7簽名和 AWS DSA公共證書驗證實例身份證明文件

1. 連線到執行個體。
2. 從執行個體中繼資料擷取PKCS7簽名，並將簽名新增至名為的新檔案，以`pkcs7`及必要的頁首和頁尾。根據執行個體所使用的IMDS版本，使用下列其中一個指令。

   IMDSv2

   ```
   $ echo "-----BEGIN PKCS7-----" >> pkcs7 \
    && TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
    && curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/dynamic/instance-identity/pkcs7 >> pkcs7 \
    && echo "" >> pkcs7 \
    && echo "-----END PKCS7-----" >> pkcs7
   ```

   IMDSv1

   ```
   $ echo "-----BEGIN PKCS7-----" >> pkcs7 \
    && curl -s http://169.254.169.254/latest/dynamic/instance-identity/pkcs7
    >> pkcs7 \
    && echo "" >> pkcs7 \
    && echo "-----END PKCS7-----" >> pkcs7
   ```

3. 在中找到您區域的DSA公共證書，AWS 執行個體身分識別文件簽名的公開並將內容添加到名為的新文件中`certificate`。
4. 使用「開啟 SSL smime」指令來驗證簽章。包含指出需要驗證簽章的 `-verify` 選項，以及指出憑證不需要驗證的 `-noverify` 選項。

```
$ openssl smime -verify -in pkcs7 -inform PEM -certfile certificate -noverify | tee
 document
```

如果簽章有效，則會出現 `Verification successful` 訊息。

此命令也會將執行個體身分文件內容寫入名為 `document` 的新檔案。您可以使用下列命令，將來自執行個體中繼資料的執行個體身分文件內容與此檔案內容比較。

```
$ openssl dgst -sha256 < document
```

```
$ curl -s -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/
dynamic/instance-identity/document | openssl dgst -sha256
```

如果無法驗證簽章，請聯絡 AWS Support。

Windows 執行個體

必要條件

此過程需要 `System.Security` Microsoft。 NET核心類。若要將類別新增至您的 PowerShell 工作階段，請執行下列命令。

```
PS C:\> Add-Type -AssemblyName System.Security
```

> **Note**
>
> 該命令僅將類添加到當前 PowerShell 會話中。如果您啟動新的工作階段，您必須再次執行命令。

使用PKCS7簽名和 AWS DSA公共證書驗證實例身份證明文件

1. 連線到執行個體。
2. 從執行個體中繼資料擷取PKCS7簽章，將其轉換為位元組陣列，然後將其新增至名為的變數`$Signature`。根據執行個體所使用的IMDS版本，使用下列其中一個指令。

IMDSv2

```
PS C:\> [string]$token = (Invoke-WebRequest -Method Put -Headers @{'X-aws-
ec2-metadata-token-ttl-seconds' = '21600'} http://169.254.169.254/latest/api/
token).Content
```

```
PS C:\> $Signature = [Convert]::FromBase64String((Invoke-WebRequest -Headers
 @{'X-aws-ec2-metadata-token' = $Token} http://169.254.169.254/latest/dynamic/
instance-identity/pkcs7).Content)
```

IMDSv1

```
PS C:\> $Signature = [Convert]::FromBase64String((Invoke-WebRequest
 http://169.254.169.254/latest/dynamic/instance-identity/pkcs7).Content)
```

3. 從執行個體中繼資料擷取純文字執行個體身分文件，將其轉換為位元組陣列，然後新增至名為 `$Document` 的變數。根據執行個體所使用的IMDS版本，使用下列其中一個指令。

   IMDSv2

   ```
   PS C:\> $Document = [Text.Encoding]::UTF8.GetBytes((Invoke-WebRequest -Headers
    @{'X-aws-ec2-metadata-token' = $Token} http://169.254.169.254/latest/dynamic/
   instance-identity/document).Content)
   ```

   IMDSv1

   ```
   PS C:\> $Document =  [Text.Encoding]::UTF8.GetBytes((Invoke-WebRequest
    http://169.254.169.254/latest/dynamic/instance-identity/document).Content)
   ```

4. 在中找到您區域的DSA公共證書，AWS 執行個體身分識別文件簽名的公開並將內容添加到名為的新文件中`certificate.pem`。
5. 從憑證檔案擷取憑證，並將其存放在名為 `$Store` 的變數中。

   ```
   PS C:\> $Store =
    [Security.Cryptography.X509Certificates.X509Certificate2Collection]::new([Security.Cryptog
   Path certificate.pem)))
   ```

6. 驗證簽章。

```
PS C:\> $SignatureDocument = [Security.Cryptography.Pkcs.SignedCms]::new()
```

```
PS C:\> $SignatureDocument.Decode($Signature)
```

```
PS C:\> $SignatureDocument.CheckSignature($Store, $true)
```

如果簽章有效，則命令不會傳回任何輸出。如果無法驗證簽章，則命令會傳回 `Exception calling "CheckSignature" with "2" argument(s): "Cannot find the original signer`。如果無法驗證您的簽章，請聯絡 AWS Support。

7. 驗證執行個體身分文件的內容。

```
PS C:
\> [Linq.Enumerable]::SequenceEqual($SignatureDocument.ContentInfo.Content, $Document)
```

如果執行個體身分文件的內容有效，則命令會傳回 `True`。如果無法驗證實例身份證明文件，請聯繫 AWS Support。

選項 2：使用 base64 編碼的簽名驗證執行個體身分文件

本主題說明如何使用 base64 編碼的簽章和 AWS RSA公用憑證來驗證執行個體身分識別文件。

Linux 執行個體

使用 base64 編碼的簽名和公用憑證驗證執行個體身分識別文件 AWS RSA

1. 連線到執行個體。
2. 從執行個體中繼資料擷取 base64 編碼的簽章，將其轉換為二進位，然後新增至名為 `signature` 的檔案。根據執行個體所使用的IMDS版本，使用下列其中一個指令。

   IMDSv2

   ```
   $ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
    && curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/dynamic/instance-identity/signature | base64 -d >> signature
   ```

IMDSv1

```
$ curl -s http://169.254.169.254/latest/dynamic/instance-identity/signature |
 base64 -d >> signature
```

3. 從執行個體中繼資料擷取純文字 執行個體身分文件，並新增至名為 document 的檔案。根據執行個體所使用的IMDS版本，使用下列其中一個指令。

   IMDSv2

   ```
   $ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-
   metadata-token-ttl-seconds: 21600"` \
    && curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/
   dynamic/instance-identity/document >> document
   ```

   IMDSv1

   ```
   $ curl -s http://169.254.169.254/latest/dynamic/instance-identity/document
    >> document
   ```

4. 在中找到您區域的RSA公共證書，AWS 執行個體身分識別文件簽名的公開並將內容添加到名為的新文件中`certificate`。
5. 從公開憑證擷取 AWS RSA公開金鑰，並將其儲存到名為的檔案中`key`。

   ```
   $ openssl x509 -pubkey -noout -in certificate >> key
   ```

6. 使用 Open SSL dgst 命令來驗證實例身份證明文件。

   ```
   $ openssl dgst -sha256 -verify key -signature signature document
   ```

   如果簽章有效，則會出現 `Verification successful` 訊息。

   此命令也會將執行個體身分文件內容寫入名為 document 的新檔案。您可以使用下列命令，將來自執行個體中繼資料的執行個體身分文件內容與此檔案內容比較。

   ```
   $ openssl dgst -sha256 < document
   ```

```
$ curl -s -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/
dynamic/instance-identity/document | openssl dgst -sha256
```

如果無法驗證簽章，請聯絡 AWS Support。

Windows 執行個體

使用 base64 編碼的簽名和公用憑證驗證執行個體身分識別文件 AWS RSA

1. 連線到執行個體。
2. 從執行個體中繼資料擷取 base64 編碼的簽章，將其轉換為位元組陣列，然後新增至名為 `$Signature` 的變數。根據執行個體所使用的IMDS版本，使用下列其中一個指令。

   IMDSv2

   ```
   PS C:\> [string]$token = (Invoke-WebRequest -Method Put -Headers @{'X-aws-
   ec2-metadata-token-ttl-seconds' = '21600'} http://169.254.169.254/latest/api/
   token).Content
   ```

   ```
   PS C:\> $Signature = [Convert]::FromBase64String((Invoke-WebRequest -Headers
    @{'X-aws-ec2-metadata-token' = $Token} http://169.254.169.254/latest/dynamic/
   instance-identity/signature).Content)
   ```

   IMDSv1

   ```
   PS C:\> $Signature = [Convert]::FromBase64String((Invoke-WebRequest
    http://169.254.169.254/latest/dynamic/instance-identity/signature).Content)
   ```

3. 從執行個體中繼資料擷取純文字執行個體身分文件，將其轉換為位元組陣列，然後新增至名為 `$Document` 的變數。根據執行個體所使用的IMDS版本，使用下列其中一個指令。

   IMDSv2

   ```
   PS C:\> $Document = [Text.Encoding]::UTF8.GetBytes((Invoke-WebRequest -Headers
    @{'X-aws-ec2-metadata-token' = $Token} http://169.254.169.254/latest/dynamic/
   instance-identity/document).Content)
   ```

IMDSv1

```
PS C:\> $Document =  [Text.Encoding]::UTF8.GetBytes((Invoke-WebRequest
 http://169.254.169.254/latest/dynamic/instance-identity/document).Content)
```

4. 在中找到您區域的RSA公共證書，AWS 執行個體身分識別文件簽名的公開並將內容添加到名為的新文件中`certificate.pem`。
5. 驗證執行個體身分文件。

```
PS C:\> [Security.Cryptography.X509Certificates.X509Certificate2]::new((Resolve-
Path certificate.pem)).PublicKey.Key.VerifyData($Document, 'SHA256', $Signature)
```

如果簽章有效，則命令會傳回 `True`。如果無法驗證簽章，請聯絡 AWS Support。

選項 3：使用 RSA -2048 簽名驗證實例身份證明文件

本主題說明如何使用 RSA -2048 簽名和 -2048 公用憑證來驗證執行個體身分識別文件。 AWS RSA

Linux 執行個體

使用 RSA -2048 簽名和 -2048 公用憑證來驗證執行個體身份證明文件 AWS RSA

1. 連線到執行個體。
2. 從執行個體中繼資料擷取 RSA -2048 簽名，並將其新增至`rsa2048`沿著所需頁首和頁尾命名的檔案。根據執行個體所使用的IMDS版本，使用下列其中一個指令。

IMDSv2

```
$ echo "-----BEGIN PKCS7-----" >> rsa2048 \
 && TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-
metadata-token-ttl-seconds: 21600"` \
 && curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/
dynamic/instance-identity/rsa2048 >> rsa2048 \
 && echo "" >> rsa2048 \
 && echo "-----END PKCS7-----" >> rsa2048
```

IMDSv1

```
$ echo "-----BEGIN PKCS7-----" >> rsa2048 \
```

```
&& curl -s http://169.254.169.254/latest/dynamic/instance-identity/rsa2048
>> rsa2048 \
&& echo "" >> rsa2048 \
&& echo "-----END PKCS7-----" >> rsa2048
```

3. 在中尋找您地區的 RSA-2048 公用憑證，AWS 執行個體身分識別文件簽名的公開並將內容新增至名為的新檔案。`certificate`
4. 使用「開啟 SSL smime」指令來驗證簽章。包含指出需要驗證簽章的 `-verify` 選項，以及指出憑證不需要驗證的 `-noverify` 選項。

```
$ openssl smime -verify -in rsa2048 -inform PEM -certfile certificate -noverify |
 tee document
```

如果簽章有效，則會出現 `Verification successful` 訊息。如果無法驗證簽章，請聯絡 AWS Support。

Windows 執行個體

必要條件

此過程需要 `System.Security` Microsoft。 NET核心類。若要將類別新增至您的 PowerShell 工作階段，請執行下列命令。

```
PS C:\> Add-Type -AssemblyName System.Security
```

> **Note**
> 該命令僅將類添加到當前 PowerShell 會話中。如果您啟動新的工作階段，您必須再次執行命令。

使用 RSA -2048 簽名和 -2048 公用憑證來驗證執行個體身份證明文件 AWS RSA

1. 連線到執行個體。
2. 從執行個體中繼資料擷取 RSA -2048 簽章，將其轉換為位元組陣列，並將其新增至名為的變數。`$Signature`根據執行個體所使用的IMDS版本，使用下列其中一個指令。

IMDSv2

```
PS C:\> [string]$token = (Invoke-WebRequest -Method Put -Headers @{'X-aws-
ec2-metadata-token-ttl-seconds' = '21600'} http://169.254.169.254/latest/api/
token).Content
```

```
PS C:\> $Signature = [Convert]::FromBase64String((Invoke-WebRequest -Headers
 @{'X-aws-ec2-metadata-token' = $Token} http://169.254.169.254/latest/dynamic/
instance-identity/rsa2048).Content)
```

IMDSv1

```
PS C:\> $Signature = [Convert]::FromBase64String((Invoke-WebRequest
 http://169.254.169.254/latest/dynamic/instance-identity/rsa2048).Content)
```

3. 從執行個體中繼資料擷取純文字執行個體身分文件，將其轉換為位元組陣列，然後新增至名為 `$Document` 的變數。根據執行個體所使用的IMDS版本，使用下列其中一個指令。

   IMDSv2

   ```
   PS C:\> $Document = [Text.Encoding]::UTF8.GetBytes((Invoke-WebRequest -Headers
    @{'X-aws-ec2-metadata-token' = $Token} http://169.254.169.254/latest/dynamic/
   instance-identity/document).Content)
   ```

   IMDSv1

   ```
   PS C:\> $Document =  [Text.Encoding]::UTF8.GetBytes((Invoke-WebRequest
    http://169.254.169.254/latest/dynamic/instance-identity/document).Content)
   ```

4. 在中尋找您地區的 RSA-2048 公用憑證，AWS 執行個體身分識別文件簽名的公開並將內容新增至名為的新檔案。`certificate.pem`

5. 從憑證檔案擷取憑證，並將其存放在名為 `$Store` 的變數中。

   ```
   PS C:\> $Store =
    [Security.Cryptography.X509Certificates.X509Certificate2Collection]::new([Security.Cryptog
   Path certificate.pem)))
   ```

6. 驗證簽章。

```
PS C:\> $SignatureDocument = [Security.Cryptography.Pkcs.SignedCms]::new()
```

```
PS C:\> $SignatureDocument.Decode($Signature)
```

```
PS C:\> $SignatureDocument.CheckSignature($Store, $true)
```

如果簽章有效，則命令不會傳回任何輸出。如果無法驗證簽章，則命令會傳回 `Exception calling "CheckSignature" with "2" argument(s): "Cannot find the original signer`。如果無法驗證您的簽章，請聯絡 AWS Support。

7. 驗證執行個體身分文件的內容。

```
PS C:
\> [Linq.Enumerable]::SequenceEqual($SignatureDocument.ContentInfo.Content, $Document)
```

如果執行個體身分文件的內容有效，則命令會傳回 `True`。如果無法驗證實例身份證明文件，請聯繫 AWS Support。

AWS 執行個體身分識別文件簽名的公開

下列 AWS 公用憑證可用來驗證執行個體身分識別文件的內容，如中所述驗證執行個體身份證。

請依據您的所在區域以及您正在使用的驗證程序，確定您使用正確的憑證。如果您正在驗證PKCS7簽名，請使用DSA憑證。如果您要驗證 Base6 編碼的簽名，請使用憑證。RSA如果您正在驗證 RSA -2048 簽名，請使用 RSA -2048 憑證。

展開下方的每個區域，以檢視特定區域的憑證。

美國東部 (維吉尼亞北部) — us-east-1

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
```

```
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUE1y2NIKCU+Rg4uu4u32koG9QEYIwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE3MzQwMVoXDTI5MDQyODE3MzQwMVowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUE1y2NIKC
U+Rg4uu4u32koG9QEYIwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQAlxSmwcWnhT4uAeSinJuz+1BTcKhVSWb5jT8pYjQb8ZoZkXXRGb09mvYeU
NeqOBr27rvRAnaQ/9LUQf72+SahDFuS4CMI8nwowytqbmwquqFr4dxA/SDADyRiF
ea1UoMuNHTY49J/1vPomqsVn7mugTp+TbjqCfOJTpu0temHcFA==
-----END CERTIFICATE-----
```


RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJALFpzEAVWaQZMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNTA4MTQw
ODU5MTJaGA8yMTk1MDExNzA4NTkxMlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
```

```
CgKCAQEAjS2vqZu9mEOhOq+0bRpAbCUiapbZMFNQqRg7kTlr7Cf+gDqXKpHPjsng
SfNz+JHQd8WPI+pmNs+q0Z2aTe23klmf2U52KH9/j1k8RlIbap/yFibFTSedmegX
E5r447GbJRsHUmuIIfZTZ/oRlpuIIO5/Vz7SOj22tdkdY2ADp7caZkNxhSP915fk
2jJMTBUOzyXUS2rBU/ulNHbTTeePjcEkvzVYPahD30TeQ+/A+uWUu89bHSQOJR8h
Um4cFApzZgN3aD5j2LrSMu2pctkQwf9CaWyVznqrsGYjYOY66LuFzSCXwqSnFBfv
fFBAFsjCgY24G2DoMyYkF3MyZlu+rwIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQUrynSPp4uqSECwy+PiO4qyJ8TWSkwgY4GA1UdIwSBhjCBg4AUrynS
Pp4uqSECwy+PiO4qyJ8TWSmhYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJALFpzEAVWaQZMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBADW/s8lXijwdP6NkEoH1m9XLrvK4YTqkNfR6
er/uRRgTx2QjFcMNrx+g87gAml11z+D0crAZ5LbEhDMs+JtZYR3ty0HkDk6SJM85
haoJNAFF7EQ/zCp1EJRIkLLsC7bcDL/Eriv1swt78/BB4RnC9W9kSp/sxd5svJMg
N9a6FAplpNRsWAnbP8JBlAP93oJzblX2LQXgykTghMkQO7NaY5hg/H5o4dMPclTK
lYGqlFUCH6A2vdrxmpKDLmTn5//5pujdD2MN0df6sZWtxwZ0osljV4rDjm9Q3VpA
NWIsDEcp3GUB4proOR+C7PNkY+VGODitBOw09qBGosCBstwyEqY=
-----END CERTIFICATE-----
```


美國東部 (俄亥俄) — us-east-2

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```

RSA


```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUVJTc+hOU+8Gk3JlqsX438Dk5c58wDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE3MTE0OVoXDTI5MDQyODE3MTE0OVowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUVJTc+hOU
+8Gk3JlqsX438Dk5c58wEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQAywJQaVNWJqW0R0T0xVOSoN1GLk9x9kKEuN67RN9CLin4dA97qa7Mr5W4P
FZ6vnh5CjOhQBRXV9xJUeYSdqVItNAUFK/fEzDdjf1nUfPlQ3OJ49u6CV01NoJ9m
usvY9kWcV46dqn2bk2MyfTTgvmeqP8fiMRPxxnVRkSzlldP5Fg==
-----END CERTIFICATE-----
```


RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAM07oeX4xevdMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNjA2MTAx
MjU4MThaGA8yMTk1MTExNDEyNTgxOFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEA6v6kGMnRmFDLxBEqXzP4npnL65OO0kmQ7w8YXQygSdmNIoScGSU5wfh9
mZdcvCxCdxgALFsFqPvH8fqiE9ttI0fEfuZvHOs8wUsIdKr0Zz0MjSx3cik4tKET
ch0EKfMnzKOgDBavraCDeX1rUDU0Rg7HFqNAOry3uqDmnqtk00XC9GenS3z/7ebJ
fIBEPAam5oYMVFpX6M6St77WdNE8wEU8SuerQughiMVx9kMB07imeVHBiELbMQ0N
lwSWRL/61fA02keGSTfSp/0m3u+lesf2VwVFhqIJs+JbsEscPxOkIRlzy8mGd/JV
ONb/DQpTedzUKLgXbw7KtO3HTG9iXQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQU2CTGYE5fTjx7gQXzdZSGPEWAJY4wgY4GA1UdIwSBhjCBg4AU2CTG
YE5fTjx7gQXzdZSGPEWAJY6hYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJAM07oeX4xevdMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBANdqkIpVypr2PveqUsAKke1wKCOSuw1UmH9k
xX1/VRoHbrI/UznrXtPQOPMmHA2LKSTedwsJuorUn3cFH6qNs8ixBDrl8pZwfKOY
```

```
IBJcTFBbI1xBEFkZoO3wczzo5+8vPQ60RVqAaYb+iCa1HFJpccC3Ovajfa4GRdNb
n6FYnluIcDbmpcQePoVQwX7W3oOYLB1QLN7fE6H1j4TBIsFdO3OuKzmaifQlwLYt
DVxVCNDabpOr6Uozd5ASm4ihPPoEoKo7Ilp0fOT6fZ41U2xWA4+HF/89UoygZSo7
K+cQ90xGxJ+gmlYbLFR5rbJOLfjrgDAb2ogbFy8LzHo2ZtSe60M=
-----END CERTIFICATE-----
```


美國西部 (加利佛尼亞北部) — us-west-1

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUK2zmY9PUSTR7rc1k2OwPYu4+g7wwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE3MDI0M1oXDTI5MDQyODE3MDI0M1owXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
```

```
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUK2zmY9PU
STR7rc1k2OwPYu4+g7wwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQA1Ng4QmN4n7iPh5CnadSOc0ZfM7by0dBePwZJyGvOHdaw6P6E/vEk76KsC
Q8p+akuzVzVPkU4kBK/TRqLp19wEWoVwhhTaxHjQ1tTRHqXIVlrkw4JrtFbeNM21
GlkSLonuzmNZdivn9WuQYeGe7nUD4w3q9GgiF3CPorJe+UxtbA==
-----END CERTIFICATE-----
```


RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJANNPkIpcyEtIMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNTEwMjkw
OTAzMDdaGA8yMTk1MDQwMzA5MDMwN1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEApHQGvHvq3SVCzDrC7575BW7GWLzcj8CLqYcL3YY7Jffupz7OjcftO57Z
4fo5Pj0CaS8DtPzh8+8vdwUSMbiJ6cDd3ooio3MnCq6DwzmsY+pY7CiI3UVG7KcH
4TriDqr1Iii7nB5MiPJ8wTeAqX89T3SYaf6Vo+4GCb3LCDGvnkZ9TrGcz2CHkJsj
AIGwgopFpwhIjVYm7obmuIxSIUv+oNH0wXgDL029Zd98SnIYQd/njiqkzE+lvXgk
4h4Tu17xZIKBgFcTtWPky+POGu81DYFqiWVEyR2JKKm2/iR1dL1YsT39kbNg47xY
aR129sS4nB5Vw3TRQA2jL0ToTIxzhQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQUgepyiONs8j+q67dmcWu+mKKDa+gwgY4GA1UdIwSBhjCBg4AUgepy
iONs8j+q67dmcWu+mKKDa+ihYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJANNPkIpcyEtIMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAGLFWyutf1u0xcAc+kmnMPqtc/Q6b79VIX0E
tNoKMI2KR8lcV8ZElXDb0NC6v8UeLpe1WBKjaWQtEjL1ifKg9hdY9RJj4RXIDSK7
33qCQ8juF4vep2U5TTBd6hfWxt1Izi88xudjixmbpUU4YKr8UPbmixldYR+BEx0u
B1KJi9l1lxvuc/Igy/xeHOAZEjAXzVvHp8Bne33VVwMiMxWECZCiJxE4I7+Y6fqJ
pLLSFFJKbNaFyXlDiJ3kXyePEZSc1xiWeyRB2ZbTi5eu7vMG4i3AYWuFVLthaBgu
lPfHafJpj/JDcqt2vKUKfur5edQ6j1CGdxqqjawhOTEqcN8m7us=
-----END CERTIFICATE-----
```


美國西部 (奧勒岡) — us-west-2

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
```

```
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUFx8PxCkbHwpD31bOyCtyz3GclbgwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE3MjM1OVoXDTI5MDQyODE3MjM1OVowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUFx8PxCkb
HwpD31bOyCtyz3GclbgwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQBzOl+9Xy1+UsbUBI95HO9mbbdnuX+aMJXgG9uFZNjgNEbMcvx+h8P9IMko
z7PzFdheQQ1NLjsHH9mSR1SyC4m9ja6BsejH5nLBWyCdjfdP3muZM4O5+r7vUa1O
dWU+hP/T7DUrPAIVMOE7mpYa+WPWJrN6BlRwQkKQ7twm9kDalA==
-----END CERTIFICATE-----
```


RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJALZL3lrQCSTMMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
```

```
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNTA4MTQw
OTAxMzJaGA8yMTk1MDExNzA5MDEzMlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEA02Y59qtAA0a6uzo7nEQcnJ26OKF+LRPwZfixBH+EbEN/Fx0gYy1jpjCP
s5+VRNg6/WbfqAsV6X2VSjUKN59ZMnMY9ALA/Ipz0n00Huxj38EBZmX/NdNqKm7C
qWu1q5kmIvYjKGiadfboU8wLwLcHo8ywvfgI6FiGGsEO9VMC56E/hL6Cohko11LW
dizyvRcvg/IidazVkJQCN/4zC9PUOVyKdhW33jXy8BTg/QH927QuNk+ZzD7HH//y
tIYxDhR6TIZsSnRjz3bOcEHxt1nsidc65mY0ejQty4hy7ioSiapw316mdbtE+RTN
fcH9FPIFKQNBpiqfAW5Ebp3Lal3/+wIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQU7coQx8Qnd75qA9XotSWT3IhvJmowgY4GA1UdIwSBhjCBg4AU7coQ
x8Qnd75qA9XotSWT3IhvJmqhYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJALZL3lrQCSTMMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAFZ1e2MnzRaXCaLwEC1pW/f0oRG8nHrlPZ9W
OYZEWbh+QanRgaikBNDtVTwARQcZm3z+HWSkaIx3cyb6vM0DSkZuiwzm1LJ9rDPc
aBm03SEt5v8mcc7sXWvgFjCnUpzosmky6JheCD4O1Cf8k0olZ93FQnTrbg62OK0h
83mGCDeVKU3hLH97FYoUq+3N/IliWFDhvibAYYKFJydZLhIdlCiiB99AM6Sg53rm
oukS3csyUxZyTU2hQfdjyo1nqW9yhvFAKjnnggiwxNKTTPZzstKW8+cnYwiiTwJN
QpVoZdt0SfbuNnmwRUMi+QbuccXweav29QeQ3ADqjgB0CZdSRKk=
-----END CERTIFICATE-----
```


非洲 (開普敦) — af-south-1

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7DCCAqwCCQCncbCtQbjuyzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xOTA2MDQxMjQ4MDVaFw00
NTA2MDQxMjQ4MDVaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbYwggErBgcqhkjOOAQBMIIBHgKBgQC12Nr1gMrHcFSZ7S/A
pQBSCMHWmn2qeoQTMVWqe50fnTd0zGFxDdIjKxUK58/8zjWG5uR4TXRzmZpGpmXB
bSufAR6BGqud2LnT/HIWGJAsnX2uOtSyNfCoJigqwhea5w+CqZ6I7iBDdnB4TtTw
qO6TlnExHFVj8LMkylZgiaE1CQIVAIhdobse4K0QnbAhCL6R2euQzloXAoGAV/21
WUuMz/79Ga0JvQcz1FNy1sT0pU9rU4TenqLQIt5iccn/7EIfNtvVO5TZKulIKq7J
gXZr0x/KIT8zsNweetLOaGehPIYRMPX0vunMMR7hN7qA7W17WZv/76adywIsnDKq
ekfe15jinaX8MsKUdyDK7Y+ifCG4PVhoM4+W2XwDgYQAAoGAIxOKbVgwLxbn6Pi2
6hBOihFv16jKxAQI0hHzXJLV0Vyv9QwnqjJJRfOCy3dB0zicLXiIxeIdYfvqJr+u
hlN8rGxEZYYJjEUKMGvsc0DW85jonXz0bNfcP0aaKH0lKKVjL+OZi5n2kn9wgdo5
```

```
F3CVnMl8BUra8A1Tr2yrrE6TVZ4wCQYHKoZIzjgEAwMvADAsAhQfa7MCJZ+/TEY5
AUr0J4wm8VzjoAIUSYZVu2NdRJ/ERPmDfhW5EsjHlCA=
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIICNjCCAZ+gAwIBAgIJAKumfZiRrNvHMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xOTExMjcw
NzE0MDVaGA8yMTk5MDUwMjA3MTQwNVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKB
gQDFd571nUzVtke3rPyRkYfvs3jh0C0EMzzG72boyUNjnfw1+m0TeFraTLKb9T6F
7TuB/ZEN+vmlYqr2+5Va8U8qLbPF0bRH+FdaKjhgWZdYXxGzQzU3ioy5W5ZM1VyB
7iUsxEAlxsybC3ziPYaHI42UiTkQNahmoroNeqVyHNnBpQIDAQABMA0GCSqGSIb3
DQEBCwUAA4GBAAJLylWyElEgOpW4B1XPyRVD4pAds8Guw2+krgqkY0HxLCdjosuH
RytGDGN+q75aAoXzW5a7SGpxLxk6Hfv0xp3RjDHsoeP0i1d8MD3hAC5ezxS4oukK
s5gbPOnokhKTMPXbTdRn5ZifCbWlx+bYN/mTYKvxho7b5SVg2o1La9aK
-----END CERTIFICATE-----
```


RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJAIFI+O5A6/ZIMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xOTA2MDQx
MjQ4MDRaGA8yMTk4MTEwNzEyNDgwNFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAy7/WHBBHOrk+20aumT07g8rxrSM0UXgki3eYgKauPCG4Xx//vwQbuZwI
oeVmR9nqnhfij2wOcQdbLandh0EGtbxerete3IoXzd1KXJb11PVmzrzyu5SPBPuP
iCeV4qdjjkXo2YWM6t9YQ911hcG96YSp89TBXFYUh3KLxfqAdTVhuC0NRGhXpyii
j/czo9njofHhqhTr7UEyPun8NVS2QWctLQ86N5zWR3Q0GRoVqqMrJs0cowHTrVw2
9Qr7QBjjBOVbyYmtYxm/DtiKprYV/e6bCAVok015X1sZDd3oCOQNoGlv5XbHJe2o
JFD8GRRy2rkWO/lNwVFDcwec6zC3QwIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQCE
goqzjpCpmMgCpszFHwvRaSMbspKtK7wNImUjrSBOfBJsfFulyg1Zgn2nDCK7kQhx
jMJmNIvXbps3yMqQ2cHUkKcKf5t+WldfeT4Vk1Rz6HSA8sd0kgVcIesIaoy2aaXU
VEB/oQziRGyKdN1d4TGYVZXG44CkrzSDvlbmfiTq5tL+kAieznVF3bzHgPZW6hKP
EXC3G/IXrXicFEe6YyE1Rakl62VncYSXiGe/i2XvsiNH3Qlmnx5XS7W0SCN0oAxW
EH9twibauv82DVg1WOkQu8EwFw8hFde9X0Rkiu0qVcuU8lJgFEvPWMDFU5sGB6ZM
gkEKTzMvlZpPbBhg99Jl
-----END CERTIFICATE-----
```

亞太區域 (香港) — ap-east-1

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7zCCAq4CCQCO7MJe5Y3VLjAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xOTAyMDMwMjIxMjFaFw00
NTAyMDMwMjIxMjFaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbgwggEsBgcqhkjOOAQBMIIBHwKBgQDvQ9RzVvf4MAwGbqfX
blCvCoVb9957OkLGn/04CowHXJ+vTBR7eyIa6AoXltsQXBOmrJswToFKKxT4gbuw
jK7s9QQX4CmTRWcEgO2RXtZSVjOhsUQMh+yf7Ht4OVL97LWnNfGsX2cwjcRWHYgI
7lvnuBNBzLQHdSEwMNq0Bk76PwIVAMan6XIEEPnwr4e6u/RNnWBGKd9FAoGBAOCG
eSNmxpW4QFu4pIlAykm6EnTZKKHT87gdXkAkfoC5fAfOxxhnE2HezZHp9Ap2tMV5
8bWNvoPHvoKCQqwfm+OUBlAxC/3vqoVkKL2mG1KgUH9+hrtpMTkwO3RREnKe7I5O
x9qDimJpOihrL4I0dYvy9xUOoz+DzFAW8+ylWVYpA4GFAAKBgQDbnBAKSxWr9QHY
6Dt+EFdGz6lAZLedeBKpaP53Z1DTO34J0C55YbJTwBTFGqPtOLxnUVDlGiD6GbmC
80f3jvogPR1mSmGsydbNbZnbUEVWrRhe+y5zJ3g9qs/DWmDW0deEFvkhWVnLJkFJ
9pdOu/ibRPH1lE2nz6pK7GbOQtLyHTAJBgcqhkjOOAQDAzAAMC0CFQCoJlwGtJQC
cLoM4p/jtVFOj26xbgIUUS4pDKyHaG/eaygLTtFpFJqzWHc=
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIICSzCCAbQCCQDtQvkVxRvK9TANBgkqhkiG9w0BAQsFADBqMQswCQYDVQQGEwJV
UzETMBEGA1UECBMKV2FzaGluZ3RvbjEQMA4GA1UEBxMHU2VhdHRsZTEYMBYGA1UE
ChMPQW1hem9uLmNvbSBJbmMuMRowGAYDVQQDExFlYzIuYW1hem9uYXdzLmNvbTAe
Fw0xOTAyMDMwMzAwMDZaFw0yOTAyMDIwMzAwMDZaMGoxCzAJBgNVBAYTAlVTMRMw
EQYDVQQIEwpXYXNoaW5ndG9uMRAwDgYDVQQHEwdTZWF0dGxlMRgwFgYDVQQKEw9B
bWF6b24uY29tIEluYy4xGjAYBgNVBAMTEWVjMi5hbWF6b25hd3MuY29tMIGfMA0G
CSqGSIb3DQEBAQUAA4GNADCBiQKBgQC1kkHXYTfc7gY5Q55JJhjTieHAgacaQkiR
Pity9QPDE3b+NXDh4UdP1xdIw73JcIIG3sG9RhWiXVCHh6KkuCTqJfPUknIKk8vs
M3RXflUpBe8Pf+P92pxqPMCz1Fr2NehS3JhhpkCZVGxxwLC5gaG0Lr4rFORubjYY
Rh84dK98VwIDAQABMA0GCSqGSIb3DQEBCwUAA4GBAA6xV9f0HMqXjPHuGILDyaNN
dKcvplNFwDTydVg32MNubAGnecoEBtUPtxBsLoVYXCOb+b5/ZMDubPF9tU/vSXuo
TpYM5Bq57gJzDRaBOntQbX9bgHiUxw6XZWaTS/6xjRJDT5p3S1E0mPI3lP/eJv4o
Ezk5zb3eIf10/sqt4756
-----END CERTIFICATE-----
```

RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJAMoxixvs3YssMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xODA3MjAw
ODQ0NDRaGA8yMTk3MTIyMzA4NDQ0NFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEA4T1PNsOg0FDrGlWePoHeOSmOJTA3HCRy5LSbYD33GFU2eBrOIxoU/+SM
rInKu3GghAMfH7WxPW3etIAZiyTDDU5RLcUq2Qwdr/ZpXAWpYocNc/CEmBFtfbxF
z4uwBIN3/drM0RSbe/wP9EcgmNUGQMMZWeAji8sMtwpOblNWAP9BniUG0Flcz6Dp
uPovwDTLdAYT3TyhzlohKL3f6O48TR5yTaV+3Ran2SGRhyJjfh3FRpP4VC+z5LnT
WPQHN74Kdq35UgrUxNhJraMGCzznolUuoR/tFMwR93401GsM9fVA7SW3jjCGF81z
PSzjy+ArKyQqIpLW1YGWDFk3sf08FQIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQDK
2/+C3nPMgtyOFX/I3Cyk+Pui44IgOwCsIdNGwuJysdqp5VIfnjegEu2zIMWJSKGO
lMZoQXjffkVZZ97J7RNDW06oB7kj3WVE8a7U4WEOfnO/CbMUf/x99CckNDwpjgW+
K8V8SzAsQDvYZs2KaE+18GFfLVF1TGUYK2rPSZMHyX+v/TIlc/qUceBycrIQ/kke
jDFsihUMLqgmOV2hXKUpIsmiWMGrFQV4AeV0iXP8L/ZhcepLf1t5SbsGdUA3AUY1
3If8s81uTheiQjwY5t9nM0SY/1Th/tL3+RaEI79VNEVfG1FQ8mgqCK0ar4m0oZJl
tmmEJM7xeURdpBBx36Di
-----END CERTIFICATE-----
```


亞太區域 (海德拉巴) — ap-south-2

DSA


```
-----BEGIN CERTIFICATE-----
MIIC8DCCArCgAwIBAgIGAXjrQ4+XMAkGByqGSM44BAMwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgMEFdhc2hpbmd0b24g
U4EddRIpUt9KnC7s5Of2EbdSPO9EAMMeP4C2USZpRV1AIlH7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdrmVClpJ+f6AR7ECLCT7up1/63xhv4O1fnxqimFQ8E
+4P208UewwI1VBNaFpEy9nXzrith1yrv8iIDGZ3RSAHHAhUAl2BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v6OuqC+VdMCz0HgmdRWVeOutRZT
+ZxBxCBgLRJFnEj6EwoFhO3zwkyjMim4TwWeotUfI0o4KOuHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/
hWuWfBpKLZl6Ae1UlZAFMO/7PSSoDgYUAAoGBAJCKGBBoxIUxqBk94JHhwZZbgvbP0DAOoHENQWxp/98lI7/
YOfYJOVMJS22aCnHDurofmo5rvNIkgXi7Rztbhu
+lko9rK6DgpmpUwBU0WZtf34aZ2IWNBwHaVhHvWAQf9/46u18dMa2YucK1Wi+Vc+M
+KldrvgXmhym6ErNlzhJyMAkGByqGSM44BAMDLwAwLAIUaaPKxaOHoYvwz709xXpsQueIq+UCFFa/
GpzoDOSokl1057NU/2hnsiW4
-----END CERTIFICATE-----
```

RSA


```
-----BEGIN CERTIFICATE-----
MIICMzCCAZygAwIBAgIGAXjwLj9CMA0GCSqGSIb3DQEBBQUAMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIDBBXYXNoaW5n
+sFcobrjvcAYm0PNRD8f4R1jAzvoLt2+qGeOTAyO1Httj6cmsYN3AP1hN5iYuppFiYsl2eNPa/
CD0Vg0BAfDFlV5rzjpA0j7TJabVh4kj7JvtD+xYMi6wEQA4x6SPONY4OeZ2+8o/
HS8nucpWDVdPRO6ciWUlMhjmDmwIDAQABMA0GCSqGSIb3DQEBBQUAA4GBAAy6sgTdRkTqELHBeWj69q60xHyUmsWqHAQ
TGGbYP0yP2qfMl0cCImzRI5W0gn8gogdervfeT7nH5ih0TWEy/QDWfkQ601L4erm4yh4YQq8vcqAPSkf04N
-----END CERTIFICATE-----
```


RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAIVWfPw/X82fMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yMjA3MDQx
NDMwMjhaGA8yMjAxMTIwODE0MzAyOFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAg29QEFriG+qFEjYW/v62nN70lMJY/Hevx5TtmU/VIYBPQa3HUGTBAbbI
2Tmy8UMpa8kZeaYeI3RAfiQWt0Ws7wUrBuO2Pdp518WDPaJUH7RWEuu1BDDkyZRW
NAMNPCn3ph7Od243IFcLGku7HVekel5poqRpSfojrMasjlf+CvixUeAJbmFoxUHK
kh5unzG2sZyO4wHXcJPQkRf5a8zSTPe9YZP1kXPPEv4p/jTSggaYPxXyS6QVaTlV
zLeLFZ0fesLPMeil3KYQtV7IKLQiEA2F6dxWnxNWQlyMHtdq6PucfEmVx17i/Xza
yNBRo0azY8WUNVkEXrRhp/pU8Nh3GQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQU9AO1aZk9RLXk2ZvRVoUxYvQy9uwwgY4GA1UdIwSBhjCBg4AU9AO1
aZk9RLXk2ZvRVoUxYvQy9uyhYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJAIVWfPw/X82fMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBADexluMRQRftqViahCnauEWGdMvLCBr8A+Yr
6hJq0guoxEk/lahxR137DnfMPuSbi1Rx5QKo7oBrWfG/zsgQUnF2IwHTzwD+i/2m
XCane6FiS5RpK3lGdILq8ZmlhQk+6iI8yoZLr0LCfTh+CLgIKHOknfR5lFzgzAiF
SI8/Q9mm+uvYtSTZECI6Zh57QZPoETAG/yl+9jiOy2lAelqa/k1i+Qo8gMfOc+Pm
dwY7o6fV+oucgRlsdey6VM45LeyILQqv0RXtVzjuowanzmCCFMjgqiO9oZAWu4Oh
+F3unijELo01vZJs8s2N3KGlo3/jtUFTX6RTKShZlAPLwBi5GMI=
-----END CERTIFICATE-----
```


亞太區域 (雅加達) — ap-southeast-3

DSA


```
-----BEGIN CERTIFICATE-----
```

```
MIIC8DCCArCgAwIBAgIGAXbVDEikMAkGByqGSM44BAMwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgMEFdhc2hpbmd0b24g
U4EddRIpUt9KnC7s5Of2EbdSPO9EAMMeP4C2USZpRV1AIlH7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdrmVClpJ+f6AR7ECLCT7up1/63xhv4O1fnxqimFQ8E
+4P208UewwI1VBNaFpEy9nXzrith1yrv8iIDGZ3RSAHHAhUAl2BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v6OuqC+VdMCz0HgmdRWVeOutRZT
+ZxBxCBgLRJFnEj6EwoFhO3zwkyjMim4TwWeotUfI0o4KOuHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/
hWuWfBpKLZl6Ae1UlZAFMO/7PSSoDgYUAAoGBAPjuiEx05N3JQ6cVwntJie67D8OuNo4jGRn
+crEtL7YO0jSVB9zGE1ga
+UgRPIaYETL293S8rTJTVgXAqdpBwfaHC6NUzre8U8iJ8FMNnlP9Gw1oUIlgQBjORyynVJexoB31TDZM
+/52g9O/bpq1QqNyKbeIgyBBlc1dAtr1QLnsMAkGByqGSM44BAMDLwAwLAIUK8E6RDIRtwK+9qnaTOBhvO/
njuQCFFocyT1OxK+UDR888oNsdgtif2Sf
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIICMzCCAZygAwIBAgIGAXbVDG2yMA0GCSqGSIb3DQEBBQUAMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIDBBXYXNoaW5n
Vbt0gQ1ebWcur2hSO7PnJifE4OPxQ7RgSAlc4/spJp1sDP+ZrS0LO1ZJfKhXf1R9S3AUwLnsc7b
+IuVXdY5LK9RKqu64nyXP5dx170zoL8loEyCSuRR2fs+04i2QsWBVP+KFNAn7P5L1EHRjkgTO8kjNKviwRV
+OkP9ab5wIDAQABMA0GCSqGSIb3DQEBBQUAA4GBAI4WUy6+DKh0JDSzQEZNyBgNlSoSuC2owtMxCwGB6nBfzzfcekWvs
+87w/g9lNwUnUt0ZHYyh2tuBG6hVJuUEwDJ/z3wDd6wQviLOTF3MITawt9P8siR1hXqLJNxpjRQFZrgHqi
-----END CERTIFICATE-----
```


RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAMtdyRcH51j9MA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yMjA0MDgx
MjM5MTZaGA8yMjAxMDkxMjEyMzkxNlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAvUsKCxoH6KXRYJLeYTWAQfaBQeCwhJaR56mfUeFHJE4g8aFjWkiN4uc1
TvOyYNnIZKTHWmzmulmdinWNbwP0GiROHb/i7ro0Hhvnptyycgt8ag8affiIbx5X
7ohdwSN2KJ6G0IKflIx7f2NEI0oAMM/9k+T1eVF+MVWzpZoiDp8frLNkqp8+RAgz
ScZsbRfwv3u/if5xJAvdg2nCkIWDMSHEVPoz0lJo7v0ZuDtWWsL1LHnL5ozvsKEk
+ZJyEi23r+U1hIT1NTBdp4yoigNQexedtwCSr7q36oOdDwvZpqYlkLi3uxZ4ta+a
01pzOSTwMLgQZSbKWQrpMvsIAPrxoQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQU1GgnGdNpbnL3lLF30Jomg7Ji9hYwgY4GA1UdIwSBhjCBg4AU1Ggn
GdNpbnL3lLF30Jomg7Ji9hahYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJAMtdyRcH51j9MBIGA1UdEwEB/wQIMAYBAf8C
```

```
AQAwDQYJKoZIhvcNAQELBQADggEBACVl00qQlatBKVeiWMrhpczsJroxDxlZTOba
6wTMZk7c3akb6XMOSZFbGaifkebPZqTHEhDlrClM2j9AIlYcCx6YCrTf4cuhn2mD
gcJN33143eOWSaeRY3ee4j+V9ne98y3kO2wLz95VrRgclPFR8po2iWGzGhwUi+FG
q8dXeCH3N0DZgQsSgQWwmdNQXZZej6RHLU/8In5trHKLY0ppnLBjn/UZQbeTyW5q
RJB3GaveXjfgFUWj2qOcDuRGaikdS+dYaLsi5z9cA3FolHzWxx9MOs8io8vKqQzV
XUrLTNWwuhZy88cOlqGPxnoRbw7TmifwPw/cunNrsjUUOgs6ZTk=
-----END CERTIFICATE-----
```


亞太區域 (馬來西亞) — AP-東南 -5

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7zCCAq4CCQC5X6U+vgoLEDAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yNDAxMDMxMjU3NTRaGA8y
MDUwMDEwMzEyNTc1NFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0
b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIg
U2VydmljZXMgTExDMIIBtjCCASsGByqGSM44BAEwggEeAoGBAIzEMPCIPFfOYCg4
BCjKGyl60wOfmwHPzSOXZ3Z2wS/LYNHUtHGwtVNePSTyCu/CuZF6gC9n/wBORtQp
+SSkn+weGc/BmUA1mp/vrN7v+aSCgKJoO+sgpa1PPOgNvUaMw6O5odsZWQCMSjkU
6RTo/PL2v/tMfiCocF4ghvyRC6hvAhUA0VoObKC2IXzXgVvRRupo4qHbcm8CgYAe
bbNuawh3rAxkFvUs9FPzW5E+xllG16Z//61PENKqonmk+zBiBdiIlS1F6ZqmTqkI
z5+qfSt1m3pb3j2W0NT7lEDFvy8Gr6Y2vohCHmL+T1u1Yy4PeqbgrFwcen7y7WoO
/KCV7Y9/ODQMMyuAzT3h5wJNweT7L5MUN8JYpZSi3QOBhAACgYBqaDuG2u6V91Qj
K2wEAElxaaRaNo/ewg/wWDmHYqoeHOR0HfuFCYgASE9f7ULqYtX1VURcgcjw9XN4
BDmPilXvfi04INPTnw4IxFJKDzzC0kVH7esVas982Po8v3megH32H9Rl87r7UG1c
ZEbkSkKVX6YKYg1PR3rfjXgdwVZv/zAJBgcqhkjOOAQDAzAAMC0CFFWeRe2fYW2i
6mMd26Wzbx87Y0DXAhUAoPCnF+5hGJw0jT9aL7QsgcfLi9Y=
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIICNjCCAZ+gAwIBAgIJAMuBl6rhZCJKMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yNDAxMDMx
MjU3NTRaGA8yMjAzMDYwOTEyNTc1NFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKB
gQDFuKydxZsordNH7bLwIluEGOkX7/CdLdpeqkDKEhQkFwzpRxaX4EAlkGh2/o7D
8qneC9cGQhqSG5WVVBrmZG7sfkFOM4m1AtY++kfv+MYto1VFgLk1xJbkpq1r4YeQ
```

```
Ul+ZsJYsZpyX/t+g8s7rW0OVcBsYx4L75bf34z38mwK8PQIDAQABMA0GCSqGSIb3
DQEBCwUAA4GBADD9C4pWL8RUvF1CJW8kExj35xmozlFlmrKs8Zpi8+Eg6q+W9dgd
xMdH95tgZtmVMDqlvVR+DK0iO1BNpqPjrqWkk2tTLivpS+sGzCE/jCl18Q28Rk71
/A3gLD7Rtbq5TKNvuFCHwYMjrTDHI6aBjIaAlDm4e2/j/OxVtHyZGTre
-----END CERTIFICATE-----
```


RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJANC3xtbPhQ2GMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yNDAxMDMx
MjU3NTRaGA8yMjAzMDYwOTEyNTc1NFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAt3aMy7Hsp4ySG3mlfi+pdTcZw6H6XNU1Y36fNdi4c+MzinQQbnqMPyt7
QLgU+XCWmcWsVo7GQF6n9N01Rh+UXXUZU4jcXlFocQPCWf9O+IIIIPXkd67kFMUV
HAxCELjfxHbC+I8e7dwOJhmdF4Bfi52Ty8zz0HdE8JDypPkTDlXuGvTgDyW7NP56
I/vlQaXLoYSbcQe5pv2a9gyBaaCM1QoeqwAHAeCNXb9Nuj9ZX3GHGJb3TuqAeKCD
5i9TscCB9XjY6Fx+zfSAobjBZwgLEtLOwJhbZnKmx4gJMaaNFipAjVT2FSS3+yev
eTYBoa1dvhk0ivQyQIPpHmihrmkWuwIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQBC
fdgyI8GjmCqiHAlh+L1bj0LdNql9zl7RXmOEzsuRdtdMumkxYXX88UtROy3fdi1i
VaEwHdAK8ThzRkesgHza/cXzqCMewaYxujSI6p6G7x99FFeGif1x0FJdj8AoeTL7
4h9bmS/6l4/NL7DJI9G7ovES/hoUA9v9TDhv+vauxXlgfrp0MPecprxBYlrc+DH2
adGCdKCp2lQ2YDKOD9TCEjYIli8XSoyevoWHUjfDYrCrCp8l4s/p7HOgYr8fJBAs
EuVy8211LVz1/X4EMBRNtNjXK9sklsxAOX14NDfBFSS0toxl3K6Tf9t/PviBl95d
hncyDAcFgDCK4w8LLlVW
-----END CERTIFICATE-----
```


亞太區域 (墨爾本) — ap-southeast-4

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7zCCAq
+gAwIBAgIGAXjWF7P2MAkGByqGSM44BAMwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgMEFdhc2hpbmd0b24gU3RhdGUxED
U4EddRIpUt9KnC7s5Of2EbdSPO9EAMMeP4C2USZpRV1AIlH7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdrmVClpJ+f6AR7ECLCT7up1/63xhv4O1fnxqimFQ8E
+4P208UewwI1VBNaFpEy9nXzrith1yrv8iIDGZ3RSAHHAhUAl2BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v6OuqC+VdMCz0HgmdRWVeOutRZT
+ZxBxCBgLRJFnEj6EwoFhO3zwkyjMim4TwWeotUfI0o4KOuHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/
```

```
hWuWfBpKLZl6Ae1UlZAFMO/7PSSoDgYQAAoGAPRXSsQP9E3dw8QXKlrgBgEVCprLHdK/bbrMas0XMu1EhOD
+q
+0PcTr8+iwbtoXlY5MCeatWIp1GrXQjVqsF8vQqxlEuRuYKbR3nq4mWwaeGlx9AG5EjQHRa3GQ44wWHOdof0M3NRI1MP
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIICMzCCAZygAwIBAgIGAXjSh40SMA0GCSqGSIb3DQEBBQUAMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIDBBXYXNoaW5n
+qWTGAbGsPeMX4hBMjAJUKys2NIRcRZaLM/BCew2FIPVjNtlaj6Gwn9ipU4Mlz3zIwAMWi1AvGMSreppt
+wV6MRtfOjh0Dvj/veJe88aEZJMozNgkJFRS
+WFWsckQeL56tf6kY6QTlNo8V/0CsQIDAQABMA0GCSqGSIb3DQEBBQUAA4GBAF7vpPghH0FRo5gu49EArRNPrIvW1egM
wcgkqIwwuXYj+1rhlL+/
iMpQWjdVGEqIZSeXn5fLmdx50eegFCwND837r9e8XYTiQS143Sxt9+Yi6BZ7U7YD8kK9NBWoJxFqUeHdpRCs0O7COjT3
-----END CERTIFICATE-----
```


RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAN4GTQ64zVs8MA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yMjA3MTMx
MzMzMDBaGA8yMjAxMTIxNzEzMzMwMFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEA2BYgeCr+Rk/jIAEDOHS7wJql62vc83QEwjuzk0qOFEReIZzlN1fBRNXK
g0T178Kd3gLYcE59wEFbTe/X5yOA1Lo95x1anSAo7R+Cisf9C2HQuJp+gVb+zx71
lniPF7gHziGpm0M8DdAU/IW+wkZwGbP4z7Hq9+bJ0P2ltvPJ5yxSgkFuDsI9VBHa
CLoprHsCHh2VdP8KcMgQQMmHe1NmBpyTk0ul/aLmQkCQEX6ZIRGOeq228fwlh/t+
Ho+jv87duihVKic6MrL32SlD+maX0LSDUydWdaOLLTGkh7oV7+bFuH6msrXUu+Ur
ZEP1r/MidCWMhfgrFzeTBzOHA97qxQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQUcHMd1cHqzmsQ5hpUK3EMLhHdsi4wgY4GA1UdIwSBhjCBg4AUcHMd
1cHqzmsQ5hpUK3EMLhHdsi6hYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJAN4GTQ64zVs8MBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAI4PFyVN+7EGSObioiPnv0LL0f70SSzUZJ8p
XO90d4rWea7jIbgZ2AKb+ErynkU9xVg7XQQ5k6KDWgp/4jYFL2dqnt/YAY4PS0un
RSrYElawxLT0BcLn4rcSDC79vQe1xGC5//wDdV6b399COAHRAK6axWYy5w32u9PL
uw0cIp3Ch8JoNwcgTHKRRGzePmBeR4PNqhHTArG4/dJk6/aUO4OpX0WzI6L67CGY
6Nex3dau+gkLCK93dTEkrXtyXHu4wB0J9zd1w+iQ0SEa9eKc78/NjEsF/FZdGrWC
t57lIMOOXJhQ1kRgSwNeZdQWV1dRakvO6sfcvVYkfj1wAvZvvAw=
-----END CERTIFICATE-----
```

亞太區域 (孟買) — ap-south-1

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUDLA+x6tTAP3LRTr0z6nOxfsozdMwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE0MTMwMVoXDTI5MDQyODE0MTMwMVowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUDLA+x6tT
AP3LRTr0z6nOxfsozdMwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQAZ7rYKoAwwiiH1M5GJbrT/BEk3OO2VrEPw8ZxgpqQ/EKlzMlOs/0Cyrmp7
UYyUgYFQe5nq37Z94rOUSeMgv/WRxaMwrLlLqD78cuF9DSkXaZIX/kECtVaUnjk8
BZx0QhoIHOpQocJUSlm/dLeMuE0+0A3HNR6JVktGsUdv9ulmKw==
```

```
-----END CERTIFICATE-----
```

RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJAPRYyD8TtmC0MA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNjAzMDcx
MDQ1MDFaGA8yMTk1MDgxMTEwNDUwMVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEA0LSS5I/eCT2PM0+qusorBx67QL26BIWQHd/yF6ARtHBb/1DdFLRqE5Dj
07Xw7eENC+T79mOxOAbeWg91KaODOzw6i9I/2/HpK0+NDEdD6sPKDA1d45jRra+v
CqAjI+nV9Vw91wv7HjMk3RcjWGziM8/hw+3YNIutt7aQzZRwIWlBpcqx3/AFd8Eu
2UsRMSHgkGUW6UzUF+h/U8218XfrauKNGmNKDYUhtmyBrHT+k6J0hQ4pN7fe6h+Z
w9RVHm24BGhlLxLHLmsOIxvbrF277uX9Dxu1HfKfu5D2kimTY7xSZDNLR2dt+kNY
/+iWdIeEFpPT0PLSILt52wP6stF+3QIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQBI
E6w+WWC2gCfoJO6c9HMyGLMFEpqZmz1n5IcQt1h9iyO7Vkm1wkJiZsMhXpk73zXf
TPxuXEacTX3SOEa07OIMCFwkusO5f6leOyFTynHCzBgZ3U0UkRVZA3WcpbNB6Dwy
h7ysVlqyT9WZd7EOYm5j5oue2G2xdei+6etgn5UjyWm6liZGrcOF6WPTdmzqa6WG
ApEqanpkQd/HM+hUYex/ZS6zEhd4CCDLgYkIjlrFbFb3pJ1OVLztIfSN5J4Oolpu
JVCfIq5u1NkpzL7ys/Ub8eYipbzI6P+yxXiUSuF0v9b98ymczMYjrSQXIf1e8In3
OP2CclCHoZ8XDQcvvKAh
-----END CERTIFICATE-----
```


亞太區域 (大阪) — ap-northeast-3

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
```

```
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUHTRhxHhBZFOGvTFKxHoy9+f5Hl8wDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE2NTQwN1oXDTI5MDQyODE2NTQwN1owXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUHTRhxHhB
ZFOGvTFKxHoy9+f5Hl8wEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQAUZx7DcYbhWNTD4BNGhr5beruT2OUoGHH9J73UKxwdqeb9bH1LIWhIZO0X
/1mjn3bWBgCwfoS8gjZwsVB6fZbNBRy8urdBZJ87xF/4JPBjt7S9oGx/zthDUYrC
yK0Y0v4G0PgiS81CvYLg09LpmYhLSJbXENlkC04v5yxdKxZxyg==
-----END CERTIFICATE-----
```


RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJAMn1yPk22ditMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNzA3MTkx
MTEyNThaGA8yMTk2MTIyMjExMTI1OFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEArznEYef8IjhrJoazI0QGZkmlmHm/4rEbyQbMNifxjsDE8YWtHNwaM91z
zmyK6Sk/tKlWxcnl3g31iq305ziyFPEewe5Qbwf1iz2cMsvfNBcTh/E6u+mBPH3J
gvGanqUJt6c4IbipdEouIjjnynyVWd4D6erLl/ENijeR1OxVpaqSW5SBK7jms49E
pw3wtbchEl3qsE42Ip4IYmWxqjgaxB7vps91n4kfyzAjUmklcqTfMfPCkzmJCRgp
Vh1C79vRQhmriVKD6BXwfZ8tG3a7mijeDn7kTsQzgO07Z2SAE63PIO48JK8HcObH
tXORUQ/XF1jzi/SIaUJZT7kq3kWl8wIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQBj
```

```
ThtO9dLvU2QmKuXAhxXjsIdlQgGG3ZGh/Vke4If1ymgLx95v2Vj9Moxk+gJuUSRL
BzFte3TT6b3jPolbECgmAorjj8NxjC17N8QAAI1d0S0gI8kqkG7V8iRyPIFekv+M
pcai1+cIv5IV5qAz8QOMGYfGdYkcoBjsgiyvMJu/2N2UbZJNGWvcEGkdjGJUYYOO
NaspCAFm+6HA/K7BD9zXB1IKsprLgqhiIUgEaW3UFEbThJT+z8UfHG9fQjzzfN/J
nT6vuY/0RRu1xAZPyh2gr5okN/s6rnmh2zmBHU1n8cbCc64MVfXe2g3EZ9Glq/9n
izPrI09hMypJDP04ugQc
-----END CERTIFICATE-----
```


亞太區域 (首爾) — ap-northeast-2

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUbBSn2UIO6vYk4iNWV0RPxJJtHlgwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTEzMzg0NloXDTI5MDQyODEzMzg0NlowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
```

```
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUbBSn2UIO
6vYk4iNWV0RPxJJtHlgwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQAmjTjalG8MGLqWTC2uYqEM8nzI3px1eo0ArvFRsyqQ3fgmWcQpxExqUqRy
l3+2134Kv8dFab04Gut5wlfRtc2OwPKKicmv/IXGN+9bKFnQFjTqif08NIzrDZch
aFT/uvxrIiM+oN2YsHq66GUhO2+xVRXDXVxM/VObFgPERbJpyA==
-----END CERTIFICATE-----
```


RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJANuCgCcHtOJhMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNTA5MTQx
NTU3NDRaGA8yMTk1MDIxNzE1NTc0NFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEA66iNv6pJPmGM20W8HbVYJSlKcAg2vUGx8xeAbzZIQdpGfkabVcUHGB6m
Gy59VXDMDlrJckDDk6dxUOhmcX9z785TtVZURq1fua9QosdbTzX4kAgHGdp4xQEs
mO6QZqg5qKjBP6xr3+PshfQ1rB8Bmwg0gXEm22CC7o77+7N7Mu2sWzWbiUR7vil4
9FjWS8XmMNwFTlShp4l1TDTevDWW/uYmC30RThM9S4QPvTZ0rAS18hHVam8BCTxa
LHaVCH/Yy52rsz0hM/FlghnSnK105ZKj+b+KIp3adBL8OMCjgc/Pxi0+j3HQLdYE
32+FaXWU84D2iP2gDT28evnstzuYTQIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQC1
mA4q+12pxy7By6g3nBk1s34PmWikNRJBwOqhF8ucGRv8aiNhRRye9lokcXomwo8r
KHbbqvtK85l0xUZp/Cx4sm4aTgcMvfJP29jGLclDzeqADIvkWEJ4+xncxSYVlS9x
+78TvF/+8h9U2LnSl64PXaKdxHy2IsHIVRN4GtoaP2Xhpa1S0M328Jykq/571nfN
1WRD1c/fQf1edgzRjhQ4whcAhv7WRRF+qTbfQJ/vDxy8lkiOsvU9XzUaZ0fZSfXX
wXxZamQbONvFcxVHY/0PSiM8nQoUmkkBQuKleDwRWvkoJKYKyr3jvXK7HIWtMrO4
jmXe0aMy3thyK6g5sJVg
-----END CERTIFICATE-----
```


亞太區域 (新加坡) — ap-southeast-1

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
```

```
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUSqP6ih+++5KF07NXngrWf26mhSUwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE0MzAxNFoXDTI5MDQyODE0MzAxNFowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUSqP6ih++
+5KF07NXngrWf26mhSUwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQAw13BxW11U/JL58j//Fmk7qqtrZTqXmaz1qm2WlIpJpW750MOcP4ux1uPy
eM0RdVZ4jHSMv5gtLAv/PjExBfw9n6vNCk+5GZG4Xec5DoapBZHXmfMo93sjxBFP
4x9rWn0GuwAVO9ukjYPevq2Rerilrq5VvppHtbATVNY2qecXDA==
-----END CERTIFICATE-----
```


RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAJVMGw5SHkcvMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNTEwMjkw
```

```
ODU3MTlaGA8yMTk1MDQwMzA4NTcxOVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAlaSSLfBl7OgmikjLReHuNhVuvM20dCsVzptUyRbut+KmIEEc24wd/xVy
2RMIrydGedkW4tUjkUyOyfET5OAyT43jTzDPHZTkRSVkYjBdcYbe9o/0Q4P7IVS3
XlvwrUu0qo9nSID0mxMnOoF1l8KAqnn10tQ0W+lNSTkasW7QVzcb+3okPEVhPAOq
MnlY3vkMQGI8zX4iOKbEcSVIzf6wuIffXMGHVC/JjwihJ2USQ8fq6oy686g54P4w
ROg415kLYcodjqThmGJPNUpAZ7MOc5Z4pymFuCHgNAZNvjhZDA842Ojecqm62zcm
Tzh/pNMNeGCRYq2EQX0aQtYOIj7bOQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQU6SSB+3qALorPMVNjToM1Bj3oJMswgY4GA1UdIwSBhjCBg4AU6SSB
+3qALorPMVNjToM1Bj3oJMuhYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJAJVMGw5SHkcvMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAF/0dWqkIEZKg5rca8o0P0VS+tolJJE/FRZO
atHOeaQbWzyac6NEwjYeeV2kY63skJ+QPuYbSuIBLM8p/uTRIvYM4LZYImLGUvoO
IdtJ8mAzq8CZ3ipdMs1hRqF5GRp8lg4w2QpX+PfhnW47iIOBiqSAUkIr3Y3BDaDn
EjeXF6qS4iPIvBaQQ0cvdddNh/pE33/ceghbkZNTYkrwMyBkQlRTTVKXFN7pCRUV
+L9FuQ9y8mP0BYZa5e1sdkwebydU+eqVzsil98ntkhpjvRkaJ5+Drs8TjGaJWlRw
5WuOr8unKj7YxdL1bv7//RtVYVVi296ldoRUYv4SCvJF11z0OdQ=
-----END CERTIFICATE-----
```


亞太區域 (雪梨) — ap-southeast-2

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
```

```
-----END CERTIFICATE-----
```

RSA


```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUFxWyAdk4oiXIOC9PxcgjYYh71mwwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE1MjE0M1oXDTI5MDQyODE1MjE0M1owXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUFxWyAdk4
oiXIOC9PxcgjYYh71mwwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQByjeQe6lr7fiIhoGdjBXYzDfkX0lGGvMIhRh57G1bbceQfaYdZd7Ptc0jl
bpycKGaTvhUdkpMOiV2Hi9dOOYawkdhyJDstmDNKu6P9+b6Kak8He5z3NU1tUR2Y
uTwcz7Ye8Nldx//ws3raErfTI7D6s9m63OX8cAJ/f8bNgikwpw==
-----END CERTIFICATE-----
```


RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAL2bOgb+dq9rMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNTEwMjkw
OTAwNTdaGA8yMTk1MDQwMzA5MDA1N1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAmRcyLWraysQS8yDC1b5Abs3TUaJabjqWu7d5gHik5Icd6dKl8EYpQSeS
vz6pLhkgO4xBbCRGlgE8LS/OijcZ5HwdrxBiKbicR1YvIPaIyEQQvF5sX6UWkGYw
Ma5IRGj4YbRmJkBybw+AAV9Icb5LJNOMWPi34OWM+2tMh+8L234v/JA6ogpdPuDr
sM6YFHMZ0NWo58MQ0FnEj2D7H58Ti//vFPl0TaaPWaAIRF85zBiJtKcFJ6vPidqK
f2/SDuAvZmyHC8ZBHg1moX9bR5FsU3QazfbW+c+JzAQWHj2AaQrGSCITxCMlS9sJ
l51DeoZBjnx8cnRe+HCaC4YoRBiqIQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQU/wHIo+r5U31VIsPoWoRVsNXGxowwgY4GA1UdIwSBhjCBg4AU/wHI
o+r5U31VIsPoWoRVsNXGxoyhYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJAL2bOgb+dq9rMBIGA1UdEwEB/wQIMAYBAf8C
```

```
AQAwDQYJKoZIhvcNAQELBQADggEBACobLvj8IxlQyORTz/9q7/VJL509/p4HAeve
92riHp6+Moi0/dSEYPeFTgdWB9W3YCNc34Ss9TJq2D7t/zLGGlbI4wYXU6VJjL0S
hCjWeIyBXUZOZKFCb0DSJeUElsTRSXSFuVrZ9EAwjLvHni3BaC9Ve34iP71ifr75
8Tpk6PEj0+JwiijFH8E4GhcV5chB0/iooU6ioQqJrMwFYnwo1cVZJD5v6D0mu9bS
TMIJLJKv4QQQqPsNdjiB7G9bfkB6trP8fUVYLHLsVlIy5lGx+tgwFEYkG1N8IOO/
2LCawwaWm8FYAFd3IZl04RImNs/IMG7VmH1bf4swHOBHgCN1uYo=
-----END CERTIFICATE-----
```


亞太區域 (東京) — ap-northeast-1

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIULgwDh7TiDrPPBJwscqDwiBHkEFQwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTEyMjMxMFoXDTI5MDQyODEyMjMxMFowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
```

```
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IULgwDh7Ti
DrPPBJwscqDwiBHkEFQwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQBtjAglBde1t4F9EHCZOj4qnY6Gigy07Ou54i+lR77MhbpzE8V28Li9l+YT
QMIn6SzJqU3/fIycIro1OVY1lHmaKYgPGSEZxBenSBHfzwDLRmC9oRp4QMe0BjOC
gepj1lUoiN7OA6PtA+ycNlsP0oJvdBjhvayLiuM3tUfLTrgHbw==
-----END CERTIFICATE-----
```


RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAL9KIB7Fgvg/MA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNTA4MTQw
OTAwMjVaGA8yMTk1MDExNzA5MDAyNVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAz0djWUcmRW85C5CiCKPFiTIvj6y2OuopFxNE5d3Wtab10bm06vnXVKXu
tz3AndG+Dg0zIL0gMlU+QmrSR0PH2PfV9iejfLak9iwdm1WbwRrCEAj5VxPe0Q+I
KeznOtxzqQ5Wo5NLE9bA61sziUAFNVsTFUzphEwRohcekYyd3bBC4v/RuAjCXHVx
40z6AIksnAOGN2VABMlTeMNvPItKOCIeRLlllSqXX1gbtL1gxSW40JWdF3WPB68E
e+/1U3F7OEr7XqmNODOL6yh92QqZ8fHjG+afOL9Y2Hc4g+P1nk4w4iohQOPABqzb
MPjK7B2Rze0f9OEc51GBQu13kxkWWQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQU5DS5IFdU/QwYbikgtWvkU3fDwRgwgY4GA1UdIwSBhjCBg4AU5DS5
IFdU/QwYbikgtWvkU3fDwRihYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJAL9KIB7Fgvg/MBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAG/N7ua8IE9IMyno0n5T57erBvLTOQ79fIJN
Mf+mKRM7qRRsdg/eumFft0rLOKo54pJ+Kim2cngCWNhkzctRHBV567AJNt4+ZDG5
hDgV0IxWO1+eaLE4qzqWP/9VrO+p3reuumgFZLVpvVpwXBBeBFUf2drUR14aWfI2
L/6VGINXYs7uP8v/2VBS7r6XZRnPBUy/R4hv5efYXnjwA9gq8+a3stC2ur8m5ySl
faKSwE4H320yAyaZWH4gpwUdbUlYgPHtm/ohRtiWPrN7KEG5Wq/REzMIjZCnxOfS
6KR6PNjlhxBsImQhmBvz6j5PLQxOxBZIpDoiK278e/1Wqm9LrBc=
-----END CERTIFICATE-----
```

加拿大 (中部) — ca-central-1

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUIrLgixJJB5C4G8z6pZ5rB0JU2aQwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE1MzU0M1oXDTI5MDQyODE1MzU0M1owXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUIrLgixJJ
B5C4G8z6pZ5rB0JU2aQwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQBHiQJmzyFAaSYs8SpiRijIDZW2RIo7qBKb/pI3rqK6yOWDlPuMr6yNI81D
IrKGGftg4Z+2KETYU4x76HSf0s//vfH3QA57qFaAwddhKYy4BhteFQl/Wex3xTlX
LiwI07kwJvJy3mS6UfQ4HcvZy219tY+0iyOWrz/jVxwq7TOkCw==
```

```
-----END CERTIFICATE-----
```

RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJAJNKhJhaJOuMMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNjA3Mjkx
MTM3MTdaGA8yMTk2MDEwMjExMzcxN1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAhDUh6j1ACSt057nSxAcwMaGr8Ez87VA2RW2HyY8l9XoHndnxmP50Cqld
+26AJtltlqHpI1YdtnZ6OrVgVhXcVtbvte0lZ3ldEzC3PMvmISBhHs6A3SWHA9ln
InHbToLX/SWqBHLOX78HkPRaG2k0COHpRy+fG9gvz8HCiQaXCbWNFDHZev9OToNI
xhXBVzIa3AgUnGMalCYZuh5AfVRCEeALG60kxMMC8IoAN7+HG+pMdqAhJxGUcMO0
LBvmTGGeWhi04MUZWfOkwn9JjQZuyLg6B1OD4Y6s0LB2P1MovmSJKGY4JcF8Qu3z
xxUbl7Bh9pvzFR5gJN1pjM2n3gJEPwIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQAJ
UNKM+gIIHNk0G0tzv6vZBT+o/vt+tIp8lEoZwaPQh1121iw/I7ZvhMLAigx7eyvf
IxUt9/nf8pxWaeGzi98RbSmbap+uxYRynqe1p5rifTamOsguuPrhVpl12OgRWLcT
rjg/K60UMXRsmg2w/cxV45pUBcyVb5h6Op5uEVAVq+CVns13ExiQL6kk3guG4+Yq
LvP1p4DZfeC33a2Rfre2IHLsJH5D4SdWcYqBsfTpf3FQThH0l0KoacGrXtsedsxs
9aRd7OzuSEJ+mBxmzxSjSwM84Ooh78DjkdpQgv967p3d+8NiSLt3/n7MgnUy6WwB
KtDujDnB+ttEHwRRngX7
-----END CERTIFICATE-----
```


加拿大西部 (卡加利) — ca-west-1

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7zCCAq
+gAwIBAgIGAYPouptUMAkGByqGSM44BAMwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgMEFdhc2hpbmd0b24gU3RhdGUxED
U4EddRIpUt9KnC7s5Of2EbdSPO9EAMMeP4C2USZpRV1AIlH7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdrmVClpJ+f6AR7ECLCT7up1/63xhv4O1fnxqimFQ8E
+4P208UewwI1VBNaFpEy9nXzrith1yrv8iIDGZ3RSAHHAhUAl2BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v6OuqC+VdMCz0HgmdRWVeOutRZT
+ZxBxCBgLRJFnEj6EwoFhO3zwkyjMim4TwWeotUfI0o4KOuHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/hWuWfBpKLZl6Ae1UlZAFMO/7PSSoDgYQAAoGAMITzTJUa6cBsIfdHN69zW/
aHjUB4r1ZfKb1FMhIp9EZtEf5n+06oXjUG2+dKRSlFQeEK333ehNZsPd6uqey6TYKtHpFb5XRLS8BpqB
+7gnbAd0CBZM5o4NWesSQ1GLnTdQcGZkYG/
QESkbadoCXQTifCujJE682hTDLIVt1d4ewwCQYHKoZIzjgEAwMvADAsAhRJc4gRS/HWTkCR2MESaQEe/
jOMNQIUNoTwLvuPrmGPupPlGiHe0veZiO8=
```

```
-----END CERTIFICATE-----
```

RSA

```
-----BEGIN CERTIFICATE-----
MIICMzCCAZygAwIBAgIGAYPou9weMA0GCSqGSIb3DQEBBQUAMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIDBBXYXNoaW5n
v4XBVH13ZCMgqlRHMqV8AWI5iO6gFn2A9sN3AZXTMqwtZeiDdebq3k6Wt7ieYvpXTg0qvgsjQIovRZWaBDBJy9x8C2hW
+w9lMQjFHkJ7Jy/
PHCJ69EzebQIDAQABMA0GCSqGSIb3DQEBBQUAA4GBAGe9Snkz1A6rHBH6/5kDtYvtPYwhx2sXNxztbhkXErFk40Nw5l4
gvDVtWG7qyb6fAqgoisyAbk8K9LzxSim2S1nmT9vD84B/t/VvwQBylc
+ej8kRxMH7fquZLp7IXfmtBzyUqu6Dpbne+chG2
-----END CERTIFICATE-----
```

RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJALyTn5IHrIZjMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yMzEyMDcx
NTM3MDFaGA8yMjAzMDUxMzE1MzcwMVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEA1GP5os424BJmGPCKOSgOclP7lzUiB85du03M4hfjzSOszsBpmBGFDLz1
owYHtIxlq3+Vi1Lt5Qlx3id/ov1QyaBPFWXVeklHVXy9vieCcI3TdjGjTl1W/8MM
m3X26QPcsnHM/Kk2wJ7s186MrqmdSsp3SCPpxv4vEG2Q9yR2bXY41hpc2rWlW8qU
D0JGXluvmmAdFnto20l1XWZ6xFen1h60DRugek/ufCbN+ljKy0xLqPoavH0Ybjsb
UpsAsBs7phaoN+X/5hIERfbp5LfVnqq54pNG5KNu4KynfW9+kA/WS4cJ6FTTN5t+
y0P1HvcL+BL2RuDy6T2bB21xw5WqtQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQURTvu/Dd4zDnmS5G5CfVLnmUBNOswgY4GA1UdIwSBhjCBg4AURTvu
/Dd4zDnmS5G5CfVLnmUBNOuhYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJALyTn5IHrIZjMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAFt523A3Aug6/F8xxyITgA8gkU0btFhlXNSP
U4U2OQ9n0tWI9WqnKNWH3KBxwY5EPitU6b3LM4xc9lDWpz7h2Pto+WhxP9LVKe6f
r8r7teTLCVZ7cfYZHzHg+flZjVpAgzE5BVfrRlj3QKpv0hYT3J1wMtI++Vorq5Nf
aPjzedehJLhmZVALwnfqfLrgv6/gmraP9Vmoa8U4D6AljNiQGYaLwyoPoRm3bUs2
v1Mh9GkEQ1b9+1pFXcqgzJJTGRuiPCyPbECI79FAnx5JM/CkGJV8HlOmjIW1qkK1
Y2qT7wzErrKLJyB53Pw15BdIM1onbDAQreZb0yZQLdoEl/tx7Uk=
-----END CERTIFICATE-----
```

中國 (北京) — cn-north-1

DSA


```
-----BEGIN CERTIFICATE-----
MIIDNjCCAh4CCQD3yZ1w1AVkTzANBgkqhkiG9w0BAQsFADBcMQswCQYDVQQGEwJV
UzEZMBcGA1UECBMQV2FzaGluZ3RvbiBTdGF0ZTEQMA4GA1UEBxMHU2VhdHRsZTEg
MB4GA1UEChMXQW1hem9uIFdlYiBTZXJ2aWNlcyBMTEMwIBcNMTUwNTEzMDk1OTE1
WhgPMjE5NDEwMTYwOTU5MTVaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNo
aW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24g
V2ViIFNlcnZpY2VzIExMQzCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEB
AMWk9vyppSmDU3AxZ2Cy2bvKeK3F1UqNpMuyeriizi+NTsZ8tQqtNloaQcqhto/l
gsw9+QSnEJeYWnmivJWOBdn9CyDpN7cpHVmeGgNJL2fvImWyWe2f2Kq/BL9l7N7C
P2ZT52/sH9orlck1n2zO8xPi7MItgPHQwu3OxsGQsAdWucdxjHGtdchulpo1uJ31
jsTAPKZ3p1/sxPXBBAgBMatPHhRBqhwHO/Twm4J3GmTLWN7oVDds4W3bPKQfnw3r
vtBj/SM4/IgQ3xJslFcl90TZbQbgxIi88R/gWTbs7GsyT2PzstU30yLdJhKfdZKz
/aIzraHvoDTWFaOdy0+OOaECAwEAATANBgkqhkiG9w0BAQsFAAOCAQEAdSzN2+0E
V1BfR3DPWJHWRf1b7zl+1X/ZseW2hYE5r6YxrLv+1VPf/L5I6kB7GEtqhZUqteY7
zAceoLrVu/7OynRyfQetJVGichaaxLNM3lcr6kcxOowb+WQQ84cwrB3keykH4gRX
KHB2rlWSxta+2panSEO1JX2q5jhcFP90rDOtZjlpYv57N/Z9iQ+dvQPJnChdq3BK
5pZlnIDnVVxqRike7BFy8tKyPj7HzoPEF5mh9Kfnn1YoSVu+61lMVv/qRjnyKfS9
c96nE98sYFj0ZVBzXw8Sq4Gh8FiVmFHbQp1peGC19idOUqxPxWsasWxQXO0azYsP
9RyWLHKxH1dMuA==
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIIDCzCCAnSgAwIBAgIJALSOMbOoU2svMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0yMzA3MDQw
ODM1MzlaFw0yODA3MDIwODM1MzlaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQzCBnzANBgkqhkiG9w0BAQEFAAOBjQAwgYkCgYEA
uhhUNlqAZdcWWB/OSDVDGk3OA99EFzOn/mJlmciQ/Xwu2dFJWmSCqEAE6gjufCjQ
q3voxAhC2CF+elKtJW/C0Sz/LYo60PUqd6iXF4h+upB9HkOOGuWHXsHBTsvgkgGA
1CGgel4U0Cdq+23eANr8N8m28UzljjSnTlrYCHtzN4sCAwEAAaOB1DCB0TALBgNV
HQ8EBAMCB4AwHQYDVR0OBBYEFBkZu3wT27NnYgrfH+xJz4HJaNJoMIGOBgNVHSME
gYYwgYOAFBkZu3wT27NnYgrfH+xJz4HJaNJooWCkXjBcMQswCQYDVQQGEwJVUzEZ
MBcGA1UECBMQV2FzaGluZ3RvbiBTdGF0ZTEQMA4GA1UEBxMHU2VhdHRsZTEgMB4G
A1UEChMXQW1hem9uIFdlYiBTZXJ2aWNlcyBMTEOCCQC0jjGzqFNrLzASBgNVHRMB
Af8ECDAGAQH/AgEAMA0GCSqGSIb3DQEBCwUAA4GBAECji43p+oPkYqmzll7e8Hgb
oADS0ph+YUz5P/bUCm61wFjlxaTfwKcuTR3ytj7bFLoW5Bm7Sa+TCl3lOGb2taon
```

```
2h+9NirRK6JYk87LMNvbS40HGPFumJL2NzEsGUeK+MRiWu+Oh5/lJGii3qw4YByx
SUDlRyNy1jJFstEZjOhs
-----END CERTIFICATE-----
```


RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJAOtrM5XLDSjCMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNTA4MTQx
MDAxNDJaGA8yMTk1MDExNzEwMDE0MlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAvVBz+WQNdPiM9S+aUULOQEriTmNDUrjLWLr7SfaOJScBzis5D5ju0jh1
+qJdkbuGKtFX5OTWTm8pWhInX+hIOoS3exC4BaANoa1A3o6quoG+Rsv72qQf8LLH
sgEi6+LMlCN9TwnRKOToEabmDKorss4zFl7VSsbQJwcBSfOcIwbdRRaW9Ab6uJHu
79L+mBR3Ea+G7vSDrVIA8goAPkae6jY9WGw9KxsOrcvNdQoEkqRVtHo4bs9fMRHU
Etphj2gh4ObXlFN92VtvzD6QBs3CcoFWgyWGvzg+dNG5VCbsiiuRdmii3kcijZ3H
Nv1wCcZoEAqH72etVhsuvNRC/xAP8wIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQA8
ezx5LRjzUU9EYWYhyYIEShFlP1qDHs7F4L46/5lc4pL8FPoQm5CZuAF31DJhYi/b
fcV7i3n++/ymQbCLC6kAg8DUB7NrcROll5ag8d/JXGzcTCnlDXLXx1905fPNa+jI
0q5quTmdmiSi0taeaKZmyUdhrB+a7ohWdSdlokEIOtbH1P+g5yll3bI2leYE6Tm8
LKbyfK/532xJPqO9abx4Ddn89ZEC6vvWVNDgTsxERg992Wi+/xoSw3XxkgAryIv1
zQ4dQ6irFmXwCWJqc6kHg/M5W+z60S/94+wGTXmp+19U6Rkq5jVMLh16XJXrXwHe
4KcgIS/aQGVgjM6wivVA
-----END CERTIFICATE-----
```


中國 (寧夏) — cn-northwest-1

DSA


```
-----BEGIN CERTIFICATE-----
MIIDNjCCAh4CCQD3yZ1w1AVkTzANBgkqhkiG9w0BAQsFADBcMQswCQYDVQQGEwJV
UzEZMBcGA1UECBMQV2FzaGluZ3RvbiBTdGF0ZTEQMA4GA1UEBxMHU2VhdHRsZTEg
MB4GA1UEChMXQW1hem9uIFdlYiBTZXJ2aWNlcyBMTEMwIBcNMTUwNTEzMDk1OTE1
WhgPMjE5NDEwMTYwOTU5MTVaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNo
aW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24g
V2ViIFNlcnZpY2VzIExMQzCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEB
AMWk9vyppSmDU3AxZ2Cy2bvKeK3F1UqNpMuyeriizi+NTsZ8tQqtNloaQcqhto/l
gsw9+QSnEJeYWnmivJWOBdn9CyDpN7cpHVmeGgNJL2fvImWyWe2f2Kq/BL9l7N7C
P2ZT52/sH9orlck1n2zO8xPi7MItgPHQwu3OxsGQsAdWucdxjHGtdchulpo1uJ31
jsTAPKZ3p1/sxPXBBAgBMatPHhRBqhwHO/Twm4J3GmTLWN7oVDds4W3bPKQfnw3r
```

```
vtBj/SM4/IgQ3xJslFcl90TZbQbgxIi88R/gWTbs7GsyT2PzstU30yLdJhKfdZKz
/aIzraHvoDTWFaOdy0+OOaECAwEAATANBgkqhkiG9w0BAQsFAAOCAQEAdSzN2+0E
V1BfR3DPWJHWRf1b7zl+1X/ZseW2hYE5r6YxrLv+1VPf/L5I6kB7GEtqhZUqteY7
zAceoLrVu/7OynRyfQetJVGichaaxLNM3lcr6kcxOowb+WQQ84cwrB3keykH4gRX
KHB2rlWSxta+2panSEO1JX2q5jhcFP90rDOtZjlpYv57N/Z9iQ+dvQPJnChdq3BK
5pZlnIDnVVxqRike7BFy8tKyPj7HzoPEF5mh9Kfnn1YoSVu+61lMVv/qRjnyKfS9
c96nE98sYFj0ZVBzXw8Sq4Gh8FiVmFHbQp1peGC19idOUqxPxWsasWxQXO0azYsP
9RyWLHKxH1dMuA==
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIIDCzCCAnSgAwIBAgIJALSOMbOoU2svMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0yMzA3MDQw
ODM1MzlaFw0yODA3MDIwODM1MzlaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQzCBnzANBgkqhkiG9w0BAQEFAAOBjQAwgYkCgYEA
uhhUNlqAZdcWWB/OSDVDGk3OA99EFzOn/mJlmciQ/Xwu2dFJWmSCqEAE6gjufCjQ
q3voxAhC2CF+elKtJW/C0Sz/LYo60PUqd6iXF4h+upB9HkOOGuWHXsHBTsvgkgGA
1CGgel4U0Cdq+23eANr8N8m28UzljjSnTlrYCHtzN4sCAwEAAaOB1DCB0TALBgNV
HQ8EBAMCB4AwHQYDVR0OBBYEFBkZu3wT27NnYgrfH+xJz4HJaNJoMIGOBgNVHSME
gYYwgYOAFBkZu3wT27NnYgrfH+xJz4HJaNJooWCkXjBcMQswCQYDVQQGEwJVUzEZ
MBcGA1UECBMQV2FzaGluZ3RvbiBTdGF0ZTEQMA4GA1UEBxMHU2VhdHRsZTEgMB4G
A1UEChMXQW1hem9uIFdlYiBTZXJ2aWNlcyBMTEOCCQC0jjGzqFNrLzASBgNVHRMB
Af8ECDAGAQH/AgEAMA0GCSqGSIb3DQEBCwUAA4GBAECji43p+oPkYqmzll7e8Hgb
oADS0ph+YUz5P/bUCm61wFjlxaTfwKcuTR3ytj7bFLoW5Bm7Sa+TCl3lOGb2taon
2h+9NirRK6JYk87LMNvbS40HGPFumJL2NzEsGUeK+MRiWu+Oh5/lJGii3qw4YByx
SUDlRyNy1jJFstEZjOhs
-----END CERTIFICATE-----
```


RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJAPu4ssY3BlzcMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNTEyMDMy
MTI5MzJaGA8yMTk1MDUwODIxMjkzMlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAsOiGi4A6+YTLzCdIyP8b8SCT2M/6PGKwzKJ5XbSBoL3gsnSWiFYqPg9c
uJPNbiy9wSA9vlyfWMd90qvTfiNrT6vewP813QdJ3EENZOx4ERcf/Wd22tV72kxD
```

```
yw1Q3I1OMH4bOItGQAxU5OtXCjBZEEUZooOkU8RoUQOU2Pql4NTiUpzWacNutAn5
HHS7MDc4lUlsJqbN+5QW6fFrcNG/0Mrib3JbwdFUNhrQ5j+Yq5h78HarnUivnX/3
Ap+oPbentv1qd7wvPJu556LZuhfqI0TohiIT1Ah+yUdN5osoaMxTHKKtf/CsSJ1F
w3qXqFJQA0VWsqjFyHXFI32I/GOupwIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQCn
Um00QHvUsJSN6KATbghowLynHn3wZSQsuS8E0COpcFJFxP2SV0NYkERbXu0n/Vhi
yq5F8v4/bRA2/xpedLWmvFs7QWlomuXhSnYFkd33Z5gnXPb9vRkLwiMSw4uXls35
qQraczUJ9EXDhrv7VmngIk9H3YsxYrlDGEqh/oz4Ze4ULOgnfkauanHikk+BUEsg
/jsTD+7e+niEzJPihHdsvKFDlud5pakEzyxovHwNJ1GS2I//yxrJFIL91mehjqEk
RLPdNse7N6UvSnuXcOokwu6l6kfzigGkJBxkcq4gre3szZFdCQcUioj7Z4xtuTL8
YMqfiDtN5cbD8R8ojw9Y
-----END CERTIFICATE-----
```


歐洲 (法蘭克福) — eu-central-1

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUFD5GsmkxRuecttwsCG763m3u63UwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE1NTUyOVoXDTI5MDQyODE1NTUyOVowXDELMAkGA1UEBhMCVVMx
```

```
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUFD5Gsmkx
RuecttwsCG763m3u63UwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQBBh0WaXlBsW56Hqk588MmJxsOrvcKfDjF57RgEDgnGnQaJcStCVWDO9UYO
JX2tdsPw+E7AjDqjsuxYaotLn3Mr3mK0sNOXq9BljBnWD4pARg89KZnZI8FN35HQ
O/LYOVHCknuPL123VmVRNs51qQA9hkPjvw21UzpDLxaUxt9Z/w==
-----END CERTIFICATE-----
```


RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAKD+v6LeR/WrMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNTA4MTQw
OTA4MTlaGA8yMTk1MDExNzA5MDgxOVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAka8FLhxs1cSJGK+Q+q/vTf8zVnDAPZ3U6oqppOW/cupCtpwMAQcky8DY
Yb62GF7+C6usniaq/9W6xPn/3o//wti0cNt6MLsiUeHqNl5H/4U/Q/fR+GA8pJ+L
npqZDG2tFi1WMvvGhGgIbScrjR4VO3TuKy+rZXMYvMRk1RXZ9gPhk6evFnviwHsE
jV5AEjxLz3duD+u/SjPp1vloxe2KuWnyC+EKInnka909sl4ZAUh+qIYfZK85DAjm
GJP4W036E9wTJQF2hZJrzsiB1MGyC1WI9veRISd30izZZL6VVXLXUtHwVHnVASrS
zZDVpzj+3yD5hRXsvFigGhY0FCVFnwIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQUxC2l6pvJaRflgu3MUdN6zTuP6YcwgY4GA1UdIwSBhjCBg4AUxC2l
6pvJaRflgu3MUdN6zTuP6YehYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJAKD+v6LeR/WrMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAIK+DtbUPppJXFqQMv1f2Gky5/82ZwgbbfXa
HBeGSii55b3tsyC3ZW5ZlMJ7Dtnr3vUkiWbV1EUaZGOUlndUFtXUMABCb/coDndw
CAr53XTv7UwGVNe/AFO/6pQDdPxXn3xBhF0mTKPrOGdvYmjZUtQMSVb9lbMWCFfs
w+SwDLnm5NF4yZchIcTs2fdpoyZpOHDXy0xgxO1gWhKTnYbaZOxkJvEvcckxVAwJ
obF8NyJla0/pWdjhlHafEXEN8lyxyTTyOa0BGTuYOBD2cTYYynauVKY4fqHUkr3v
Z6fboaHEd4RFamShM8uvSu6eEFD+qRmvqlcodbpsSOhuGNLzhOQ=
-----END CERTIFICATE-----
```

歐洲 (愛爾蘭) — eu-west-1

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUakDaQ1Zqy87Hy9ESXA1pFC116HkwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE2MTgxMFoXDTI5MDQyODE2MTgxMFowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUakDaQ1Zq
y87Hy9ESXA1pFC116HkwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQADIkn/MqaLGPuK5+prZZ5Ox4bBZLPtreO2C7r0pqU2kPM2lVPyYYydkvP0
lgSmmsErGu/oL9JNztDe2oCA+kNy17ehcsf8cw0uP861czNFKCeU8b7FgBbL+sIm
qi33rAq6owWGi/5uEcfCR+JP7W+oSYVir5r/yDmWzx+BVH5S/g==
```

```
-----END CERTIFICATE-----
```

RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAOrmqHuaUt0vMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNTEwMjkw
OTA2MTlaGA8yMTk1MDQwMzA5MDYxOVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAjE7nVu+aHLtzp9FYV25Qs1mvJ1JXD7J0iQ1Gs/RirW9a5ZECCtc4ssnf
zQHq2JRVr0GRchvDrbm1HaP/avtFQR/Thvfltwu9AROVT22dUOTvERdkNzveoFCy
hf52Rqf0DMrLXG8ZmQPPXPDFAv+sVMWCDftcChxRYZ6mP9O+TpgYNT1krD5PdvJU
7HcXrkNHDYqbsg8A+Mu2hzl0QkvUET83Csg1ibeK54HP9w+FsD6F5W+6ZSHGJ88l
FI+qYKs7xsjJQYgXWfEt6bbckWs1kZIaIOyMzYdPF6ClYzEec/UhIe/uJyUUNfpT
VIsI5OltBbcPF4c7Y20jOIwwI2SgOQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQUF2DgPUZivKQR/Zl8mB/MxIkjZDUwgY4GA1UdIwSBhjCBg4AUF2Dg
PUZivKQR/Zl8mB/MxIkjZDWhYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJAOrmqHuaUt0vMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAGm6+57W5brzJ3+T8/XsIdLTuiBSe5ALgSqI
qnO5usUKAeQsa+kZIJPyEri5i8LEodh46DAF1RlXTMYgXXxl0YggX88XPmPtok17
l4hib/D9/lu4IaFIyLzYNSzsETYWKWoGVe7ZFz60MTRTwY2u8YgJ5dec7gQgPSGj
avB0vTIgoW41G58sfw5b+wjXCsh0nROon79RcQFFhGnvup0MZ+JbljyhZUYFzCli
31jPZiKzqWa87xh2DbAyvj2KZrZtTe2LQ48Z4G8wWytJzxEeZdREe4NoETf+Mu5G
4CqoaPR05KWkdNUdGNwXewydb3+agdCgfTs+uAjeXKNdSpbhMYg=
-----END CERTIFICATE-----
```


歐洲 (倫敦) — eu-west-2

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
```

```
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUCgCV/DPxYNND/swDgEKGiC5I+EwwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE2MjkxNFoXDTI5MDQyODE2MjkxNFowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUCgCV/DPx
YNND/swDgEKGiC5I+EwwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQATPu/sOE2esNa4+XPEGKlEJSgqzyBSQLQc+VWo6FAJhGG9fp7D97jhHeLC
5vwfmtTAfnGBxadfAOT3ASkxnOZhXtnRna460LtnNHm7ArCVgXKJo7uBn6ViXtFh
uEEw4y6p9YaLQna+VC8Xtgw6WKq2JXuKzuhuNKSFaGGw9vRcHg==
-----END CERTIFICATE-----
```


RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJANBx0E2bOCEPMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNjA4MTEx
NDU2NDJaGA8yMTk2MDExNTE0NTY0MlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEArYS3mJLGaMrh2DmiPLbqr4Z+xWXTzBWCjOwpsuHE9H6dWUUyl2Bgnu+Z
d8QvW306Yleec45M4F2RA3J4hWHtShzsMlOJVRt+YulGeTf9OCPr26QmIFfs5nD4
```

```
fgsJQEry2MBSGA9Fxq3Cw6qkWcrOPsCR+bHOU0XykdKl0MnIbpBf0kTfciAupQEA
dEHnM2J1L2iI0NTLBgKxy5PXLH9weX20BFauNmHH9/J07OpwL20SN5f8TxcM9+pj
Lbk8h1V4KdIwVQpdWkbDL9BCGlYjyadQJxSxz1J343NzrnDM0M4h4HtVaKOS7bQo
Bqt2ruopLRCYgcuFHck/1348iAmbRQIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQBG
wujwU1Otpi3iBgmhjMClgZyMMn0aQIxMigoFNqXMUNx1Mq/e/Tx+SNaOEAu0n2FF
aiYjvY0/hXOx75ewzZvM7/zJWIdLdsgewpUqOBH4DXFhbSk2TxggSPb0WRqTBxq5
Ed7F7+7GRIeBbRzdLqmISDnfqey8ufW0ks51XcQNomDIRG5s9XZ5KHviDCar8FgL
HngBCdFI04CMagM+pwTO9XN1Ivt+NzUj208ca3oP1IwEAd5KhIhPLcihBQA5/Lpi
h1s3170z1JQ1HZbDrH1pgp+8hSI0DwwDVb3IIH8kPR/J0Qn+hvOl2HOpaUg2Ly0E
pt1RCZe+W7/dF4zsbqwK
-----END CERTIFICATE-----
```


歐洲 (米蘭) — eu-south-1

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7TCCAqwCCQCMElHPdwG37jAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xOTA0MjkyMDM1MjJaFw00
NTA0MjkyMDM1MjJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbYwggErBgcqhkjOOAQBMIIBHgKBgQDAkoL4YfdMI/MrQ0oL
NPfeEk94eiCQA5xNOnU7+2eVQtEqjFbDADFENh1p3sh9Q9OoheLFH8qpSfNDWn/0
ktCS909ApTY6Esx1ExjGSeQq/U+SC2JSuuTT4WFMKJ63a/czMtFkEPPnVIjJJJmT
HJSKSsVUgpdDIRvJXuyB0zdB+wIVALQ3OLaVGdlPMNfS1nD/Yyn+32wnAoGAPBQ3
7XHg5NLOS4326eFRUT+4ornQFjJjP6dp3pOBEzpImNmZTtkCNNUKE4Go9hv5T4lh
R0pODvWv0CBupMAZVBP9ObplXPCyEIZtuDqVa7ukPOUpQNgQhLLAqkigTyXVOSmt
ECBj9tu5WNP/x3iTZTHJ+g0rhIqpgh012UwJpKADgYQAAoGAV1OEQPYQUg5/M3xf
6vE7jKTxxyFWEyjKfJK7PZCzOIGrE/swgACy4PYQW+AwcUweSlK/Hx2OaZVUKzWo
wDUbeu65DcRdw2rSwCbBTU342sitFo/iGCV/Gjf+BaiAJtxniZze7J1ob8vOBeLv
uaMQmgOYeZ5e0fl04GtqPl+lhcQwCQYHKoZIzjgEAwMwADAtAhQdoeWLrkm0K49+
AeBK+j6m2h9SKQIVAIBNhS2a8cQVABDCQXVXrc0tOmO8
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIICNjCCAZ+gAwIBAgIJAOZ3GEIaDcugMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xOTEwMjQx
NTE5MDlaGA8yMTk5MDMyOTE1MTkwOVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
```

```
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKB
gQCjiPgW3vsXRj4JoA16WQDyoPc/eh3QBARaApJEc4nPIGoUolpAXcjFhWplo2O+
ivgfCsc4AU9OpYdAPha3spLey/bhHPRi1JZHRNqScKP0hzsCNmKhfnZTIEQCFvsp
DRp4zr91/WS06/flJFBYJ6JHhp0KwM81XQG59lV6kkoW7QIDAQABMA0GCSqGSIb3
DQEBCwUAA4GBAGLLrY3P+HH6C57dYgtJkuGZGT2+rMkk2n81/abzTJvsqRqGRrWv
XRKRXlKdM/dfiuYGokDGxiC0Mg6TYy6wvsR2qRhtXW1OtZkiHWcQCnOttz+8vpew
wx8JGMvowtuKB1iMsbwyRqZkFYLcvH+Opfb/Aayi20/ChQLdI6M2R5VU
-----END CERTIFICATE-----
```


RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJAO/+DgYF78KwMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xOTA0Mjky
MDM1MjJaGA8yMTk4MTAwMjIwMzUyMlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAv1ZLV+Z/P6INq+R1qLkzETBg7sFGKPiwHekbpuB6lrRxKHhj8V9vaReM
lnv1Ur5LAPpMPYDsuJ4WoUbPYAqVqyMAo7ikJHCCM1cXgZJefgN6z9bpS+uA3YVh
V/0ipHh/X2hc2S9wvxKWiSHu6Aq9GVpqL035tJQD+NJuqFd+nXrtcw4yGtmvA6wl
5Bjn8WdsP3xOTKjrByYY1BhXpP/f1ohU9jE9dstsRXLa+XTgTPWcWdCS2oRTWPGR
c5Aeh47nnDsyQfP9gLxHeYeQItV/BD9kU/2Hn6mnRg/B9/TYH8qzlRTzLapXp4/5
iNwusrTNexGl8BgvAPrfhjDpdgYuTwIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQB7
5ya11K/hKgvaRTvZwVV8GlVZt0CGPtNvOi4AR/UN6TMm51BzUB5nurB4z0R2MoYO
Uts9sLGvSFALJ4otoB77hyNpH3drttU1CVVwal/yK/RQLSon/IoUkaGEbqalu+mH
nYad5IG4tEbmepX456XXcO58MKmnczNbPyw3FRzUZQtI/sf94qBwJ1Xo6XbzPKMy
xjL57LHIZCssD+XPifXay69OFlsCIgLim11HgPkRIHEOXLSf3dsW9r+4CjoZqB/Z
jj/P4TLCxbYCLkvglwaMjgEWF40Img0fhx7yT2X92MiSrs3oncv/IqfdVTiN8OXq
jgnq1bf+EZEZKvb6UCQV
-----END CERTIFICATE-----
```


歐洲 (巴黎) — eu-west-3

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
```

```
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUaC9fX57UDr6u1vBvsCsECKBZQyIwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE2MzczOFoXDTI5MDQyODE2MzczOFowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUaC9fX57U
Dr6u1vBvsCsECKBZQyIwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQCARv1bQEDaMEzYI0nPlu8GHcMXgmgA94HyrXhMMcaIlQwocGBs6VILGVhM
TXP2r3JFaPEpmXSQNQHvGA13clKwAZbni8wtzv6qXb4L4muF34iQRHF0nYrEDoK7
mMPR8+oXKKuPO/mv/XKo6XAV5DDERdSYHX5kkA2R9wtvyZjPnQ==
-----END CERTIFICATE-----
```


RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJALWSfgHuT/ARMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNzA1MzEx
MTE4MTZaGA8yMTk2MTEwMzExMTgxNlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
```

```
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAy5V7KDqnEvF3DrSProFcgu/oL+QYD62b1U+Naq8aPuljJe127Sm9WnWA
EBdOSASkOaQ9fzjCPoG5SGgWKxYoZjsevHpmzjVv9+Ci+F57bSuMbjgUbvbRIFUB
bxQojVoXQPHgK5v433ODxkQ4sjRyUbf4YV1AFdfU7zabC698YgPVOExGhXPlTvco
8mlc631ubw2g52j0lzaozUkHPSbknTomhQIvO6kUfX0e0TDMH4jLDG2ZIrUB1L4r
OWKG4KetduFrRZyDHF6ILZu+s6ywiMicUd+2UllDFC6oas+a8D11hmO/rpWU/ieV
jj4rWAFrsebpn+Nhgy96iiVUGS2LuQIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQDE
iYv6FQ6knXCg+svlcaQG9q59xUC5z8HvJZ1+SxzPKKC4PKQdKvIIfE8GxVXqlZG1
cl5WKTFDMapnzb9RV/DTaVzWx3cMYT77vm1Hl1XGjhx611CGcENH1egI3lOTILsa
+KfopuJEQQ9TDMAIkGjhA+KieU/U5Ctv9fdej6d0GC6OEuwKkTNzPWue6UMq8d4H
2xqJboWsE1t4nybEosvZfQJcZ8jyIYcYBnsG13vCLM+ixjuU5MVVQNMY/gBJzqJB
V+U0QiGiuT5cYgY/QihxdHt99zwGaE0ZBC7213NKrlNuLSrqhDI2NLu8NsExqOFy
OmY0v/xVmQUQl26jJXaM
-----END CERTIFICATE-----
```


歐洲 (西班牙) — eu-south-2

DSA


```
-----BEGIN CERTIFICATE-----
MIIC8DCCAq
+gAwIBAgIGAXjwLk46MAkGByqGSM44BAMwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgMEFdhc2hpbmd0b24gU3RhdGUxED
U4EddRIpUt9KnC7s5Of2EbdSPO9EAMMeP4C2USZpRV1AIlH7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdrmVClpJ+f6AR7ECLCT7up1/63xhv4O1fnxqimFQ8E
+4P208UewwI1VBNaFpEy9nXzrith1yrv8iIDGZ3RSAHHAhUAl2BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v6OuqC+VdMCz0HgmdRWVeOutRZT
+ZxBxCBgLRJFnEj6EwoFhO3zwkyjMim4TwWeotUfI0o4KOuHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/hWuWfBpKLZl6Ae1UlZAFMO/7PSSoDgYQAAoGAGG2m8EKmaf5qQqj3Z
+rzSaTaXE3B/R/4A2VuGqRYR7MljPtwdmU6/3CPjCACcZmTIcOAKbFiDHqadQgBZXfzGpzw8Zo
+eYmmk5fXycgnj57PYH1dIWU6I7mCbAah5MZMcmHaTmIsomGrhcnWB8d8qOU7oZ0UWK4lbiAQs1MihoUwCQYHKoZIzjg
WmbaU7YM5GwCFCvIJOesO5hZ8PHC52dAR8WWC6oe
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIICMzCCAZygAwIBAgIGAXjwLkiaMA0GCSqGSIb3DQEBBQUAMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIDBBXYXNoaW5n
VvR1+45Aey5zn3vPk6xBm5o9grSDL6D2iAuprQnfVXn8CIbSDbWFhA3fi5ippjKkh3sl8VyCvCOUXKdOaNrYBrPRkrdH
+3m/
rxIUZ2IK1fDlC6sWAjddf6sBrV2w2a78H0H8EwuwiSgttURBjwJ7KPPJCqaqrQIDAQABMA0GCSqGSIb3DQEBBQUAA4GB
+FzqQDzun/
```

```
iMMzcFucmLMl5BxEblrFXOz7IIuOeiGkndmrqUeDCykztLku45s7hxdNy4ltTuVAaE5aNBdw5J8U1mRvsKvHLy2ThH6h
+hBgiphYp84DUbWVYeP8YqLEJSqscKscWC
-----END CERTIFICATE-----
```


RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJALWSmO6DvSpQMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yMjA3MTgx
MzU4NDNaGA8yMjAxMTIyMjEzNTg0M1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAuAAhuSpsHC00/fD2zNlBDpNLRndi9qbHsNeuz3WqN7Samj2aSrM2hS+i
hUxx0BspZj0tZCOsbpPZ+i74NOEQtFeqQoEGvKhB1nJiF4y5I81HDhs5qHvoIivm
7rbbik3zgm1PqS/DmDjVQaXPcD31Rd9ILwBmWEwJqHigyNVlxYtCzTQcrlBrvNZM
dnNgCDAdX/HBEFxx9O12xeu0bSt0s+PJWZ1RTbYrNe7LIH6ntUqHxP/ziQ5trXEZ
uqy7aWk1L8uK4jmyNph0lbaqBa3Y6pYmU1nC27UE4i3fnPB0LSiAr+SrwVvX1g4z
ilo8kr+tbIF+JmcgYLBvO8Jwp+EUqQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQUwvGzKJL9A5LReJ4Fxo5K6I20xcowgY4GA1UdIwSBhjCBg4AUwvGz
KJL9A5LReJ4Fxo5K6I20xcqhYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJALWSmO6DvSpQMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAJAZd31jyoTGLawAD2+v/vQsaB9vZIx5EImi
G8YGkd61uFWeNhAmtrwyE/i6FDSIphDrMHBkvw/D3BsqK+Ev/JOK/VYuaYDx/8fp
H4cwp9jC57CXzdIDREWNf6M9PsHFg2WA9XNNtClOZL5WJiJwel8eDSg+sqJUxE0l
MW+QChq/20F6niyaRK4bXrZq14as7h+F9u3A9xHEOVP7Zk9C2ehrBXzCMLSDt3GV
fEuMea2RxMhozwz34Hkdb6j18qoCfygubulovRNQjKw/cEmgPR16KfZPP5caILVt
9qkYPvePmbiVswZDee73cDymJYxLqILpOZwyXvUH8StiH42FHZQ=
-----END CERTIFICATE-----
```


歐洲 (斯德哥爾摩) — eu-north-1

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
```

```
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUN1c9U6U/xiVDFgJcYKZB4NkH1QEwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE2MDYwM1oXDTI5MDQyODE2MDYwM1owXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUN1c9U6U/
xiVDFgJcYKZB4NkH1QEwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQBTIQdoFSDRHkpqNPUbZ9WXR2O5v/9bpmHojMYZb3Hw46wsaRso7STiGGX/
tRqjIkPUIXsdhZ3+7S/RmhFznmZc8e0bjU4n5vi9CJtQSt+1u4E17+V2bF+D3h/7
wcfE0l3414Q8JaTDtfEf/aF3F0uyBvr4MDMd7mFvAMmDmBPSlA==
-----END CERTIFICATE-----
```


RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJALc/uRxg++EnMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xODA0MTAx
NDAwMTFaGA8yMTk3MDkxMzE0MDAxMVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
```

```
CgKCAQEAzwCGJEJIxqtr2PD2a1mA6LhRzKhTBa1AZsg3eYfpETXIVlrpojMfvVoN
qHvGshWLgrGTT6os/3gsaADheSaJKavxwX3X6tJA8fvEGqr3a1C1MffH9hBWbQqC
LbfUTAbkwis4GdTUwOwPjT1Cm3u9R/VzilCNwkj7iQ65AFAI8Enmsw3UGldEsop4
yChKB3KW3WI0FTh0+gD0YtjrqqYJxpGOYBpJp5vwdd3fZ4t1vidmDMs7liv4f9Bx
p0oSmUobU4GUlFhBchK1DukICVQdnOVzdMonYm7s+HtpFbVHR8yf6QoixBKGdSal
mBf7+y0ixjCn0pnC0VLVooGo4mi17QIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQDG
4ONZiixgk2sjJctwbyD5WKLTH6+mxYcDw+3y/F0fWz561YORhP2FNnPOmEkf0Sl/
Jqk4svzJbCbQeMzRoyaya/46d7UioXMHRZam5IaGBhOdQbi97R4VsQjwQj0RmQsq
yDueDyuKTwWLK9KnvI+ZA6e6bRkdNGflK4N8GGKQ+fBhPwVELkbT9f16OJkezeeN
S+F/gDADGJgmPXfjogICb4Kvshq0H5Lm/xZlDULF2g/cYhyNY6EOI/eS5m1I7R8p
D/m6WoyZdpInxJfxW616OMkxQMRVsruLTNGtby3u1g6ScjmpFtvAMhYejBSdzKG4
FEyxIdEjoeO1jhTsck3R
-----END CERTIFICATE-----
```


歐洲 (蘇黎世) — eu-central-2

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7zCCAq
+gAwIBAgIGAXjXiKJnMAkGByqGSM44BAMwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgMEFdhc2hpbmd0b24gU3RhdGUxED
U4EddRIpUt9KnC7s5Of2EbdSPO9EAMMeP4C2USZpRV1AIlH7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdrmVClpJ+f6AR7ECLCT7up1/63xhv4O1fnxqimFQ8E
+4P208UewwI1VBNaFpEy9nXzrith1yrv8iIDGZ3RSAHHAhUAl2BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v6OuqC+VdMCz0HgmdRWVeOutRZT
+ZxBxCBgLRJFnEj6EwoFhO3zwkyjMim4TwWeotUfI0o4KOuHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/hWuWfBpKLZl6Ae1UlZAFMO/7PSSoDgYQAAoGAYNjaCNg/
cfgQOl1BUj5ClUulqwZ9Q+SfDzPZh9D2C0VbiRANiZoxrV8RdgmzzC5T7VcriVwjwvta2Ch//
b+sZ86E5h0XWWr+BeEjD9cu3eDj12XB5sWEbNHNx49p5Tmtu5r2LDtlL8X/
Rpfalu2Z2OJgjFJWGf7hRwxe456n
+lowCQYHKoZIzjgEAwMvADAsAhRChsLcj4U5CVb2cp5M0RE1XbXmhAIUeGSnH+aiUQIWmPEFja+itWDufIk=
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIICMzCCAZygAwIBAgIGAXjSGFGiMA0GCSqGSIb3DQEBBQUAMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIDBBXYXNoaW5n
opKZAUusJx2hpgU3pUHhlp9ATh/VeVD582jTd9IY
+8t5MDa6Z3fGliByEiXz0LEHdi8MBacLREu1TwIDAQABMA0GCSqGSIb3DQEBBQUAA4GBAILlpoE3k9o7KdALAxsFJNit
+g3RMzdbiFM+7MA63Nv5fsf+0xgcjSNBElvPCDKFvTJl4QQhToy056llO5GvdS9RK
+H8xrP2mrqngApoKTApv93vHBixgFSn5KrczRO0YSm3OjkqbydU7DFlmkXXR7GYE+5jbHvQHYiT1J5sMu
-----END CERTIFICATE-----
```

RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJALvT012pxTxNMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yMjA3MTgx
NTEyMDdaGA8yMjAxMTIyMjE1MTIwN1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAyn+Lsnq1ykrfYlZkk6aAAYNReNd9Iw8AUwCBkgOr2eBiBBepYxHwU85N
++moQ+j0EV2VaahBeTLShGZZS1HsyK8+cYT2QzpghIoamcYhrPXyIxlWiRQlaqSg
OFiE9bsqL3rCF5Vz+tOiTe5W/7ojfOFls6++g7ZpobwJlpMbuJepqyeHMPyjvO5A
age81lJewc4bxo2ntaW0HCqNksqfYB78j6X6kn3PFpX7FaYAwZA+Xx6C7UCY7rNi
UdQzfAo8htfJi4chz7frpUdQ9kl3IOQrsLshBB5fFUjl09NiFipCGBwi+8ZMeSn1
5qwBI01BWXPfG7WX6Owyjhmh6JtE1wIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQU8HN4vvJrsZgPQeksMBgJb9xR1yYwgY4GA1UdIwSBhjCBg4AU8HN4
vvJrsZgPQeksMBgJb9xR1yahYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJALvT012pxTxNMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAGlHYDtcHpfBvdHx9HeQE8HgNugJUPdEqxun
t9U33p8VFrs+uLPtrOd9HDJEGvvs5h84EUie/oGJxRt7V1Vlid1PvHf6cRmpjgqY
YdggAVkZtY/PnFVmzf2bMVlSQPrqCl7U0zaw2Kvnj4zgX0rZyCetgrRZSUSxotyp
978Wy9ccXwVSeYG/YAr5rJpS6ZH7eRQvUY0IzwFNeaOPgOTEVpcjWlV6+MQEvsEx
W85q+s6AVr49eppEx8SLJsl0C23yB+L+t32tAveQImRWtJMpzZ5cxh/sYgDVeoC0
85H1NK/7H9fAzT1cPu1oHSnB0xYzzHGOAmXmusMfwUk8fL1RQkE=
-----END CERTIFICATE-----
```


以色列 (特拉維夫) — il-central-1

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7zCCAq+gAwIBAgIGAX0QPi
+9MAkGByqGSM44BAMwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgMEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcMB1NlYX
U4EddRIpUt9KnC7s5Of2EbdSPO9EAMMeP4C2USZpRV1AIlH7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdrmVClpJ+f6AR7ECLCT7up1/63xhv4O1fnxqimFQ8E
+4P208UewwI1VBNaFpEy9nXzrith1yrv8iIDGZ3RSAHHAhUAl2BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v6OuqC+VdMCz0HgmdRWVeOutRZT
+ZxBxCBgLRJFnEj6EwoFhO3zwkyjMim4TwWeotUfI0o4KOuHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/
hWuWfBpKLZl6Ae1UlZAFMO/7PSSoDgYQAAoGAbazCL5XXyPmcw3+oMYQUF5/9YogW6D0FZbYuyPgjOoUwWdl6fjlzWca
```

```
pq+l1ezuK2DFOzNTEyPEwwCQYHKoZIzjgEAwMvADAsAhRt1jkpXsvrS
+xTo2M9h2s2uLAhEQIUOZ2FcnTSrshF2EIdixZZwtNv66Q=
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIICMzCCAZygAwIBAgIGAX0QQGVLMA0GCSqGSIb3DQEBBQUAMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIDBBXYXNoaW5n
+S8vOy5hpLoRe4RkOrY0cM3bN07GdEMlin5mU0y1t8y3ct4YewvmkgT42kTyMM
+t1K4S0xsqjXxxS716uGYh7eWtkxrCihj8AbXN/6pa095h
+7TZyl2n83keiNUzM2KoqQVMwIDAQABMA0GCSqGSIb3DQEBBQUAA4GBADwA6VVEIIZD2YL00F12po40xDLzIc9XvqFPS
FmU7H8s62/jD6cOR1A1cClIyZUe1yT1ZbPySCs43J+Thr8i8FSRxzDBSZZi5foW
-----END CERTIFICATE-----
```


RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAOVp1h2I9wW7MA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yMjA3MTUx
MjQ0MTJaGA8yMjAxMTIxOTEyNDQxMlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEA13PkyWvl61iV/SYfO1UFO76UpDfPm2SF/Rz/o33cm699X++EYPxTnoEc
vmWeS0I7eDXc4OCUiToG0sEx0klE0CX1Z1tK6qJ+zgWQLZ9SZEC9H0NsSA6LhrHu
Nq0dzeK3LjhdfcX46/4GqdiptpdTuM4m/hOQ5yx4JMQ/n1sdpv4M5VLRWwWW9Lem
ufb79Id7O9SispxgRsz1KXIjp7N9S4BY7itSXz97uSyzTqEjWZ6mDUhTu3t21GKC
6flALGTTrG2yghEhz53rkvLsvwzjPSSlT6LIf0mrRPzHaf+EdaKoasElE1SHh+ZH
9mI81HywpE+HZ+W+5hBCvjYp90Y1fwIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQU58tN2JO+yEGq5JbIXxGi4vRVPyIwgY4GA1UdIwSBhjCBg4AU58tN
2JO+yEGq5JbIXxGi4vRVPyKhYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJAOVp1h2I9wW7MBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBANBN0e1EqNy4+IU2yQzMJ+Wy5ZIOtTP6GSBR
7muVY1bDeAwtNTEOpwgrZVlC7/xq5Q0LC1y0Z70hHXEf8au7qStaAoUtxzvhTAZI
NC0lwoFU56UFw4N0vZII17iqEfoqRC4PpI3OxqEJHFy0VLlvAzJoKB4QLLqDAYVA
LXCiOLoVT+y9tRYsxw5MyO0Bi6fxQIIADl2bE9xkunTN1Jkkwqo3LxNy/ryz4QWR
8K7jHUItifv4h/hxBKpHEquN8CkdvM9oeGl7I8PFrSFEpGr1euDXY0euZzzYiDBV
m6GpTJgzpVsEuIX52dPcPemwQncoIfZyhWDW85MJUnby2WTEcFo=
-----END CERTIFICATE-----
```

中東 (巴林) — me-south-1

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7jCCAq4CCQCVWIgSmP8RhTAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xOTAyMDUxMzA2MjFaFw00
NTAyMDUxMzA2MjFaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbgwggEsBgcqhkjOOAQBMIIBHwKBgQDcwojQfgWdV1QliO0B
8n6cLZ38VE7ZmrjZ9OQV//Gst6S1h7euhC23YppKXi1zovefSDwFU54zi3/oJ++q
PHlP1WGL8IZ34BUgRTtG4TVolvp0smjkMvyRu5hIdKtzjV93Ccx15gVgyk+o1IEG
fZ2Kbw/Dd8JfoPS7KaSCmJKxXQIVAIZbIaDFRGa2qcMkW2HWASyNDl7bAoGBANtz
IdhfMq+l2I5iofY2oj3HI21Kj3LtZrWEg3W+/4rVhL3lTm0Nne1rl9yGujrjQwy5
Zp9V4A/w9w2O10Lx4K6hj34Eefy/aQnZwNdNhv/FQP7Az0fju+Yl6L13OOHQrL0z
Q+9cF7zEosekEnBQx3v6psNknKgD3Shgx+GO/LpCA4GFAAKBgQCVS7m77nuNAlZ8
wvUqcooxXMPkxJFl54NxAsAul9KP9KN4svm0O3Zrb7t2FOtXRM8zU3TqMpryq1o5
mpMPsZDg6RXo9BF7Hn0DoZ6PJTamkFA6md+NyTJWJKvXC7iJ8fGDBJqTciUHuCKr
12AztQ8bFWsrTgTzPE3p6U5ckcgV1TAJBgcqhkjOOAQDAy8AMCwCFB2NZGWm5EDl
86ayV3c1PEDukgQIAhQow38rQkN/VwHVeSW9DqEshXHjuQ==
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIIDPDCCAqWgAwIBAgIJAMl6uIV/zqJFMA0GCSqGSIb3DQEBCwUAMHIxCzAJBgNV
BAYTAlVTMRMwEQYDVQQIDApXYXNoaW5ndG9uMRAwDgYDVQQHDAdTZWF0dGxlMSAw
HgYDVQQKDBdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzEaMBgGA1UEAwwRZWMyLmFt
YXpvbmF3cy5jb20wIBcNMTkwNDI2MTQzMjQ3WhgPMjE5ODA5MjkxNDMyNDdaMHIx
CzAJBgNVBAYTAlVTMRMwEQYDVQQIDApXYXNoaW5ndG9uMRAwDgYDVQQHDAdTZWF0
dGxlMSAwHgYDVQQKDBdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzEaMBgGA1UEAwwR
ZWMyLmFtYXpvbmF3cy5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBALVN
CDTZEnIeoX1SEYqq6k1BV0ZlpY5y3KnoOreCAE589TwS4MX5+8Fzd6AmACmugeBP
Qk7Hm6b2+g/d4tWycyxLaQlcq81DB1GmXehRkZRgGeRge1ePWd1TUA0I8P/QBT7S
gUePm/kANSFU+P7s7u1NNl+vynyi0wUUrw7/wIZTAgMBAAGjgdcwgdQwHQYDVR0O
BBYEFILtMd+T4YgH1cgc+hVsVOV+480FMIGkBgNVHSMEgZwwgZmAFILtMd+T4YgH
1cgc+hVsVOV+480FoXakdDByMQswCQYDVQQGEwJVUzETMBEGA1UECAwKV2FzaGlu
Z3RvbjEQMA4GA1UEBwwHU2VhdHRsZTEgMB4GA1UECgwXQW1hem9uIFdlYiBTZXJ2
aWNlcyBMTEMxGjAYBgNVBAMMEWVjMi5hbWF6b25hd3MuY29tggkAyXq4hX/OokUw
DAYDVR0TBAUwAwEB/zANBgkqhkiG9w0BAQsFAAOBgQBhkNTBIFgWFd+ZhC/LhRUY
4OjEiykmbEp6hlzQ79T0Tfbn5A4NYDI2icBP0+hmf6qSnIhwJF6typyd1yPK5Fqt
NTpxxcXmUKquX+pHmIkK1LKDO8rNE84jqxrxRsfDi6by82fjVYf2pgjJW8R1FAw+
```

```
mL5WQRFexbfB5aXhcMo0AA==
-----END CERTIFICATE-----
```


RSA-2048


```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJANZkFlQR2rKqMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xOTAyMDUx
MzA2MjBaGA8yMTk4MDcxMTEzMDYyMFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAy4Vnit2eBpEjKgOKBmyupJzJAiT4fr74tuGJNwwa+Is2vH12jMZn9Il1
UpvvEUYTIboIgISpf6SJ5LmV5rCv4jT4a1Wm0kjfNbiIlkUi8SxZrPypcw24m6ke
BVuxQZrZDs+xDUYIZifTmdgD50u5YE+TLg+YmXKnVgxBU6WZjbuK2INohi71aPBw
2zWUR7Gr/ggIpf635JLU3KIBLNEmrkXCVSnDFlsK4eeCrB7+UNak+4BwgpuykSGG
Op9+2vsuNqFeU1l9daQeG9roHR+4rIWSPa0opmMxv5nctgypOrE6zKXx2dNXQldd
VULv+WH7s6Vm4+yBeG8ctPYH5GOo+QIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQBs
ZcViiZdFdpcXESZP/KmZNDxB/kktlIEIhsQ+MNn29jayE5oLmtGjHj5dtA3XNKlr
f6PVygVTKbtQLQqunRT83e8+7iCZMKI5ev7pITUQVvTUwI+Fc01JkYZxRFlVBuFA
WGZO+98kxCS4n6tTwVt+nSuJr9BJRVC17apfHBgSS8c5OWna0VU/Cc9ka4eAfQR4
7pYSDU3wSRE01cs30q34lXZ629IyFirSJ5TTOIc0osNL7vwMQYj8HOn4OBYqxKy8
ZJyvfXsIPh0Na76PaBIs6ZlqAOflLrjGzxBPiwRM/XrGmF8ze4KzoUqJEnK13O6A
KHKgfiigQZ1+gv5FlyXH
-----END CERTIFICATE-----
```


中東 (UAE) — me-central-1

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7zCCAq
+gAwIBAgIGAXjXhqnnMAkGByqGSM44BAMwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgMEFdhc2hpbmd0b24gU3RhdGUxED
U4EddRIpUt9KnC7s5Of2EbdSPO9EAMMeP4C2USZpRV1AIlH7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdrmVClpJ+f6AR7ECLCT7up1/63xhv4O1fnxqimFQ8E
+4P208UewwI1VBNaFpEy9nXzrith1yrv8iIDGZ3RSAHHAhUAl2BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v6OuqC+VdMCz0HgmdRWVeOutRZT
+ZxBxCBgLRJFnEj6EwoFhO3zwkyjMim4TwWeotUfI0o4KOuHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/hWuWfBpKLZl6Ae1UlZAFMO/7PSSoDgYQAAoGAW+csuHsWp/7/
pv8CTKFwxsYudxuR6rbWaHCykIeAydXL9AWnphK6yp1ODEMBFl68Xq8Hp23sOWyf8moOhqCom9+0+ovuUFdpvCie86bp
TOZU568Ty1ff3dDWbdRzeNQRHodRG+XEQSizMkAreeWt4kBa+PUwCQYHKoZIzjgEAwMvADAsAhQD3Z
+XGmzKmgaLgGcVX/Qf1+Tn4QIUH1cgksBSVKbWj81tovBMJeKgdYo=
```

```
-----END CERTIFICATE-----
```

RSA

```
-----BEGIN CERTIFICATE-----
MIICMzCCAZygAwIBAgIGAXjRrnDjMA0GCSqGSIb3DQEBBQUAMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIDBBXYXNoaW5n
KyA6zyruJQrYy00a6wqLA7eeUzk3bMiTkLsTeDQfrkaZMfBAjGaaOymRo1C3qzE4rIenmahvUplu9ZmLwL1idWXMRX2R
+d2SeoKOKQWoc2UOFZMHYxDue7zkyk1CIRaBukTeY13/
RIrlc6X61zJ5BBtZXlHwayjQIDAQABMA0GCSqGSIb3DQEBBQUAA4GBABTqTy3R6RXKPW45FA+cgo7YZEj/
Cnz5YaoUivRRdX2A83BHuBTvJE2+WX00FTEj4hRVjameE1nENoO8Z7fUVloAFDlDo69fhkJeSvn51D1WRrPnoWGgEfr1
B+Wqm3kVEz/QNcz6npmA6
-----END CERTIFICATE-----
```

RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAM4h7b1CVhqqMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yMjA0MTEx
MDE1MDNaGA8yMjAxMDkxNTEwMTUwM1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEApybTWfMOhSoMpqPo72eqAmnn1dXGZM+G8EoZXzwHwT/+IHEXNB4q5N6k
tudYLre1bJxuzEw+iProSHjmb9bB9YscRTofjVhBlt35Fc+i8BaMeH94SR/eE8QO
mll8gnLNW3d62lyuhzuyv1e5wVlRqzYw+X2zRH4/wRDOCOpzjKoHIgyPKsMgwsw5
aTZhNMsGxZN9dbkf0iCGeQLDytwU/JTh/HqvSr3VfUOapTJJiyAxoCtZWgp1/7wC
RvOCSMRJobpUqxZgl/VsttwNkikSFz1wGkcYeSQvk+odbnYQckA8tdddoVI56eD4
qtREQvfpMAX5v7fcqLexl5d5vH8uZQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQUOadrbTs+OhzwoAgUJ7RqQNdwufkwgY4GA1UdIwSBhjCBg4AUOadr
bTs+OhzwoAgUJ7RqQNdwufmhYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJAM4h7b1CVhqqMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAICTdAOGEOnII8HaGCpCB8us/hGFaLptJaAf
D5SJAyVy66/mdfjGzE1BKkKxnbxemEVUIzbRidOnyilB+pKwN3edAjTZtWdpVA0V
R/G/qQPmcVljtycBz4VC6SuOUYflGzLH1GZ6GJWbuDtFzw8r7HGdRN1wrEPe3UF2
sMpuVezqnRUdvVRoVQP4jFgNsE7kNvtn2NiPhb/CtrxpcwIQ7r6YeoHcBSheuV1Z
xZDHynC3KUprQGx1+Z9QqPrDf180MaoqAlTl4+W6Pr2NJYrVUFGS/ivYshMg574l
CPU6r4wWZSKwEUXq4BInYX6z6iclp/p/J5QnJp2mAwyi6M+I13Y=
-----END CERTIFICATE-----
```

南美洲 (聖保羅) — sa-east-1

DSA


```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUX4Bh4MQ86Roh37VDRRX1MNOB3TcwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE2NDYwOVoXDTI5MDQyODE2NDYwOVowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUX4Bh4MQ8
6Roh37VDRRX1MNOB3TcwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQBnhocfH6ZIX6F5K9+Y9V4HFk8vSaaKL5ytw/P5td1h9ej94KF3xkZ5fyjN
URvGQv3kNmNJBoNarcP9I7JIMjsNPmVzqWawyCEGCZImoARxSS3Fc5EAs2PyBfcD
9nCtzMTaKO09Xyq0wqXVYn1xJsE5d5yBDsGrzaTHKjxo61+ezQ==
```

```
-----END CERTIFICATE-----
```

RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAMcyoxx4U0xxMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNTA4MTQw
ODU4MDJaGA8yMTk1MDExNzA4NTgwMlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAw45IhGZVbQcy1fHBqzROhO8CsrDzxj/WP4cRbJo/2DAnimVrCCDs5O86
FA39Zo1xsDuJHDlwMKqeXYXkJXHYbcPWc6EYYAnR+PlLG+aNSOGUzsy202S03hT0
B20hWPCqpPp39itIRhG4id6nbNRJOzLm6evHuepMAHR4/OV7hyGOiGaV/v9zqiNA
pMCLhbh2xk0PO35HCVBuWt3HUjsgeks2eEsu9Ws6H3JXTCfiqp0TjyRWapM29OhA
cRJfJ/d/+wBTz1fkWOZ7TF+EWRIN5ITEadlDTPnF1r8kBRuDcS/lIGFwrOOHLo4C
cKoNgXkhTqDDBDu6oNBb2rS0K+sz3QIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQUqBy7D847Ya/w321Dfr+rBJGsGTwwgY4GA1UdIwSBhjCBg4AUqBy7
D847Ya/w321Dfr+rBJGsGTyhYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJAMcyoxx4U0xxMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBACOoWSBf7b9AlcNrl4lr3QWWSc7k90/tUZal
PlT0G3Obl2x9T/ZiBsQpbUvs0lfotG0XqGVVHcIxF38EbVwbw9KJGXbGSCJSEJkW
vGCtc/jYMHXfhx67Szmftm/MTYNvnzsyQQ3v8y3Rdah+xe1NPdpFrwmfL6xe3pFF
cY33KdHA/3PNLdn9CaEsHmcmj3ctaaXLFIzZhQyyjtsrgGfTLvXeXRokktvsLDS/
YgKedQ+jFjzVJqgr4NjfY/Wt7/8kbbdhzaqlB5pCPjLLzv0zp/XmO6k+JvOePOGh
JzGk5t1QrSju+MqNPFk3+1O7o910Vrhqw1QRB0gr1ExrviLbyfU=
-----END CERTIFICATE-----
```

AWS GovCloud (美國東部) — -1 us-gov-east

DSA

```
-----BEGIN CERTIFICATE-----
 MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
 FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
 VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
 ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
 IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
 cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
 ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
 VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
```

```
 hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
 k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
 hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
 lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
 MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
 MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
 vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
 7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
 -----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
 MIIDITCCAoqgAwIBAgIULVyrqjjwZ461qelPCiShB1KCCj4wDQYJKoZIhvcNAQEL
 BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
 BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
 MB4XDTI0MDUwNzE1MjIzNloXDTI5MDUwNjE1MjIzNlowXDELMAkGA1UEBhMCVVMx
 GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
 BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
 A4GNADCBiQKBgQCpohwYUVPH9I7Vbkb3WMe/JB0Y/bmfVj3VpcK445YBRO9K80al
 esjgBc2tAX4KYg4Lht4EBKccLHTzaNi51YEGX1aLNrSmxhz1+WtzNLNUsyY3zD9z
 vwX/3k1+JB2dRA+m+Cpwx4mjzZyAeQtHtegVaAytkmqtxQrSCexBxvqRqQIDAQAB
 o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQU1ZXneBYnPvYXkHVlVjg7918V
 gE8wgZkGA1UdIwSBkTCBjoAU1ZXneBYnPvYXkHVlVjg7918VgE+hYKReMFwxCzAJ
 BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
 ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IULVyrqjjw
 Z461qelPCiShB1KCCj4wEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
 AAOBgQBfAL/YZv0y3zmVbXjyxQCsDloeDCJjFKIu3ameEckeIWJbST9LMto0zViZ
 puIAf05x6GQiEqfBMk+YMxJfcTmJB4Ebaj4egFlslJPSHyC2xuydHlr3B04INOH5
 Z2oCM68u6GGbj0jZjg7GJonkReG9N72kDva/ukwZKgq8zErQVQ==
 -----END CERTIFICATE-----
```


RSA-2048


```
-----BEGIN CERTIFICATE-----
 MIIDOzCCAiOgAwIBAgIJALPB6hxFhay8MA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
 BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
 dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xODA0MTAx
 MjMyNDlaGA8yMTk3MDkxMzEyMzI0OVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
 EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
 YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
 CgKCAQEAva9xsI9237KYb/SPWmeCVzi7giKNron8hoRDwlwwMC9+uHPd53UxzKLb
 pTgtJWAPkZVxEdl2Gdhwr3SULoKcKmkqE6ltVFrVuPT33La1UufguT9k8ZDDuO9C
```

```
 hQNHUdSVEuVrK3bLjaSsMOS7Uxmnn7lYT990IReowvnBNBsBlcabfQTBV04xfUG0
 /m0XUiUFjOxDBqbNzkEIblW7vK7ydSJtFMSljga54UAVXibQt9EAIF7B8k9l2iLa
 mu9yEjyQy+ZQICTuAvPUEWe6va2CHVY9gYQLA31/zU0VBKZPTNExjaqK4j8bKs1/
 7dOV1so39sIGBz21cUBec1o+yCS5SwIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQBt
 hO2W/Lm+Nk0qsXW6mqQFsAou0cASc/vtGNCyBfoFNX6aKXsVCHxq2aq2TUKWENs+
 mKmYu1lZVhBOmLshyllh3RRoL3Ohp3jCwXytkWQ7ElcGjDzNGc0FArzB8xFyQNdK
 MNvXDi/ErzgrHGSpcvmGHiOhMf3UzChMWbIr6udoDlMbSIO7+8F+jUJkh4Xl1lKb
 YeN5fsLZp7T/6YvbFSPpmbn1YoE2vKtuGKxObRrhU3h4JHdp1Zel1pZ6lh5iM0ec
 SD11SximGIYCjfZpRqI3q50mbxCd7ckULz+UUPwLrfOds4VrVVSj+x0ZdY19Plv2
 9shw5ez6Cn7E3IfzqNHO
 -----END CERTIFICATE-----
```


AWS GovCloud (美國西部) — -1 us-gov-west

DSA


```
-----BEGIN CERTIFICATE-----
 MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
 FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
 VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
 ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
 IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
 cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
 ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
 VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
 hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
 k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
 hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
 lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
 MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
 MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
 vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
 7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
 -----END CERTIFICATE-----
```


RSA


```
-----BEGIN CERTIFICATE-----
 MIIDITCCAoqgAwIBAgIUe5wGF3jfb7lUHzvDxmM/ktGCLwwwDQYJKoZIhvcNAQEL
 BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
 BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
 MB4XDTI0MDUwNzE3MzAzMloXDTI5MDUwNjE3MzAzMlowXDELMAkGA1UEBhMCVVMx
```

```
 GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
 BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
 A4GNADCBiQKBgQCpohwYUVPH9I7Vbkb3WMe/JB0Y/bmfVj3VpcK445YBRO9K80al
 esjgBc2tAX4KYg4Lht4EBKccLHTzaNi51YEGX1aLNrSmxhz1+WtzNLNUsyY3zD9z
 vwX/3k1+JB2dRA+m+Cpwx4mjzZyAeQtHtegVaAytkmqtxQrSCexBxvqRqQIDAQAB
 o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQU1ZXneBYnPvYXkHVlVjg7918V
 gE8wgZkGA1UdIwSBkTCBjoAU1ZXneBYnPvYXkHVlVjg7918VgE+hYKReMFwxCzAJ
 BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
 ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUe5wGF3jf
 b7lUHzvDxmM/ktGCLwwwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
 AAOBgQCbTdpx1Iob9SwUReY4exMnlwQlmkTLyA8tYGWzchCJOJJEPfsW0ryy1A0H
 YIuvyUty3rJdp9ib8h3GZR71BkZnNddHhy06kPs4p8ewF8+d8OWtOJQcI+ZnFfG4
 KyM4rUsBrljpG2aOCm12iACEyrvgJJrS8VZwUDZS6mZEnn/lhA==
 -----END CERTIFICATE-----
```


RSA-2048


```
-----BEGIN CERTIFICATE-----
 MIIDOzCCAiOgAwIBAgIJANCOF0Q6ohnuMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
 BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
 dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNTA5MTAx
 OTQyNDdaGA8yMTk1MDIxMzE5NDI0N1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
 EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
 YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
 CgKCAQEAzIcGTzNqie3f1olrrqcfzGfbymSM2QfbTzDIOG6xXXeFrCDAmOq0wUhi
 3fRCuoeHlKOWAPu76B9os71+zgF22dIDEVkpqHCjBrGzDQZXXUwOzhm+PmBUI8Z1
 qvbVD4ZYhjCujWWzrsX6Z4yEK7PEFjtf4M4W8euw0RmiNwjy+knIFa+VxK6aQv94
 lW98URFP2fD84xedHp6ozZlr3+RZSIFZsOiyxYsgiwTbesRMI0Y7LnkKGCIHQ/XJ
 OwSISWaCddbu59BZeADnyhl4f+pWaSQpQQ1DpXvZAVBYvCH97J1oAxLfH8xcwgSQ
 /se3wtn095VBt5b7qTVjOvy6vKZazwIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQA/
 S8+a9csfASkdtQUOLsBynAbsBCH9Gykq2m8JS7YE4TGvqlpnWehz78rFTzQwmz4D
 fwq8byPkl6DjdF9utqZ0JUo/Fxelxom0h6oievtBlSkmZJNbgc2WYm1zi6ptViup
 Y+4S2+vWZyg/X1PXD7wyRWuETmykk73uEyeWFBYKCHWsO9sI+62O4Vf8Jkuj/cie
 1NSJX8fkervfLrZSHBYhxLbL+actVEo00tiyZz8GnhgWx5faCY38D/k4Y/j5Vz99
 7lUX/+fWHT3+lTL8ZZK7fOQWh6NQpI0wTP9KtWqfOUwMIbgFQPoxkP00TWRmdmPz
 WOwTObEf9ouTnjG9OZ20
 -----END CERTIFICATE-----
```


EC2執行個體上的精確時脈和時間同步

Amazon EC2 執行個體上一致且準確的時間參考對於許多伺服器任務和程序而言至關重要。在識別問題發生的時間和事件的時間順序方面，系統日誌中的時間戳記扮演著至關重要的角色。當您使用 AWS

CLI 或向執行個體提出 AWS SDK要求時，這些工具會代表您簽署要求。如果執行個體的日期和時間設定不正確，可能會導致簽名日期和請求日期之間出現差異，導致 AWS 拒絕您的請求。

為了解決這一重要方面，Amazon 提供了 Amazon 時間同步服務，該服務可從所有EC2實例訪問並被各個實例使用 AWS 服務。該服務使用衛星連接和原子參考時鐘的艦隊，以提供協調世界時（ ）UTC全球標準的準確和當前時間讀數。 AWS 區域

為了獲得最佳效能，我們建議您在EC2執行個體上使用本機 Amazon 時間同步服務。若要備份到執行個體上的本機 Amazon 時間同步服務，或將 Amazon 外部的資源連接EC2到 Amazon 時間同步服務，您可以使用位於的公有 Amazon 時間同步服務`time.aws.com`。公用的 Amazon 時間同步服務 (如本機 Amazon 時間同步服務) 會自動塗抹新增到UTC的任何閏秒。公用 Amazon 時間同步服務由我們的衛星連接和原子參考時鐘在全球範圍內提供支援。 AWS 區域

閏秒

1972 年推出的閏秒是偶爾調整UTC時間的一秒鐘，以因應國際原子時間（TAI）和太陽時間（Ut1）之間的差異而考慮地球自轉的不規則性。為了代表客戶管理閏秒，我們在 Amazon Time Sync Service 中設計了閏秒消彌機制。如需詳細資訊，請參閱閏秒搶先報：即將到來的閏秒和 AWS。

國際度量衡局第 27 屆大會決議，將在 2035 年之前取消閏秒，我們全力支持這項決議。

為了支援此轉換，我們仍計劃透過本機NTP連線或我們的公用NTP集區存取 Amazon Time Sync 服務時，在閏秒事件期間塗抹時間 (`time.aws.com`)。但是，PTP硬體時鐘並不提供塗抹的時間選項。如果發生閏秒，PTP硬體時鐘會UTC依照標準新增閏秒。在大多數情況下，消彌的閏秒和閏秒時間來源相同。但由於閏秒事件期間會有所不同，所以不建議在閏秒事件期間，在您的時間用戶端組態中同時使用消彌和未消彌的時間來源。

主題

- 在EC2執行個體上設定時間參考以使用本機 Amazon 時間同步服務
- 在EC2執行個體或任何連接網際網路的裝置上設定時間參考，以使用公用的 Amazon 時間同步服務
- 比較 Linux 執行個體的時間戳記
- 變更執行個體的時區

相關資源

- AWS 運算部落格：關於時間：Amazon 執行個體上的微秒精確時鐘 EC2
- AWS 雲端操作和遷移部落格：使用 Amazon 時間同步服務和 Amazon 管理 Amazon EC2 執行個體時鐘準確性 CloudWatch — 第 1 部分
- （Linux）https://chrony-project.org/

在EC2執行個體上設定時間參考以使用本機 Amazon 時間同步服務

本機 Amazon 時間同步服務可以使用網路時間通訊協定 (NTP)，或在支援的執行個體上提供本機精確時間協定 (PTP) 硬體時鐘。PTP硬體時鐘支援NTP連線 (Linux 和 Windows 執行個體) 或直接PTP連線 (僅限 Linux 執行個體)。NTP和直接PTP連接使用相同的高度準確的時間來源，但直接PTP連接比NTP連接更準確。與 Amazon 時間同步服務的NTP連線支援飛躍塗抹，而與PTP硬體時鐘的PTP連線不會塗抹時間。如需詳細資訊，請參閱閏秒。

您的執行個體可以存取本機 Amazon Time Sync Service，如下所示：

- 透過NTP下列 IP 位址端點：
  - IPv4: `169.254.169.123`
  - IPv6: `fd00:ec2::123` (僅可由 AWS Nitro 系統上建置的執行個體存取。)
- （僅限 Linux）通過直接PTP連接連接到本地PTP硬件時鐘：
  - `PHC0`

Amazon Linux AMIs AMIs、Windows 和大多數合作夥伴將您的執行個體AMIs設定為預設使用NTPIPv4端點。這是大多數客戶工作負載的建議設定。從這些執行個體啟動的執行個體不需要進一步的設定，AMIs除非您想要使用IPv6端點或直接連線至PTP硬體時鐘。

NTP和PTP連線不需要變更任何VPC設定，而且您的執行個體不需要存取網際網路。

> **Note**
>
> 只有 Linux 執行個體可以使用直接PTP連線來連線到本機PTP硬體時鐘。Windows 執行個體用NTP來連線到本機PTP硬體時鐘。

主題

- Connect 到 Amazon 時間同步服務的IPv4端點

- Connect 到 Amazon 時間同步服務的IPv6端點
- Connect 到PTP硬件時鐘

Connect 到 Amazon 時間同步服務的IPv4端點

本節說明如何將執行個體設定為透過IPv4端點使用本機 Amazon 時間同步服務。

請遵循於執行個體作業系統的指示。

Linux

AL2023 和最新版本的 Amazon Linux 2 和 Amazon Linux 設定AMIs為預設使用 Amazon 時間同步服務IPv4端點。從這些執行個體啟動的執行個體不需要進一步AMIs的組態，您可以略過下列程序。

如果您使用的是AMI預設情況下未設定 Amazon 時間同步服務，請使用下列其中一個程序，使用用`chrony`戶端在執行個體上設定 Amazon 時間同步服務。須編輯 `chrony` 組態檔，新增 Amazon Time Sync Service 的伺服器項目。

請遵循於執行個體作業系統的指示。

Amazon Linux

使用時間連接到 Amazon Linux 上的 Amazon 時間同步服務的IPv4端點

1. Connect 至執行個體並解除安裝NTP服務。

```
[ec2-user ~]$ sudo yum erase 'ntp*'
```

2. 安裝 `chrony` 套裝服務。

```
[ec2-user ~]$ sudo yum install chrony
```

3. 使用文字編輯器 (例如 `/etc/chrony.conf` 或 vim) 開啟 nano 檔案。確認檔案是否包含下列程式碼行：

```
server 169.254.169.123 prefer iburst minpoll 4 maxpoll 4
```

如果該行存在，則表示 Amazon 時間同步服務已設定為使用 Amazon 時間同步服務的IPv4端點，您可以繼續下一步。如果沒有，請將該行程式碼新增在檔案中已有之任何其他 `server` 或 `pool` 陳述式之後，然後儲存您的變更。

4. 重新啟動 chrony 協助程式 (chronyd)。

```
[ec2-user ~]$ sudo service chronyd restart
```

```
Starting chronyd:                                          [  OK  ]
```

> **Note**
>
> 在RHEL和 CentOS（最多版本 6）上，服務名稱chrony代替。chronyd

5. 使用 chkconfig 命令設定 chronyd 在每次系統開機時啟動。

```
[ec2-user ~]$ sudo chkconfig chronyd on
```

6. 確認chrony正在使用169.254.169.123IPv4端點同步時間。

```
[ec2-user ~]$ chronyc sources -v
```

```
210 Number of sources = 7

         .-- Source mode  '^' = server, '=' = peer, '#' = local clock.
        / .- Source state '*' = current synced, '+' = combined , '-' = not
 combined,
       | /   '?' = unreachable, 'x' = time may be in error, '~' = time too
 variable.
       ||                                                 .- xxxx [ yyyy ] +/-
 zzzz
       ||      Reachability register (octal) -.           |  xxxx = adjusted
 offset,
       ||      Log2(Polling interval) --.      |          |  yyyy = measured
 offset,
       ||                                \     |          |  zzzz = estimated
 error.
       ||                                 |    |           \
       MS Name/IP address         Stratum Poll Reach LastRx Last sample


 ===============================================================================
       ^* 169.254.169.123               3   6    17    43    -30us[ -226us] +/-
  287us
```

```
        ^- ec2-12-34-231-12.eu-west>     2   6    17    43   -388us[ -388us] +/-
   11ms
        ^- tshirt.heanet.ie              1   6    17    44   +178us[  +25us] +/-
 1959us
        ^? tbag.heanet.ie                0   6     0     -     +0ns[   +0ns] +/-
    0ns
        ^? bray.walcz.net                0   6     0     -     +0ns[   +0ns] +/-
    0ns
        ^? 2a05:d018:c43:e312:ce77:>     0   6     0     -     +0ns[   +0ns] +/-
    0ns
        ^? 2a05:d018:dab:2701:b70:b>     0   6     0     -     +0ns[   +0ns] +/-
    0ns
```

在傳回的輸出中，^* 表示偏好的時間來源。

7. 確認時間同步指標是否由 chrony 回報。

```
[ec2-user ~]$ chronyc tracking
```

```
Reference ID    : A9FEA97B (169.254.169.123)
        Stratum         : 4
        Ref time (UTC)  : Wed Nov 22 13:18:34 2017
        System time     : 0.000000626 seconds slow of NTP time
        Last offset     : +0.002852759 seconds
        RMS offset      : 0.002852759 seconds
        Frequency       : 1.187 ppm fast
        Residual freq   : +0.020 ppm
        Skew            : 24.388 ppm
        Root delay      : 0.000504752 seconds
        Root dispersion : 0.001112565 seconds
        Update interval : 64.4 seconds
        Leap status     : Normal
```

Ubuntu

使用時間連接到 Ubuntu 上的 Amazon 時間同步服務的IPv4端點

1. 連線至您的執行個體，然後使用 apt 來安裝 chrony 套件。

```
ubuntu:~$ sudo apt install chrony
```

> **Note**
>
> 如有必要，請執行 `sudo apt update` 先更新您的執行個體。

2. 使用文字編輯器 (例如 `/etc/chrony/chrony.conf` 或 vim) 開啟 nano 檔案。將下列程式碼新增在檔案中已有之任何其他 `server` 或 `pool` 的陳述式之前，然後儲存您的變更：

```
server 169.254.169.123 prefer iburst minpoll 4 maxpoll 4
```

3. 重新啟動 `chrony` 服務。

```
ubuntu:~$ sudo /etc/init.d/chrony restart
```

```
Restarting chrony (via systemctl): chrony.service.
```

4. 確認chrony正在使用169.254.169.123IPv4端點同步時間。

```
ubuntu:~$ chronyc sources -v
```

```
210 Number of sources = 7

              .-- Source mode  '^' = server, '=' = peer, '#' = local clock.
             / .- Source state '*' = current synced, '+' = combined , '-' = not
 combined,
            | /   '?' = unreachable, 'x' = time may be in error, '~' = time too
 variable.
            ||                                                 .- xxxx [ yyyy ]
 +/- zzzz
            ||      Reachability register (octal) -.           |  xxxx =
 adjusted offset,
            ||      Log2(Polling interval) --.      |          |  yyyy =
 measured offset,
            ||                                \     |          |  zzzz =
 estimated error.
            ||                                 |    |           \
            MS Name/IP address         Stratum Poll Reach LastRx Last sample

 ===============================================================================
            ^* 169.254.169.123               3   6    17    12    +15us[  +57us]
 +/-  320us
```

```
            ^- tbag.heanet.ie               1   6    17    13  -3488us[-3446us]
 +/- 1779us
            ^- ec2-12-34-231-12.eu-west-    2   6    17    13   +893us[ +935us]
 +/- 7710us
            ^? 2a05:d018:c43:e312:ce77:6    0   6     0   10y     +0ns[   +0ns]
 +/-    0ns
            ^? 2a05:d018:d34:9000:d8c6:5    0   6     0   10y     +0ns[   +0ns]
 +/-    0ns
            ^? tshirt.heanet.ie             0   6     0   10y     +0ns[   +0ns]
 +/-    0ns
            ^? bray.walcz.net               0   6     0   10y     +0ns[   +0ns]
 +/-    0ns
```

在傳回的輸出中，以 `^*` 開頭的行表示偏好的時間來源。

5. 確認時間同步指標是否由 `chrony` 回報。

```
ubuntu:~$ chronyc tracking
```

```
Reference ID    : 169.254.169.123 (169.254.169.123)
            Stratum         : 4
            Ref time (UTC)  : Wed Nov 29 07:41:57 2017
            System time     : 0.000000011 seconds slow of NTP time
            Last offset     : +0.000041659 seconds
            RMS offset      : 0.000041659 seconds
            Frequency       : 10.141 ppm slow
            Residual freq   : +7.557 ppm
            Skew            : 2.329 ppm
            Root delay      : 0.000544 seconds
            Root dispersion : 0.000631 seconds
            Update interval : 2.0 seconds
            Leap status     : Normal
```

SUSE Linux

從 SUSE Linux 企業伺服器 15 開始，`chrony`是的預設實作NTP。

使用時間連接到 SUSE Linux 上的 Amazon 時間同步服務的IPv4端點

1. 使用文字編輯器 (例如 `/etc/chrony.conf` 或 vim) 開啟 nano 檔案。
2. 確認檔案是否包含下列程式碼行：

```
server 169.254.169.123 prefer iburst minpoll 4 maxpoll 4
```

如果此程式碼不存在，請新增它。

3. 將任何其他伺服器或集區的程式碼行變更為註解。
4. 開啟 yast 並啟用 chrony 服務。

Windows

從 2018 年 8 月發行版本開始，Windows 預設會AMIs使用 Amazon 時間同步服務。從這些執行個體啟動的執行個體不需要進一步AMIs的組態，您可以略過下列程序。

如果您使用的是AMI預設情況下未設定 Amazon 時間同步服務的，請先驗證您目前的NTP組態。如果您的執行個體已在使用 Amazon 時間同步服務的IPv4端點，則無需進一步設定。如果您的執行個體未使用 Amazon 時間同步服務，請完成變更NTP伺服器以使用 Amazon 時間同步服務的程序。

驗證組NTP態

1. 從執行個體中，開啟命令提示視窗。
2. 輸入下列命令以取得目前的NTP組態：

```
w32tm /query /configuration
```

此命令會傳回 Windows 執行個體目前的組態設定，並顯示是否已連接至 Amazon Time Sync Service。

3. (選用) 輸入下列命令，以取得目前組態的狀態：

```
w32tm /query /status
```

這個命令會傳回執行個體上次與NTP伺服器同步的時間，以及輪詢間隔等資訊。

若要變更NTP伺服器以使用 Amazon 時間同步服務

1. 從命令提示視窗，執行下列命令：

```
w32tm /config /manualpeerlist:169.254.169.123 /syncfromflags:manual /update
```

2. 使用下列命令，確認新設定：

```
w32tm /query /configuration
```

在傳回的輸出中，確認是否`NtpServer`顯示`169.254.169.123`IPv4端點。

Amazon 視窗的預設網路時間通訊協定 (NTP) 設定 AMIs

Amazon 機器映像 (AMIs) 通常會遵守 out-of-the-box預設值，除非需要變更才能在EC2基礎設施上運作。下列設定已確認在虛擬化環境中運作良好，以及可將任何時鐘漂移保持在一秒的準確度內：

- 更新間隔 — 控制時間服務調整系統時間以達到精確度的頻率。 AWS 設定每兩分鐘執行一次的更新間隔。
- NTP伺服器 — 從 2018 年 8 月發行版本開始，依預設AMIs使用 Amazon 時間同步服務。 AWS 區域您可以從 169.254.169. IPv4 123 端點上的任何一個端點存取此次服務。此外，0x9 標記表示時間服務正作為用戶端，並使用 `SpecialPollInterval` 來判定以設定的時間伺服器進行查入的頻率。
- 類型 — "NTP" 表示服務充當獨立NTP用戶端，而不是做為網域的一部分。
- 啟用和 InputProvider — 計時服務已啟用，並提供作業系統的時間。
- 特殊輪詢間隔 — 每 900 秒 (15 分鐘) 對設定的NTP伺服器進行檢查。

登錄檔路徑	金鑰名稱	資料
HKLM：\ 系統\\ 服務Current ControlSet\ w32 時間\ Config	UpdateInterval	120
HKLM：\ 系統\\ 服務Current ControlSet\ w32 時間\ 參數	NtpServer	169.254.169.123,0x9
HKLM：\ 系統\\ 服務Current ControlSet\ w32 時間\ 參數	Type	NTP
HKLM：\ 系統\Curren tControlSet\ 服務\ w32 時間\\ TimeProviders NtpClient	已啟用	1

登錄檔路徑	金鑰名稱	資料
HKLM：\ 系統\Curren tControlSet\ 服務\ w32 時間\\ TimeProviders NtpClient	InputProvider	1
HKLM：\ 系統\Curren tControlSet\ 服務\ w32 時間\\ TimeProviders NtpClient	SpecialPollInterval	900

Connect 到 Amazon 時間同步服務的IPv6端點

本節說明如果您將執行個體設定為透過IPv6端點使用本機 Amazon 時間同步服務，中所述的步驟有何Connect 到 Amazon 時間同步服務的IPv4端點不同。它不會說明整個 Amazon Time Sync Service 組態程序。

IPv6端點只能在 AWS Nitro 系統上建置的執行個體上存取。

> **Note**
> 我們不建議同時使用IPv4和IPv6端點項目。IPv4和IPv6NTP封包來自執行個體的相同本機伺服器。不需要同時設定IPv4和IPv6端點，也不會改善執行個體上的時間準確性。

請遵循於執行個體作業系統的指示。

Linux

根據您使用的 Linux 發行版，當您到達編輯時間 .conf 檔案的步驟時，您將使用 Amazon 時間同步服務 (`fd00:ec2::123`) 的IPv6端點，而不是IPv4端點 ()：`169.254.169.123`

```
server fd00:ec2::123 prefer iburst minpoll 4 maxpoll 4
```

儲存檔案並確認`chrony`是否正在使用`fd00:ec2::123`IPv6端點同步處理時間：

```
[ec2-user ~]$ chronyc sources -v
```

在輸出中，如果您看到`fd00:ec2::123`IPv6端點，表示組態已完成。

Windows

當您到達變更NTP伺服器以使用 Amazon 時間同步服務的步驟時，您將使用 Amazon 時間同步服務 (`fd00:ec2::123`) 的IPv6端點，而不是IPv4端點 (`169.254.169.123`)：

```
w32tm /config /manualpeerlist:fd00:ec2::123 /syncfromflags:manual /update
```

確認您的新設定正在使用`fd00:ec2::123`IPv6端點同步處理時間：

```
w32tm /query /configuration
```

在輸出中，確認是否`NtpServer`顯示`fd00:ec2::123`IPv6端點。

Connect 到PTP硬件時鐘

PTP硬體時脈是 AWS Nitro System 的一部分，因此無需使用任何客戶資源，即可在支援的裸機和虛擬化EC2執行個體上直接存取。

PTP硬體時鐘的NTP端點與一般 Amazon 時間同步服務的端點相同。如果您的執行個PTP體有硬體時鐘，且您已設定NTP連線 (IPv4或IPv6端點)，您的執行個體時間就會自動從PTP硬體時脈取得NTP。

對於 Linux 實例，您可以配置直接PTP連接，這將為您提供比NTP連接更準確的時間。Windows 執行個體僅支援與PTP硬體時鐘的NTP連線。

要求

當符合下列需求時，即可在執行個體上使用PTP硬體時鐘：

- 支援 AWS 區域：美國東部 (維吉尼亞北部) 和亞太區域 (東京)
- 支援的執行個體系列：
  - 一般用途：兆 7 安培，兆 7 克，兆平方米，M7
  - 計算最佳化：C7 安培、C7gD、C7 i
  - 最佳化記憶體：7 安培、R7 克、R7 公升
- (僅限 Linux) 在支援的作業系統上安裝的ENA驅動程式版本 2.10.0 或更新版本。如需有關支援作業系統的詳細資訊，請參閱的驅動程式先決條件GitHub。

(僅限 Linux) 設定與PTP硬體時鐘的直接PTP連線

本節說明如何將 Linux 執行個體設定為透過直接PTP連線透過PTP硬體時鐘使用本機 Amazon 時間同步服務。它需要在`chrony`配置文件中添加PTP硬件時鐘的服務器條目。

設定與PTP硬體時鐘的直接PTP連線 (僅限 Linux 執行個體)

1. Connect 至您的 Linux 執行個體，然後執行下列動作：

   a. 安裝彈性網路介面卡 (ENA) 2.10.0 版或更新版本的 Linux 核心驅動程式。

   b. 啟用PTP硬體時鐘。

   如需安裝指示，請參閱上的彈性網路介面卡 (ENA) 系列的 Linux 核心驅動程式GitHub。

2. 確認 `/dev/ptp0` 裝置顯示在執行個體上。

```
[ec2-user ~]$ ls /dev/ptp0
```

   預期的輸出如下：如果`/dev/ptp0`不在輸出中，則表示ENA驅動程式未正確安裝。請參閱此程序中的步驟 1 以安裝驅動程式。

```
/dev/ptp0
```

3. 使用文字編輯器編輯 `/etc/chrony.conf`，然後在檔案中的任何位置新增以下行。

```
refclock PHC /dev/ptp0 poll 0 delay 0.000010 prefer
```

4. 重新啟動計時。

```
[ec2-user ~]$ sudo systemctl restart chronyd
```

5. 確認 chrony 正在使用PTP硬體時鐘來同步此執行個體上的時間。

```
[ec2-user ~]$ chronyc sources
```

   預期的輸出結果

```
MS Name/IP address         Stratum Poll Reach LastRx Last sample
===============================================================================
#* PHC0                          0   0    377    1   +2ns[ +1ns] +/-   5031ns
```

   在傳回的輸出中，`*`指出偏好的時間來源。 `PHC0`對應於PTP硬件時鐘。重新啟動 chrony 之後，可能需要等待幾秒鐘，星號才會出現。

在EC2執行個體或任何連接網際網路的裝置上設定時間參考，以使用公用的 Amazon 時間同步服務

您可以將執行個體或任何連接網際網路的裝置 (例如本機電腦或內部部署伺服器) 設定為使用可透過網際網路從 `time.aws.com` 存取的公有 Amazon Time Sync Service。您可以使用公有的 Amazon 時間同步服務做為本機 Amazon 時間同步服務的備份，以及將資源連接 AWS 到 Amazon 時間同步服務以外的資源。

> **Note**
>
> 為了獲得最佳效能，我們建議您在執行個體上使用本機 Amazon 時間同步服務，並且僅使用公有 Amazon 時間同步服務做為備份。

使用執行個體或裝置作業系統的指示。

Linux

若要將 Linux 執行個體或裝置設定為使用公有 Amazon Time Sync Service，請使用 chrony 或 ntpd

1. 使用文字編輯器編輯 `/etc/chrony.conf` (若使用 chrony) 或 `/etc/ntp.conf` (若使用 ntpd)，如下所示：

   a. 若要防止執行個體或裝置嘗試混合已消彌和未消彌的伺服器，請刪除以 `server` 開頭的行或變成註解，但連接本機 Amazon Time Sync Service 的任何現有連線除外。

   > **Important**
   >
   > 如果您要將EC2執行個體設定為連線到公用 Amazon 時間同步服務，請勿移除以下行，這一行將您的執行個體設定為連線到本機 Amazon 時間同步服務。本機 Amazon Time Sync Service 是更直接的連線，可提供更高的時鐘準確度。公有 Amazon Time Sync Service 只能當成備案。
   >
   > ```
   > server 169.254.169.123 prefer iburst minpoll 4 maxpoll 4
   > ```

   b. 新增以下行以連接至公有 Amazon Time Sync Service。

   ```
   pool time.aws.com iburst
   ```

2. 使用以下其中一個命令重新啟動常駐程式。

   - chrony

   ```
   sudo service chronyd force-reload
   ```

   - ntpd

   ```
   sudo service ntp reload
   ```

macOS

將您的 macOS 執行個體或裝置設定為使用公有 Amazon Time Sync Service

1. 開啟 System Preferences (系統偏好設定)。
2. 選擇 Date & Time (日期與時間)，然後選擇 Date & Time (日期與時間) 標籤。
3. 若要進行變更，請選擇鎖定圖示，然後在出現提示時輸入您的密碼。
4. 對於 Set date and time automatically (自動設定日期與時間)，輸入 **`time.aws.com`**。

Windows

將您的 Windows 執行個體或裝置設定為使用公有 Amazon Time Sync Service

1. 開啟 Control Panel (主控台)。
2. 選擇 Date and Time (日期與時間) 圖示。
3. 選擇 Internet Time (網際網路時間) 標籤。如果您的 PC 是網域的一部分，就無法使用此索引標籤。在這種情況下，系統會與網域控制器同步時間。您可以將控制器設定為使用公有 Amazon Time Sync Service。
4. 選擇 Change settings (變更設定)。
5. 選取 Synchronize with an Internet time server (與網際網路時間伺服器同步) 的核取方塊。
6. 在 Server (伺服器) 旁，輸入 **`time.aws.com`**。

將您的 Windows Server 執行個體或裝置設定為使用公有 Amazon Time Sync Service

- 遵照 Microsoft 的指示，更新您的登錄檔。

比較 Linux 執行個體的時間戳記

如果您使用的是 Amazon 時間同步服務，則可以將 Amazon EC2 Linux 執行個體上的時間戳記與比較， ClockBound 以判斷事件的真實時間。 ClockBound 測量EC2實例的時鐘準確性，並允許您檢查給定的時間戳是否處於過去還是 future 與實例的當前時鐘相關。這項資訊對於判斷執行個體間事件和交易的順序和一致性非常重要，而不受每個EC2執行個體的地理位置影響。

ClockBound 是一個開源守護進程和庫。要進一步了解 ClockBound，包括安裝說明，請參閱 ClockBound（詳見）GitHub。

ClockBound 僅支援 Linux 執行個體。

如果您使用直接PTP連接到PTP硬件時鐘，則您的時間守護進程（例如chrony）將低估時鐘錯誤綁定。這是因為PTP硬件時鐘沒有像這樣傳遞正確的chrony錯誤綁定信息。NTP因此，您的時鐘同步處理程式會假設時脈準確無誤UTC，因此具有錯誤繫結0。為了測量完整的錯誤綁定，Nitro 系統會計算PTP硬件時鐘的錯誤綁定，並通過ENA驅動程序`sysfs`文件系統將其提供給您的EC2實例。您可以直接將其作為一個值讀取，以納秒為單位。

檢索綁定的PTP硬件時鐘錯誤

1. 首先通過使用以下命令之一獲取PTP硬件時鐘設備的正確位置。根據AMI用來啟動執行個體的不同，命令中的路徑會有所不同。

   - 針對 Amazon Linux 2：

     ```
     cat /sys/class/net/eth0/device/uevent | grep PCI_SLOT_NAME
     ```

   - 對於 Amazon

     ```
     cat /sys/class/net/ens5/device/uevent | grep PCI_SLOT_NAME
     ```

   輸出是PCI插槽名稱，這是PTP硬件時鐘設備的位置。在此範例中，位置為`0000:00:03.0`。

   ```
   PCI_SLOT_NAME=0000:00:03.0
   ```

2. 要檢索綁定的PTP硬件時鐘錯誤，請運行以下命令。包括上一個步驟的PCI插槽名稱。

   ```
   cat /sys/bus/pci/devices/0000:00:03.0/phc_error_bound
   ```

輸出是PTP硬件時鐘的時鐘錯誤綁定，以納秒為單位。

要計算在使用直接PTP連接到PTP硬件時鐘時，在特定時間點綁定的正確時鐘錯誤，您必須添加從chrony輪詢PTP硬件 ClockBound 時鐘chrony或當時綁定的時鐘錯誤。有關測量和監控時鐘準確性的詳細資訊，請參閱使用 Amazon 時間同步服務和 Amazon 管理 Amazon EC2 執行個體時鐘準確性 CloudWatch — 第 1 部分。

變更執行個體的時區

Amazon EC2 執行個體預設會設定為 UTC (協調世界時間) 時區。您可以將執行個體上的時間變更為當地時區或網路中的另一個時區。

請遵循於執行個體作業系統的指示。

Linux

> **Important**
>
> 此資訊適用於 Amazon Linux。如需其他分發的詳細資訊，請參閱其特定文件。

若要變更 AL2 023 或 Amazon Linux 2 執行個體上的時區

1. 檢視系統目前的時區設定。

```
[ec2-user ~]$ timedatectl
```

2. 列出可用的時區。

```
[ec2-user ~]$ timedatectl list-timezones
```

3. 設定選擇的時區。

```
[ec2-user ~]$ sudo timedatectl set-timezone America/Vancouver
```

4. (選用) 再次執行 timedatectl 命令確認目前的時區已更新為新的時區。

```
[ec2-user ~]$ timedatectl
```

Windows

變更 Windows 執行個體的時區

1. 從執行個體中，開啟命令提示視窗。
2. 找出執行個體使用的時區。若要取得時區清單，請使用下列命令：

```
tzutil /l
```

此命令會傳回所有可用時區的清單，格式如下：

```
display name
time zone ID
```

3. 找出要指派給執行個體的時區 ID。
4. 使用下列命令指派其他時區：

```
tzutil /s "Pacific Standard Time"
```

新的時區應該會立即生效。

> **Note**
>
> 您可以使用下列指令來指派UTC時區：
>
> ```
> tzutil /s "UTC"
> ```

為了防止您的時區在您為 Windows 伺服器設定之後變更

當您變更 Windows 執行個體上的時區時，必須確定在系統重新啟動之後仍然會持續保留時區。否則，執行個體重新啟動時，會恢復為使用UTC時間。您可以透過新增RealTimeIsUniversal登錄機碼來保留您的時區設定。依預設，此機碼會在所有最新一代的執行個體上設定。若要確認是否已設定 RealTimeIsUniversal 登錄機碼，請參閱下列程序中的步驟 4。如果未設定機碼，請從頭開始執行下列步驟。

若要設定 RealTimeIsUniversal 登錄機碼

1. 從執行個體中，開啟命令提示視窗。
2. 使用下列命令，新增登錄機碼：

```
reg add "HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\TimeZoneInformation" /
v RealTimeIsUniversal /d 1 /t REG_DWORD /f
```

3. 如果您使用的是在 2013 年 2 月 22 日之前建立的視窗伺服器 2008 AMI (而非視窗伺服器 2008 R2)，我們建議您更新至最新的 AWS 視窗AMI。如果您正在使用正在AMI運行的視窗服務器 2008 R2（不是視窗服務器 2008），您必須驗證 Microsoft 修補程序KB2922223已安裝。如果未安裝此 hotfix，我們建議您更新至最新的 AWS Windows AMI。
4. (選用) 使用下列命令，確認執行個體已順利儲存機碼：

```
reg query "HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control
\TimeZoneInformation" /s
```

此命令會傳回 TimeZoneInformation 登錄機碼的子機碼。您應該會在清單底端看到 RealTimeIsUniversal 機碼，類似以下：

```
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\TimeZoneInformation
    Bias                            REG_DWORD     0x1e0
    DaylightBias                    REG_DWORD     0xffffffc4
    DaylightName                    REG_SZ        @tzres.dll,-211
    DaylightStart                   REG_BINARY    00000300020002000000000000000000
    StandardBias                    REG_DWORD     0x0
    StandardName                    REG_SZ        @tzres.dll,-212
    StandardStart                   REG_BINARY    00000B0001000200000000000000000
    TimeZoneKeyName                 REG_SZ        Pacific Standard Time
    DynamicDaylightTimeDisabled     REG_DWORD     0x0
    ActiveTimeBias                  REG_DWORD     0x1a4
    RealTimeIsUniversal             REG_DWORD     0x1
```

管理EC2執行個體的裝置驅動程式

裝置驅動程式是與 Amazon EC2 執行個體虛擬化硬體通訊的軟體元件。為了防止系統錯誤，性能問題和其他意外行為，保留驅動程序非常重要 up-to-date。對於可能對系統性能產生強烈影響的驅動程序尤

其如此，具體取決於您的使用情況，例如網絡，圖形和存儲設備驅動程序。新的驅動程式版本可以包含瑕疵修正，或引入您可能想要針對目前執行中執行個體使用的擴充功能。

網絡驅動

Linux 發行版可以在內核中合併諸如彈性網絡適配器（ENA）或彈性結構適配器（EFA）之類的網絡功能。但是，在不同發行版中實現內核驅動程序功能的時間可能會有所不同。

ENA和 EFA Linux 核心驅動程式可從 Amazon 驅動程式 GitHub 儲存庫取得。如需詳細資訊和可用驅動程式的連結，請參閱上的 Amazon 驅動程式 GitHub。

如需ENA驅動程式的詳細資訊，請參閱在EC2執行個體ENA上啟用增強型聯網。如需有關EFA驅動程式的詳細資訊，請參閱本指南一適用於 Amazon 上的機器學習HPC工作負載的彈性光纖 EC2節中的入門主題。

若要在 Windows 執行個體上安裝或更新網路驅動程式，請參閱下列主題：

- 在視窗上安裝ENA驅動程式
- 安裝最新的 AWS 光伏驅動程式

  如需詳細資訊，請參閱Windows 執行個體的全虛擬驅動程式。

> **Note**
>
> EFA在視窗執行個體上不受支援。

顯示卡驅動

若要安裝或更新圖形驅動程式，請參閱下列主題：

- AMDEC2執行個體的驅動程式
- NVIDIAAmazon EC2 實例的驅動程序

儲存裝置驅動

若要安裝或更新儲存區驅動程式，請參閱下列主題：

- 對於 Linux 執行個體，請參閱 Amazon EBS 使用者指南中的安裝或升級NVMe驅動程式。
- 如需視窗執行個體，請參閱AWS NVMe驅動程式。

AMDEC2執行個體的驅動程式

已連接AMDGPU的執行個體 (例如 G4ad 執行個體) 必須安裝適當的AMD驅動程式。根據您的需求，您可以使用預先安裝AMI的驅動程式，或從 Amazon S3 下載驅動程式。

若要在已NVIDIAGPU連接的執行個體 (例如 G4dn 執行個體) 上安裝NVIDIA驅動程式，請參閱NVIDIA司機相反的說明。

內容

- AMD用於企業驅動程序的 Radeon Pro 軟件
- AMIs安裝了AMD驅動程序
- AMD驅動下載

AMD用於企業驅動程序的 Radeon Pro 軟件

適用於企業驅動程序的 AMD Radeon Pro 軟件旨在為專業級圖形用例提供支持。使用驅動程式，您可以為每個執行個體配置兩個 4K 顯示器GPU。

支援 APIs

- OpenGL、OpenCL
- Vulkan
- AMD先進的媒體框架
- 視訊加速 API
- DirectX 9 和更新版本
- Microsoft 硬體媒體基礎轉換

AMIs安裝了AMD驅動程序

AWS 提供不同的 Amazon 機器映像（AMIs）隨附的AMD驅動程序安裝。提供AMD驅動程式的開放式 Marketplace 產品。

AMD驅動下載

如果您沒有使用已安裝AMD驅動程式AMI的驅動程式，您可以下載AMD驅動程式並將其安裝在您的執行個體上。只有下列作業系統版本支援AMD驅動程式：

- Amazon Linux 2 與內核版本 4.14

> **Note**
> AMD驅動程式版本 amdgpu-pro 20.20-1184451 和更新的驅動程式版本需要核心版本 5.15 或更新版本。

- Windows Server 2016
- Windows Server 2019

這些下載僅供 AWS 客戶使用。通過下載，您同意使用下載的軟件僅用AMIs於開發與 AMD Radeon Pro V520 硬件一起使用。安裝軟體時，您必須遵守「軟AMD體使用者授權合約」的條款。

在您的 Linux 執行個體上安裝AMD驅動程式

1. 連線至您的 Linux 執行個體。
2. AWS CLI 在您的 Linux 執行個體上安裝並設定預設認證。如需詳細資訊，請參閱《AWS Command Line Interface 使用者指南》中的安裝 AWS CLI。

   > **Important**
   > 您的使用者或角色必須擁有包含 AmazonS3 ReadOnlyAccess 政策的授與權限。如需詳細資訊，請參閱 AWSAmazon 簡單儲存服務使用者指南ReadOnlyAccess中的受管政策：AmazonS3。

3. 如果尚未安裝 gcc 和 make，請進行安裝。

   ```
   $ sudo yum install gcc make
   ```

4. 更新套件快取，並取得執行個體的套件更新。

   - 針對 Amazon Linux 2：

     ```
     $ sudo amazon-linux-extras install epel -y
     ```

```
$ sudo yum update -y
```

- 對於

```
$ wget https://repo.radeon.com/.preview/a0e4ef1dffbc95b4abb54e891f265e61/amdgpu-install/5.5.02.05.2/ubuntu/jammy/amdgpu-install_5.5.02.05.50502-1_all.deb
$ sudo apt install ./amdgpu-install_5.5.02.05.50502-1_all.deb
$ sudo sed -i 's#repo.radeon.com#&/.preview/a0e4ef1dffbc95b4abb54e891f265e61#' /etc/apt/sources.list.d/{amdgpu.list,rocm.list,amdgpu-proprietary.list}
```

- 對於其他版本：

```
$ sudo dpkg --add-architecture i386
$ sudo apt-get update -y && sudo apt upgrade -y
```

- CentOS 系統：

```
$ sudo yum install epel-release -y
$ sudo yum update -y
```

5. 重新啟動執行個體。

```
$ sudo reboot
```

6. 重新啟動後重新連線至執行個體。
7. 下載最新的AMD驅動程式。

> **Note**
>
> 略過此步驟

```
$ aws s3 cp --recursive s3://ec2-amd-linux-drivers/latest/ .
```

8. 解壓縮檔案。

- Amazon Linux 2 和 CentOS：

```
$ tar -xf amdgpu-pro-*rhel*.tar.xz
```

- Ubuntu 系統：

> **Note**
>
> 略過此步驟

```
$ tar -xf amdgpu-pro*ubuntu*.xz
```

9. 變更為解壓縮驅動程式的資料夾。
10. 新增缺少的用於驅動程式安裝的模組。
    - Amazon Linux 2 和 CentOS：

      跳過此步驟。
    - Ubuntu 系統：

      > **Note**
      >
      > 略過此步驟

      ```
      $ sudo apt install linux-modules-extra-$(uname -r) -y
      ```
11. 執行自主安裝指令碼以安裝完整的圖形堆疊。
    - 對於

      ```
      $ sudo amdgpu-install --usecase=workstation --vulkan=pro --opencl=rocr,legacy -y
      ```
    - 對於 Amazon Linux 2 和 CentOS 和其他版本：

      ```
      $ ./amdgpu-pro-install -y --opencl=pal,legacy
      ```
12. 重新啟動執行個體。

    ```
    $ sudo reboot
    ```
13. 確認驅動程式可正常運作。

```
$ dmesg | grep amdgpu
```

回應應如下所示：

```
Initialized amdgpu
```

在 Windows 執行個體上安裝AMD驅動程式

1. Connect 至您的 Windows 執行個體並開啟 PowerShell 視窗。
2. 設定 Windows 執行個體 AWS Tools for Windows PowerShell 上的預設認證。如需詳細資訊，請參閱 AWS Tools for Windows PowerShell 使用者指南中 AWS Tools for Windows PowerShell入門。

> **Important**
>
> 您的使用者或角色必須擁有包含 AmazonS3 ReadOnlyAccess 政策的授與權限。如需詳細資訊，請參閱 AWSAmazon 簡單儲存服務使用者指南ReadOnlyAccess中的受管政策：AmazonS3。

3. 根據您的 Windows 版本設置 key prefix：
   - 視窗 10 和視窗 11

     ```
     $KeyPrefix = "latest/AMD_GPU_WINDOWS10"
     ```

   - Windows Server 2016

     ```
     $KeyPrefix = "archives"
     ```

   - Windows Server 2019

     ```
     $KeyPrefix = "latest/AMD_GPU_WINDOWS_2K19 # use "archives" for Windows Server
      2016"
     ```

   - Windows Server 2022

     ```
     $KeyPrefix = "latest/AMD_GPU_WINDOWS_2K22"
     ```

4. 使用以下PowerShell 命令將驅動程式從 Amazon S3 下載到您的桌面。

```
$Bucket = "ec2-amd-windows-drivers"
$LocalPath = "$home\Desktop\AMD"
$Objects = Get-S3Object -BucketName $Bucket -KeyPrefix $KeyPrefix -Region us-east-1
foreach ($Object in $Objects) {
$LocalFileName = $Object.Key
if ($LocalFileName -ne '' -and $Object.Size -ne 0) {
    $LocalFilePath = Join-Path $LocalPath $LocalFileName
    Copy-S3Object -BucketName $Bucket -Key $Object.Key -LocalFile $LocalFilePath -
Region us-east-1
    }
}
```

5. 解壓縮下載的驅動程式檔案，並使用下列PowerShell 指令執行安裝程式。

```
Expand-Archive $LocalFilePath -DestinationPath "$home\Desktop\AMD\$KeyPrefix" -
Verbose
```

現在，請檢查新目錄的內容。可以使用`Get-ChildItem` PowerShell 指令擷取目錄名稱。

```
Get-ChildItem "$home\Desktop\AMD\$KeyPrefix"
```

輸出格式應類似以下內容：

```
Directory: C:\Users\Administrator\Desktop\AMD\latest

Mode                LastWriteTime         Length Name
----                -------------         ------ ----
d-----       10/13/2021  12:52 AM                210414a-365562C-Retail_End_User.2
```

安裝驅動程式：

```
pnputil /add-driver $home\Desktop\AMD\$KeyPrefix\*.inf /install /subdirs
```

6. 依照指示安裝驅動程式，並視需要將執行個體重新開機。
7. 若要確認GPU是否正常運作，請檢查裝置管理員。您應該看到列為顯示適配器的「AMDRadeon Pro V520 M GPU x」。
8. 若要協助您充分利用高達 4K 解析度的四個顯示器，請設定高效能顯示通訊協定NICEDCV。

NVIDIAAmazon EC2 實例的驅動程序

連接NVIDIAGPU的執行個體 (例如 P3 或 G4dn 執行個體) 必須安裝適當的NVIDIA驅動程式。視執行個體類型而定，您可以下載公用NVIDIA驅動程式、從 Amazon S3 下載僅供 AWS 客戶使用的驅動程式，或AMI搭配預先安裝的驅動程式使用。

若要在已連接AMDGPU的執行個體 (例如 G4ad 執行個體) 上安裝AMD驅動程式，請參閱。AMD驅動若要安裝NVIDIA驅動程式，請參閱NVIDIA司機。

內容

- NVIDIA驅動程式類型
- 依執行個體類型分類的可用驅動程式
- 安裝選項
  - 選項 1：安AMIs裝NVIDIA驅動程式
  - 選項 2：公開NVIDIA驅動程式
  - 選項 3：GRID驅動程式 (G6、G6、G5、G4DN 和 G3 執行個體)
  - 選項 4：NVIDIA遊戲驅動程式 (G5 和 G4dn 執行個體)
- 安裝其他版本的 CUDA

NVIDIA驅動程式類型

以下是可與GPU基礎執行個體搭配使用的主要NVIDIA驅動程式類型。

Tesla 驅動程式

這些驅動程式主要用於運算工作負載，這些工作負載用GPUs於運算工作，例如機器學習的平行化浮點計算和高效能運算應用程式的快速傅立葉轉換。

GRID司機

這些驅動程式經過認證，可為轉譯 3D 模型或高解析度影片等內容的專業視覺化應用程式提供最佳效能。您可以設定GRID驅動程式以支援兩種模式。Quadro 虛擬工作站提供每GPU個 4K 顯示器的存取權。GRID vApps 提供RDSH應用程序託管功能。

遊戲驅動程式

這些驅動程式包含遊戲的最佳化功能，而且經常更新，可提供效能增強功能。它們支持每個單個 4K 顯示器GPU。

設定的模式

在 Windows 上，特斯拉驅動程序配置為在特斯拉計算集群（TCC）模式下運行。GRID和遊戲驅動程式設定為在 Windows 顯示驅動程式模型 (WDDM) 模式下執行。在TCC模式下，該卡專用於計算工作負載。在WDDM模式下，該卡同時支持計算和圖形工作負載。

NVIDIA控制面板

NVIDIA控制面板支持GRID和遊戲驅動程序。Tesla 驅動程式不支援此面板。

支援 APIs Tesla GRID、和遊戲驅動程式

- OpenCL、OpenGL 和 Vulkan
- NVIDIACUDA以及相關的程式庫 (例如銅DNN、TensorRT JPEG、NV 和 cu) BLAS
- NVENC用於視頻編碼和NVDEC視頻解碼
- 僅限視窗APIs：直接輸入、直接 2D、DirectX 視訊加速、DirectX 光線追蹤

依執行個體類型分類的可用驅動程式

下表摘要說明每個GPU執行個體類型支援的NVIDIA驅動程式。

執行個體類型	Tesla 驅動程式	GRID司機	遊戲驅動程式
G3	是	是	否
G4dn	是	是	是
G5	是	是	是
G5g	是 [1]	否	否
G6	是	是	否
G6e	是	是	否
GR6	是	是	否
P2	是	否	否
P3	是	否	否

執行個體類型	Tesla 驅動程式	GRID司機	遊戲驅動程式
P4d	是	否	否
P4de	是	否	否

[1] 此 Tesla 驅動程序還支持特定於ARM64平台的優化圖形應用程序

[2] AMIs 僅使用 Marketplace

安裝選項

使用下列其中一個選項取得GPU執行個體所需的NVIDIA驅動程式。

選項

- 選項 1：安AMIs裝NVIDIA驅動程式
- 選項 2：公開NVIDIA驅動程式
- 選項 3：GRID驅動程式 (G6、G6、G5、G4DN 和 G3 執行個體)
- 選項 4：NVIDIA遊戲驅動程式 (G5 和 G4dn 執行個體)

選項 1：安AMIs裝NVIDIA驅動程式

AWS 並NVIDIA提供安裝NVIDIA驅動程序附帶的不同 Amazon 機器映像（AMIs）。

- 具有 Tesla 驅動程式的 Marketplace 產品
- 提供GRID驅動程式的 Marketplace 產品
- 具有遊戲驅動程式的 Marketplace 產品

若要檢閱依賴您作業系統 (OS) 平台的考量事項，請選擇適用於您的AMI.

Linux

若要更新使用其中一個驅動程式安裝的驅動程式版本AMIs，您必須從執行個體解除安裝NVIDIA套件，以避免版本衝突。使用此命令解除安裝NVIDIA套件：

```
[ec2-user ~]$ sudo yum erase nvidia cuda
```

該CUDA工具包包具有對NVIDIA驅動程序的依賴關係。解除安裝NVIDIA套件會清除CUDA工具組。安裝NVIDIA驅動程式後，您必須重新安裝CUDA工具組。

Windows

如果您AMI使用其中一個 AWS Marketplace 供應項目建立自訂 Windows，則AMI必須是使用 Windows Sysprep 建立的標準化映像檔，以確保GRID驅動程式可正常運作。如需詳細資訊，請參閱EC2AMI使用視窗系統創建一個 Amazon。

選項 2：公開NVIDIA驅動程式

提供的選項 AWS 隨驅動程序必要的許可證一起提供。或者，您可以安裝公用驅動程式並攜帶自己的授權。要安裝公共驅動程序，請按照此處所述從NVIDIA站點下載它。

或者，您可以使用提供的選項， AWS 而不是公共驅動程序。若要在 P3 執行個體上使用GRID驅動程式，請使用選項 1 中所述的。 AWS Marketplace AMIs若要在 G6、G6e、G6、G5、G4dn 或 G3 執行個體上使用GRID驅動程式，請使用選項 1 中所述的，或依照中所 AWS Marketplace AMIs述安裝由提供的驅動程式。NVIDIA AWS 選項 3：GRID驅動程式 (G6、G6、G5、G4DN 和 G3 執行個體)

下載公用NVIDIA驅動程式

登入您的執行個體，然後從 http://www.nvidia.com/Download/Find.aspx 下載適用於執行個體類型的 64 位元NVIDIA驅動程式。對於產品類型、產品系列及產品，請使用下表中的選項。

執行個體	產品類型	產品系列	產品
G3	Tesla	M-Class	M60
G4dn	Tesla	T 系列	T4
G5	Tesla	A 系列	A10
G5G	Tesla	T 系列	NVIDIAT4G
三級賽	Tesla	L 型系列	L4
G6e	Tesla	L 型系列	L40
六	Tesla	L 型系列	L4
P2	Tesla	K 系列	K80

執行個體	產品類型	產品系列	產品
P3	Tesla	V 系列	V100
P4d	Tesla	A 系列	A100
P4de	Tesla	A 系列	A100
五年級	Tesla	H 系列	H100

[1] G5 執行個體需要 470.00 或更新版本的驅動程式。

[2] 個 G5G 執行個體需要驅動程式版本 470.82.01 或更新版本。操作系統是 Linux 系統.

[3] G6 和 Gr6 執行個體需要驅動程式版本 525.0 或更新版本。

[4] 個 G6e 執行個體需要驅動程式版本 535.0 或更新版本。

[5] 個 P5 執行個體需要 530 或更新版本的驅動程式。

若要在 Linux 作業系統上安裝NVIDIA驅動程式，請參閱《NVIDIA驅動程式安裝快速入門指南》。

若要在 Windows 上安裝NVIDIA驅動程式，請依照下列步驟執行：

1. 開啟您下載驅動程式的所在資料夾，並啟動安裝檔案。依照指示安裝驅動程式，並視需要將執行個體重新開機。
2. 使用 Device Manager 停用標有警告圖示的名為 Microsoft Basic Display Adapter 的顯示轉接器。安裝這些 Windows 功能：Media Foundation 和 Quality Windows Audio Video Experience (高品質 Windows 音訊/視訊體驗)。

> **Important**
>
> 請勿停用名為 Microsoft Remote Display Adapter 的顯示轉接器。如果停用 Microsoft Remote Display Adapter，您的連線可能會中斷，並在重新開機後會嘗試連線至執行個體可能會失敗。

3. 檢查裝置管理員以確認GPU是否正常運作。
4. 若要從中獲得最佳效能GPU，請完成中的最佳化步驟優化 Amazon EC2 執行個體上的GPU設定。

選項 3：GRID驅動程式 (G6、G6、G5、G4DN 和 G3 執行個體)

這些下載僅供 AWS 客戶使用。若要遵守NVIDIAGRID雲端使用者授權合約 (EULA) 所述 AWS 解決方案的要求，即表示您同意僅將下載的軟體用AMIs於開發 NVIDIA L4、NVIDIA A10G、Tesla T4 或 NVIDIA T NVIDIA esla M60 硬體。安裝軟體後，您將受到NVIDIAGRID雲端使用者授權合約的條款約束。如需有關您作業系統之NVIDIAGRID驅動程式版本的資訊，請參閱NVIDIA網站上的 NVIDIA® Virtual GPU (vGPU) 軟體文件。

考量事項

- G6 和 Gr6 執行個體需要 GRID 17.1 或更新版本。
- G5 執行個體需要 GRID 13.1 或更新版本 (或 GRID 12.4 或更新版本)。
- G3 執行個體需要 AWS 提供DNS解決方案，GRID授權才能運作。
- IMDSv2僅支援 14.0 或更新版本的NVIDIA驅動程式。
- 對於 Windows 執行個體，如果您從自訂視窗啟動執行個體AMI，則AMI必須是使用 Windows Sysprep 建立的標準化映像檔，以確保GRID驅動程式可正常運作。如需詳細資訊，請參閱EC2AMI使用視窗系統創建一個 Amazon。
- GRID17.0 及更新版本不支援視窗伺服器 2019。
- GRID14.2 及更新版本不支援視窗伺服器 2016。
- GRIDG3 執行個體不支援 17.0 及更新版本。

Amazon Linux 和 Amazon Linux 2

在執行個體上安裝NVIDIAGRID驅動程式

1. 連線至您的 Linux 執行個體。
2. AWS CLI 在您的 Linux 執行個體上安裝並設定預設認證。如需詳細資訊，請參閱《AWS Command Line Interface 使用者指南》中的安裝 AWS CLI。

   > **Important**
   >
   > 您的使用者或角色必須擁有包含 AmazonS3 ReadOnlyAccess 政策的授與權限。如需詳細資訊，請參閱 AWSAmazon 簡單儲存服務使用者指南ReadOnlyAccess中的受管政策：AmazonS3。

3. 如果尚未安裝 gcc 和 make，請進行安裝。

```
[ec2-user ~]$ sudo yum install gcc make
```

4. 更新套件快取，並取得執行個體的套件更新。

```
[ec2-user ~]$ sudo yum update -y
```

5. 將執行個體重新開機以載入最新的核心版本。

```
[ec2-user ~]$ sudo reboot
```

6. 重新開機後，請重新連線至您的執行個體。
7. 針對您目前執行的核心版本，安裝 gcc 編譯器與核心標頭套件。

```
[ec2-user ~]$ sudo yum install -y kernel-devel-$(uname -r)
```

8. 使用下列指令下載GRID驅動程式安裝公用程式：

```
[ec2-user ~]$ aws s3 cp --recursive s3://ec2-linux-nvidia-drivers/latest/ .
```

此值區中會儲存多個版本的GRID驅動程式。您可以使用下列命令查看所有可用版本。

```
[ec2-user ~]$ aws s3 ls --recursive s3://ec2-linux-nvidia-drivers/
```

9. 請使用以下命令新增許可來執行驅動程式安裝公用程式。

```
[ec2-user ~]$ chmod +x NVIDIA-Linux-x86_64*.run
```

10. 執行自我安裝指令碼，如下所示，以安裝您下載的GRID驅動程式。例如：

```
[ec2-user ~]$ sudo /bin/sh ./NVIDIA-Linux-x86_64*.run
```

> **Note**
>
> 如果您將 Amazon Linux 2 與核心版本 5.10 搭配使用，請使用下列指令來安裝GRID驅動程式。
>
> ```
> [ec2-user ~]$ sudo CC=/usr/bin/gcc10-cc ./NVIDIA-Linux-x86_64*.run
> ```

出現提示時，請接受授權合約，並依需求指定安裝選項 (您可以接受預設選項)。

11. 確認驅動程式可正常運作。下列指令的回應會列出安裝的NVIDIA驅動程式版本以及有關的詳細資訊GPUs。

```
[ec2-user ~]$ nvidia-smi -q | head
```

12. 如果您在 G4dn、G5 或 G5G 執行個GPU體上使用 14.x 或更新版本的 NVIDIA v 軟體，請使用下列命令停GSP用。有關更多信息，有關為什麼需要這樣做，請訪問NVIDIA的文檔。

```
[ec2-user ~]$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
[ec2-user ~]$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /
etc/modprobe.d/nvidia.conf
```

13. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

14. (選用) 視您的使用案例而定，您可能會完成下列選用步驟。如果您不需要這項功能，請勿完成這些步驟。

    a. 若要協助您充分利用四個高達 4K 解析度的顯示器，請設定高效能顯示通訊協定NICEDCV。

    b. NVIDIAQuadro 虛擬工作站模式預設為啟用狀態。若要啟動GRID虛擬應用程式的RDSH應用程式託管功能，請完成中的GRID虛擬應用程式在 Amazon EC2 GPU 執行個體上啟用NVIDIAGRID虛擬應用程式啟動步驟

CentOS 7 和 Red Hat Enterprise Linux 7

在執行個體上安裝NVIDIAGRID驅動程式

1. 連線至您的 Linux 執行個體。如果尚未安裝 gcc 和 make，請進行安裝。

```
[ec2-user ~]$ sudo yum install -y gcc make
```

2. 更新套件快取，並取得執行個體的套件更新。

```
[ec2-user ~]$ sudo yum update -y
```

3. 將執行個體重新開機以載入最新的核心版本。

```
[ec2-user ~]$ sudo reboot
```

4. 重新開機後，請重新連線至您的執行個體。
5. 為您目前執行的核心版本安裝核心標頭套件。

```
[ec2-user ~]$ sudo yum install -y kernel-devel-$(uname -r)
```

6. 停用NVIDIA顯示卡的nouveau開放原始碼驅動程式。

   a. 將 nouveau 新增到 /etc/modprobe.d/blacklist.conf 黑名單檔案。複製下列程式碼區塊並貼至終端機。

   ```
   [ec2-user ~]$ cat << EOF | sudo tee --append /etc/modprobe.d/blacklist.conf
   blacklist vga16fb
   blacklist nouveau
   blacklist rivafb
   blacklist nvidiafb
   blacklist rivatv
   EOF
   ```

   b. 編輯 /etc/default/grub 檔案並新增下行：

   ```
   GRUB_CMDLINE_LINUX="rdblacklist=nouveau"
   ```

   c. 重建 Grub 組態。

   ```
   [ec2-user ~]$ sudo grub2-mkconfig -o /boot/grub2/grub.cfg
   ```

7. 使用下列指令下載GRID驅動程式安裝公用程式：

```
[ec2-user ~]$ aws s3 cp --recursive s3://ec2-linux-nvidia-drivers/latest/ .
```

此值區中會儲存多個版本的GRID驅動程式。您可以使用下列命令查看所有可用版本。

```
[ec2-user ~]$ aws s3 ls --recursive s3://ec2-linux-nvidia-drivers/
```

8. 請使用以下命令新增許可來執行驅動程式安裝公用程式。

```
[ec2-user ~]$ chmod +x NVIDIA-Linux-x86_64*.run
```

9. 執行自我安裝指令碼，如下所示，以安裝您下載的GRID驅動程式。例如：

```
[ec2-user ~]$ sudo /bin/sh ./NVIDIA-Linux-x86_64*.run
```

出現提示時，請接受授權合約，並依需求指定安裝選項 (您可以接受預設選項)。

10. 確認驅動程式可正常運作。下列指令的回應會列出安裝的NVIDIA驅動程式版本以及有關的詳細資訊GPUs。

```
[ec2-user ~]$ nvidia-smi -q | head
```

11. 如果您在 G4dn、G5 或 G5G 執行個GPU體上使用 14.x 或更新版本的 NVIDIA v 軟體，請使用下列命令停GSP用。有關更多信息，有關為什麼需要這樣做，請訪問NVIDIA的文檔。

```
[ec2-user ~]$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
[ec2-user ~]$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /
etc/modprobe.d/nvidia.conf
```

12. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

13. (選用) 視您的使用案例而定，您可能會完成下列選用步驟。如果您不需要這項功能，請勿完成這些步驟。

    a. 若要協助您充分利用四個高達 4K 解析度的顯示器，請設定高效能顯示通訊協定NICEDCV。

    b. NVIDIAQuadro 虛擬工作站模式預設為啟用狀態。若要啟動GRID虛擬應用程式的RDSH應用程式託管功能，請完成中的GRID虛擬應用程式在 Amazon EC2 GPU 執行個體上啟用NVIDIAGRID虛擬應用程式啟動步驟

    c. 安裝GUI桌上型電腦/工作站套件。

```
[ec2-user ~]$ sudo yum groupinstall -y "Server with GUI"
```

CentOS Stream 8 和 Red Hat Enterprise Linux 8

在執行個體上安裝NVIDIAGRID驅動程式

1. 連線至您的 Linux 執行個體。如果尚未安裝 gcc 和 make，請進行安裝。

```
[ec2-user ~]$ sudo yum install -y gcc make
```

2. 更新套件快取，並取得執行個體的套件更新。

```
[ec2-user ~]$ sudo yum update -y
```

3. 將執行個體重新開機以載入最新的核心版本。

```
[ec2-user ~]$ sudo reboot
```

4. 重新開機後，請重新連線至您的執行個體。
5. 為您目前執行的核心版本安裝核心標頭套件。

```
[ec2-user ~]$ sudo dnf install -y elfutils-libelf-devel libglvnd-devel kernel-
devel-$(uname -r)
```

6. 使用下列指令下載GRID驅動程式安裝公用程式：

```
[ec2-user ~]$ aws s3 cp --recursive s3://ec2-linux-nvidia-drivers/latest/ .
```

此值區中會儲存多個版本的GRID驅動程式。您可以使用下列命令查看所有可用版本。

```
[ec2-user ~]$ aws s3 ls --recursive s3://ec2-linux-nvidia-drivers/
```

7. 請使用以下命令新增許可來執行驅動程式安裝公用程式。

```
[ec2-user ~]$ chmod +x NVIDIA-Linux-x86_64*.run
```

8. 執行自我安裝指令碼，如下所示，以安裝您下載的GRID驅動程式。例如：

```
[ec2-user ~]$ sudo /bin/sh ./NVIDIA-Linux-x86_64*.run
```

出現提示時，請接受授權合約，並依需求指定安裝選項 (您可以接受預設選項)。

9. 確認驅動程式可正常運作。下列指令的回應會列出安裝的NVIDIA驅動程式版本以及有關的詳細資訊GPUs。

```
[ec2-user ~]$ nvidia-smi -q | head
```

10. 如果您在 G4dn、G5 或 G5G 執行個GPU體上使用 14.x 或更新版本的 NVIDIA v 軟體，請使用下列命令停GSP用。有關更多信息，有關為什麼需要這樣做，請訪問NVIDIA的文檔。

```
[ec2-user ~]$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
[ec2-user ~]$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /
etc/modprobe.d/nvidia.conf
```

11. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

12. (選用) 視您的使用案例而定，您可能會完成下列選用步驟。如果您不需要這項功能，請勿完成這些步驟。

    a. 若要協助您充分利用四個高達 4K 解析度的顯示器，請設定高效能顯示通訊協定NICEDCV。

    b. NVIDIAQuadro 虛擬工作站模式預設為啟用狀態。若要啟動GRID虛擬應用程式的RDSH應用程式託管功能，請完成中的GRID虛擬應用程式在 Amazon EC2 GPU 執行個體上啟用NVIDIAGRID虛擬應用程式啟動步驟

    c. 安裝工GUI作站套件。

    ```
    [ec2-user ~]$ sudo dnf groupinstall -y workstation
    ```

Rocky Linux 8

若要在 Linux 執行個體上安裝NVIDIAGRID驅動程式

1. 連線至您的 Linux 執行個體。如果尚未安裝 gcc 和 make，請進行安裝。

```
[ec2-user ~]$ sudo yum install -y gcc make
```

2. 更新套件快取，並取得執行個體的套件更新。

```
[ec2-user ~]$ sudo yum update -y
```

3. 將執行個體重新開機以載入最新的核心版本。

```
[ec2-user ~]$ sudo reboot
```

4. 重新開機後，請重新連線至您的執行個體。
5. 為您目前執行的核心版本安裝核心標頭套件。

```
[ec2-user ~]$ sudo dnf install -y elfutils-libelf-devel libglvnd-devel kernel-
devel-$(uname -r)
```

6. 使用下列指令下載GRID驅動程式安裝公用程式：

```
[ec2-user ~]$ aws s3 cp --recursive s3://ec2-linux-nvidia-drivers/latest/ .
```

此值區中會儲存多個版本的GRID驅動程式。您可以使用下列命令查看所有可用版本。

```
[ec2-user ~]$ aws s3 ls --recursive s3://ec2-linux-nvidia-drivers/
```

7. 請使用以下命令新增許可來執行驅動程式安裝公用程式。

```
[ec2-user ~]$ chmod +x NVIDIA-Linux-x86_64*.run
```

8. 執行自我安裝指令碼，如下所示，以安裝您下載的GRID驅動程式。例如：

```
[ec2-user ~]$ sudo /bin/sh ./NVIDIA-Linux-x86_64*.run
```

出現提示時，請接受授權合約，並依需求指定安裝選項 (您可以接受預設選項)。

9. 確認驅動程式可正常運作。下列指令的回應會列出安裝的NVIDIA驅動程式版本以及有關的詳細資訊GPUs。

```
[ec2-user ~]$ nvidia-smi -q | head
```

10. 如果您在 G4dn、G5 或 G5G 執行個GPU體上使用 14.x 或更新版本的 NVIDIA v 軟體，請使用下列命令停GSP用。有關更多信息，有關為什麼需要這樣做，請訪問NVIDIA的文檔。

```
[ec2-user ~]$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
[ec2-user ~]$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /
etc/modprobe.d/nvidia.conf
```

11. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

12. (選用) 視您的使用案例而定，您可能會完成下列選用步驟。如果您不需要這項功能，請勿完成這些步驟。

    a. 若要協助您充分利用四個高達 4K 解析度的顯示器，請設定高效能顯示通訊協定NICEDCV。

    b. NVIDIAQuadro 虛擬工作站模式預設為啟用狀態。若要啟動GRID虛擬應用程式的RDSH應用程式託管功能，請完成中的GRID虛擬應用程式在 Amazon EC2 GPU 執行個體上啟用NVIDIAGRID虛擬應用程式啟動步驟

Ubuntu 和 Debian

在執行個體上安裝NVIDIAGRID驅動程式

1. 連線至您的 Linux 執行個體。如果尚未安裝 gcc 和 make，請進行安裝。

```
[ec2-user ~]$ sudo apt-get install -y gcc make
```

2. 更新套件快取，並取得執行個體的套件更新。

```
$ sudo apt-get update -y
```

3. (Ubuntu) 升級 `linux-aws` 套件以接收最新版本。

```
$ sudo apt-get upgrade -y linux-aws
```

   (Debian) 升級套件以接收最新版本。

```
$ sudo apt-get upgrade -y
```

4. 將執行個體重新開機以載入最新的核心版本。

```
$ sudo reboot
```

5. 重新開機後，請重新連線至您的執行個體。
6. 針對您目前執行的核心版本，安裝 gcc 編譯器與核心標頭套件。

```
$ sudo apt-get install -y linux-headers-$(uname -r)
```

7. 停用NVIDIA顯示卡的nouveau開放原始碼驅動程式。

   a. 將 nouveau 新增到 /etc/modprobe.d/blacklist.conf 黑名單檔案。複製下列程式碼區塊並貼至終端機。

   ```
   $ cat << EOF | sudo tee --append /etc/modprobe.d/blacklist.conf
   blacklist vga16fb
   blacklist nouveau
   blacklist rivafb
   blacklist nvidiafb
   blacklist rivatv
   EOF
   ```

   b. 編輯 /etc/default/grub 檔案並新增下行：

   ```
   GRUB_CMDLINE_LINUX="rdblacklist=nouveau"
   ```

   c. 重建 Grub 組態。

   ```
   $ sudo update-grub
   ```

8. 使用下列指令下載GRID驅動程式安裝公用程式：

```
$ aws s3 cp --recursive s3://ec2-linux-nvidia-drivers/latest/ .
```

此值區中會儲存多個版本的GRID驅動程式。您可以使用下列命令查看所有可用版本。

```
$ aws s3 ls --recursive s3://ec2-linux-nvidia-drivers/
```

9. 請使用以下命令新增許可來執行驅動程式安裝公用程式。

```
$ chmod +x NVIDIA-Linux-x86_64*.run
```

10. 執行自我安裝指令碼，如下所示，以安裝您下載的GRID驅動程式。例如：

```
$ sudo /bin/sh ./NVIDIA-Linux-x86_64*.run
```

出現提示時，請接受授權合約，並依需求指定安裝選項 (您可以接受預設選項)。

11. 確認驅動程式可正常運作。下列指令的回應會列出安裝的NVIDIA驅動程式版本以及有關的詳細資訊GPUs。

```
$ nvidia-smi -q | head
```

12. 如果您在 G4dn、G5 或 G5G 執行個GPU體上使用 14.x 或更新版本的 NVIDIA v 軟體，請使用下列命令停GSP用。有關更多信息，有關為什麼需要這樣做，請訪問NVIDIA的文檔。

```
$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /etc/
modprobe.d/nvidia.conf
```

13. 重新啟動執行個體。

```
$ sudo reboot
```

14. (選用) 視您的使用案例而定，您可能會完成下列選用步驟。如果您不需要這項功能，請勿完成這些步驟。

    a. 若要協助您充分利用四個高達 4K 解析度的顯示器，請設定高效能顯示通訊協定NICEDCV。

    b. NVIDIAQuadro 虛擬工作站模式預設為啟用狀態。若要啟動GRID虛擬應用程式的RDSH應用程式託管功能，請完成中的GRID虛擬應用程式在 Amazon EC2 GPU 執行個體上啟用NVIDIAGRID虛擬應用程式啟動步驟

    c. 安裝GUI桌上型電腦/工作站套件。

    ```
    $ sudo apt-get install -y lightdm ubuntu-desktop
    ```

視窗作業系統

若要在 Windows 執行個體上安裝NVIDIAGRID驅動程式

1. Connect 至您的 Windows 執行個體並開啟 PowerShell 視窗。

2. 設定 Windows 執行個體 AWS Tools for Windows PowerShell 上的預設認證。如需詳細資訊，請參閱 AWS Tools for Windows PowerShell 使用者指南中 AWS Tools for Windows PowerShell入門。

> **Important**
>
> 您的使用者或角色必須擁有包含 AmazonS3 ReadOnlyAccess 政策的授與權限。如需詳細資訊，請參閱 AWSAmazon 簡單儲存服務使用者指南ReadOnlyAccess中的受管政策：AmazonS3。

3. 使用下列 PowerShell 命令，將驅動程式和NVIDIAGRID雲端使用者授權合約從 Amazon S3 下載到桌面。

```
$Bucket = "ec2-windows-nvidia-drivers"
$KeyPrefix = "latest"
$LocalPath = "$home\Desktop\NVIDIA"
$Objects = Get-S3Object -BucketName $Bucket -KeyPrefix $KeyPrefix -Region us-east-1
foreach ($Object in $Objects) {
$LocalFileName = $Object.Key
if ($LocalFileName -ne '' -and $Object.Size -ne 0) {
    $LocalFilePath = Join-Path $LocalPath $LocalFileName
    Copy-S3Object -BucketName $Bucket -Key $Object.Key -LocalFile $LocalFilePath -
Region us-east-1
}
}
```

此值區中會儲存多個版本的NVIDIAGRID驅動程式。您可以移除 `-KeyPrefix $KeyPrefix` 選項，來下載儲存貯體中的所有可用 Windows 版本。如需有關您作業系統之NVIDIAGRID驅動程式版本的資訊，請參閱NVIDIA網站上的 NVIDIA® Virtual GPU (vGPU) 軟體文件。

從 11.0 GRID 版開始，您可以將下面的驅動程式用`latest`於 G3 和 G4dn 執行個體。我們不會將 11.0 之後的版本新增至 `g4/latest`，但會將版本 11.0 和 G4dn 專用的舊版保留在 `g4/latest` 下。

G5 執行個體需要 GRID 13.1 或更新版本 (或 GRID 12.4 或更新版本)。

4. 導覽至桌面並按兩下安裝檔案予以啟動 (選擇與您的執行個體 OS 版本對應的驅動程式版本)。依照指示安裝驅動程式，並視需要將執行個體重新開機。若要確認GPU是否正常運作，請檢查裝置管理員。

5. (選擇性) 使用下列指令停用控制台中的授權頁面，以防止使用者意外變更產品類型 (預設為啟用「NVIDIAGRID虛擬工作站」)。如需詳細資訊，請參閱GRID授權使用指南。

   PowerShell

   執行下列 PowerShell 命令以建立登錄值，以停用控制台中的授權頁面。 AWS Windows AWS Tools for PowerShell 中的AMIs預設值為 32 位元版本，且此命令會失敗。請改用作業系統PowerShell隨附的 64 位元版本。

```
New-Item -Path "HKLM:\SOFTWARE\NVIDIA Corporation\Global" -Name GridLicensing
New-ItemProperty -Path "HKLM:\SOFTWARE\NVIDIA Corporation\Global\GridLicensing" -
Name "NvCplDisableManageLicensePage" -PropertyType "DWord" -Value "1"
```

   命令提示

   執行下列登錄命令以建立登錄值來停用控制面板中的授權頁面。您可以使用 [命令提示字元] 視窗或 64 位元版本的執行它PowerShell。

```
reg add "HKLM\SOFTWARE\NVIDIA Corporation\Global\GridLicensing" /v
 NvCplDisableManageLicensePage /t REG_DWORD /d 1
```

6. (選用) 視您的使用案例而定，您可能會完成下列選用步驟。如果您不需要這項功能，請勿完成這些步驟。

   a. 若要協助您充分利用高達 4K 解析度的四個顯示器，請設定高效能顯示通訊協定NICEDCV。

   b. NVIDIAQuadro 虛擬工作站模式預設為啟用狀態。若要啟動GRID虛擬應用程式的RDSH應用程式託管功能，請完成中的GRID虛擬應用程式在 Amazon EC2 GPU 執行個體上啟用NVIDIAGRID虛擬應用程式啟動步驟

選項 4：NVIDIA遊戲驅動程式 (G5 和 G4dn 執行個體)

這些驅動程式僅供 AWS 客戶使用。下載這些軟體即表示您同意使用下載的軟體，僅用AMIs於開發NVIDIA A10G 和 T NVIDIA esla T4 硬體。安裝軟體後，您將受到NVIDIAGRID雲端使用者授權合約的條款約束。

考量事項

- G3 執行個體需要 AWS 提供DNS解決方案，GRID授權才能運作。
- IMDSv2僅支援 495.x 或更新版本的NVIDIA驅動程式。

先決條件

在安裝NVIDIA遊戲驅動程式之前，請確認您已在執行個體上安 AWS CLI 裝並設定預設憑證。如需詳細資訊，請參閱《AWS Command Line Interface 使用者指南》中的安裝 AWS CLI。

> **Important**
> 您的使用者或角色必須擁有包含 AmazonS3 ReadOnlyAccess 政策的授與權限。如需詳細資訊，請參閱 AWSAmazon 簡單儲存服務使用者指南ReadOnlyAccess中的受管政策：AmazonS3。

Amazon Linux 和 Amazon Linux 2

在執行個體上安裝NVIDIA遊戲驅動程式

1. 連線至您的 Linux 執行個體。
2. 如果尚未安裝 gcc 和 make，請進行安裝。

```
[ec2-user ~]$ sudo yum install gcc make
```

3. 更新套件快取，並取得執行個體的套件更新。

```
[ec2-user ~]$ sudo yum update -y
```

4. 將執行個體重新開機以載入最新的核心版本。

```
[ec2-user ~]$ sudo reboot
```

5. 重新開機後，請重新連線至您的執行個體。
6. 為您目前執行的核心版本安裝核心標頭套件。

```
[ec2-user ~]$ sudo yum install -y kernel-devel-$(uname -r)
```

7. 使用下列命令下載遊戲驅動程式安裝公用程式：

```
[ec2-user ~]$ aws s3 cp --recursive s3://nvidia-gaming/linux/latest/ .
```

此儲存貯體會保留遊戲驅動程式的多個版本。您可以使用下列命令查看所有可用版本：

```
[ec2-user ~]$ aws s3 ls --recursive s3://nvidia-gaming/linux/
```

8. 從下載的 `.zip` 封存中擷取遊戲驅動程序安裝公用程式。

```
[ec2-user ~]$ unzip latest-driver-name.zip -d nvidia-drivers
```

9. 請使用以下命令新增許可來執行驅動程式安裝公用程式：

```
[ec2-user ~]$ chmod +x nvidia-drivers/NVIDIA-Linux-x86_64*-grid.run
```

10. 使用以下命令執行安裝程式：

```
[ec2-user ~]$ sudo ./nvidia-drivers/NVIDIA-Linux-x86_64*.run
```

> **Note**
>
> 如果您將 Amazon Linux 2 與核心版本 5.10 搭配使用，請使用下列指令來安裝NVIDIA遊戲驅動程式。
>
> ```
> [ec2-user ~]$ sudo CC=/usr/bin/gcc10-cc ./NVIDIA-Linux-x86_64*.run
> ```

出現提示時，請接受授權合約，並依需求指定安裝選項 (您可以接受預設選項)。

11. 使用以下命令建立所需的組態檔案。

```
[ec2-user ~]$ cat << EOF | sudo tee -a /etc/nvidia/gridd.conf
vGamingMarketplace=2
EOF
```

12. 使用以下命令下載認證檔案及重新命名。
    - 對於 460.39 版本或更新版本：

      ```
      [ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2023_9_22.cert"
      ```

    - 對於 440.68 到 445.48 版本：

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2020_04.cert"
```

- 對於較早版本

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2019_09.cert"
```

13. 如果您在 G4dn、G5 或 G5G 執行個體上使用NVIDIA驅動程式版本 510.x 或更新版本，請使用下列指令停用。GSP有關更多信息，有關為什麼需要這樣做，請訪問NVIDIA的文檔。

```
[ec2-user ~]$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
[ec2-user ~]$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /
etc/modprobe.d/nvidia.conf
```

14. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

15. (選擇性) 若要協助利用高達 4K 解析度的單一顯示器，請設定高效能顯示通訊協定NICEDCV。

CentOS 7 和 Red Hat Enterprise Linux 7

在執行個體上安裝NVIDIA遊戲驅動程式

1. 連線至您的 Linux 執行個體。如果尚未安裝 gcc 和 make，請進行安裝。

```
[ec2-user ~]$ sudo yum install -y gcc make
```

2. 更新套件快取，並取得執行個體的套件更新。

```
[ec2-user ~]$ sudo yum update -y
```

3. 將執行個體重新開機以載入最新的核心版本。

```
[ec2-user ~]$ sudo reboot
```

4. 重新開機後，請重新連線至您的執行個體。

5. 為您目前執行的核心版本安裝核心標頭套件。

```
[ec2-user ~]$ sudo yum install -y unzip kernel-devel-$(uname -r)
```

6. 停用NVIDIA顯示卡的nouveau開放原始碼驅動程式。

   a. 將 nouveau 新增到 /etc/modprobe.d/blacklist.conf 黑名單檔案。複製下列程式碼區塊並貼至終端機。

   ```
   [ec2-user ~]$ cat << EOF | sudo tee --append /etc/modprobe.d/blacklist.conf
   blacklist vga16fb
   blacklist nouveau
   blacklist rivafb
   blacklist nvidiafb
   blacklist rivatv
   EOF
   ```

   b. 編輯 /etc/default/grub 檔案並新增下行：

   ```
   GRUB_CMDLINE_LINUX="rdblacklist=nouveau"
   ```

   c. 重建 Grub 組態。

   ```
   [ec2-user ~]$ sudo grub2-mkconfig -o /boot/grub2/grub.cfg
   ```

7. 使用下列命令下載遊戲驅動程式安裝公用程式：

```
[ec2-user ~]$ aws s3 cp --recursive s3://nvidia-gaming/linux/latest/ .
```

此儲存貯體會保留遊戲驅動程式的多個版本。您可以使用下列命令查看所有可用版本：

```
[ec2-user ~]$ aws s3 ls --recursive s3://nvidia-gaming/linux/
```

8. 從下載的 .zip 封存中擷取遊戲驅動程序安裝公用程式。

```
[ec2-user ~]$ unzip *Gaming-Linux-Guest-Drivers.zip -d nvidia-drivers
```

9. 請使用以下命令新增許可來執行驅動程式安裝公用程式：

```
[ec2-user ~]$ chmod +x nvidia-drivers/NVIDIA-Linux-x86_64*-grid.run
```

10. 使用以下命令執行安裝程式：

```
[ec2-user ~]$ sudo nvidia-drivers/NVIDIA-Linux-x86_64*.run
```

出現提示時，請接受授權合約，並依需求指定安裝選項 (您可以接受預設選項)。

11. 使用以下命令建立所需的組態檔案。

```
[ec2-user ~]$ cat << EOF | sudo tee -a /etc/nvidia/gridd.conf
vGamingMarketplace=2
EOF
```

12. 使用以下命令下載認證檔案及重新命名。
    - 對於 460.39 版本或更新版本：

      ```
      [ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2023_9_22.cert"
      ```

    - 對於 440.68 到 445.48 版本：

      ```
      [ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2020_04.cert"
      ```

    - 對於較早版本

      ```
      [ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2019_09.cert"
      ```

13. 如果您在 G4dn、G5 或 G5G 執行個體上使用NVIDIA驅動程式版本 510.x 或更新版本，請使用下列指令停用。GSP有關更多信息，有關為什麼需要這樣做，請訪問NVIDIA的文檔。

```
[ec2-user ~]$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
[ec2-user ~]$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /etc/modprobe.d/nvidia.conf
```

14. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

15. (選擇性) 若要協助利用高達 4K 解析度的單一顯示器，請設定高效能顯示通訊協定NICEDCV。如果您不需要此功能，請勿完成此步驟。

CentOS Stream 8 和 Red Hat Enterprise Linux 8

在執行個體上安裝NVIDIA遊戲驅動程式

1. 連線至您的 Linux 執行個體。如果尚未安裝 gcc 和 make，請進行安裝。

```
[ec2-user ~]$ sudo yum install -y gcc make
```

2. 更新套件快取，並取得執行個體的套件更新。

```
[ec2-user ~]$ sudo yum update -y
```

3. 將執行個體重新開機以載入最新的核心版本。

```
[ec2-user ~]$ sudo reboot
```

4. 重新開機後，請重新連線至您的執行個體。
5. 為您目前執行的核心版本安裝核心標頭套件。

```
[ec2-user ~]$ sudo yum install -y unzip kernel-devel-$(uname -r)
```

6. 使用下列命令下載遊戲驅動程式安裝公用程式：

```
[ec2-user ~]$ aws s3 cp --recursive s3://nvidia-gaming/linux/latest/ .
```

此儲存貯體會保留遊戲驅動程式的多個版本。您可以使用下列命令查看所有可用版本：

```
[ec2-user ~]$ aws s3 ls --recursive s3://nvidia-gaming/linux/
```

7. 從下載的 `.zip` 封存中擷取遊戲驅動程序安裝公用程式。

```
[ec2-user ~]$ unzip *Gaming-Linux-Guest-Drivers.zip -d nvidia-drivers
```

8. 請使用以下命令新增許可來執行驅動程式安裝公用程式：

```
[ec2-user ~]$ chmod +x nvidia-drivers/NVIDIA-Linux-x86_64*-grid.run
```

9. 使用以下命令執行安裝程式：

```
[ec2-user ~]$ sudo nvidia-drivers/NVIDIA-Linux-x86_64*.run
```

出現提示時，請接受授權合約，並依需求指定安裝選項 (您可以接受預設選項)。

10. 使用以下命令建立所需的組態檔案。

```
[ec2-user ~]$ cat << EOF | sudo tee -a /etc/nvidia/gridd.conf
vGamingMarketplace=2
EOF
```

11. 使用以下命令下載認證檔案及重新命名。
    - 對於 460.39 版本或更新版本：

      ```
      [ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2023_9_22.cert"
      ```

    - 對於 440.68 到 445.48 版本：

      ```
      [ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2020_04.cert"
      ```

    - 對於較早版本

      ```
      [ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2019_09.cert"
      ```

12. 如果您在 G4dn、G5 或 G5G 執行個體上使用NVIDIA驅動程式版本 510.x 或更新版本，請使用下列指令停用。GSP有關更多信息，有關為什麼需要這樣做，請訪問NVIDIA的文檔。

```
[ec2-user ~]$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
[ec2-user ~]$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /etc/modprobe.d/nvidia.conf
```

13. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

14. (選擇性) 若要協助利用高達 4K 解析度的單一顯示器，請設定高效能顯示通訊協定NICEDCV。

Rocky Linux 8

在執行個體上安裝NVIDIA遊戲驅動程式

1. 連線至您的 Linux 執行個體。如果尚未安裝 gcc 和 make，請進行安裝。

```
[ec2-user ~]$ sudo yum install -y gcc make
```

2. 更新套件快取，並取得執行個體的套件更新。

```
[ec2-user ~]$ sudo yum update -y
```

3. 將執行個體重新開機以載入最新的核心版本。

```
[ec2-user ~]$ sudo reboot
```

4. 重新開機後，請重新連線至您的執行個體。
5. 為您目前執行的核心版本安裝核心標頭套件。

```
[ec2-user ~]$ sudo dnf install -y unzip elfutils-libelf-devel libglvnd-devel
 kernel-devel-$(uname -r)
```

6. 使用下列命令下載遊戲驅動程式安裝公用程式：

```
[ec2-user ~]$ aws s3 cp --recursive s3://nvidia-gaming/linux/latest/ .
```

此儲存貯體會保留遊戲驅動程式的多個版本。您可以使用下列命令查看所有可用版本：

```
[ec2-user ~]$ aws s3 ls --recursive s3://nvidia-gaming/linux/
```

7. 從下載的 `.zip` 封存中擷取遊戲驅動程序安裝公用程式。

```
[ec2-user ~]$ unzip *Gaming-Linux-Guest-Drivers.zip -d nvidia-drivers
```

8. 請使用以下命令新增許可來執行驅動程式安裝公用程式：

```
[ec2-user ~]$ chmod +x nvidia-drivers/NVIDIA-Linux-x86_64*-grid.run
```

9. 使用以下命令執行安裝程式：

```
[ec2-user ~]$ sudo nvidia-drivers/NVIDIA-Linux-x86_64*.run
```

出現提示時，請接受授權合約，並依需求指定安裝選項 (您可以接受預設選項)。

10. 使用以下命令建立所需的組態檔案。

```
[ec2-user ~]$ cat << EOF | sudo tee -a /etc/nvidia/gridd.conf
vGamingMarketplace=2
EOF
```

11. 使用以下命令下載認證檔案及重新命名。
    - 對於 460.39 版本或更新版本：

      ```
      [ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2023_9_22.cert"
      ```

    - 對於 440.68 到 445.48 版本：

      ```
      [ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2020_04.cert"
      ```

    - 對於較早版本

      ```
      [ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2019_09.cert"
      ```

12. 如果您在 G4dn、G5 或 G5G 執行個體上使用NVIDIA驅動程式版本 510.x 或更新版本，請使用下列指令停用。GSP有關更多信息，有關為什麼需要這樣做，請訪問NVIDIA的文檔。

```
[ec2-user ~]$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
[ec2-user ~]$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /etc/modprobe.d/nvidia.conf
```

13. 重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

14. (選擇性) 若要協助利用高達 4K 解析度的單一顯示器，請設定高效能顯示通訊協定NICEDCV。

Ubuntu 和 Debian

在執行個體上安裝NVIDIA遊戲驅動程式

1. 連線至您的 Linux 執行個體。如果尚未安裝 gcc 和 make，請進行安裝。

```
[ec2-user ~]$ sudo yum install -y gcc make
```

2. 更新套件快取，並取得執行個體的套件更新。

```
$ sudo apt-get update -y
```

3. 升級 `linux-aws` 套件以接收最新版本。

```
$ sudo apt-get upgrade -y linux-aws
```

4. 將執行個體重新開機以載入最新的核心版本。

```
$ sudo reboot
```

5. 重新開機後，請重新連線至您的執行個體。
6. 為您目前執行的核心版本安裝核心標頭套件。

```
$ sudo apt-get install -y unzip linux-headers-$(uname -r)
```

7. 停用NVIDIA顯示卡的nouveau開放原始碼驅動程式。

   a. 將 `nouveau` 新增到 `/etc/modprobe.d/blacklist.conf` 黑名單檔案。複製下列程式碼區塊並貼至終端機。

```
$ cat << EOF | sudo tee --append /etc/modprobe.d/blacklist.conf
blacklist vga16fb
blacklist nouveau
blacklist rivafb
blacklist nvidiafb
blacklist rivatv
EOF
```

   b. 編輯 `/etc/default/grub` 檔案並新增下行：

```
GRUB_CMDLINE_LINUX="rdblacklist=nouveau"
```

c. 重建 Grub 組態。

```
$ sudo update-grub
```

8. 使用下列命令下載遊戲驅動程式安裝公用程式：

```
$ aws s3 cp --recursive s3://nvidia-gaming/linux/latest/ .
```

此儲存貯體會保留遊戲驅動程式的多個版本。您可以使用下列命令查看所有可用版本：

```
$ aws s3 ls --recursive s3://nvidia-gaming/linux/
```

9. 從下載的 `.zip` 封存中擷取遊戲驅動程序安裝公用程式。

```
$ unzip *Gaming-Linux-Guest-Drivers.zip -d nvidia-drivers
```

10. 請使用以下命令新增許可來執行驅動程式安裝公用程式：

```
$ chmod +x nvidia-drivers/NVIDIA-Linux-x86_64*-grid.run
```

11. 使用以下命令執行安裝程式：

```
$ sudo nvidia-drivers/NVIDIA-Linux-x86_64*.run
```

出現提示時，請接受授權合約，並依需求指定安裝選項 (您可以接受預設選項)。

12. 使用以下命令建立所需的組態檔案。

```
$ cat << EOF | sudo tee -a /etc/nvidia/gridd.conf
vGamingMarketplace=2
EOF
```

13. 使用以下命令下載認證檔案及重新命名。

- 對於 460.39 版本或更新版本：

```
$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2023_9_22.cert"
```

- 對於 440.68 到 445.48 版本：

```
$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2020_04.cert"
```

- 對於較早版本

```
$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2019_09.cert"
```

14. 如果您在 G4dn、G5 或 G5G 執行個體上使用NVIDIA驅動程式版本 510.x 或更新版本，請使用下列指令停用。GSP有關更多信息，有關為什麼需要這樣做，請訪問NVIDIA的文檔。

```
$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /etc/
modprobe.d/nvidia.conf
```

15. 重新啟動執行個體。

```
$ sudo reboot
```

16. (選擇性) 若要協助利用高達 4K 解析度的單一顯示器，請設定高效能顯示通訊協定NICEDCV。如果您不需要此功能，請勿完成此步驟。

視窗作業系統

在您的執行個體上安裝NVIDIA遊戲驅動程式之前，除了針對所有遊戲驅動程式提到的考量之外，您還必須確定符合下列先決條件。

- 如果您使用自訂視窗啟動 Windows 執行個體AMI，則AMI必須是使用 Windows Sysprep 建立的標準化映像檔，以確保遊戲驅動程式正常運作。如需詳細資訊，請參閱EC2AMI使用視窗系統創建一個Amazon。
- 設定 Windows 執行個體 AWS Tools for Windows PowerShell 上的預設認證。如需詳細資訊，請參閱 AWS Tools for Windows PowerShell 使用者指南中 AWS Tools for Windows PowerShell入門。

若要在 Windows 執行個體上安裝NVIDIA遊戲驅動程式

1. Connect 至您的 Windows 執行個體並開啟 PowerShell 視窗。
2. 使用下列 PowerShell指令下載並安裝遊戲驅動程式。

```
$Bucket = "nvidia-gaming"
$KeyPrefix = "windows/latest"
$LocalPath = "$home\Desktop\NVIDIA"
$Objects = Get-S3Object -BucketName $Bucket -KeyPrefix $KeyPrefix -Region us-east-1
foreach ($Object in $Objects) {
$LocalFileName = $Object.Key
if ($LocalFileName -ne '' -and $Object.Size -ne 0) {
    $LocalFilePath = Join-Path $LocalPath $LocalFileName
    Copy-S3Object -BucketName $Bucket -Key $Object.Key -LocalFile $LocalFilePath -
Region us-east-1
}
}
```

多個版本的NVIDIAGRID驅動程式存放在此 S3 儲存貯體中。如果將 `$KeyPrefix` 變數的值從 "windows/latest" 變更為 "windows"，則可以下載儲存貯體中的所有可用版本。

3. 導覽至桌面並按兩下安裝檔案予以啟動 (選擇與您的執行個體 OS 版本對應的驅動程式版本)。依照指示安裝驅動程式，並視需要將執行個體重新開機。若要確認GPU是否正常運作，請檢查裝置管理員。
4. 使用下列其中一種方法來註冊驅動程式。

   Version 527.27 or above

   使用 64 位元版本或 [命令提示字元] 視窗建立下列登錄機碼。 PowerShell

   key：`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\nvlddmkm\Global`

   名稱： vGamingMarketplace

   類型：DWord

   value (值)：2

   PowerShell

   執行下列 PowerShell 命令以建立此登錄值。 AWS Windows AWS Tools for PowerShell 中的 AMIs預設值為 32 位元版本，且此命令會失敗。請改用作業系統 PowerShell 隨附的 64 位元版本。

```
New-ItemProperty -Path "HKLM:\SYSTEM\CurrentControlSet\Services\nvlddmkm\Global"
 -Name "vGamingMarketplace" -PropertyType "DWord" -Value "2"
```

命令提示

執行下列登錄命令來建立此登錄值。您可以使用 [命令提示字元] 視窗或 64 位元版本的執行它 PowerShell。

```
reg add "HKLM\SYSTEM\CurrentControlSet\Services\nvlddmkm\Global" /v
 vGamingMarketplace /t REG_DWORD /d 2
```

Earlier versions

使用 64 位元版本或 [命令提示字元] 視窗建立下列登錄機碼。 PowerShell

key：`HKEY_LOCAL_MACHINE\SOFTWARE\NVIDIA Corporation\Global`

名稱： vGamingMarketplace

類型：DWord

value (值)：2

PowerShell

執行下列 PowerShell 命令以建立此登錄值。 AWS Windows AWS Tools for PowerShell 中的 AMIs預設值為 32 位元版本，且此命令會失敗。請改用作業系統 PowerShell 隨附的 64 位元版本。

```
New-ItemProperty -Path "HKLM:\SOFTWARE\NVIDIA Corporation\Global" -Name
 "vGamingMarketplace" -PropertyType "DWord" -Value "2"
```

命令提示

執行下列登錄命令，使用「命令提示字元」視窗來建立此登錄機碼。您也可以在 64 位元版本的中使用此命令PowerShell。

```
reg add "HKLM\SOFTWARE\NVIDIA Corporation\Global" /v vGamingMarketplace /t
 REG_DWORD /d 2
```

5. 在中執行下列命令 PowerShell。這會下載認證檔案、重新命名檔案 `GridSwCert.txt`，並將檔案移至系統磁碟機上的 Public Documents (公用文件) 資料夾。通常，資料夾路徑為 `C:\Users\Public\Documents`。

   - 對於 461.40 版本或更新版本：

     ```
     Invoke-WebRequest -Uri "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertWindows_2023_9_22.cert" -OutFile "$Env:PUBLIC\Documents\GridSwCert.txt"
     ```

   - 對於 445.87 版本：

     ```
     Invoke-WebRequest -Uri "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Windows_2020_04.cert" -OutFile "$Env:PUBLIC\Documents\GridSwCert.txt"
     ```

   - 對於較早版本

     ```
     Invoke-WebRequest -Uri "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Windows_2019_09.cert" -OutFile "$Env:PUBLIC\Documents\GridSwCert.txt"
     ```

   > **Note**
   >
   > 如果您在下載檔案時收到錯誤訊息，而且您使用的是 Windows 伺服器 2016 或更早版本，則可能需要為 PowerShell 終端機啟用 TLS 1.2。您可以使用以下命令為當前 PowerShell 會話啟用 TLS 1.2，然後再試一次：
   >
   > ```
   > [Net.ServicePointManager]::SecurityProtocol =
   > [Net.SecurityProtocolType]::Tls12
   > ```

6. 將執行個體重新開機。

7. 使用以下命令驗證NVIDIA遊戲許可證。

   ```
   C:\Windows\System32\DriverStore\FileRepository\nvgridsw_aws.inf_*\nvidia-smi.exe -q
   ```

   輸出格式應類似以下內容。

   ```
   vGPU Software Licensed Product
   ```

```
Product Name                : NVIDIA Cloud Gaming
License Status              : Licensed (Expiry: N/A)
```

8. (選擇性) 若要協助利用高達 4K 解析度的單一顯示器，請設定高效能顯示通訊協定NICEDCV。如果您不需要此功能，請勿完成此步驟。

安裝其他版本的 CUDA

在執行個體上安裝NVIDIA圖形驅動程式之後，您可以安裝圖形驅動程式隨附的版本以外的版本。CUDA下列程序示範如何在執行個體CUDA上設定多個版本的。

在 Linux 上安裝CUDA工具包

請依照下列步驟在 Linux 上安裝CUDA工具組：

1. 連線至您的 Linux 執行個體。
2. 打開NVIDIA網站，然後選擇所需CUDA的版本。
3. 選取執行個體上作業系統的架構、發行版本和版本。針對安裝程式類型，選取 runfile (local)。
4. 依照指示下載安裝指令碼。
5. 將執行權限新增至您使用下列命令下載的安裝指令碼。

```
[ec2-user ~]$ chmod +x downloaded_installer_file
```

6. 執行安裝指令碼，如下所示以安裝CUDA工具組，並將CUDA版本號碼新增至工具組路徑。

```
[ec2-user ~]$ sudo sh downloaded_installer_file --silent --override --toolkit --
samples --toolkitpath=/usr/local/cuda-version --samplespath=/usr/local/cuda --no-
opengl-libs
```

7. (選擇性) 如下設定預設CUDA版本。

```
[ec2-user ~]$ sudo ln -s /usr/local/cuda-version /usr/local/cuda
```

在視窗上安裝CUDA工具包

請依照下列步驟在 Windows 上安裝CUDA工具組：

若要安裝工CUDA具組

1. 連接至 Windows 執行個體。
2. 打開NVIDIA網站，然後選擇所需CUDA的版本。
3. 在Installer Type (安裝程式類型)中，選取 exe (local)，然後選取 Download (下載)。
4. 使用瀏覽器執行下載的安裝檔案。依照指示安裝工CUDA具組。您可能需要重新啟動執行個體。

在 EC2 Windows 執行個體上安裝ENA驅動程式

如果您的執行個體不是以 Amazon 提供的最新 Windows Amazon 機器映像 (AMIs) 為基礎，請使用下列程序在您的執行個體上安裝目前的ENA驅動程式。您應該在方便重新啟動執行個體時，執行這項更新。如果安裝指令碼未自動重新啟動執行個體，則建議您重新啟動執行個體作為最後一步。

如果在執行個體執行時使用執行個體儲存體磁碟區來儲存資料，則當停止執行個體時會清除該資料。停止執行個體之前，請確認您已將所需的任何資料從執行個體存放磁碟區複製到持續性儲存，例如 Amazon EBS 或 Amazon S3。

必要條件

若要安裝或升級ENA驅動程式，您的 Windows 執行個體必須符合下列先決條件：

- 已安裝 3.0 或更新 PowerShell 版本

步驟 1：備份資料

建議您建立備份AMI，以防您無法透過「裝置管理員」回復變更。若要使用建立備份 AMI AWS Management Console，請依照下列步驟執行：

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇執行個體。
3. 選取需要升級驅動程式的執行個體，並從執行個體狀態選單中選擇停止執行個體。
4. 在停止執行個體後，再次選取執行個體。若要建立備份，請從動作選單中選擇映像和範本，然後選擇建立映像。
5. 若要重新啟動執行個體，請從執行個體狀態選單中選擇啟動執行個體。

步驟 2：安裝或升級ENA驅動程式

您可以使用 AWS Systems Manager 散發者或PowerShell 指令程式來安裝或升級ENA驅動程式。如需進一步指示，請選取與您要使用之方法相符的索引標籤。

Systems Manager Distributor

您可以使用 Systems Manager Distributor 功能，將套件部署至 Systems Manager 管理的節點。使用系統管理員代理商時，您可以安裝ENA驅動程式套件一次，或安裝排程更新。如需有關如何安裝ENA驅動程式套件 (`AwsEnaNetworkDriver`) 與 Systems Manager 配送者的詳細資訊，請參閱AWS Systems Manager 使用者指南中的安裝或更新套件。

PowerShell

本節說明如何使用 PowerShell cmdlet 在執行個體上下載及安裝ENA驅動程式套件。

選項 1：下載並解壓縮最新版本

1. 連線到您的執行個體，並以本機管理員登入。
2. 使用 invoke-webrequest cmdlet 下載最新的驅動程式套件：

```
PS C:\> invoke-webrequest https://ec2-windows-drivers-
downloads.s3.amazonaws.com/ENA/Latest/AwsEnaNetworkDriver.zip -
outfile $env:USERPROFILE\AwsEnaNetworkDriver.zip
```

> **Note**
>
> 如果您在下載檔案時收到錯誤訊息，而且您使用的是 Windows 伺服器 2016 或更早版本，則可能需要為 PowerShell 終端機啟用 TLS 1.2。您可以使用以下命令為當前 PowerShell 會話啟用 TLS 1.2，然後再試一次：
>
> ```
> [Net.ServicePointManager]::SecurityProtocol =
> [Net.SecurityProtocolType]::Tls12
> ```
>
> 或者，可以從執行個體的瀏覽器視窗下載最新的驅動程式套件。

3. 使用 expand-archive cmdlet 來解壓縮下載至執行個體的 zip 封存檔：

```
PS C:\> expand-archive $env:userprofile\AwsEnaNetworkDriver.zip -
DestinationPath $env:userprofile\AwsEnaNetworkDriver
```

選項 2：下載並解壓縮特定版本

1. 連線到您的執行個體，並以本機管理員登入。
2. 從ENA驅動程式版本紀錄表格中的版本連結下載您想要的特定版本的ENA驅動程式套件。
3. 將 zip 封存檔解壓縮至執行個體。

安裝ENA驅動程序 PowerShell

無論您是下載最新的驅動程式還是特定版本，安裝步驟都相同。若要安裝ENA驅動程式，請依照下列步驟執行。

1. 若要安裝驅動程式，請從`install.ps1` PowerShell 執行個體上的`AwsEnaNetworkDriver`目錄執行指令碼。如果出現錯誤，請確定您使用的是 PowerShell 3.0 或更新版本。
2. 如果安裝程式未自動重新啟動執行個體，請執行指Restart-Computer PowerShell 令程式。

```
PS C:\> Restart-Computer
```

步驟 3 (選用)：安裝後驗證ENA驅動程式版本

若要確保ENA驅動程式套件已成功安裝在執行個體上，您可以依照下列步驟驗證新版本：

1. 連線到您的執行個體，並以本機管理員登入。
2. 若要開啟 Windows Device Manager (Windows 裝置管理員)，請在 Run (執行) 對話方塊中輸入 `devmgmt.msc`。
3. 選擇確定。這將開啟 Device Manager (裝置管理員) 視窗。
4. 選取 Network adapters (網路轉接器) 左側的箭頭以展開清單。
5. 選取名稱，或開啟 Amazon Elastic Network Adapter (Amazon 彈性網路轉接器) 的內容功能表，然後選擇 Properties (屬性)。這會開啟 Amazon 彈性網路介面卡屬性對話方塊。

> **Note**
>
> ENA介面卡都使用相同的驅動程式。如果您有多個ENA介面卡，您可以選取其中任何一個來更新所有ENA介面卡的驅動程式。

6. 若要驗證目前安裝的版本，請開啟驅動程式索引標籤並檢查驅動程式版本。如果目前版本與目標版本不相符，則請參閱 排除彈性網路介面卡 Windows 驅動程式。

復原ENA驅動程式安裝

如果安裝出現任何問題，則您可能需要復原驅動程序。請依照下列步驟復原至執行個體上安裝的舊版ENA驅動程式。

1. 連線到您的執行個體，並以本機管理員登入。
2. 若要開啟 Windows Device Manager (Windows 裝置管理員)，請在 Run (執行) 對話方塊中輸入 `devmgmt.msc`。
3. 選擇確定。這將開啟 Device Manager (裝置管理員) 視窗。
4. 選取 Network adapters (網路轉接器) 左側的箭頭以展開清單。
5. 選取名稱，或開啟 Amazon Elastic Network Adapter (Amazon 彈性網路轉接器) 的內容功能表，然後選擇 Properties (屬性)。這會開啟 Amazon 彈性網路介面卡屬性對話方塊。

> **Note**
>
> ENA介面卡都使用相同的驅動程式。如果您有多個ENA介面卡，您可以選取其中任何一個來更新所有ENA介面卡的驅動程式。

6. 若要復原驅動程式，請開啟驅動程式索引標籤並選擇復原驅動程式。這會開啟驅動程式套件復原視窗。

> **Note**
>
> 如果驅動程式索引標籤未顯示復原驅動程式動作，或者動作無法使用，則表示執行個體上的驅動程式存放區不包含先前安裝的驅動程式套件。若要疑難排解此問題故障診斷方案，請參閱並展開「安裝未預期的ENA驅動程式版本」一節。如需有關裝置驅動程式套件選取程序的詳細資訊，請參閱 Microsoft 文件網站上的 Windows 如何為裝置選取驅動程式套件。

追蹤ENA驅動程式版本

視窗AMIs包含 ENA Windows 驅動程式，以啟用增強型網路功能。

下表顯示每個 Windows 伺服器版本要下載的對應ENA驅動程式版本。

Windows Server 版本	ENA 驅動程式版本
Windows Server 2022	2.4.0 和更新版本
Windows Server 2019	最新
Windows Server 2016	最新
Windows Server 2012 R2	2.6.0 和更舊版本
Windows Server 2012	2.6.0 和更舊版本
Windows Server 2008 R2	2.2.3 和舊版

ENAAmazon 的 Windows 驅動程序發布通知 SNS

Amazon SNS 可以在 EC2 Windows 驅動程序的新版本發布時通知您。使用下列程序訂閱這些通知。

訂閱 EC2通知

1. 在 https://console.aws.amazon.com/sns/v3/ home 上打開 Amazon SNS 控制台。
2. 如有必要，請在導覽列中將「區域」變更為美國東部 (維吉尼亞北部)。您必須選取此區域，因為您訂閱的SNS通知位於此區域中。
3. 在導覽窗格中，選擇訂閱。
4. 選擇 Create subscription (建立訂閱)。
5. 在 Create subscription (建立訂閱) 對話方塊中，執行下列動作：
   a. 對於主題 ARN，請複製以下 Amazon 資源名稱 (ARN)：

      arn:aws:sns:us-east-1:801119661308:ec2-windows-drivers
   b. 針對 rotocol (通訊協定)，選擇 `Email`。
   c. 針對 Endpoint (端點)，請輸入可用於接收通知的電子郵件地址。
   d. 選擇 Create subscription (建立訂閱)。
6. 您將會收到確認電子郵件。開啟電子郵件並遵循指示完成訂閱。

每當發布新的 EC2 Windows 驅動程序時，我們都會向訂閱者發送通知。如果您不想再接收這些通知，請使用下列程序來取消訂閱。

退訂 Amazon EC2 視窗驅動程式通知

1. 在 https://console.aws.amazon.com/sns/v3/ home 上打開 Amazon SNS 控制台。
2. 在導覽窗格中，選擇 Subscriptions (訂閱)。
3. 選取訂閱的核取方塊，然後選取 Actions (動作)，Delete subscriptions (刪除訂閱)。出現確認提示時，請選擇 Delete (刪除)。

ENA驅動程式版本紀錄

下表摘要說明各版本的變更。

驅動程式版本	詳細資訊	發行日期
2.7.0	新功能 • 刪除了對視窗服務器 2012（視窗 8）和視窗服務器 2012 R2（視窗 8.1）的支持。這些作業系統版本已經到來的支援結束 AWS。在 Windows 伺服器 2012 及更早版本上，驅動程式安裝將失敗。 • 添加了對卸載 IPv6 Tx 校驗和計算到設備的支持。 • 增加了廣泛的低延遲排隊（LLQ）支持。這是根據裝置建議動態啟用的。您可以使用新的「寬LLQ」登錄機碼覆寫此設定。 • 已新增 Rx 溢位造成封包丟棄的報告，表示 Rx 環中傳入封包的空間不足。 • 增加了對來自設備的次優配置通知的支持。請參閱 Windows 事件檢視器59000中的事件識別碼。 錯誤修正 •	2024年5月1日

驅動程式版本	詳細資訊	發行日期
	避免標頭超過最大低延遲佇列 (LLQ) 標頭大小的 Tx 封包造成不必要的裝置重設。	

驅動程式版本	詳細資訊	發行日期

2.6.0	新功能
	• 為支援 ENA Express 的執行個體類型新增下列網路效能測量結果。
	  • `ena_srd_mode`
	  • `ena_srd_tx_pkts`
	  • `ena_srd_eligible_tx_pkts`
	  • `ena_srd_rx_pkts`
	  • `ena_srd_resource_utilization`
	• 為 Nitro 型執行個體類型新增 `conntrack_allowance_available` 網路效能指標。
	• 新增由於偵測到 RX 資料損毀而重設新介面卡的原因。
	• 更新驅動程式記錄的基礎設施。
	
	錯誤修正
	• 在網路效能測量結果更新失敗的CPU情況下，防止介面卡重設。
	• 防止對裝置心跳的中斷狀況進行錯誤偵測。
	• 修復驅動程式安裝指令碼以支援進行降級操作。
	• 修復接收錯誤計數統計的問題。
		2023 年 6 月 20 日

驅動程式版本	詳細資訊	發行日期
2.5.0	公告 ENAWindows 驅動程式 2.5.0 版已經復原，因為無法在 Windows 網域控制站上初始化。Windows Client 和 Windows Server 不受影響。	2023 年 2 月 17 日
2.4.0	新功能 • 新增對 Windows Server 2022 的支援。 • 移除對 Windows Server 2008 R2 的支援。 • 將「低延遲佇列」(LLQ) 設定為「永遠開啟」，以提升第六代 Amazon EC2 執行個體的效能。 錯誤修正 • 修正無法將網路效能度量發佈至 Windows 的效能計數器 (PCW) 系統。 • 修復登錄機碼讀取操作期間的記憶體流失問題。 • 防止在轉接器重設過程中發生無法復原的錯誤時，出現無限重設迴圈。	2022 年 4 月 28 日

驅動程式版本	詳細資訊	發行日期

2.2.4	公告
	
	ENAWindows 驅動程式 2.2.4 版已經復原，因為第六代 EC2執行個體可能會降低效能。我們建議您使用下列其中一個方法，降級您的驅動程式：
	
	• 安裝舊版
	1. 從此表格中的連結下載舊版套件 (版本 2.2.3)。
	2. 執行安install.ps1 PowerShell 裝指令碼。
	
	如需有關安裝前和安裝後步驟的詳情，請參閱 啟用 Windows 上的增強型網路。
	
	使用 Amazon EC2 Systems Manager 進行批量更新
	
	• 使用下列參數透過SSM文件`AWS-ConfigureAWSPackage` 執行大量更新：
	  • 名稱： AwsEnaNetworkDriver
	  • 版本：2.2.3
		2021 年 10 月 26 日

驅動程式版本	詳細資訊	發行日期
2.2.3	新功能 • 新增對具有最高 400 Gbps 執行個體聯網之新 Nitro 卡的支援。 錯誤修正 • 修復了ENA驅動程序系統時間更改和系統時間查詢之間的競爭狀況，這會導致硬件無法響應的誤陽性檢測。 視窗ENA驅動程式版本 2.2.3 是支援視窗伺服器 2008 R2 的最終版本。目前可用的執行個體類型ENA將繼續受到 Windows 伺服器 2008 R2 的支援，而驅動程式可透過下載取得。未來執行個體類型不會支援 Windows Server 2008 R2，而且您無法將 Windows Server 2008 R2 啟動、匯入或遷移到未來執行個體類型。	2021 年 3 月 25 日
2.2.2	新功能 • 添加對查詢網路介面卡效能度量的支援，以 CloudWatch 及適用於 Windows 消費者的效能計數器。 錯誤修正 • 修正裸機執行個體的效能問題。	2020 年 12 月 21 日
2.2.1	新功能 • 新增方法以允許主機查詢彈性網路轉接器來取得網路效能指標。	2020 年 10 月 1 日

驅動程式版本	詳細資訊	發行日期
2.2.0	新功能 • 新增對下一代硬體類型的支援。 • 改善從停止休眠恢復之後的執行個體啟動時間，並消除誤報錯誤訊息。ENA 效能最佳化 • 最佳化傳入流量的處理。 • 改善低資源環境中的共用記憶體管理。 錯誤修正 • 在少數情況下，驅動程序無法重ENA置，避免在設備移除時系統崩潰。	2020 年 8 月 12 日
2.1.5	錯誤修正 • 修正裸機執行個體上偶發性的網路介面卡初始化失敗。	2020 年 6 月 23 日
2.1.4	錯誤修正 • 防止由於從網絡堆棧到達損壞的LSO數據包元數據導致的連接問題。 • 預防因為存取已釋放封包記憶體造成罕見競爭條件情況，而導致系統當機。	2019 年 11 月 25 日

驅動程式版本	詳細資訊	發行日期
2.1.2	新功能 • 增加了對供應商 ID 報告的支持，以允許MAC基於操作系統生成UUIDs。 錯誤修正 • 改善初始化期間的DHCP網路組態效能。 • 當最大傳輸單元 (MTU) 超過 4K 時，正確計算入站IPv6流量的 L4 校驗值。 • 一般性改進增加穩定性和次要錯誤修正。	2019 年 11 月 4 日
2.1.1	錯誤修正 • 防止從作業系統到達的高度分散TCPLSO封包丟棄。 • IPSec在IPv6網絡中正確處理封裝安全有效負載（ESP）協議。	2019 年 9 月 16 日

驅動程式版本	詳細資訊	發行日期

2.1.0	ENAWindows 驅動程式 v2.1 引入了新的ENA裝置功能、提供效能提升、增加新功能，並包含多項穩定性改進。
	• 新功能
	  • 對巨型訊框組態使用標準化的 Windows 登錄機碼。
	  • 允許透過ENA驅動程式內容設定 VLAN ID GUI。
	  • 改善復原流程
	  • 改善失敗識別機制。
	  • 已新增對可調整復原參數的支援。
	  • 對於具有 8 個以上的較新EC2執行個體，最多可 Support 援 32 個 I/O 佇列vCPUs。
	  • 驅動程式記憶體使用量減少最高達 90%。
	• 效能最佳化
	  • 減少傳輸路徑延遲。
	  • 支援接收檢查總和卸載。
	  • 對負載繁重系統的效能最佳化 (鎖定機制的最佳化使用量)。
	  • 進一步增強功能，可降低CPU使用率並改善負載下的系統回應
	• 錯誤修正
	  • 修正由於非接續 Tx 標頭無效的剖析造成的損毀。
	  •
		2019 年 7 月 1 日

驅動程式版本	詳細資訊	發行日期
	修正裸機執行個體上彈性網絡介面分離期間的驅動程式 1.5 版損毀。 • 修復LSO偽頭校驗和計算錯誤。IPv6 • 修正初始化失敗時可能的記憶體資源流失。 • 禁用TCP/UDP校驗和卸載片段。IPv4 • 修正了VLAN配置。VLAN當只有VLAN優先級應該被禁用時被錯誤地禁用。 • 啟用事件檢視器對自訂驅動程式訊息的正確剖析。 • 由於無效的時間戳記處理，修正無法初始化驅動程式。 • 修復數據處理和ENA設備禁用之間的競爭情況。	
1.5.0	• 已提升穩定性並修正效能。 • 接收緩衝區現在可以在的「進階內容」中設定為 8192 的ENANIC值。 • 預設 Receive Buffers (接收緩衝區) 為 1,000。	2018 年 10 月 4 日
1.2.3	包含從 Windows Server 2008 R2 到 Windows Server 2016 的可靠性修正並統一支援。	2018 年 2 月 13 日
1.0.8	初始版本。包括在AMIs視窗服務器 2008 R2, 視窗服務器 2012RTM, 視窗服務器 2012 R2, 和視窗服務器 2016.	2016 年 7 月

Windows 執行個體的全虛擬驅動程式

Windows AMIs 包含一組允許訪問虛擬化硬件的驅動程序。Amazon 使用這些驅動程序EC2將實例存儲和 Amazon EBS 卷映射到其設備。下表顯示不同驅動程式的關鍵差異。

	RedHat 光伏	Citrix PV	AWS 光伏
執行個體類型	不支援所有執行個體類型。若您指定不支援的執行個體類型，執行個體便會故障。	支援 Xen 執行個體類型。	支援 Xen 執行個體類型。
已連接磁碟區	支援最多 16 個已連接磁碟區。	支援超過 16 個已連接磁碟區。	支援超過 16 個已連接磁碟區。
網路	驅動程式存在網路連線在高負載下重設的已知問題，例如，快速的FTP檔案傳輸。		驅動程式會在位於相容的執行個體類型上時，自動在網路介面卡上設定 Jumbo Frame。當執行個體位於叢集置放群組中時，這會在叢集置放群組中的執行個體之間提供較佳的網路效能。如需詳細資訊，請參閱Amazon EC2 執行個體的放置群組。

下表顯示您應該在 Amazon 上每個版本的 Windows 服務器上運行哪些光伏驅動程序EC2。

Windows Server 版本	PV 驅動程式版本
Windows Server 2022	AWS 光伏最新版
Windows Server 2019	AWS 光伏最新版
Windows Server 2016	AWS 光伏最新版
Windows Server 2012 R2	AWS 光伏最新版
Windows Server 2012	AWS 光伏最新版
Windows Server 2008 R2	AWS 光伏版本 8.3.5 版
Windows Server 2008	Citrix PV 5.9
Windows Server 2003	Citrix PV 5.9

目錄

- AWS 光伏驅動器
- Citrix PV 驅動程式
- RedHat 光伏驅動器
- 訂閱 通知
- 升級 EC2 Windows 執行個體上的光伏驅動
- 排解 Windows 執行個體上的光伏驅動

AWS 光伏驅動器

P AWS V 驅動程式會儲存在目`%ProgramFiles%\Amazon\Xentools`錄中。此目錄還包含公用符號和指令行工具`xenstore_client.exe`，可讓您存取中的項目 XenStore。例如，下列 PowerShell 命令會從 Hypervisor 傳回目前的時間：

```
PS C:\> [DateTime]::FromFileTimeUTC((gwmi -n root\wmi -cl
 AWSXenStoreBase).XenTime).ToString("hh:mm:ss")
11:17:00
```

AWS 光伏驅動程式元件列於下的 Windows 登錄中`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services`。這些驅動程式元件如下：xenbus、xeniface、xennet、xenvbd 及 xenvif。

AWS 光伏驅動程式也有一個名為的 Windows 服務 LiteAgent，該服務以使用者模式執行。它會處理 Xen 產生執行個體 AWS APIs上的關機和重新啟動事件等工作。您可以藉由從命令列執行 `Services.msc` 來存取及管理服務。在 Nitro 產生執行個體上執行時，不會使用 AWS PV 驅動程式，且 LiteAgent 服務會從驅動程式版本 8.2.4 開始自我停止。更新至最新的 AWS PV 驅動程式也會更新 LiteAgent 並改善所有執行個體世代的可靠性。

安裝最新的 AWS 光伏驅動程式

Amazon Windows AMIs 包含一組允許存取虛擬化硬體的驅動程式。Amazon 使用這些驅動程序EC2將實例存儲和 Amazon EBS 卷映射到其設備。我們建議您安裝最新的驅動程式，以改善 EC2 Windows 執行個體的穩定性和效能。

安裝選項

- 您可以使用 AWS Systems Manager 來自動更新 PV 驅動程式。如需詳細資訊，請參閱AWS Systems Manager 使用指南中的逐步解說：在 EC2 Windows 執行個體上自動更新 PV 驅動程式 (主控台)。
- 您可以下載驅動程式套件並手動執行安裝程式。請務必檢查 `readme.txt` 檔案是否有系統需求。如需有關下載和安裝 AWS PV 驅動程式，或是升級網域控制器的資訊，請參閱手動升級 Windows 伺服器執行個體 (AWS PV 升級)。

AWS 光伏驅動器封裝歷史

下表顯示每個驅動程式版本對 AWS PV 驅動程式的變更。

套件版本	詳細資訊	發行日期
8.4.3	修正了套件安裝程式中的錯誤，以改善升級體驗。	2023 年 1 月 24 日
8.4.2	解決爭用情況的穩定性修復。	2022 年 4 月 13 日
8.4.1	改進的套件安裝程式。	2022 年 1 月 7 日

套件版本	詳細資訊	發行日期
8.4.0	• 穩定性修正，以解決磁碟 IO 的罕見卡住情況。 • 穩定性修正可解決EBS磁碟區分離期間發生當機的罕見情況。 • 新增功能，可將負載分配到多個核心，以處理運用超過 20,000 個工作負載，IOPS並因瓶頸而遭受降級的情況。若要啟用此功能，請參閱利用超過 20,000 個磁碟IOPS因CPU瓶頸而降級的工作負載。 • AWS 視窗伺服器 2008 R2 上的 PV 8.4 安裝將會失敗。 AWS 光伏版本 8.3.5 及更早版本在視窗伺服器 2008 R2 上受到支援。	2021 年 3 月 2 日
8.3.5	改進的套件安裝程式。	2022 年 1 月 7 日
8.3.4	改善網路裝置連接的可靠性。	2020 年 8 月 4 日
8.3.3	• 更新為 XenStore-facing 元件，以防止錯誤處理路徑期間的錯誤檢查。 • 更新到存儲組件，以避免提交無效SRB時崩潰。 若要在 Windows Server 2008 R2 執行個體上更新此驅動程式，您必須先驗證已安裝適當的修補程式來處理下列 Microsoft Security Advisory：Microsoft Security Advisory 3033929。	2020 年 2 月 4 日
8.3.2	強化的聯網元件可靠性	2019 年 7 月 30 日
8.3.1	改善儲存體元件的效能和健全性。	2019 年 6 月 12 日
8.2.7	已改善效率，支援遷移至最新一代執行個體類型。	2019 年 5 月 20 日
8.2.6	提升損毀傾印路徑的效率。	2019 年 1 月 15 日

套件版本	詳細資訊	發行日期
8.2.5	其他安全強化功能。 PowerShell 安裝程序現在可以在包。	2018 年 12 月 12 日
8.2.4	可靠性改進功能。	2018 年 10 月 2 日
8.2.3	錯誤修正與效能改進。 將磁EBS碟區 ID 報告為磁碟EBS區的磁碟序號。這可啟用如 S2D 等叢集情境。	2018 年 5 月 29 日
8.2.1	網路及儲存體效能改善及多個強固性修正。 若要確認已安裝此版本，請參閱下列 Windows 登錄值：`HKLM\Software\Amazon\PVDriver\Version 8.2.1` 。	2018 年 3 月 8 日
7.4.6	穩定性修復，使 AWS 光伏驅動器更具彈性。	2017 年 4 月 26 日
7.4.3	新增 Windows Server 2016 支援。 所有支援的 Windows OS 版本穩定性修正。 *AWS 光伏驅動程式 7.4.3 版的簽章將於 2019 年 3 月 29 日到期。我們建議您更新至最新的 AWS PV 驅動程式。	2016 年 11 月 18 日
7.4.2	X1 執行個體類型支援的穩定性修正。	2016 年 8 月 2 日
7.4.1	• AWS 光伏存儲驅動程序的性能改進。 • AWS PV 儲存驅動程式中的穩定性修正：修正執行個體發生錯誤檢查碼 0x0 DEAD 000 的系統當機問題。 • AWS PV 網路驅動程式的穩定性修正。 • 新增 Windows Server 2008R2 支援。	2016 年 7 月 12 日

套件版本	詳細資訊	發行日期
7.3.2	• 改善記錄日誌及診斷。 • AWS 光伏存儲驅動程序的穩定性修復。在某些案例中，磁碟在重新連接到執行個體之後，可能不會在 Windows 中出現。 • 新增 Windows Server 2012 支援。	2015 年 6 月 24 日
7.3.1	TRIM更新：修復與TRIM請求相關的問題。此修正程式可在管理大量TRIM要求時穩定執行個體並改善執行個體效能。	
7.3.0	TRIM支持： AWS PV 驅動程序現在將TRIM請求發送到虛擬機器管理程序。臨時磁盤將在基礎存儲支持TRIM（ ）的情況下正確處理TRIM請求。SSD請注意，自 2015 年 3 月起不支援EBSTRIM基礎儲存裝置。	
7.2.5	• AWS PV 儲存驅動程式中的穩定性修正：在某些情況下，AWS 光伏驅動程式可能會解除參照無效記憶體並導致系統故障。 • 生成崩潰轉儲時的穩定性修復：在某些情況下， AWS PV 驅動程序可能會在寫入崩潰轉儲時卡在競爭情況下。在此版本之前，僅能透過強制驅動程式停止並重新啟動，遺失記憶體傾印來解決。	
7.2.4	裝置 ID 持續性：此驅動程式修正會遮罩平台PCI裝置 ID，並強制系統一律顯示相同的裝置 ID，即使執行個體已移動也是如此。更一般的來說，修正會影響虛擬化處理程序呈現虛擬裝置的方式。此修正程式也包含對 AWS PV 驅動程式的共同安裝程式所做的修改，以便系統持續保留對應的虛擬裝置	
7.2.2	• 以目錄服務還原模式 (DSRM) 模式載入 AWS PV 驅動程式：目錄服務還原模式是 Windows Server 網域控制站的安全模式開機選項。 • 重新連接虛擬網路介面卡裝置時保留裝置 ID：此修正程式會強制系統檢查MAC位址對應並保留裝置 ID。此修正會確保介面卡在重新連接時保留其靜態設定。	

套件版本	詳細資訊	發行日期
7.2.1	• 於安全模式中執行：修正驅動程式無法在安全模式中載入的問題。之前，P AWS V 驅動程式只會在正常執行的系統中實例化。 • 新增磁碟至 Microsoft Windows 儲存集區 (Storage Pools)：先前我們會合成 Page 83 查詢。修正會停用 Page 83 支援。請注意，這不會影響目前在叢集環境中使用的儲存集區，因為 PV 磁碟並非有效的叢集磁碟。	
7.2.0	基本：P AWS V 基本版本。	

Citrix PV 驅動程式

Citrix PV 驅動程式存放在 `%ProgramFiles%\Citrix\XenTools` (32 位元執行個體) 或 `%ProgramFiles(x86)%\Citrix\XenTools` (64 位元執行個體) 目錄。

Citrix PV 驅動程式元件會在 Windows 登錄中的 `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services` 下列出。這些驅動程式元件如下：xenevtchn、xeniface、xennet、Xennet6、xensvc、xenvbd 及 xenvif。

Citrix 也有一個名為的驅動程式元件 XenGuestAgent，該元件會以 Windows 服務的形式執行。它處理的工作，例如關機和重新啟動事件API。您可以藉由從命令列執行 `Services.msc` 來存取及管理服務。

如果您在執行某些工作負載時遇到網路錯誤，您可能需要停用 Citrix PV 驅動程式的TCP卸載功能。如需詳細資訊，請參閱TCP卸載。

RedHat 光伏驅動器

RedHat 舊版執行個體支援驅動程式，但由於驅動程式限制，RAM因此不建議在超過 12GB 的較新執行個體上使用。具有超過 12GB 執行中 RedHat 驅動程式的RAM執行個體可能無法開機並變得無法存取。我們建議您將 RedHat 驅動程式升級到 Citrix 光伏驅動程式，然後將 Citrix 光伏驅動器升級為 AWS 光伏驅動器。

RedHat 驅動程式的來源檔案位於 `%ProgramFiles%\RedHat` (32 位元執行個體) 或 `%ProgramFiles(x86)%\RedHat` (64 位元執行個體) 目錄中。這兩個驅動程序是`rhelnet` RedHat 半虛擬化網絡驅動程序和`rhelscsi` RedHat SCSI微型端口驅動程序。

訂閱 通知

Amazon SNS 可以在 EC2 Windows 驅動程序的新版本發布時通知您。使用下列其中一種方法訂閱這些通知。

> **Note**
>
> 您必須為您訂閱的SNS主題指定「地區」。

從主控台訂閱EC2通知

1. 在 https://console.aws.amazon.com/sns/v3/ home 上打開 Amazon SNS 控制台。
2. 如有必要，請在導覽列中將「區域」變更為美國東部 (維吉尼亞北部)。您必須選取此區域，因為您訂閱的SNS通知位於此區域中。
3. 在導覽窗格中，選擇訂閱。
4. 選擇 Create subscription (建立訂閱)。
5. 在 Create subscription (建立訂閱) 對話方塊中，執行下列動作：
   a. 對於主題 ARN，請複製以下 Amazon 資源名稱 (ARN)：

      arn:aws:sns:us-east-1:801119661308:ec2-windows-drivers
   b. 針對 Protocol (通訊協定)，選擇 `Email`。
   c. 針對 Endpoint (端點)，輸入可用於接收通知的電子郵件地址。
   d. 選擇 Create subscription (建立訂閱)。
6. 您將會收到確認電子郵件。開啟電子郵件並遵循指示完成訂閱。

使用訂閱EC2通知 AWS CLI

若要使用訂閱EC2通知 AWS CLI，請使用下列命令。

```
aws sns subscribe --topic-arn arn:aws:sns:us-east-1:801119661308:ec2-
windows-drivers --region us-east-1 --protocol email --notification-
endpoint YourUserName@YourDomainName.ext
```

使用訂閱EC2通知 AWS Tools for PowerShell

若要使用 Windows 適用的工具訂閱EC2通知 PowerShell，請使用下列命令。

```
Connect-SNSNotification -TopicArn 'arn:aws:sns:us-east-1:801119661308:ec2-windows-drivers' -Region us-east-1 -Protocol email -Endpoint 'YourUserName@YourDomainName.ext'
```

每當發布新的 EC2 Windows 驅動程序時，我們都會向訂閱者發送通知。如果您不想再接收這些通知，請使用下列程序來取消訂閱。

退訂 Amazon EC2 視窗驅動程式通知

1. 在 https://console.aws.amazon.com/sns/v3/ home 上打開 Amazon SNS 控制台。
2. 在導覽窗格中，選擇 Subscriptions (訂閱)。
3. 選取訂閱的核取方塊，然後選取 Actions (動作)，Delete subscriptions (刪除訂閱)。出現確認提示時，請選擇 Delete (刪除)。

升級 EC2 Windows 執行個體上的光伏驅動

我們建議您安裝最新的 PV 驅動程式，以改善 EC2 Windows 執行個體的穩定性和效能。此頁面上的指示可協助您下載驅動程式套件並執行安裝程式。

確認 Windows 執行個體使用的驅動程式

開啟 Control Panel (控制台) 中的 Network Connections (網路連線)，並檢視 Local Area Connection (區域連線)。檢查驅動程式是否為下列項目的其中之一：

- AWS 光伏網路裝置
- Citrix PV 乙太網路卡
- RedHat 光伏NIC驅動器

或者，您可以檢查 `pnputil -e` 命令的輸出。

系統要求

請務必檢查下載中的 `readme.txt` 檔案是否有系統需求。

目錄

- 使用代理商升級 Windows 伺服器執行個體 (AWS PV 升級)
- 手動升級 Windows 伺服器執行個體 (AWS PV 升級)

- 升級網域控制站 (AWS PV 升級)
- 升級 Windows Server 2008 及 2008 R2 執行個體 (RedHat 升級至 Citrix PV)
- 升級您的 Citrix Xen 客體代理程式服務

使用代理商升級 Windows 伺服器執行個體 (AWS PV 升級)

您可以使用「代理商」功能來安裝或升級 AWS PV 驅動程式套件。 AWS Systems Manager安裝或升級可以執行一次，也可以按照排程安裝或更新。此代理商套件不支援「安裝類型 **`In-place update`**」選項。

> **Important**
>
> 如果您的執行個體為網域控制站，請參閱升級網域控制站 (AWS PV 升級)。網域控制器執行個體的升級程序與標準版本的 Windows 不同。

1. 我們建議您建立備份，以防您需要復原變更。

   > **Tip**
   >
   > 而不是AMI從 Amazon EC2 控制台創建，您可以使用 Systems Manager 自動化創建AMI使用 `AWS-CreateImage` runbook。若要取得更多資訊，請參閱AWS Systems Manager 自動化工作流程簿參考使用者指南AWS-CreateImage中。

   a. 停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。停止執行個體之前，請確認您已將所需的任何資料從執行個體存放磁碟區複製到持續性儲存，例如 Amazon EBS 或 Amazon S3。
   b. 在導覽窗格中，選擇 Instances (執行個體)。
   c. 選取需要升級驅動程式的執行個體，並依序選取 Instance state (執行個體狀態)、Stop instance (停止執行個體)。
   d. 停止執行個體後，請選取執行個體，依序選取 Actions (動作)、Image and templates (映像和範本)，然後選取 Create image (建立映像)。
   e. 選擇 Instance state (執行個體狀態)、Start instance (啟動執行個體)。
2. 使用遠端桌面連線到執行個體。如需詳細資訊，請參閱the section called “使 Connect 用RDP戶端連線”。

3. 建議您讓所有非系統磁碟離線，並且在磁碟管理中記下次要磁碟的任何磁碟機代號映射，之後再執行此升級。如果您要執行 AWS PV 驅動程式的就地更新，則不需要執行此步驟。我們也建議在服務主控台中將非必要的服務設為 Manual (手動) 啟動。
4. 如需有關如何使用「代理商」安裝或升級 AWS PV 驅動程式套件的指示，請參閱「使用AWS Systems Manager 者指南」中的「安裝或更新套件」中的程序。
5. 對於「名稱」，選擇AWSPVDriver。
6. 對於安裝類型，選取解除安裝並重新安裝。
7. 視需要設定套裝軟體的其他參數，並使用中參照的程序執行安裝或升級Step 4。

   執行散發程式套件之後，執行個體會自動重新開機，然後升級驅動程式。執行個體最多可能會在 15 分鐘內無法使用。
8. 升級完成後，執行個體在 Amazon EC2 主控台中通過兩項運作狀態檢查後，請使用遠端桌面連線至執行個體，確認已安裝新驅動程式。
9. 連線完成後，執行下列 PowerShell 命令：

```
Get-ItemProperty HKLM:\SOFTWARE\Amazon\PVDriver
```

10. 確認驅動程式版本與驅動程式版本歷史記錄表格中列出的最新版本相同。如需詳細資訊，請參閱AWS 光伏驅動器封裝歷史開啟磁碟管理以檢閱任何離線輔助磁碟區，並將其與中所述磁碟機代號相對應的線上Step 3。

如果您先前停TCP卸載用 Netsh 的 Citrix 光伏驅動程式，我們建議您在升級到 AWS PV 驅動程式後重新啟用此功能。TCP AWS 光伏驅動程式中不存在 Citrix 驅動程式的卸載問題。因此，TCP卸載可提供更好的 AWS PV 驅動器效能。

如果您之前已將靜態 IP 位址或DNS組態套用至網路介面，則可能需要在升級 AWS PV 驅動程式之後重新套用靜態 IP 位址或DNS組態。

手動升級 Windows 伺服器執行個體 (AWS PV 升級)

請使用下列程序在視窗伺服器 2008 R2、視窗伺服器 2012 年 R2、視窗伺服器 2012 年 R2、視窗伺服器 2012 年 R2、視窗伺服器 2012 年 R2、視窗伺服器 2016 年、視窗伺服器 2016 年、視窗伺服器 2019 或視窗伺服器 2022 上執行 AWS 光伏驅動程式的就地升級，或將光伏驅動程式 AWS 此升級不適用於 RedHat 驅動程式或其他版本的 Windows 伺服器。

某些舊版 Windows 伺服器無法使用最新的驅動程式。若要確認您的作業系統所使用的驅動程式版本，請參閱 Windows 執行個體的全虛擬驅動程式 頁面中的驅動程式版本表。

> **Important**
>
> 如果您的執行個體為網域控制站，請參閱升級網域控制站 (AWS PV 升級)。網域控制器執行個體的升級程序與標準版本的 Windows 不同。

手動升級 AWS PV 驅動程式

1. 我們建議您建立備份，以防您需要復原變更。

   > **Tip**
   >
   > 而不是AMI從 Amazon EC2 控制台創建，您可以使用 Systems Manager 自動化創建AMI使用 `AWS-CreateImage` runbook。若要取得更多資訊，請參閱AWS Systems Manager 自動化工作流程簿參考使用者指南AWS-CreateImage中。

   a. 停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。停止執行個體之前，請確認您已將所需的任何資料從執行個體存放磁碟區複製到持續性儲存，例如 Amazon EBS 或 Amazon S3。

   b. 在導覽窗格中，選擇 Instances (執行個體)。

   c. 選取需要升級驅動程式的執行個體，並依序選取 Instance state (執行個體狀態)、Stop instance (停止執行個體)。

   d. 停止執行個體後，請選取執行個體，依序選取 Actions (動作)、Image and templates (映像和範本)，然後選取 Create image (建立映像)。

   e. 選擇 Instance state (執行個體狀態)、Start instance (啟動執行個體)。

2. 使用遠端桌面連線到執行個體。

3. 建議您讓所有非系統磁碟離線，並且在磁碟管理中記下次要磁碟的任何磁碟機代號映射，之後再執行此升級。如果您要執行 AWS PV 驅動程式的就地更新，則不需要執行此步驟。我們也建議在服務主控台中將非必要的服務設為 Manual (手動) 啟動。

4. 將最新的驅動程式套件下載到執行個體。

   或者，執行下列 PowerShell 命令：

   ```
   Invoke-WebRequest https://s3.amazonaws.com/ec2-windows-drivers-downloads/AWSPV/
   Latest/AWSPVDriver.zip -outfile $env:USERPROFILE\pv_driver.zip
   ```

```
Expand-Archive $env:userprofile\pv_driver.zip -DestinationPath
 $env:userprofile\pv_drivers
```

> **Note**
>
> 如果您在下載檔案時收到錯誤訊息，而且您使用的是 Windows 伺服器 2016 或更早版本，則可能需要為 PowerShell 終端機啟用 TLS 1.2。您可以使用以下命令為當前 PowerShell 會話啟用 TLS 1.2，然後再試一次：
>
> ```
> [Net.ServicePointManager]::SecurityProtocol =
> [Net.SecurityProtocolType]::Tls12
> ```

5. 解壓縮資料夾的內容，然後執行 `AWSPVDriverSetup.msi`。

執行之後MSI，執行個體會自動重新開機，然後升級驅動程式。執行個體最多可能會在 15 分鐘內無法使用。升級完成且執行個體在 Amazon EC2 主控台中通過兩項運作狀態檢查之後，您可以使用遠端桌面連線至執行個體，然後執行下列 PowerShell 命令來驗證是否已安裝新驅動程式：

```
Get-ItemProperty HKLM:\SOFTWARE\Amazon\PVDriver
```

確認驅動程式版本與驅動程式版本歷史記錄表格中列出的最新版本相同。如需詳細資訊，請參閱AWS光伏驅動器封裝歷史開啟磁碟管理以檢閱任何離線輔助磁碟區，並將其與中所述磁碟機代號相對應的線上Step 3。

如果您先前停TCP卸載用 Netsh 的 Citrix 光伏驅動程式，我們建議您在升級到 AWS PV 驅動程式後重新啟用此功能。TCP AWS 光伏驅動程式中不存在 Citrix 驅動程式的卸載問題。因此，TCP卸載可提供更好的 AWS PV 驅動器效能。

如果您之前已將靜態 IP 位址或DNS組態套用至網路介面，則可能需要在升級 AWS PV 驅動程式之後重新套用靜態 IP 位址或DNS組態。

升級網域控制站 (AWS PV 升級)

在網域控制站上使用下列程序來執行 PV 驅動程式的就地升級，或從 Citrix AWS PV 驅動程式升級為 AWS PV 驅動程式。

升級網域控制站

1. 我們建議您建立網域控制站的備份，以防您需要復原變更。不支援使用AMI做為備份。如需詳細資訊，請參閱 Microsoft 文件中的虛擬網域控制站的備份和還原考慮。
2. 執行下列命令，將 Windows 設定為開機進入目錄服務還原模式 (DSRM)。

> **Warning**
> 執行此命令之前，請確認您知道DSRM密碼。您將需要此資訊，才能在升級完成且執行個體自動重新開機後登入您的執行個體。

```
bcdedit /set {default} safeboot dsrepair
```

PowerShell:

```
PS C:\> bcdedit /set "{default}" safeboot dsrepair
```

系統必須開機，DSRM因為升級公用程式會移除 Citrix PV 儲存驅動程式，以便安裝 AWS PV 驅動程式。因此，建議您在磁碟管理中記下次要磁碟的任何磁碟機代號和資料夾映射。當 Citrix PV 儲存體驅動程式不存在時，將無法偵測到輔助磁碟機。使用次要磁碟機上NTDS資料夾的網域控制站將無法開機，因為未偵測到次要磁碟。

> **Warning**
> 在您執行此命令後，請不要手動重新開機系統。系統將無法連接，因為 Citrix 光伏驅動程式不支援。DSRM

3. 執行下列命令，將 **`DisableDCCheck`** 新增到登錄：

```
reg add HKLM\SOFTWARE\Wow6432Node\Amazon\AWSPVDriverSetup /v DisableDCCheck /t
 REG_SZ /d true
```

4. 將最新的驅動程式套件下載到執行個體。
5. 解壓縮資料夾的內容，然後執行 `AWSPVDriverSetup.msi`。

   執行之後MSI，執行個體會自動重新開機，然後升級驅動程式。執行個體最多可能會在 15 分鐘內無法使用。

6. 升級完成且執行個體在 Amazon EC2 主控台中通過兩項運作狀態檢查後，請使用遠端桌面連線至執行個體。開啟磁碟管理來檢閱任何離線的輔助磁碟區，並根據稍早記下的對應磁碟機代號和資料夾映射讓磁碟區回到線上狀態。

   您必須以下列格式指定使用者名稱主機名稱\ admin 來連線至執行個體。例如，Win2K12TestBox\管理員。

7. 執行下列命令以移除開DSRM機組態：

```
bcdedit /deletevalue safeboot
```

8. 重新啟動執行個體。
9. 若要完成升級程序，請確認已安裝新的驅動程式。在裝置管理員中，於 Storage Controllers (儲存控制器) 下，尋找 AWS PV Storage Host Adapter (AWS PV 儲存體主機控制器)。確認驅動程式版本與驅動程式版本歷史記錄表格中列出的最新版本相同。如需詳細資訊，請參閱AWS 光伏驅動器封裝歷史。
10. 執行下列命令，從登錄刪除 **DisableDCCheck**：

```
reg delete HKLM\SOFTWARE\Wow6432Node\Amazon\AWSPVDriverSetup /v DisableDCCheck
```

> **Note**
>
> 如果您先前停TCP卸載用 Netsh 的 Citrix 光伏驅動程式，我們建議您在升級到 AWS PV 驅動程式後重新啟用此功能。TCP AWS 光伏驅動程式中不存在 Citrix 驅動程式的卸載問題。因此，TCP卸載可提供更好的 AWS PV 驅動器效能。

升級 Windows Server 2008 及 2008 R2 執行個體 (RedHat 升級至 Citrix PV)

在開始將 RedHat 驅動程式升級到 Citrix PV 驅動程式之前，請務必執行以下操作：

- 安裝最新版本的EC2Config服務。如需詳細資訊，請參閱安裝 EC2Config 的最新版本。
- 請確認您已安裝視窗 PowerShell 3.0。若要驗證您已安裝的版本，請在 PowerShell視窗中執行下列命令：

```
PS C:\> $PSVersionTable.PSVersion
```

視窗 PowerShell 3.0 捆綁在視窗管理框架（WMF）3.0 版安裝包中。如果您需要安裝視窗 PowerShell 3.0，請參閱 Microsoft 下載中心的視窗管理架構 3.0。

- 備份執行個體的重要資訊，或AMI從執行個體建立。若要取得有關建立的更多資訊AMI，請參閱創建一個 Amazon 支EBS持 AMI。

> **Tip**
>
> 而不是AMI從 Amazon EC2 控制台創建，您可以使用 Systems Manager 自動化創建AMI使用 `AWS-CreateImage` runbook。若要取得更多資訊，請參閱AWS Systems Manager 自動化工作流程簿參考使用者指南AWS-CreateImage中。

如果您創建了一個AMI，請確保執行以下操作：

- 記下您的密碼。
- 請勿手動執行 Sysprep 工具或使用服務。EC2Config
- 將您的乙太網路介面卡設定為使用自動取得 IP 位址DHCP。

升級 RedHat 驅動程式

1. 連線到您的執行個體，並以本機管理員登入。如需連線至執行個體的詳細資訊，請參閱使用 Connect 至您的 Windows 執行個體 RDP。
2. 在您的執行個體中，下載 Citrix PV 升級套件。
3. 將升級套件的內容解壓縮到您選擇的位置。
4. 按兩下 Upgrade.bat 檔案。若您收到安全警告，請選擇 Run (執行)。
5. 在 Upgrade Drivers (升級驅動程式) 對話方塊中，檢閱資訊並於準備好開始升級時選擇 Yes (是)。
6. 在「適用於 Windows 解除安裝程式的 Red Hat 準虛擬化 Xen 驅動程式」對話方塊中，選擇「是」以移除軟體。 RedHat 您的執行個體將會重新開機。

> **Note**
>
> 若您沒有看到解除安裝程式對話方塊，請在 Windows 任務列中選擇 Red Hat Paravirtualize (Red Hat 全虛擬化)。

7. 檢查執行個體已重新開機並準備好可供使用。

   a. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。

   b. 在 Instances (執行個體) 頁面上，依序選取 Actions (動作)、Monitor and troubleshoot (監視和故障診斷)，然後選取 Get system log (取得系統記錄檔)。

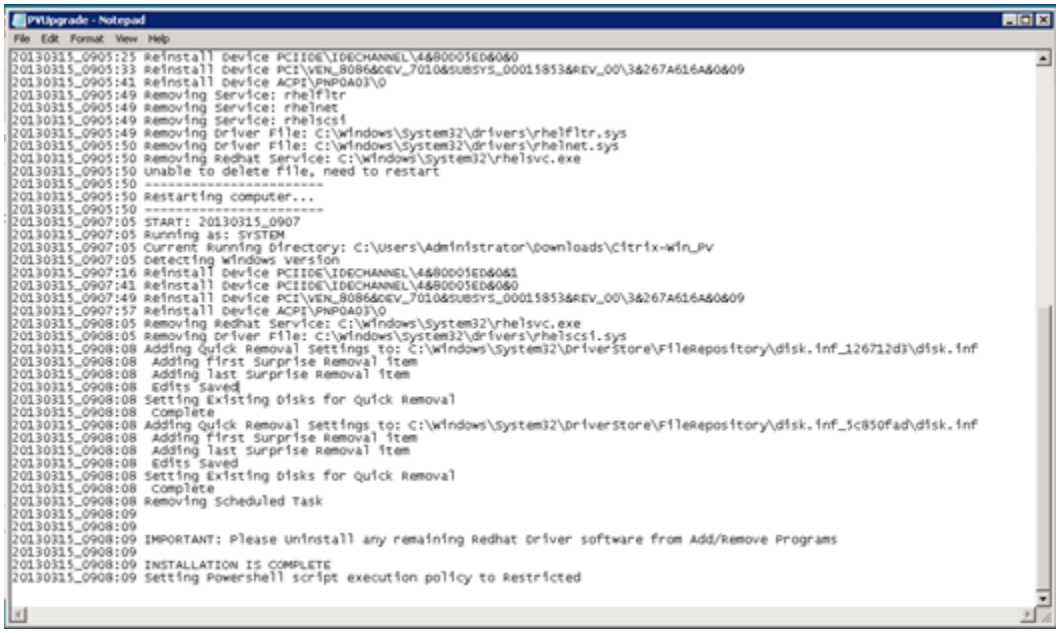
   c. 升級操作應該會重新開機伺服器 3 或 4 次。您可以在日誌檔案中，查看顯示的 `Windows is Ready to use` 次數來了解。

```
    Microsoft Windows NT 6.0.6002 Service Pack 2 (en-US)
 Ec2Config service v2.1.9.0
  RedHat PV NIC Driver v1.3.10.0
2013/03/15 17:11:01Z: Waiting for meta-data accessibility...
2013/03/15 17:11:02Z: Meta-data is now available.
<RDPCERTIFICATE>
<THUMBPRINT>D6BFD64F21359516C781CA7DF2821C5EFC35648A</THUMBPRINT>
</RDPCERTIFICATE>
<Username>Administrator</Username>
<Password>
L79ThJPF8LyIL38IZht0FBrjet3vnT2csTiU/XGVMRCH7kQtBnznAnXrKd1sirX1xl9BwVMsd9b38jFJqv01IUpgNNJRZoCDc7IbUw
</Password>
2013/03/15 17:11:30Z: Product activation was successful.
2013/03/15 17:11:32Z: Message: Windows is Ready to use
Microsoft Windows NT 6.0.6002 Service Pack 2 (en-US)
 Ec2Config service v2.1.9.0
2013/03/15 21:04:24Z: There was an exception writing driver information to console: System.Exception:
   at Ec2Config.Service1.Go()
2013/03/15 21:04:35Z: Waiting for meta-data accessibility...
2013/03/15 21:04:40Z: Meta-data is now available.
<RDPCERTIFICATE>
<THUMBPRINT>D6BFD64F21359516C781CA7DF2821C5EFC35648A</THUMBPRINT>
</RDPCERTIFICATE>
2013/03/15 21:05:08Z: Product activation was successful.
2013/03/15 21:05:09Z: Message: Windows is Ready to use
Microsoft Windows NT 6.0.6002 Service Pack 2 (en-US)
 Ec2Config service v2.1.9.0
  Citrix PV Ethernet Adapter v5.9.960.49119
2013/03/15 21:07:20Z: Waiting for meta-data accessibility...
2013/03/15 21:07:21Z: Meta-data is now available.
<RDPCERTIFICATE>
<THUMBPRINT>D6BFD64F21359516C781CA7DF2821C5EFC35648A</THUMBPRINT>
</RDPCERTIFICATE>
2013/03/15 21:07:27Z: Message: Windows is Ready to use
```

8. 連線到您的執行個體，並以本機管理員登入。

9. 關閉 Red Hat Paravirtualized Xen Drivers for Windows uninstaller (Windows Red Hat 全虛擬化 Xen 驅動程式解除安裝程式) 對話方塊。

10. 確認安裝已完成。導覽至您先前解壓縮的 `Citrix-WIN_PV` 資料夾，開啟 `PVUpgrade.log` 檔案，然後檢查是否有 `INSTALLATION IS COMPLETE` 的文字。



升級您的 Citrix Xen 客體代理程式服務

若您在 Windows Server 上使用 Citrix PV 驅動程式，您可以升級 Citrix Xen 客體代理程式服務。此 Windows 服務會從處理的工作，例如關機和重新啟動事件API。您可以在任何版本的 Windows Server 上執行此升級套件，只要執行個體是執行 Citrix PV 驅動程式即可。

> **Important**
>
> 對於 Windows 伺服器 2008 R2 及更新版本，我們建議您升級至包含客體代理程式更新的 AWS 光伏驅動程式。

在開始升級驅動程式之前，請務必備份執行個體上的重要資訊，或AMI從執行個體建立一個資訊。若要取得有關建立的更多資訊AMI，請參閱創建一個 Amazon 支EBS持 AMI。

> **Tip**
>
> 而不是AMI從 Amazon EC2 控制台創建，您可以使用 Systems Manager 自動化創建AMI使用 `AWS-CreateImage` runbook。若要取得更多資訊，請參閱AWS Systems Manager 自動化工作流程簿參考使用者指南AWS-CreateImage中。

如果您創建了一個AMI，請確保執行以下操作：

- 請勿在服務中啟用 Sysprep 工具。EC2Config
- 記下您的密碼。
- 將乙太網路介面卡設定為DHCP。

升級您的 Citrix Xen 客體代理程式服務

1. 連線到您的執行個體，並以本機管理員登入。如需連線至執行個體的詳細資訊，請參閱使用 Connect 至您的 Windows 執行個體 RDP。
2. 在您的執行個體上，下載 Citrix 升級套件。
3. 將升級套件的內容解壓縮到您選擇的位置。
4. 按兩下 Upgrade.bat 檔案。若您收到安全警告，請選擇 Run (執行)。
5. 在 Upgrade Drivers (升級驅動程式) 對話方塊中，檢閱資訊並於準備好開始升級時選擇 Yes (是)。
6. 當升級完成時，將會開啟 `PVUpgrade.log` 檔案，其中包含 `UPGRADE IS COMPLETE` 的文字。
7. 將執行個體重新開機。

排解 Windows 執行個體上的光伏驅動

以下是舊版 Amazon EC2 映像和 PV 驅動程式可能遇到的問題的解決方案。

目錄

- Windows Server 2012 R2 在執行個體重新開機後遺失網路及儲存體連線能力
- TCP卸載
- 時間同步
- 利用超過 20,000 個磁碟IOPS因CPU瓶頸而降級的工作負載

Windows Server 2012 R2 在執行個體重新開機後遺失網路及儲存體連線能力

> **Important**
>
> 只有AMIs在 2014 年 9 月之前提供時，才會發生這個問題。

2014 年 9 月 10 日之前提供的 Windows 伺服器 2012 R2 Amazon 電腦映像 (AMIs) 可能會在執行個體重新開機後失去網路和儲存連線能力。 AWS Management Console 系統記錄檔中的錯誤指出：「無法偵測控制台輸出的 PV 驅動程式詳細資料」。連線能力遺失是由「隨插即用清理」功能造成的。此功能

會每 30 天掃描並停用非使用中的系統裝置。此功能會錯誤地將EC2網路裝置識別為非作用中，並將其從系統中移除。當發生此狀況時，執行個體便會在重新開機後遺失網路連線能力。

針對您懷疑受到此問題影響的系統，您可以下載及執行就地驅動程式升級。若您無法執行就地驅動程式升級，您可以執行協助程式指令碼。指令碼會判斷您的執行個體是否受到影響。如果受到影響，且尚未移除 Amazon EC2 網路裝置，則指令碼會停用隨插即用清理掃描。若已移除網路裝置，指令碼會修復裝置，停用隨插即用清理掃描，讓您的執行個體重新開機並啟用網路連線能力。

內容

- 選擇如何修正問題
- 方法 1 - 增強聯網
- 方法 2 - 登錄組態
- 執行補救指令碼

選擇如何修正問題

有兩種方法可以還原受此問題影響執行個體的網路及儲存體連線能力。選擇下列其中一種方法：

方法	必要條件	程序概觀
方法 1 - 增強聯網	增強型聯網只能在需要 C3 執行個體類型的虛擬私有雲 (VPC) 中使用。若伺服器目前使用的並非 C3 執行個體類型，您必須暫時變更它。	您將伺服器執行個體類型變更為 C3 執行個體。增強聯網便可讓您連線到受影響的執行個體並修正問題。在修正問題之後，您再將執行個體變更回原始的執行個體類型。此方法通常比方法 2 更快，也較不容易導致使用者錯誤。執行 C3 執行個體時，您將需要支付額外的費用。
方法 2 - 登錄組態	建立或存取第二個伺服器的能力。變更登錄設定的能力。	您將根磁碟區從受影響的執行個體分離，再將其連接到不同的執行個體、連線，並在登錄中進行變更。執行額外的伺服器時，您將需要支付額外的費用。此方法比方法 1 慢，但此

方法	必要條件	程序概觀
		方法可在方法 1 無法解決問題時使用。

方法 1 - 增強聯網

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 尋找受影響的執行個體。選取執行個體，並依序選取 Instance state (執行個體狀態)、Stop instance (停止執行個體)。

   > **Warning**
   > 停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。若要保留執行個體存放區磁碟區的資料，請將資料備份至持久性儲存。

4. 在執行個體停止後，建立備份。選取執行個體，依序選取 Actions (動作)、Image and templates (映像和範本)，然後選取 Create image (建立映像)。
5. 將執行個體類型變更為任何 C3 執行個體類型。
6. 啟動實例。
7. 使用遠端桌面 Connect 至執行個體，然後將 AWS PV 驅動程式升級套件下載至執行個體。
8. 解壓縮資料夾的內容，然後執行 `AWSPVDriverSetup.msi`。

   執行之後MSI，執行個體會自動重新開機，然後升級驅動程式。執行個體最多可能會在 15 分鐘內無法使用。

9. 升級完成且執行個體在 Amazon EC2 主控台中通過兩項運作狀態檢查後，請使用遠端桌面連線至執行個體，並驗證是否已安裝新驅動程式。在裝置管理員中，於 Storage Controllers (儲存控制器) 下，尋找 AWS PV Storage Host Adapter (AWS PV 儲存體主機控制器)。確認驅動程式版本與驅動程式版本歷史記錄表格中列出的最新版本相同。如需詳細資訊，請參閱AWS 光伏驅動器封裝歷史。
10. 停止執行個體，並將執行個體變更回原始的執行個體類型。
11. 啟動執行個體，繼續一般的使用。

方法 2 - 登錄組態

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 尋找受影響的執行個體。選取執行個體，並依序選取 Instance state (執行個體狀態)、Stop instance (停止執行個體)。

> **Warning**
>
> 停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。若要保留執行個體存放區磁碟區的資料，請將資料備份至持久性儲存。

4. 選擇 Launch Instance (啟動執行個體)，在與受影響的執行個體相同的可用區域中建立暫時性的 Windows Server 2008 或 Windows Server 2012 執行個體。請不要建立 Windows Server 2012 R2 執行個體

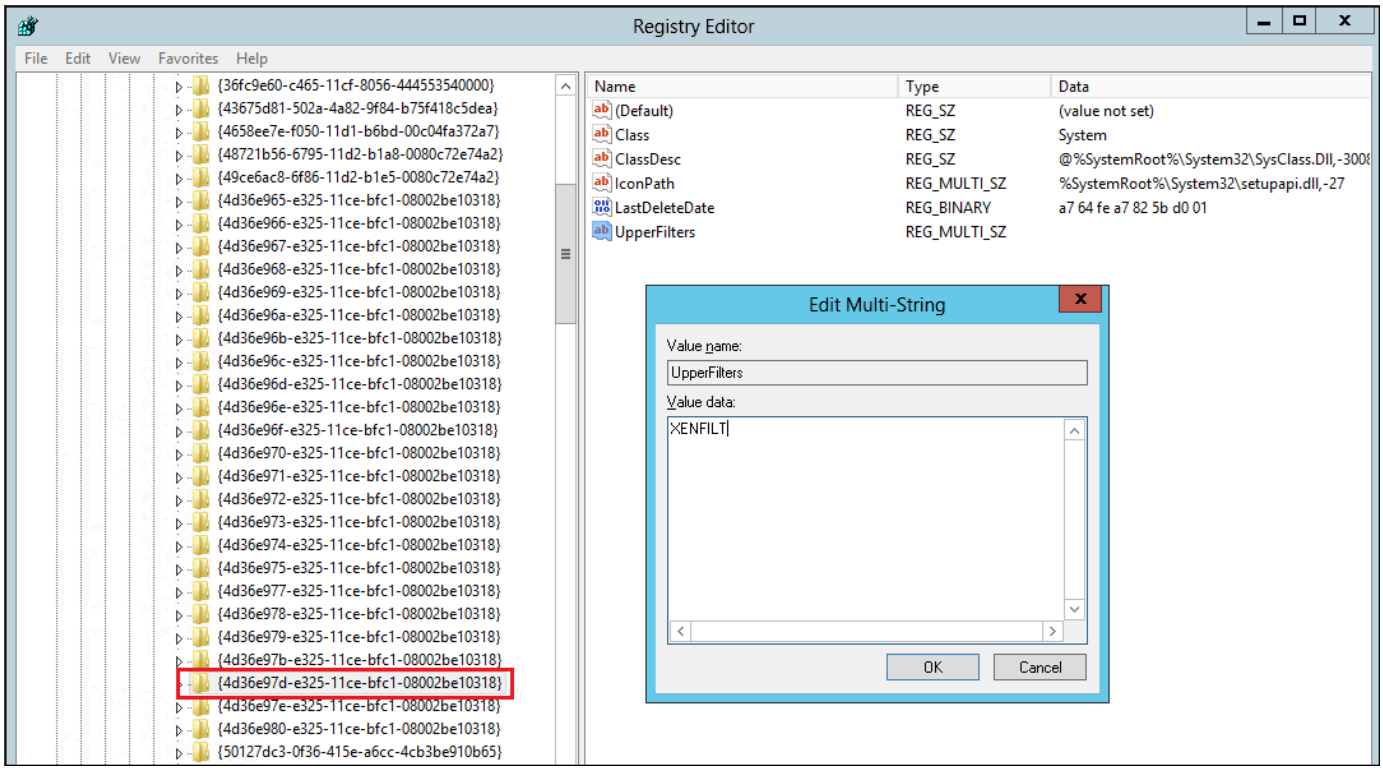
> **Important**
>
> 若您沒有在與受影響的執行個體相同的可用區域內建立執行個體，您將無法將受影響執行個體的根磁碟區連接到新的執行個體。

5. 在導覽窗格中，選擇 Volumes (磁碟區)。
6. 尋找受影響執行個體的根磁碟區。分開磁碟區，再將磁碟區連接至先前建立的暫時執行個體。使用預設裝置名稱 (xvdf) 連接它。
7. 使用遠端桌面連接至暫時性執行個體，然後使用 Disk Management 公用程式將磁碟區變成可用。
8. 在暫時性執行個體上，開啟 Run (執行) 對話方塊，輸入 **`regedit`**，然後按 Enter 鍵。
9. 在 [登錄編輯程式] 瀏覽窗格中，選擇 [HKEY_Local_Machine]，然後從 [檔案] 功能表選擇 [載入 Hive]。
10. 在 Load Hive (載入 Hive) 對話方塊中，導覽至 Affected Volume (受影響的裝置)\Windows\System32\config\System 然後在 Key Name (索引鍵名稱) 對話方塊中輸入暫時性的名稱。例如，輸入 OldSys。
11. 在登錄編輯程式的導覽窗格中，尋找下列索引鍵：

    HKEY_LOCAL_MACHINE***`your_temporary_key_name`**\ ControlSet 001\ 控制項\ 類別\ 類別

    HKEY_LOCAL_MACHINE***`your_temporary_key_name`**\ ControlSet 001\ 控制項\ 類別\ 類別

12. 針對每個機碼 UpperFilters，連按兩下，輸入值XENFILT，然後選擇 [確定]。



13. 尋找下列索引鍵：

    HKEY_LOCAL_MACHINE***your_temporary_key_name**\ ControlSet 001\ 服務\XENBUS\ 參數

14. 使用名稱 ActiveDevice 和下列值建立新字串 (REG_SZ)：

    PCI\ VEN _5853& _0001& DEV SUBSYS REV

15. 尋找下列索引鍵：

    HKEY_LOCAL_MACHINE***your_temporary_key_name**\ ControlSet 001\ 服務項目\ XENBUS

16. 將 Count 從 0 變更為 1。
17. 尋找並刪除下列索引鍵：

    HKEY_LOCAL_MACHINE***your_temporary_key_name**\ ControlSet 001\ 服務\ 未來\ StartOverride

    HKEY_LOCAL_MACHINE ***your_temporary_key_name**\ ControlSet 001\ 服務\ 超級文件\ StartOverride

18. 在登錄編輯程式導覽窗格中，選擇一開始開啟登錄編輯程式時建立的暫時性機碼。
19. 從 File (檔案) 選單中，選擇 Unload Hive (解除載入 Hive)。

20. 在磁碟管理公用程式中，選擇先前連接的磁碟機，開啟內容 (按右鍵) 選單，然後選擇 Offline (離線)。
21. 在 Amazon EC2 主控台中，將受影響的磁碟區與暫存執行個體分離，然後使用裝置名稱 /dev/sda1 重新連接到 Windows 伺服器 2012 R2 執行個體。您必須指定此裝置名稱，將磁碟區指定為根磁碟區。
22. 啟動實例。
23. 使用遠端桌面 Connect 至執行個體，然後將 AWS PV 驅動程式升級套件下載至執行個體。
24. 解壓縮資料夾的內容，然後執行 `AWSPVDriverSetup.msi`。

    執行之後MSI，執行個體會自動重新開機，然後升級驅動程式。執行個體最多可能會在 15 分鐘內無法使用。
25. 升級完成且執行個體在 Amazon EC2 主控台中通過兩項運作狀態檢查後，請使用遠端桌面連線至執行個體，並驗證是否已安裝新驅動程式。在裝置管理員中，於 Storage Controllers (儲存控制器) 下，尋找 AWS PV Storage Host Adapter (AWS PV 儲存體主機控制器)。確認驅動程式版本與驅動程式版本歷史記錄表格中列出的最新版本相同。如需詳細資訊，請參閱AWS 光伏驅動器封裝歷史。
26. 刪除或停止您在此程序中建立的暫時性執行個體。

執行補救指令碼

若您無法執行就地驅動程式升級或遷移至較新的執行個體，您可以執行補救指令碼來修正隨插即用清理任務造成的問題。

執行補救指令碼

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取您要執行補救指令碼的執行個體。依序選擇 Instance state (執行個體狀態)、Stop instance (停止執行個體)。

> **Warning**
>
> 停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。若要保留執行個體存放區磁碟區的資料，請將資料備份至持久性儲存。

4. 在執行個體停止後，建立備份。選取執行個體，依序選取 Actions (動作)、Image and templates (映像和範本)，然後選取 Create image (建立映像)。
5. 依序選擇 Instance state (執行個體狀態)、Start instance (啟動執行個體)。
6. 使用遠端桌面 Connect 至執行個體，然後將 RemediateDriverIssue .zip 資料夾下載至執行個體。
7. 解壓縮資料夾的內容。
8. 根據 Readme.txt 檔案中的說明執行補救指令碼。檔案位於您解 RemediateDriverIssue壓縮 .zip 的資料夾中。

TCP卸載

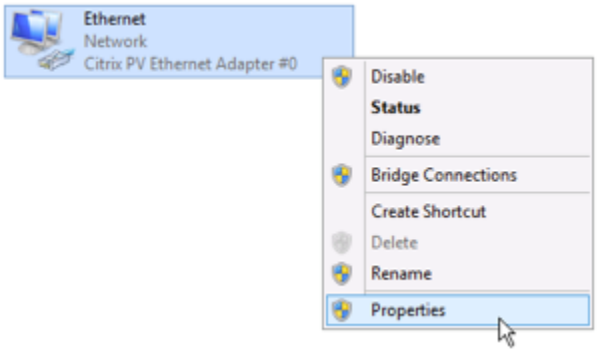
> **Important**
>
> 此問題不適用於執行 AWS PV 或 Intel 網路驅動程式的執行個體。

依預設，Windows 中的 Citrix 光伏驅動程式會啟用TCP卸載功能。AMIs如果您遇到傳輸層級錯誤或封包傳輸錯誤 (如 Windows 效能監視器上所見) (例如，當您執行特定SQL工作負載時)，您可能需要停用此功能。

> **Warning**
>
> 停用TCP卸載可能會降低執行個體的網路效能。

若要停用TCP卸載視窗伺服器 2012 年和 2008

1. 連線到您的執行個體，並以本機管理員登入。
2. 若您使用 Windows Server 2012，請按 Ctrl+Esc 存取 Start (開始) 畫面，然後選擇 Control Panel (控制台)。若您使用 Windows Server 2008，請選取 Start (開始)，然後選取 Control Panel (控制台)。
3. 依序選擇 Network and Internet (網路和網際網路) 和 Network and Sharing Center (網路和共用中心)。
4. 選擇 Change adapter settings (變更介面卡設定)。
5. 以滑鼠右鍵按一下 Citrix PV Ethernet Adapter #0 (Citrix PV 乙太網路卡 #0)，然後選取 Properties (屬性)。



6. 在 Local Area Connection Properties (區域連線屬性) 對話方塊中，選擇 Configure (設定) 以開啟 Citrix PV Ethernet Adapter #0 Properties (Citrix PV 乙太網路卡 #0 屬性) 對話方塊。
7. 在「進階」索引標籤上，停用每個屬性，除了「正確TCP/UDP總和檢查碼值」。若要停用屬性，請從 Property (屬性) 中選取，然後從 Value (值) 選取 Disabled (停用)。
8. 選擇 OK (確定)。
9. 從命令列視窗中執行下列命令。

```
netsh int ip set global taskoffload=disabled
netsh int tcp set global chimney=disabled
netsh int tcp set global rss=disabled
netsh int tcp set global netdma=disabled
```

10. 重新啟動執行個體。

時間同步

在 2013.02.13 視窗發行之前AMI，Citrix Xen 客體代理程式可能會錯誤地設定系統時間。這可能會導致您的DHCP租約過期。若您在連線到您的執行個體時發生問題，您可能需要更新代理程式。

若要判斷您是否已具有更新後的 Citrix Xen 客體代理程式，請檢查 `C:\Program Files\Citrix\XenGuestAgent.exe` 檔案是否來自 2013 年 3 月。若此檔案的日期在該日期之前，請更新 Citrix Xen 客體代理程式服務。如需詳細資訊，請參閱升級您的 Citrix Xen 客體代理程式服務。

利用超過 20,000 個磁碟IOPS因CPU瓶頸而降級的工作負載

如果您使用執行 AWS PV 驅動程式運用率超過 20,000 的 Windows 執行個體，並且您遇到錯誤檢查碼 IOPS，則可能會受到此問題的影響`0x9E: USER_MODE_HEALTH_MONITOR`。

AWS 光伏驅動程式中的磁碟讀取和寫入 (IOs) 分為兩個階段：IO 準備和 IO 完成。依預設，準備階段會在單一任意核心上執行。完成階段會在核心 `0` 上執行。處理 IO 所需的運算量因其大小和其他屬性

而有所差異。一些在準備階段IOs使用更多的計算，而另一些則在完成階段。當執行個體磁碟機超過 20,000 個時IOPS，準備或完成階段可能會導致瓶頸，CPU而執行的容量為 100%。準備或完成階段是否成為瓶頸，取決於應用程式所IOs使用的屬性。

從 AWS PV 驅動器 8.4.0 開始，準備階段和完成階段的負載可以分散到多個核心，從而消除瓶頸。每個應用程式使用不同的 IO 屬性。因此，套用下列其中一個組態可能會提升、降低或不會影響應用程式的效能。套用任何這些組態之後，請監控應用程式，以確認其符合您所需的效能。

1. 必要條件

   開始此疑難排解程序之前，請先確認下列先決條件：

   - 您的執行個體使用 AWS PV 驅動程式 8.4.0 版或更新版本。若要升級，請參閱升級 EC2 Windows 執行個體上的光伏驅動。
   - 您可以RDP存取執行個體。如需使用連線至 Windows 執行個體的步驟RDP，請參閱使用用RDP戶端 Connect 至您的 Windows 執行個體。
   - 您擁有執行處理的管理員存取權。

2. 觀察執行個體的CPU負載

   您可以使用 Windows 工作管理員檢視每個負載，CPU以判斷磁碟 IO 的潛在瓶頸。

   1. 確認您的應用程式正在執行，並處理與生產工作負載類似的流量。
   2. 使用 Connect 至您的執行個體RDP。
   3. 選擇執行個體上的 Start (開始) 功能表。
   4. 請在 Start (開始) 功能表中輸入 `Task Manager`，以開啟 Task Manager (任務管理員)。
   5. 如果 Task Manager (任務管理員) 顯示 Summary View (摘要檢視)，請選擇 More details (更多詳細資訊)，以展開詳細檢視。
   6. 選擇 Performance (效能) 索引標籤。
   7. CPU在左窗格中選取。
   8. 在主側窗格中的圖形上按一下滑鼠右鍵，然後選取 Change graph to (將圖形變更為) > Logical processors (邏輯處理器)，以顯示每個個別核心。
   9. 根據執行個體上有多少核心，您可能會看到一段時間內顯示CPU負載的線條，或者您可能只會看到一個數字。
      - 如果您看到一段時間內顯示負載的圖表，請查找框幾乎完全著色的CPUs位置。
      - 如果您在每個核心上看到一個數字，請尋找一致顯示 95% 以上的核心。

   10請注意核心 `0` 或不同的核心正在經歷繁重的負載。

3. 選擇要套用的組態

組態名稱	套用此組態的時間	備註
Default configuration	工作負載驅動量低於 20,000IOPS，或其他組態無法改善效能或穩定性。	針對此組態，IO 會發生在幾個核心上，這可能會藉由增加快取區域性並減少內容切換，從較小工作負載獲益。
Allow driver to choose whether to distribute completion	工作量驅動超過 20,000IOPS，核心觀察到中等或高負載0。	建議所有使用 PV 8.4.0 或更新版本的 Xen 執行個體使用此設定，並利用超過 20,000 個IOPS，無論是否遇到問題。
Distribute both preparation and completion	工作負載驅動量超過 20,000 個IOPS，要么允許驅動程序選擇分發並沒有提高性能，或者不0是遇到高負載以外的核心。	此組態會啟用 IO 準備和 IO 完成的分發。

> **Note**
>
> 未同時分發 IO 完成 (設定`DpcRedirection` 而不設定`NotifierDistributed`) 的情況下，我們建議您不要分發 IO 準備，因為當準備階段平行執行時，完成階段會對準備階段過載較為敏感。

登錄金鑰值

- NotifierDistributed

  值 0 或不存在 – 完成階段將在核心 0 上執行。

  值 1 – 驅動程式選擇執行完成階段或核心 0，或者每個連接磁碟有另一個核心。

  值 2 – 驅動程式會在每個連接磁碟的另一個核心上執行完成階段。

- DpcRedirection

  值 0 或不存在 – 準備階段將會在單一、任意核心上執行。

  值 1 – 準備階段分發在多個核心之間。

預設組態

使用 8.4.0 之前的 AWS PV 驅動程式版本套用預設組態，或者如果在套用本節中的其他組態之一後發現效能或穩定性降低，則套用預設組態。

1. 使用 Connect 至您的執行個體RDP。
2. 以系統管理員身分開啟新的 PowerShell 命令提示字元。
3. 執行下列命令以移除 `NotifierDistributed` 和 `DpcRedirection` 登錄金鑰。

```
Remove-ItemProperty -Path HKLM:\System\CurrentControlSet\Services\xenvbd
\Parameters -Name NotifierDistributed
```

```
Remove-ItemProperty -Path HKLM:\System\CurrentControlSet\Services\xenvbd
\Parameters -Name DpcRedirection
```

4. 將執行個體重新開機。

允許驅動程式選擇是否要分配完成

設定 `NotiferDistributed` 登錄金鑰，以允許 PV 儲存驅動程式選擇是否要分發 IO 完成。

1. 使用 Connect 至您的執行個體RDP。
2. 以系統管理員身分開啟新的 PowerShell 命令提示字元。
3. 執行下列命令以設定 `NotiferDistributed` 登錄金鑰。

```
Set-ItemProperty -Type DWORD -Path HKLM:\System\CurrentControlSet\Services\xenvbd
\Parameters -Value 0x00000001 -Name NotifierDistributed
```

4. 將執行個體重新開機。

分配準備和完成

設定 `NotifierDistributed` 和 `DpcRedirection` 登錄金鑰，以始終分發發準備和完成階段。

1. 使用 Connect 至您的執行個體RDP。
2. 以系統管理員身分開啟新的 PowerShell 命令提示字元。
3. 執行下列命令以設定 `NotifierDistributed` 和 `DpcRedirection` 登錄金鑰。

```
Set-ItemProperty -Type DWORD -Path HKLM:\System\CurrentControlSet\Services\xenvbd
\Parameters -Value 0x00000002 -Name NotifierDistributed
```

```
Set-ItemProperty -Type DWORD -Path HKLM:\System\CurrentControlSet\Services\xenvbd
\Parameters -Value 0x00000001 -Name DpcRedirection
```

4. 將執行個體重新開機。

AWS NVMe驅動程式

Amazon EBS 磁碟區和執行個體存放區磁碟區會以區NVMe塊裝置的形式公開在 AWS Nitro 系統上建置的執行個體上 若要針對以區NVMe塊裝置形式公開的磁碟區充分利用 Amazon EBS 功能的效能和功能，執行個體必須安裝 AWS NVMe驅動程式。默認情況下，所有當前一代 AWS Windows AMIs 都帶有安裝的 AWS NVMe驅動程序。

有關EBS和的更多信息NVMe，請參閱 Amazon EBS 和 Amazon 用EBS戶指南NVMe中的。如需SSD執行個體存放區的詳細資訊NVMe，請參閱SSDAmazon EC2 執行個體的執行個體存放量。

使用安裝或升級 AWS NVMe驅動程式 PowerShell

如果您不是使用 Amazon AMIs 提供的最新 AWS Windows，請使用下列程序安裝目前的 AWS NVMe驅動程式。您應該在方便重新啟動執行個體時，執行這項更新。安裝指令碼會重新啟動您的執行個體，或是您必須重新啟動，以作為最後一個步驟。

先決條件

PowerShell 3.0 或更新版本

下載並安裝最新的 AWS NVMe驅動程式

1. 我們建議您按照以下步驟創建備份，以防需要復原更改。AMI
   a. 停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。停止執行個體之前，請確認您已將所需的任何資料從執行個體存放磁碟區複製到持續性儲存，例如 Amazon EBS 或 Amazon S3。
   b. 在導覽窗格中，選擇 Instances (執行個體)。
   c. 選取需要升級驅動程式的執行個體，並依序選取 Instance state (執行個體狀態)、Stop instance (停止執行個體)。
   d. 停止執行個體後，請選取執行個體，依序選取 Actions (動作)、Image and templates (映像和範本)，然後選取 Create image (建立映像)。
   e. 選擇 Instance state (執行個體狀態)、Start instance (啟動執行個體)。
2. 連線到您的執行個體，並以本機管理員登入。
3. 使用下列其中一個選項下載驅動程式並解壓縮至您的執行個體：
   - 使用瀏覽器：
     a. 將最新的驅動程式套件下載到執行個體。
     b. 將 zip 封存檔解壓縮。
   - 使用 PowerShell：

```
Invoke-WebRequest https://s3.amazonaws.com/ec2-windows-drivers-downloads/NVMe/
Latest/AWSNVMe.zip -outfile $env:USERPROFILE\nvme_driver.zip
Expand-Archive $env:userprofile\nvme_driver.zip -DestinationPath
 $env:userprofile\nvme_driver
```

> **Note**
>
> 如果您在下載檔案時收到錯誤訊息，而且您使用的是 Windows 伺服器 2016 或更早版本，則可能需要為 PowerShell 終端機啟用 TLS 1.2。您可以使用以下命令為當前 PowerShell 會話啟用 TLS 1.2，然後再試一次：
>
> ```
> [Net.ServicePointManager]::SecurityProtocol =
> [Net.SecurityProtocolType]::Tls12
> ```

4. 從nvme_driver目錄 (.\install.ps1) 執行install.ps1 PowerShell 指令碼，將驅動程式安裝到執行個體。如果出現錯誤，請確保您使用的是 PowerShell 3.0 或更高版本。

   a. (選擇性) 從 AWS NVMe版本開始1.5.0，Windows Server 2016 及更新版本支援小型電腦系統介面 (SCSI) 持續性保留。此功能新增對 Windows 伺服器容錯移轉叢集與共用 Amazon EBS 儲存的支援。依預設，安裝期間系統不會啟用此功能。

      您可以在執行 install.ps1 指令碼以安裝驅動程式時啟用此功能，方法是將 EnableSCSIPersistentReservations 參數指定為 $true 值。

      ```
      PS C:\> .\install.ps1 -EnableSCSIPersistentReservations $true
      ```

      您可以在執行 install.ps1 指令碼以安裝驅動程式時停用此功能，方法是將 EnableSCSIPersistentReservations 參數指定為 $false 值。

      ```
      PS C:\> .\install.ps1 -EnableSCSIPersistentReservations $false
      ```

   b. 從開始 AWS NVMe1.5.0，install.ps1指令碼一律會與驅動程式一起安裝ebsnvme-id工具。

      (選用) 對於版本 1.4.0、1.4.1 和 1.4.2，install.ps1 指令碼允許您指定 ebsnvme-id 工具是否應與驅動程式一起安裝。

      i. 若要安裝 ebsnvme-id 工具，請指定 InstallEBSNVMeIdTool ‘Yes’。
      ii. 如果您不想安裝工具，請指定 InstallEBSNVMeIdTool ‘No’。

         如果您未指定 InstallEBSNVMeIdTool，並且工具已存在於 C:\ProgramData\Amazon\Tools，則套件預設會升級此工具。如果工具不存在，install.ps1 預設不會升級此工具。

         如果您不想將工具安裝為套件的一部分，而且想要稍後安裝它，則可以在驅動程式套件中找到最新版本或工具。或者，您可以從 Amazon S3 下載版本 1.0.0：

         下載 ebsnvme-id 工具。

5. 如果安裝程式並未重新啟動您的執行個體，請重新啟動執行個體。

向代理商安裝或升級 AWS NVMe驅動程式

您可以使用的 AWS Systems Manager「代理商」功能來安裝NVMe驅動程式套件一次或排程更新。

1. 如需如何使用代理商安裝NVMe驅動程式套件的指示，請參閱 Amazon EC2 Systems Manager 使用者指南中的安裝或更新套件中的程序。
2. 對於安裝類型，選取解除安裝並重新安裝。
3. 對於「名稱」，選擇AWSNVMe。
4. (選擇性) 對於其他引數，您可以指定值來自訂安裝。這些值必須使用有效的JSON語法進行格式化。如需如何為`aws configure`套件傳遞其他引數的範例，請參閱 Amazon EC2 Systems Manager 文件。

   a. 從開始 AWS NVMe`1.5.0`，驅動程式支援 Windows 伺服器 2016 年及更新版本的SCSI永久保留。依預設，安裝期間系統不會啟用此功能。

      - 若要啟用此功能，請指定`{"SSM_EnableSCSIPersistentReservations": "true"}`。
      - 如果您不想啟用此功能，請指定`{"SSM_EnableSCSIPersistentReservations": "false"}`。

   b. 從開始 AWS NVMe`1.5.0`，指`install.ps1`令碼將一律安裝`ebsnvme-id`工具。

      (選用) 對於版本 `1.4.0`、`1.4.1` 和 `1.4.2`，`install.ps1` 指令碼允許您指定 ebsnvme-id 工具是否應與驅動程式一起安裝。

      - 若要安裝 Ebsnvme-id 工具，請指定。`{"SSM_InstallEBSNVMeIdTool": "Yes"}`
      - 如果您不想安裝工具，請指定 `{"SSM_InstallEBSNVMeIdTool": "No"}`。

        如果您未針對 Additional Arguments (其他引數) 指定 `SSM_InstallEBSNVMeIdTool`，並且工具已存在於 `C:\ProgramData\Amazon\Tools`，則套件預設會升級此工具。如果工具不存在，套件預設不會升級此工具。

        如果您不想將工具安裝為套件的一部分，而且想要稍後安裝它，則可以在驅動程式套件中找到最新版本的工具。或者，您可以從 Amazon S3 下載版本 `1.0.0`：

        下載 `ebsnvme-id` 工具。
5. 如果安裝程式並未重新啟動您的執行個體，請重新啟動執行個體。

設定SCSI持續保留

在安裝 AWS NVMe驅動程式版本`1.5.0`或更新版本之後，您可以使用 Windows 伺服器 2016 及更新版本的 Windows 登錄來啟用或停用SCSI持續性保留。您必須重新啟動這些登錄檔的執行個體，變更才會生效。

您可以使用下列將值設定為的指令`EnableSCSIPersistentReservations`來啟用SCSI持續保留`1`。

```
PS C:\> $registryPath = "HKLM:\SYSTEM\CurrentControlSet\Services\AWSNVMe\Parameters
\Device"
Set-ItemProperty -Path $registryPath -Name EnableSCSIPersistentReservations -Value 1
```

您可以使用下列將值設定為的命令`EnableSCSIPersistentReservations`來停用SCSI持續保留`0`。

```
PS C:\> $registryPath = "HKLM:\SYSTEM\CurrentControlSet\Services\AWSNVMe\Parameters
\Device"
Set-ItemProperty -Path $registryPath -Name EnableSCSIPersistentReservations -Value 0
```

AWS NVMe驅動程式版本紀錄

下表顯示了在 Amazon 上每個版本的 Windows 服務器上運行的 AWS NVMe驅動程序EC2。

Windows Server 版本	AWS NVMe 驅動程式版本
Windows Server 2022	最新版本
Windows Server 2019	最新版本
Windows Server 2016	最新版本
Windows Server 2012 R2	最新版本
Windows Server 2012	最新版本
Windows Server 2008 R2	版本 1.3.2 及更早版本
Windows Server 2008	版本 1.3.2 及更早版本

下表說明 AWS NVMe驅動程式的已發行版本。

套件版本	驅動程式版本	詳細資訊	發行日期
1.5.1	1.5.0	修正安裝指令碼以為 ebsnvme-id 工具建立資料夾 (如果不存在)。	2023 年 11 月 17 日
1.5.0	1.5.0	新增對執行 Windows 伺服器 2016 及更新版本之執行個體的小型電腦系統介面 (SCSI) 持續保留的支援。ebsnvme-id 工具 (`ebsnvme-id.exe`) 現在預設安裝。	2023 年 8 月 31 日
1.4.2	1.4.2	修正 D3 執行個體上 AWS NVMe 驅動程式 不支援執行個體儲存磁碟區的錯誤。	2023 年 3 月 16 日
1.4.1	1.4.1	報告命名空間偏好的寫入粒度 (NPGW) 支援此選擇性NVMe功能的EBS磁碟區。如需詳細資訊，請參閱NVMe基本規格 1.4 版中的第 8.25 節「透過 I/O 大小和對齊遵循來改善效能」。	2022 年 5 月 20 日
1.4.0	1.4.0	• 增加了對允許IOCTLs應用程序與NVMe設備交互的支持。此支援可讓應用程式從NVMe裝置取得`IdentifyController` `IdentifyNamespace` 、及`NameSpace` 列出。如需詳細資訊，請參閱 Microsoft 文件中的通訊協定特定查詢。 • AWSNVMe在視窗伺服器 2008 R2 上安裝 1.4.0 將會失敗。 AWSNVMe 視窗伺服器 2008 R2 支援版本 1.3.2 及更早版本。 • 1.4.0 驅動程式版本和最新的 `ebsnvme-id` 工具 (`ebsnvme-id.exe`) 會組合在單一套件中。這種組合可讓您從單一套件同時安裝驅動程式和工具。如需詳細資訊，請參閱使用安裝或升級 AWS NVMe驅動程式 PowerShell。 • 錯誤修正與可靠性改進。	2021 年 11 月 23 日

套件版本	驅動程式版本	詳細資訊	發行日期
1.3.2	1.3.2	修正修改EBS磁碟區主動處理 IO 的問題，這可能會導致資料損毀。未修改線上EBS磁碟區 (例如調整大小或變更類型) 的客戶不會受到影響。	2019 年 9 月 10 日
1.3.1	1.3.1	可靠性改善。	2019 年 5 月 21 日
1.3.0	1.3.0	裝置最佳化改善。	2018 年 8 月 31 日
1.2.0	1.2.0	針對所有支援執行個體 (包括裸機執行個體) 上的 AWS NVMe裝置提升效能和可靠性。	2018 年 6 月 13 日
1.0.0	1.0.0	AWS NVMe支援執行 Windows 伺服器執行個體類型的驅動程式	2018 年 2 月 12 日

訂閱 通知

Amazon SNS 可以在 EC2 Windows 驅動程序的新版本發布時通知您。使用下列程序訂閱這些通知。

從主控台訂閱EC2通知

1. 在 https://console.aws.amazon.com/sns/v3/ home 上打開 Amazon SNS 控制台。
2. 如有必要，請在導覽列中將「區域」變更為美國東部 (維吉尼亞北部)。您必須選取此區域，因為您訂閱的SNS通知位於此區域中。
3. 在導覽窗格中，選擇訂閱。
4. 選擇 Create subscription (建立訂閱)。
5. 在 Create subscription (建立訂閱) 對話方塊中，執行下列動作：
   a. 對於主題 ARN，請複製以下 Amazon 資源名稱 (ARN)：

      arn:aws:sns:us-east-1:801119661308:ec2-windows-drivers
   b. 針對 Protocol (通訊協定)，選擇 `Email`。
   c. 針對 Endpoint (端點)，輸入可用於接收通知的電子郵件地址。

   d. 選擇 Create subscription (建立訂閱)。

6. 您將會收到確認電子郵件。開啟電子郵件並遵循指示完成訂閱。

每當發布新的 EC2 Windows 驅動程序時，我們都會向訂閱者發送通知。如果您不想再接收這些通知，請使用下列程序來取消訂閱。

取消訂閱 Amazon EC2 視窗驅動程式通知

1. 在 https://console.aws.amazon.com/sns/v3/ home 上打開 Amazon SNS 控制台。
2. 在導覽窗格中，選擇 Subscriptions (訂閱)。
3. 選取訂閱的核取方塊，然後選取 Actions (動作)，Delete subscriptions (刪除訂閱)。出現確認提示時，請選擇 Delete (刪除)。

若要使用訂閱EC2通知 AWS CLI

若要使用訂閱EC2通知 AWS CLI，請使用下列命令。

```
aws sns subscribe --topic-arn arn:aws:sns:us-east-1:801119661308:ec2-windows-drivers --
protocol email --notification-endpoint YourUserName@YourDomainName.ext
```

若要使用訂閱EC2通知 AWS Tools for Windows PowerShell

若要使用訂閱EC2通知 AWS Tools for Windows PowerShell，請使用下列命令。

```
Connect-SNSNotification -TopicArn 'arn:aws:sns:us-east-1:801119661308:ec2-windows-
drivers' -Protocol email -Region us-east-1 -Endpoint 'YourUserName@YourDomainName.ext'
```

設定您的EC2視 Amazon 執行個體

啟動 Windows 執行個體之後，您可以以系統管理員身分登入，以針對 Windows 功能和系統設定執行其他設定。

您可以設定 Windows 啟動代理程式和其他 Windows 特定功能，如下所示。

啟動代理程式

每個 AWS Windows AMI (以及上提供的許多其他功AMIs能 AWS Marketplace) 都包含預先設定為預設設定的 Windows 啟動代理程式。啟動代理程式會在執行個體啟動期間執行工作，並在執行個體停止、稍後啟動或重新啟動時執行。

EC2快速啟動

每個 Amazon EC2 Windows 執行個體都必須執行標準的 Windows 作業系統 (OS) 啟動步驟，其中包括多次重新開機，而且通常需要 15 分鐘或更長時間才能完成。啟用EC2快速啟動功能的 Amazon EC2 Windows Server AMIs 會完成其中一些步驟並預先重新開機，以減少啟動執行個體所需的時間。

> **Note**
>
> 適用於 Windows 執行個體的 Amazon 彈性圖形已於 2024 年 1 月 8 日結束使用壽命。對於需要圖形加速的工作負載，建議您使用加速執行個體類型。如需詳細資訊，請參閱 Amazon 執行個體類型指南中的加速運算EC2執行個體類型規格。

視窗特定的系統設定

下列清單包含一些僅適用於 Windows 作業系統的系統設定：

變更視窗管理員密碼

當您連線至 Windows 執行個體時，您必須指定使用者帳戶和有存取執行個體許可的密碼。第一次連線至執行個體時，您必須使用系統管理員帳戶並提供預設密碼。當您第一次連線至執行個體時，建議您變更預設的管理員密碼。

新增視窗系統元件

Windows Server 作業系統包含許多選用元件。在每個 AWS Windows 伺服器中包含所有選用元件AMI是不實際的。相反地，我們會提供安裝媒體EBS快照，其中包含在 Windows 執行個體上設定或安裝元件所需的檔案。

在 Windows 上安裝 WSL

視窗子系統 (WSL) 是一個免費的下載，你可以在你的 Windows 實例上安裝。透過安裝WSL，您可以直接在 Windows 執行個體上執行原生 Linux 命令列工具，並使用 Linux 工具來編寫指令碼，以及

傳統的 Windows 桌面平台。您可以在單一 Windows 執行個體上輕鬆地在 Linux 和 Windows 之間進行切換，這在開發環境中可能很有用。

視窗啟動代理程式 EC2

每個 AWS Windows 都AMI包含預先設定為預設設定的 Windows 啟動代理程式。啟動代理程式會在執行個體啟動期間執行工作，並在執行個體停止、稍後啟動或重新啟動時執行。如需有關特定代理程式的資訊，請參閱下列清單中的詳細資訊頁面。

如需有關 AWS 視窗的詳細資訊AMIs，請參閱視AWS 窗AMI參考資料。

- 使用 EC2Launch v2 代理程式在 EC2 Windows 執行個體啟動期間執行工作
- 在 EC2 Windows 執行個體啟動期間，使用 EC2Launch v1 代理程式執行工作
- 在EC2舊版 Windows 作業系統執行個體啟動期間，使用此EC2Config服務執行工作

內容

- 比較 Amazon EC2 啟動代理
- 設定 EC2 Windows 啟動代理程式的DNS尾碼
- 訂閱 EC2 Windows 啟動代理程式通知
- 針對視窗執行個體遷移至 EC2Launch v2
- 適用於 EC2Launch v2 和EC2Config代理程式的 Windows 服務管理

比較 Amazon EC2 啟動代理

下表顯示了 EC2Launch v1 和 EC2Launch v2 之間EC2Config的主要功能差異。

功能	EC2Config	EC2Launch v1	EC2Launch v2
Run as (執行身分)	Windows 服務	PowerShell 腳本	Windows 服務
支援	僅限舊版作業系統	Windows 2016 視窗（LTSC和SAC）	Windows 2016 視窗（LTSC和SAC） Windows 2022

功能	EC2Config	EC2Launch v1	EC2Launch v2
組態檔案	XML	JSON	JSON/YAML
設定管理員使用者名稱	否	否	是
使用者資料大小	16 KB	16 KB	60 KB (壓縮)
本地用戶數據出爐 AMI	否	否	是，不可設定。
使用者資料中的任務組態	否	否	是
可設定桌布	否	否	是
自訂任務執行順序	否	否	是
可設定的工作	15	9	啟動時為 20 個
支援 Windows 事件檢視器	是	否	是
事件檢視器事件類型的數目	2	0	30

> **Note**
>
> EC2Config文件僅供歷史參考之用。它執行的作業系統版本已不再受到 Microsoft 的支援。強烈建議您升級至最新的啟動服務。

設定 EC2 Windows 啟動代理程式的DNS尾碼

使用 Amazon EC2 啟動代理程式，您可以設定 Windows 執行個體用於網域名稱解析的DNS尾碼清單。啟動代理程式會覆寫`System\CurrentControlSet\Services\Tcpip\Parameters\SearchList`登錄機碼中的標準 Windows 設定，方法是將下列值新增至DNS尾碼搜尋清單：

- 執行個體的網域
- 執行個體網域下放所產生的尾碼
- NV 域名
- 每張網路介面卡指定的網域

所有啟動代理程式都支援DNS尾碼組態 如需詳細資訊，請參閱您的特定啟動代理程式版本：

- 如需有關`setDnsSuffix`工作以及如何在 EC2Launch v2 中設定DNS尾碼的資訊，請參閱。setDnsSuffix
- 如需有關DNS尾碼清單設定以及如何啟用或停用 EC2Launch v1 的遞增功能的詳細資訊，請參閱。在您的 Windows 執行個體上設定 EC2Launch v1 代理程式
- 若要取得有關DNS字尾清單設定以及如何啟用或停用的遞增功能的資訊EC2Config，請參閱。EC2Config設定檔案

域名下放

網域名稱遞減是一種 Active Directory 行為，允許子網域中的電腦存取父網域中的資源，而無需使用完整網域名稱。根據預設，網域名稱下放會繼續執行，直到網域名稱進度中只剩下兩個節點為止。

如果執行個體已連線至網域，Launch 代理程式會對網域名稱執行遞減，並將結果新增至**`System\CurrentControlSet\Services\Tcpip\Parameters\SearchList`**登錄機碼中維護的DNS尾碼搜尋清單。代理程式會使用下列登錄機碼中的設定來判斷遞增行為。

- **`System\CurrentControlSet\Services\Tcpip\Parameters\UseDomainNameDevolution`**
  - 未設置時，禁用下放
  - 設定為時`1`，啟用下積分 (預設)
  - 設定為時`0`，停用下放
- **`System\CurrentControlSet\Services\Dnscache\Parameters\DomainNameDevolutionLevel`**
  - 未設定時，請使用圖層 2 (預設值)
  - 設定為`3`或更大時，使用值設定等級

當您停用遞增功能或將遞增設定變更為較高層級時，System\CurrentControlSet\Services\Tcpip\Parameters\SearchList登錄機碼樣式會包含先前新增的尾碼。它們不會自動移除。您可以手動更新清單，也可以清除清單，讓代理程式執行程序以設定新清單。

> **Note**
>
> 要清除註冊表中的DNS後綴列表，可以運行以下命令。
>
> ```
> PS C:\> Invoke-CimMethod -ClassName Win32_NetworkAdapterConfiguration -
> MethodName "SetDNSSuffixSearchOrder" -Arguments @{ DNSDomainSuffixSearchOrder =
> $null } | Out-Null
> ```

下放示例

下列範例顯示網域名稱在下放過程中的進展情況。

corp.example.com

- 進展到 example.com

locale.region.corp.example.com

1. 進展到 region.corp.example.com
2. 進展到 corp.example.com
3. 進展到 example.com

locale.region.corp.example.com與設置 DomainNameDevolutionLevel=3

1. 進展到 region.corp.example.com
2. 進展到。corp.example.com由於等級設定，進度會在此停止。

訂閱 EC2 Windows 啟動代理程式通知

Amazon SNS 可以在發佈新版本的EC2啟動代理程式時通知您。使用下列程序訂閱這些通知。

訂閱 EC2Config通知

1. 在 https://console.aws.amazon.com/sns/v3/ home 上打開 Amazon SNS 控制台。
2. 如有必要，請在導覽列中將「區域」變更為美國東部 (維吉尼亞北部)。您必須選取此區域，因為您訂閱的SNS通知是在此區域中建立的。
3. 在導覽窗格中，選擇訂閱。
4. 選擇 Create subscription (建立訂閱)。
5. 在 Create subscription (建立訂閱) 對話方塊中，執行下列動作：
   a. 對於「主題」ARN，請使用下列與您要接收通知的代理程式相符的 Amazon 資源名稱 (ARN)：
      - EC2Launch第二版：

        ```
        arn:aws:sns:us-east-1:309726204594:amazon-ec2launch-v2
        ```

      - EC2Launch或 EC2Config：

        ```
        arn:aws:sns:us-east-1:801119661308:ec2-windows-ec2config
        ```

   b. 針對 rotocol (通訊協定)，選擇 `Email`。
   c. 在 Endpoint 中，輸入您要接收通知的電子郵件地址。
   d. 選擇 Create subscription (建立訂閱)。
6. 您會收到要求確認訂閱的電子郵件。開啟電子郵件並遵循指示完成訂閱。

每當發行新版本的啟動代理程式時，我們都會傳送通知給訂閱者。如果您不想再接收這些通知，請使用下列程序來取消訂閱。

退訂啟動代理程式通知

1. 打開 Amazon SNS 控制台。
2. 在導覽窗格中，選擇 Subscriptions (訂閱)。
3. 選取訂閱，然後選取 Actions (動作)、Delete subscriptions (刪除訂閱)。出現確認提示時，請選擇 Delete (刪除)。

針對視窗執行個體遷移至 EC2Launch v2

EC2Launch移轉工具會透過解除安裝EC2Config並安裝 EC2Launch v2 來升級已安裝的啟動代理程式 (和 EC2Launch v1)。先前啟動服務的適用組態會自動移轉至新服務。移轉工具不會偵測任何連結至 EC2Launch v1 指令碼的排程工作，因此不會在 EC2Launch v2 中自動設定這些工作。若要設定這些工作，請編輯`agent-config.yml`檔案，或使用 EC2Launchv2 設定對話方塊。例如，如果執行個體有執行的排程工作`InitializeDisks.ps1`，則在執行移轉工具之後，您必須在 EC2Launch v2 設定對話方塊中指定要初始化的磁碟區。請參閱 使用 EC2Launch v2 設定對話方塊變更設定 程序的步驟 6。

您可以下載移轉工具或隨SSMRunCommand 文件一起安裝。

您可以從下列位置下載此工具：

> **Note**
>
> 我們即將汰除 32 位元的遷移工具連結。建議您使用 64 位元連結移轉至 EC2Launch v2。如果您需要 32 位元的啟動代理程式，請使用 EC2Config.

- 64 位 — https://s3.amazonaws.com/amazon-ec2launch-v2-utils/MigrationTool/視窗/amd64 /最新/EC2LaunchMigrationTool.
- 32 位元 — https://s3.amazonaws.com/amazon-ec2launch-v2-utils/MigrationTool/視窗/386 /最新/zip EC2LaunchMigrationTool

> **Note**
>
> 您必須以系統管理員身分執行 EC2Launch v2 移轉工具。EC2Launchv2 會在您執行移轉工具之後安裝為服務。此工具不會立即執行。依預設，工具會在執行個體啟動期間執行，並在執行個體停止後啟動或重新啟動時執行。

使用該`AWSEC2Launch-RunMigration`SSM文檔通過SSM運行命令遷移到最新的 EC2Launch v2 版本。文件不需要任何參數。如需有關使用SSM執行命令的詳細資訊，請參閱 AWS Systems Manager 執行命令。

移轉工具會將下列組態從套用EC2Config至 EC2Launch v2。

- 如果`Ec2DynamicBootVolumeSize`設定為`false`，則移除 EC2Launch v2 `boot` 階段

- 如果`Ec2SetPassword`設定為`Enabled`，則將 EC2Launch v2 密碼類型設定為 `random`
- 如果`Ec2SetPassword`設定為`Disabled`，則將 EC2Launch v2 密碼類型設定為 `donothing`
- 如果`SetDnsSuffixList`設定為`false`，則移除 EC2Launch v2 `setDnsSuffix` 工作
- 如果`EC2SetComputerName`設置為 true，則將 EC2Launch v2 `setHostName` 任務添加到`yaml`配置

移轉工具會從 EC2Launch v1 套用下列組態至 EC2Launch v2。

- 如果`ExtendBootVolumeSize`設定為`false`，則移除 EC2Launch v2 `boot` 階段
- 如果`AdminPasswordType`設定為`Random`，則將 EC2Launch v2 密碼類型設定為 `random`
- 如果`AdminPasswordType`設定為`Specify`，則將 EC2Launch v2password 類型設定為，`static`並將密碼資料設定為 `AdminPassword`
- 如果`SetWallpaper`設定為`false`，則移除 EC2Launch v2 `setWallpaper` 工作
- 如果`AddDnsSuffixList`設定為`false`，則移除 EC2Launch v2 `setDnsSuffix` 工作
- 如果`SetComputerName`設定為`true`，則增加 EC2Launch v2 `setHostName` 工作

適用於 EC2Launch v2 和EC2Config代理程式的 Windows 服務管理

如果您以具有系統管理權限的使用者身分登入執行個體，您可以管理 EC2Launch v2 和EC2Config啟動代理程式，就像處理任何其他 Windows 服務一樣。EC2Launchv1 是預設情況下透過排程工作管理的一組 PowerShell 指令碼。本節涵蓋 EC2Launch v2 和EC2Config.

若要將更新的設定套用至執行個體，您可以從 Microsoft 管理主控台 (MMC) 介面停止並重新啟動 EC2Launch v2 代理程式或服務啟動代理程式。EC2Config同樣地，當您安裝新版啟動代理程式時，必須先停止代理程式，然後在安裝完成時重新啟動代理程式。

> **Note**
>
> 您必須以管理員身分開啟「MMC服務」介面，才能選取這些動作。若要執行此操作，您可以從內容功能表中選取「以管理員身分執行」。或者，若要使用鍵盤開啟介面，請依照下列步驟執行：
>
> 1. 使用按`Tab`鍵或方向鍵，從 [系統管理工具] 功能表中選取 [服務] 功能表項目。
> 2. 使用下列鍵盤組合以系統管理員身分開啟：`CtrlShift+` + `Enter`。

下列程序列出在執行個體上停止和啟動啟動代理程式的步驟。

停止啟動代理程式

1. 啟動和連線至 Windows 執行個體。
2. 從 Windows「開始」功能表中選取「管理工具」。
3. 如本節開頭所述，以系統管理員身分開啟「服務」主控台。
4. 在服務清單中，選取執行個體上執行的代理程式 (EC2Launch或 EC2Config)，然後從 [動作] 功能表選擇 [停止]。或者，您可以使用內容功能表停止代理程式。

重新啟動啟動代理

1. 啟動和連線至 Windows 執行個體。
2. 從 Windows「開始」功能表中選取「管理工具」。
3. 如本節開頭所述，以系統管理員身分開啟「服務」主控台。
4. 在服務清單中，選取執行個體上執行的代理程式 (EC2Launch或 EC2Config)，然後從 [動作] 功能表選擇 [啟動] 或 [重新啟動]。或者，您可以使用內容功能表重新啟動代理程式。

如果您不需要更新組態設定、建立自己的組態設定或使用 AMI AWS Systems Manager，您可以刪除或解除安裝啟動代理程式。

刪除

移除服務將移除其登錄子機碼。

卸載

解除安裝服務將移除檔案、登錄子機碼和任何服務捷徑。

刪除啟動代理程式

1. 啟動和連線至 Windows 執行個體。
2. 啟動命令提示字元視窗。
3. 執行下列其中一個命令來刪除啟動代理程式。
   - 執行下列命令以刪除EC2Launch或 EC2Launch v2：

```
sc delete ec2launch
```

- 執行下列命令以刪除EC2Config服務：

```
sc delete ec2config
```

解除安裝啟動代理

1. 啟動和連線至 Windows 執行個體。
2. 從 Windows 的「開始」功能表中選擇「視窗系統」，然後選擇「控制
3. 選擇 [程式和功能]，開啟執行個體上安裝的程式清單。
4. 從清單中選取您的啟動代理程式 (Amazon EC2Launch 或 EC2ConfigService)，然後從 [檔案] 功能表選擇 [解除安裝]。或者，您可以使用關聯式功能表。

> **Note**
>
> 您可以在 [版本] 欄中查看安裝的啟動代理程式版本。

使用 EC2Launch v2 代理程式在 EC2 Windows 執行個體啟動期間執行工作

預設情況下，從 AWS Windows 伺服器 2022 啟動的所有支援 Amazon EC2 執行個體都AMIs包含 EC2Launch v2 啟動代理程式 (`EC2Launch.exe`)。我們也提供視窗伺服器 2016 年和 2019 年AMIs安裝 EC2Launch v2 做為預設啟動代理程式。這些AMIs是除了視窗伺服器 2016 年和 2019 年之外提供的，其中AMIs包括EC2Launch第 1 版。您可以搜尋預設包AMIs含 EC2Launch v2 的 Windows，方法是在 Amazon EC2 主控台的搜尋AMIs頁面中輸入下列前置詞：`EC2LaunchV2-Windows_Server-*`。

若要比較啟動代理程式版本功能，請參閱比較 Amazon EC2 啟動代理。

EC2Launchv2 會在執行個體啟動期間執行工作，並在執行個體停止並稍後啟動或重新啟動時執行。EC2Launchv2 也可以按需執行任務。其中部分作業為自動啟用，其他作業則必須手動啟用。EC2Launchv2 服務支援所EC2Config有EC2Launch功能。

此服務會使用組態檔來控制其操作。您可以使用圖形化工具或直接將組態檔編輯為單一 .yml 檔案 (`agent-config.yml`) 來更新組態檔。服務二進位檔案位於 `%ProgramFiles%\Amazon\EC2Launch` 目錄中。

EC2Launchv2 會發佈 Windows 事件記錄檔，協助您疑難排解錯誤並設定觸發器。如需詳細資訊，請參閱Windows 事件日誌。

EC2Launchv2 代理程式支援下列 Windows 伺服器作業系統 (作業系統) 版本：

支援的 OS 版本

- Windows Server 2022
- Windows Server 2019 (長期服務通道和半年通道)
- Windows Server 2016

EC2Launch第二個概念

下面的概念是有用的考慮 EC2Launch V2 時理解。

任務

您可以調用任務以在執行個體上執行動作。您可以在 `agent-config.yml` 檔案中或透過使用者資料來設定任務。如需 EC2Launch v2 的可用工作清單，請參閱 EC2Launchv2 工作。如需了解任務組態結構描述和詳細資訊，請參閱 EC2Launchv2 任務配置。

階段

階段是 EC2Launch v2 代理程式所執行之工作的邏輯群組。某些工作只能在特定階段中執行。其他人可以在多個階段運行。使用 `agent-config.yml` 時，必須指定階段清單，以及要在每個階段中執行的任務清單。

此服務會以下列順序執行各階段：

階段 1：開機

階段 2：網路

第三階段： PreReady

Windows 已準備就緒

PreReady 階段完成後，服務會將`Windows is ready`訊息傳送至 Amazon EC2 主控台。

第四階段： PostReady

使用者資料會在PostReady階段中執行。有些指令碼版本會在`agent-config.yml`檔案PostReady階段之前執行，有些版本會在之後執行，如下所示：

`agent-config.yml` 之前

- YAML用戶數據版本 1.1
- XML使用者資料

`agent-config.yml` 之後

- YAML用戶數據版本 1.0（為了向後兼容的舊版本）

如需階段和任務的範例，請參閱 範例：`agent-config.yml`。

使用使用者資料時，您必須指定代理程式要執行的任務清單。此階段是隱含的。如需任務範例，請參閱 範例：使用者資料。

EC2Launchv2 會依照您在使用者資料中`agent-config.yml`和使用者資料中指定的順序執行工作清單。階段按順序執行。下一個階段會在前一個階段完成之後開始。任務也會依序執行。

頻率

任務頻率確定何時執行任務，這取決於開機內容。大多數任務只有一個允許的頻率。您可以指定 `executeScript` 任務的頻率。

您將在 EC2Launchv2 任務配置 中看到以下頻率。

- 一次 — 工作會在第一次啟動時執行一次 (完成 Sysprep)。AMI
- 始終 – 每次啟動代理程式執行時，任務都會執行。啟動代理程式會執行於下列時間：
  - 執行個體啟動或重新啟動
  - 該EC2Launch服務運行
  - `EC2Launch.exe run` 被叫用

agent-config

`agent-config`是位於 EC2Launch v2 組態資料夾中的檔案。其中包括開機、 PreReady網路和 PostReady 階段的組態。此檔案可用來指定在第一次啟動或後續啟動時應執行的工作執行個體組態。AMI

依預設，EC2Launchv2 安裝會安裝一個`agent-config`檔案，其中包含在標準 Amazon 視窗中使用的建議組態AMIs。您可以更新配置文件以更改 EC2Launch v2 指定的AMI默認啟動體驗。

使用者資料

使用者資料是啟動執行個體時可設定的資料。您可以更新使用者資料，以動態變更自訂AMIs或快速入門的設定AMIs方式。EC2LaunchV2 支持 60 kB 的用戶數據輸入長度。使用者資料僅包含 UserData 階段，因此會在`agent-config`檔案之後執行。您可以在使用啟動執行個體精靈啟動執行個體時輸入使用者資料，或從EC2主控台修改使用者資料。如需處理使用者資料的資訊，請參閱 Amazon 如何EC2處理 Windows 實例的用戶數據。

EC2Launchv2 任務概述

EC2Launchv2 可以在每次開機時執行下列工作：

- 設定新的和選擇性的自訂桌布來呈現執行個體的相關資訊。
- 設定在本機電腦上建立的系統管理員帳戶的屬性。
- 將DNS後綴添加到搜索後綴列表中。只有不存在的尾碼才會加入到清單中。
- 為任何額外的磁碟區設定磁碟機代號，並將它們延伸至使用可用空間。
- 將檔案從配置寫入磁碟。
- 運行 EC2Launch v2 配置文件或從中指定的腳本`user-data`。腳本`user-data`可以是純文本或壓縮，並以 base64 格式提供。
- 執行具有指定參數的程式。
- 設定電腦名稱。
- 將執行個體資訊傳送至 Amazon EC2 主控台。
- 將RDP憑證指紋傳送到 Amazon 主EC2控台。
- 動態擴充作業系統的分割區，以包含任何尚未分割的空間。
- 執行使用者資料。如需指定使用者資料的詳細資訊，請參閱EC2Launchv2 任務配置。
- 設定非持久性靜態路由以到達中繼資料服務和 AWS KMS 伺服器。
- 將非開機分割區設定為`mbr`或`gpt`。
- 在 Sysprep 後啟動 Systems Manager 服務。
- 最佳化ENA設定。
- 啟用「開啟」以SSH供更新版本使用。
- 啟用巨型訊框。
- 設置系統報告與 v2 一起EC2Launch運行。
- 發佈 Windows 事件日誌。

EC2Launchv2 目錄結構

EC2Launchv2 應該安裝在以下目錄中：

- 服務二進位檔案： `%ProgramFiles%\Amazon\EC2Launch`
- 服務資料 (設定、日誌檔和狀態檔)：`%ProgramData%\Amazon\EC2Launch`

> **Note**
>
> 根據預設，Windows 會將檔案和資料夾隱藏在 `C:\ProgramData` 下。若要檢視 EC2Launch v2 目錄和檔案，您必須在 Windows 檔案總管中輸入路徑，或變更資料夾內容以顯示隱藏的檔案和資料夾。

`%ProgramFiles%\Amazon\EC2Launch` 目錄包含二進位檔案和支援的程式庫。它包括以下子目錄：

- `settings`
  - `EC2LaunchSettingsUI.exe` – 用於修改 `agent-config.yml` 檔案的使用者介面
  - `YamlDotNet.dll`— 用DLL於支持用戶界面中的某些操作
- `tools`
  - `ebsnvme-id.exe`— 用於檢查實例上EBS卷的元數據的工具
  - `AWSAcpiSpcrReader.exe`— 用於確定要使用的正確COM端口的工具
  - `EC2LaunchEventMessage.dll`— DLL 用於支援的 Windows 事件記錄EC2Launch。
- `service`
  - `EC2LaunchService.exe` – 啟動代理程式以服務方式執行時，所啟動的 Windows 服務可執行檔。
- `EC2Launch.exe`-主EC2Launch可執行文件
- `EC2LaunchAgentAttribution.txt`— 在 EC2 Launch 中使用的程式碼歸因

`%ProgramData%\Amazon\EC2Launch` 目錄包含下列子目錄。服務產生的所有資料，包括日誌、組態和狀態，都會儲存在此目錄中。

- `config`— 配置

服務組態檔案儲存在此目錄中為 `agent-config.yml`。此檔案可以更新，以修改、新增或移除服務所執行的預設任務。在此目錄中建立檔案的許可限制為系統管理員帳戶，以防止權限提升。

- `log`— 實例日誌

  服務 (`agent.log`)、主控台 ()、效能 (`console.log`)、錯誤 (`bench.logerr.log`) 和遙測的記錄檔 (儲存`telemetry.log`在此目錄中。日誌會在後續執行服務時附加。

- `state`— 服務狀態資料

  服務用來判斷應該執行哪些任務的狀態會儲存在此處。有一個 `.run-once` 檔案會指出服務是否已在 Sysprep 之後執行 (因此，下次執行時會略過頻率為一次的任務)。此子目錄包含 `state.json` 和 `previous-state.json` 以追蹤每個任務的狀態。

- `sysprep`— 系統普雷

  此目錄包含的檔案可用來決定 Sysprep 在建立可重複使用的自訂 Windows 時，要執行哪AMI些作業。

- `wallpaper`-壁紙

  此桌布影像儲存在此目錄中。

遙測

遙測是其他資訊，可協助您進一 AWS 步瞭解您的需求、診斷問題並提供可改善使用體驗的功能 AWS 服務。

EC2Launchv2 版本`2.0.592`及更新版本會收集遙測資料，例如使用量度和錯誤。這些資料是從執行 EC2Launch v2 的 Amazon EC2 執行個體收集而來。這包括所有AMIs擁有的視窗 AWS。

EC2Launchv2 會收集下列類型的遙測：

- 用量資訊 – 代理程式命令、安裝方法和排程的執行頻率。
- 錯誤和診斷資訊 — 代理程式安裝錯誤代碼、執行錯誤代碼和錯誤呼叫堆疊。

收集資料的範例：

```
2021/07/15 21:44:12Z: EC2LaunchTelemetry: IsAgentScheduledPerBoot=true
2021/07/15 21:44:12Z: EC2LaunchTelemetry: IsUserDataScheduledPerBoot=true
2021/07/15 21:44:12Z: EC2LaunchTelemetry: AgentCommandCode=1
2021/07/15 21:44:12Z: EC2LaunchTelemetry: AgentCommandErrorCode=5
```

```
2021/07/15 21:44:12Z: EC2LaunchTelemetry: AgentInstallCode=2
2021/07/15 21:44:12Z: EC2LaunchTelemetry: AgentInstallErrorCode=0
```

預設為啟用遙測。您可隨時停用遙測收集。如果啟用遙測，EC2Launchv2 會傳送遙測資料，而不需要其他客戶通知。

遙測可見性

啟用遙測後，它會顯示在 Amazon EC2 主控台輸出中，如下所示。

```
2021/07/15 21:44:12Z: Telemetry: <Data>
```

停用執行個體的遙測

若要停用單一執行個體的遙測，您可以設定系統環境變數，或使用MSI來修改安裝。

若要藉由設定系統環境變數來停用遙測，請以管理員身分執行下列命令。

```
setx /M EC2LAUNCH_TELEMETRY 0
```

若要使用停用遙測MSI，請在下載之後執行下列命令MSI。

```
msiexec /i ".\AmazonEC2Launch.msi" Remove="Telemetry" /q
```

EC2Launchv2 的更多主題

- 安裝最新版本的 EC2Launch v2
- 設定視窗執行個體的 EC2Launch V2 設定
- EC2Launchv2 啟動工作的工作定義
- 疑難排解 EC2Launch v2 代理程式的問題
- EC2Launchv2 版本歷史記錄

安裝最新版本的 EC2Launch v2

您可以使用下列其中一種方法在EC2執行個體上安裝 EC2Launch v2 代理程式：

- 從 Amazon S3 下載代理程式並使用視窗進行安裝 PowerShell。如需下載URLs，請參閱EC2Launch 在 Amazon S3 上下載 v2。

- 與SSM代理商一起安裝。
- 當您建立自訂EC2映像檔時，從映像產生器元件安裝。
- 從預先安裝 EC2Launch v2 的執行AMI個體啟動您的執行個體。

> **Warning**
> Amazon EC2Launch .msi 會解除安裝先前版本的EC2啟動服務，例如 EC2Launch (v1) 和. EC2Config

如需了解安裝步驟，請選取符合您偏好方法的索引標籤。

Windows PowerShell

若要使用 Windows 安裝最新版的 EC2Launch v2 代理程式 PowerShell，請依照下列步驟執行。

1. 建立您的本機目錄。

```
New-Item -Path "$env:USERPROFILE\Desktop\EC2Launchv2" -ItemType Directory
```

2. URL為您的下載位置設定。使用URL您將使用的 Amazon S3 執行以下命令。如需下載URLs，請參閱 EC2Launch在 Amazon S3 上下載 v2

```
$Url = "Amazon S3 URL/AmazonEC2Launch.msi"
```

3. 使用以下複合命令來下載和安裝代理程式：

```
$DownloadFile = "$env:USERPROFILE\Desktop\EC2Launchv2\" + $(Split-Path -Path
 $Url -Leaf)
Invoke-WebRequest -Uri $Url -OutFile $DownloadFile
msiexec /i "$DownloadFile"
```

> **Note**
> 如果您在下載檔案時收到錯誤訊息，而且您使用的是 Windows 伺服器 2016 或更早版本，則可能需要為 PowerShell 終端機啟用 TLS 1.2。您可以使用以下命令為當前 PowerShell 會話啟用 TLS 1.2，然後再試一次：

```
[Net.ServicePointManager]::SecurityProtocol =
 [Net.SecurityProtocolType]::Tls12
```

4. 若要驗證安裝，請檢查 msi 檔案是否存在於執行個體 (`C:\ProgramData\Amazon\EC2Launch`) 的 EC2Launch v2 目錄中。

AWS Systems Manager Distributor

若要使用 AWS Systems Manager 快速設定為 EC2Launch v2 設定自動更新，請參閱使用經銷商快速設定自動安裝和更新。

您也可以從「 AWS Systems Manager 代理商」執行`AWSEC2Launch-Agent`套件的一次性安裝。如需有關如何從 Systems Manager Distributor 安裝套件的指示，請參閱《AWS Systems Manager 使用者指南》中的安裝或更新套件。

EC2 Image Builder component

您可以在使EC2用映像 Image Builder 建立自訂映像時安裝`ec2launch-v2-windows`元件。如需有關如何使用映像產生器建立自訂映EC2像的指示，請參閱映像產生器使用指南中的使用 EC2 Image Builder 主控台精靈建立 EC2 Image Builder 管線。

AMI

EC2Launchv2 默認情況下預先安裝在以下視窗服務器 2022 和 UEFIAMIs：

- Windows_Server-2022-English-Full-Base
- Windows_Server-2022-English-Core-Base
- 視窗服務器 2022 AMIs 與所有其他語言
- 視窗伺服器 2022 AMIs SQL 已安裝
- 視窗伺服器 2022-英文核心 _ 最佳化 EKS

EC2Launchv2 也預先安裝在以下視窗伺服器AMIs上。您可以AMIs從 Amazon EC2 主控台或使用下列搜尋前置詞找到這些內容：`EC2LaunchV2-`在 AWS CLI.

- EC2LaunchV2-視窗伺服器 2019-英文核心基礎
- EC2LaunchV2-視窗服務器 2019-英語全基
- EC2LaunchV2-視窗伺服器 2016-英文核心基礎

- EC2LaunchV2-視窗伺服器 2016-英文全基
- EC2LaunchV2-視窗伺服器 2012_R2_-英文RTM完整版
- EC2LaunchV2-視窗 _ 伺RTM服器 2012_-英文全基

使用 AWS Systems Manager 代理商快速設定自動安裝和更新 EC2Launch v2

透過「 AWS Systems Manager 代理商快速設定」，您可以設定 EC2Launch v2 的自動更新。下列程序會在您的執行個體上設定 Systems Manager 關聯，以您指定的頻率自動更新 EC2Launch v2 代理程式。「代理商快速設定」所建立的「關聯」可以包含 AWS 帳戶 和區域內的執行環境，或是 AWS 組織內的執行環境。如需有關設定組織的詳細資訊，請參閱《AWS Organizations 使用指南》中的〈教學課程：建立和設定組織〉。

開始之前，請確定您的執行個體符合所有先決條件。

必要條件

若要使用「代理商快速設定」設定自動更新，您的執行個體必須符合下列先決條件。

- 您至少有一個支援 EC2Launch v2 的執行中執行個體。請參閱支援的作業系統EC2Launch v2。
- 您已在執行個體上執行 Systems Manager 設定工作。如需詳細資訊，請參閱《AWS Systems Manager 使用指南》中的〈設定 Systems Manager〉。
- EC2Launchv2 必須是執行個體上唯一安裝的啟動代理程式。如果您安裝了一個以上的啟動代理程式，您的「代理商快速設定」組態將會失敗。使用散發商快速設定、解除安裝EC2Config或 EC2Launch v1 啟動代理程式 (如果存在) 來設定 EC2Launch v2 之前。

設定 EC2Launch v2 的代理商快速設定

若要使用「代理商快速設定」建立 EC2Launch v2 的組態，請在完成散發者套件部署的步驟時使用下列設定：

- 軟體套件：Amazon EC2Launch v2 代理程式。
- 更新頻率：從清單中選取頻率。
- 目標：從可用的部署選項中選擇。

若要檢查組態的狀態，請瀏覽至中的「Systems Manager 快速設定組態」標籤 AWS Management Console。

1. 在開啟 AWS Systems Manager 主控台https://console.aws.amazon.com/systems-manager/。

2. 在導覽窗格中，選擇快速設定。
3. 在組態索引標籤中，選擇與您建立的模型組態相關聯的列。「組態」索引標籤會列出您的組態，並包含重要詳細資料摘要，例如「區域」、「部署狀態」和「關聯狀態」。

> **Note**
>
> 每個 EC2Launch v2 代理商組態的關聯名稱都以下列前置詞開頭：`AWS-QuickSetup-Distributor-EC2Launch-Agent-`

4. 若要檢視詳細資訊，請選取組態並選擇檢視詳細資料。

如需詳細資訊和疑難排解步驟，請參閱AWS Systems Manager 使用指南中的疑難排解快速設定結果。

EC2Launch在 Amazon S3 上下載 v2

若要安裝最新版本的 EC2Launch v2，請從下列其中一個位置下載安裝程式：

> **Note**
>
> 我們即將汰除 32 位元的安裝連結。我們建議您使用 64 位元安裝連結來安裝 EC2Launch v2。如果您需要 32 位元的啟動代理程式，請使用 EC2Config.

- 64 位 — https://s3.amazonaws.com/amazon-ec2launch-v2/視窗 EC2Launch /AMD64 /最新/亞馬 Amazon .msi
- 32 位 — https://s3.amazonaws.com/amazon-ec2launch-v2/視窗 EC2Launch /386 /最新/亞馬 Amazon .微星

設定安裝選項

當您安裝或升級 EC2Launch v2 時，您可以使用 EC2Launch v2 install 對話方塊或命令列殼層中的msiexec指令來設定安裝選項。

第一次在執行個體上執行 EC2Launch v2 安裝程式時，會在執行個體上初始化啟動代理程式設定，如下所示：

- 安裝程式會建立本機路徑，並將啟動代理程式檔案寫入該路徑。這有時被稱為全新安裝。
- 安裝程式會建立 `EC2LAUNCH_TELEMETRY` 環境變數 (若尚無)，並根據您的組態進行設定。

如需有關設定的詳細資訊，請選取符合您將使用的設定方法的索引標籤。

Amazon EC2Launch Setup dialog

當您安裝或升級 EC2Launch v2 時，您可以透過 EC2Launch v2 安裝對話方塊設定下列安裝選項。

基本安裝選項

傳送遙測

在您將此功能納入安裝對話方塊中時，安裝程式會將 `EC2LAUNCH_TELEMETRY` 環境變數的值設定為 `1`。若您停用傳送遙測，安裝程式會將環境變數設定為 `0`。

EC2Launchv2 代理程式執行時，它會讀取`EC2LAUNCH_TELEMETRY`環境變數，以判斷是否要上傳遙測資料。若值等於 `1`，則會上傳資料。若不是，則不會上傳。

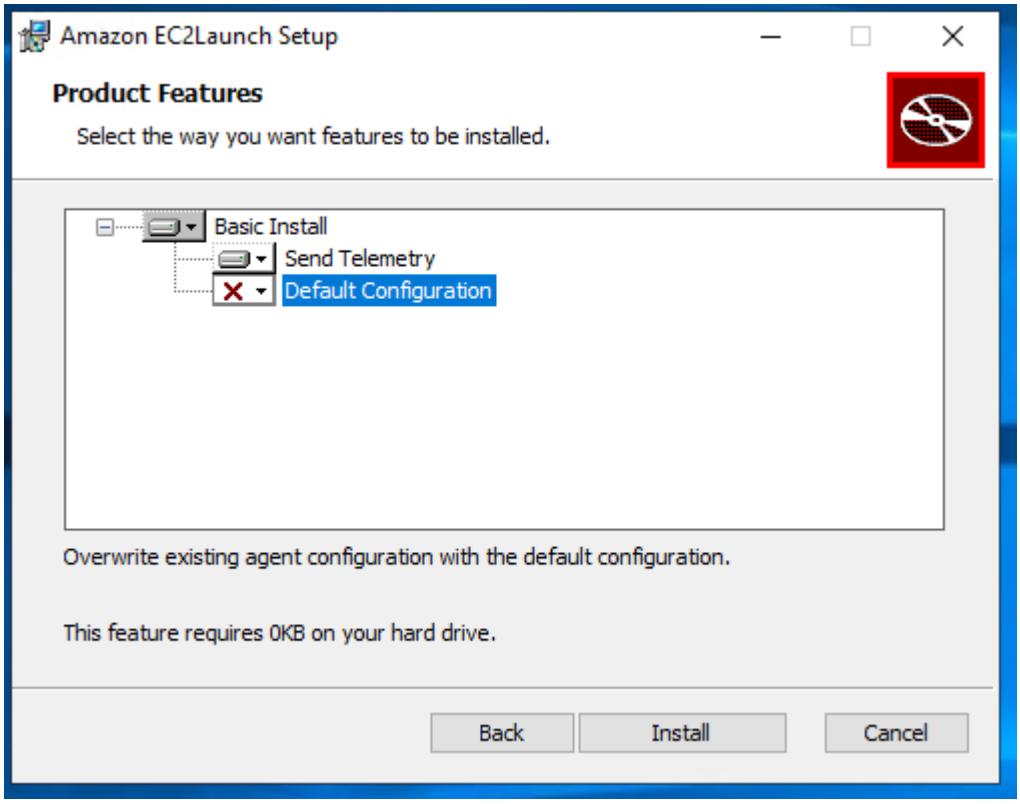
預設組態

EC2Launchv2 的預設組態是覆寫本機啟動代理程式 (如果已存在)。首次在執行個體上執行安裝時，預設組態會執行全新安裝。若您在初始安裝時停用預設組態，則安裝會失敗。

若再次在執行個體上執行安裝，您可以停用預設組態，執行不會取代 `%ProgramData%/Amazon/EC2Launch/config/agent-config.yml` 檔案的升級。

範例：使用遙測升級 EC2Launch v2

下列範例顯示設定為升級目前安裝及啟用遙測的 EC2Launch v2 安裝程式對話方塊。此組態會在不取代代理程式組態檔案的情況下執行安裝，並將 `EC2LAUNCH_TELEMETRY` 環境變數設定為 `1`。



Command line

當您安裝或升級 EC2Launch v2 時，您可以使用命令列殼層中的msiexec命令來設定下列安裝選項。

ADDLOCAL 參數值

基本 (必填)

安裝啟動代理程式。如果 ADDLOCAL 參數中不存在此值，則安裝結束。

全新

當您在 ADDLOCAL 參數中納入 Clean 值時，安裝程式會將代理程式組態檔案寫入以下位置：%ProgramData%/Amazon/EC2Launch/config/agent-config.yml。如果代理程式組態檔案已存在，則會覆寫檔案。

當您將 Clean 值放在 ADDLOCAL 參數之外時，安裝程式會執行不會取代代理程式組態檔案的升級。

遙測

當您在 ADDLOCAL 參數中納入 Telemetry 值時，安裝程式會將 EC2LAUNCH_TELEMETRY 環境變數設為 1。

當您將 `Telemetry` 值放在 `ADDLOCAL` 參數之外時，安裝程式會將環境變數設為 `0`。

EC2Launchv2 代理程式執行時，它會讀取`EC2LAUNCH_TELEMETRY`環境變數，以判斷是否要上傳遙測資料。若值等於 `1`，則會上傳資料。若不是，則不會上傳。

範例：使用遙測來安裝 EC2Launch v2

```
& msiexec /i "C:\Users\Administrator\Desktop\EC2Launchv2\AmazonEC2Launch.msi"
 ADDLOCAL="Basic,Clean,Telemetry" /q
```

驗證 EC2Launch V2 版本

請使用下列其中一個程序來驗證執行個體上安裝的 EC2Launch v2 版本。

Windows PowerShell

使用視窗驗證安裝的 EC2Launch v2 版本 PowerShell，如下所示。

1. 從您的執行個體啟動AMI並連線至執行個體。
2. 在中執行下列命令 PowerShell 以驗證安裝的 EC2Launch v2 版本：

```
& "C:\Program Files\Amazon\EC2Launch\EC2Launch.exe" version
```

Windows Control Panel

在 Windows 控制台中驗證安裝的 EC2Launch v2 版本，如下所示。

1. 從您的執行個體啟動AMI並連線至執行個體。
2. 開啟 Windows 控制台，然後選擇程式和功能。
3. 從已安裝程式清單中，找出 `Amazon EC2Launch`。其版本編號將出現在 Version (版本) 欄中。

若要檢視視 AWS 窗的最新更新AMIs，請參閱視窗AMI參考中的AWS Windows AMI 版本歷程記錄。

如需 EC2Launch v2 的最新版本，請參閱EC2Launchv2 版本歷史記錄。

如需 EC2Launch v2 移轉工具的最新版本，請參閱EC2Launchv2 遷移工具版本歷史記錄。

當 EC2Launch v2 服務的新版本發行時，您可以收到通知。如需詳細資訊，請參閱訂閱 EC2 Windows 啟動代理程式通知。

設定視窗執行個體的 EC2Launch V2 設定

本節包含如何設定 EC2Launch v2 設定的相關資訊。

主題包括：

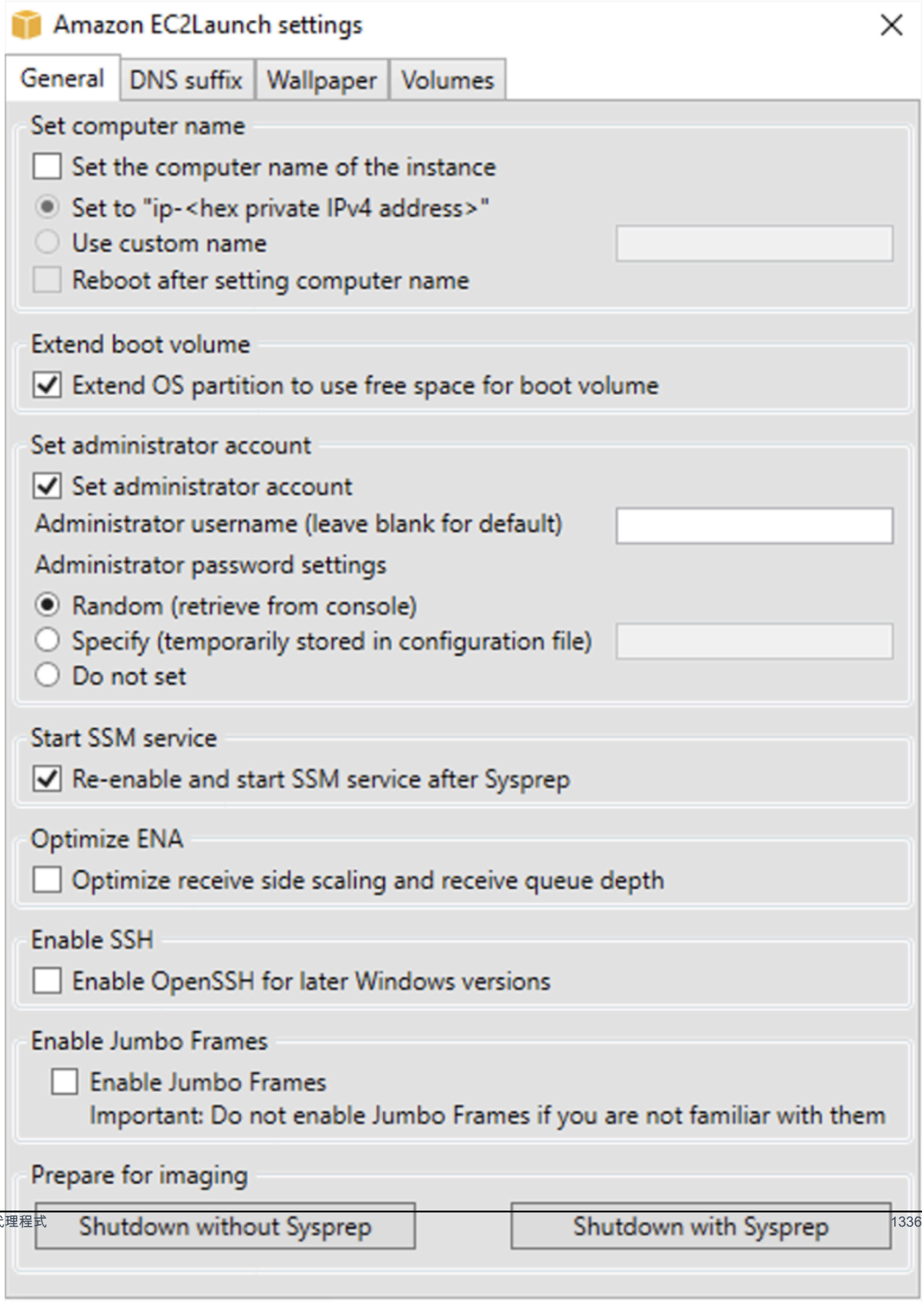
- 使用 EC2Launch v2 設定對話方塊變更設定
- EC2Launch使用 CLI
- EC2Launchv2 任務配置
- EC2Launchv2 退出代碼並重新啟動
- EC2LaunchV2 和系統

使用 EC2Launch v2 設定對話方塊變更設定

下列程序說明如何使用 EC2Launch v2 設定對話方塊來啟用或停用設定。

> **Note**
> 如果您在代理程式 config.yml 檔案中設定不當的自訂任務，並嘗試開啟 Amazon EC2Launch 設定對話方塊，將會收到錯誤訊息。有關架構範例，請參閱 範例：`agent-config.yml`。

1. 啟動和連線至 Windows 執行個體。
2. 從 [開始] 功能表中，選擇 [所有程式]，然後瀏覽至 EC2Launch[設定]。


Amazon EC2Launch settings
General
DNS suffix
Wallpaper
Volumes
Set computer name
Set the computer name of the instance
Set to "ip-<hex private IPv4 address>"
Use custom name
Reboot after setting computer name
Extend boot volume
Extend OS partition to use free space for boot volume
Set administrator account
Set administrator account
Administrator username (leave blank for default)
Administrator password settings
Random (retrieve from console)
Specify (temporarily stored in configuration file)
Do not set
Start SSM service
Re-enable and start SSM service after Sysprep
Optimize ENA
Optimize receive side scaling and receive queue depth
Enable SSH
Enable OpenSSH for later Windows versions
Enable Jumbo Frames
Enable Jumbo Frames
Important: Do not enable Jumbo Frames if you are not familiar with them
Prepare for imaging
Shutdown without Sysprep
Shutdown with Sysprep

3. 在EC2Launch設定對話方塊的「一般」頁籤上，您可以啟用或停用下列設定。

   a. Set Computer Name (設定電腦名稱)

      如果啟用此設定 (預設為停用)，則會在每次開機時將目前主機名稱與所需主機名稱進行比較。如果主機名稱不相符，則會重設主機名稱，然後系統會選擇性地重新開機以取得新的主機名稱。如果未指定自訂主機名稱，則會使用十六進位格式的私人IPv4位址產生該名稱，例如，。`ip-AC1F4E6`若要避免現有的主機名稱遭到修改，請勿啟用此設定。

   b. 延伸開機磁碟區

      此設定將動態擴充 `Disk 0/Volume 0`，以包含任何尚未分割的空間。此設定在執行個體是從自訂大小的根裝置磁碟區啟動時很有用。

   c. 設定管理員帳戶

      啟用後，您可以為本機電腦上建立的管理員帳戶設定使用者名稱和密碼屬性。如果未啟用此功能，遵循 Sysprep 的系統不會建立系統管理員帳戶。只有在 `adminPassword` 是 `adminPasswordtype` 時，才要在 `Specify` 中提供密碼。

      密碼類型的定義如下：

      i. `Random`

         EC2Launch生成密碼並使用用戶的密鑰對其進行加密。系統會在執行個體啟動後停用此設定，以便執行個體重新開機或停止再啟動時，能持續使用這個密碼。

      ii. `Specify`

         EC2Launch使用您在中指定的密碼`adminPassword`。如果密碼不符合系統需求，請改為EC2Launch產生隨機密碼。系統會將該密碼以純文字存放於 `agent-config.yml`，並在 Sysprep 設定管理員密碼後予以刪除。EC2Launch使用用戶的密鑰加密密碼。

      iii. `Do not set`

         EC2Launch會使用您在 unattend.xml 檔案中指定的密碼。如果您未在 unattend.xml 中指定密碼，會停用管理員帳戶。

   d. 啟動SSM服務

      選取此選項時，Systems Manager 服務會啟用，以啟動以下 Sysprep。EC2Launchv2 會執行先前所述的所有工作，而SSM代理程式會處 Systems Manager 功能的要求，例如執行命令和狀態管理員。

您可以使用執行命令來升級現有執行個體，以使用最新版本的 EC2Launch v2 服務和SSM代理程式。如需詳細資訊，請參閱《AWS 系統管SSM理員使用指南》中的使用執行命令更新代理程式。

e. 最佳化 ENA

選取時，會ENA設定設定以確保「ENA接收端調整」和「接收佇列深度」設定已針對最佳化 AWS。如需詳細資訊，請參閱設定接收端調整調整CPU相似。

f. 啟用 SSH

此設定會SSH針對較新的 Windows 版本啟用「開啟」，以允許遠端系統管理。

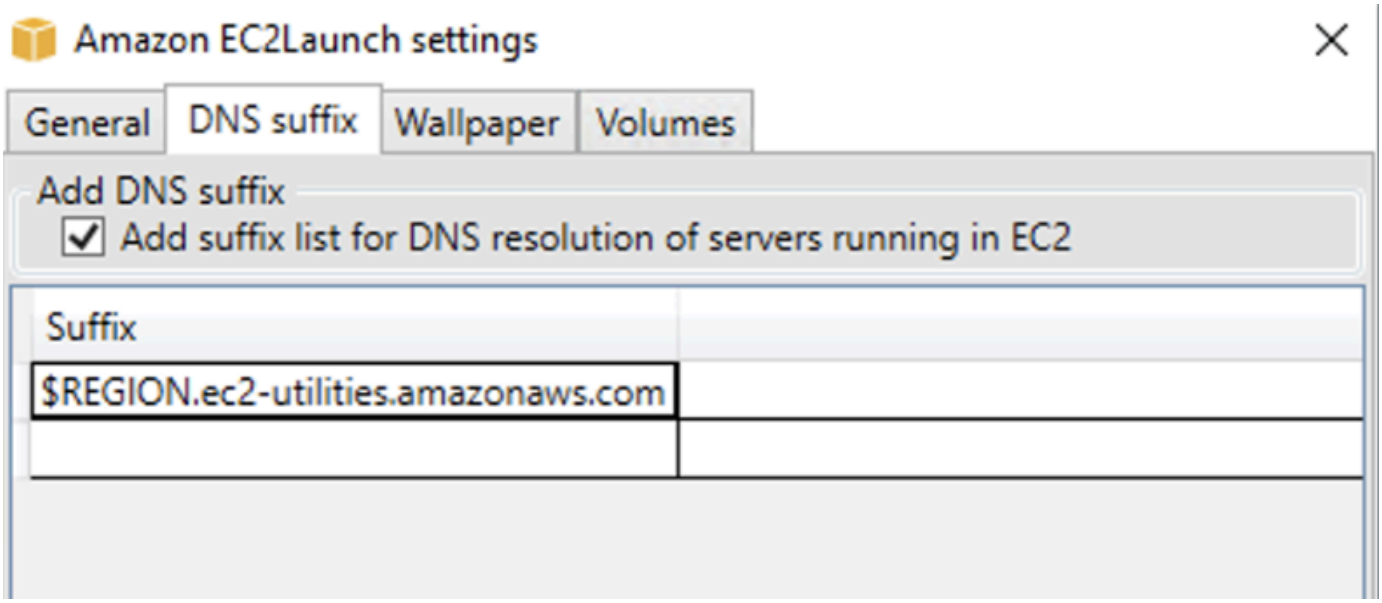
g. 啟用巨型訊框

選取以啟用「巨型訊框」。「巨型訊框」可能會對您的網路通訊產生意想不到的影響，因此在啟用之前，請務必了解「巨型訊框」對您的系統的影響。如需「巨型訊框」的詳細資訊，請參閱 巨型框MTU架。

h. 準備建立映像

選取是否要讓EC2執行個體在使用或不使用 Sysprep 的情況下關閉。當您想要使用 EC2Launch v2 執行 Sysprep 時，請選擇使用 Sysprep 關機。

4. 在 [字DNS尾] 索引標籤上，您可以選取是否要新增DNS尾碼清單來DNS解析中執行的伺服器 EC2，而不提供完整網域名稱。DNS後綴可以包含變量`$REGION`和. `$AZ` 只有不存在的尾碼才會新增到清單中。

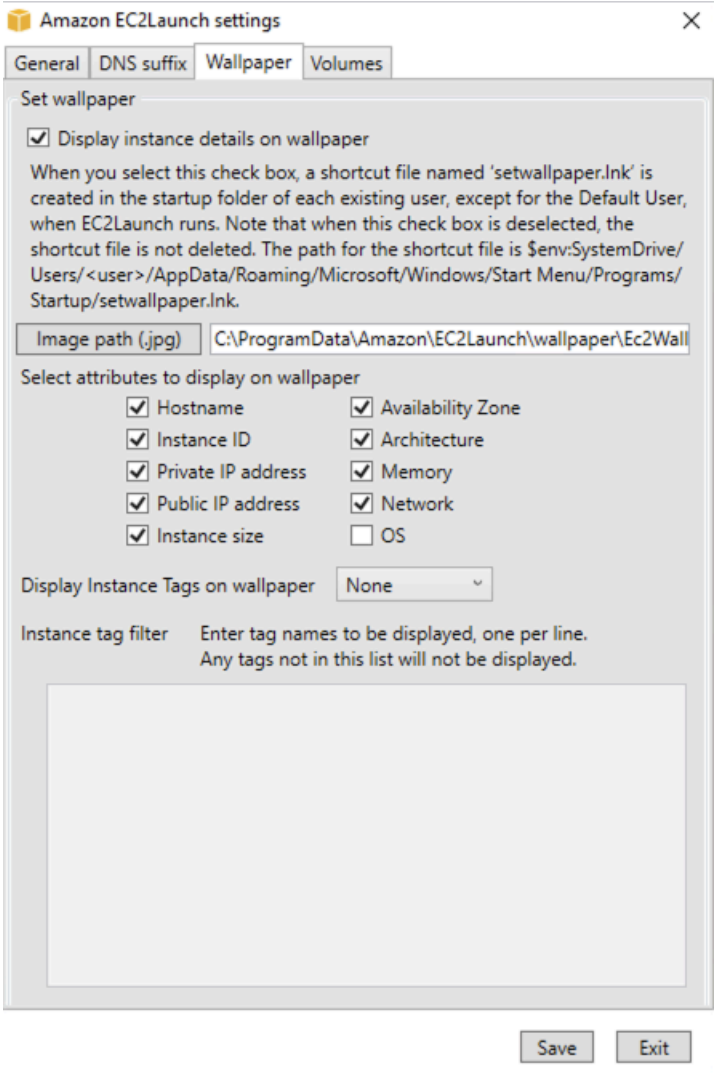


5. 在桌布索引標籤中，您可以使用背景圖片來設定執行個體桌布，並指定要顯示之桌布的執行個體詳細資訊。Amazon 每次登錄時都EC2會生成詳細信息。

   您可以使用下列控制項來設定桌布。

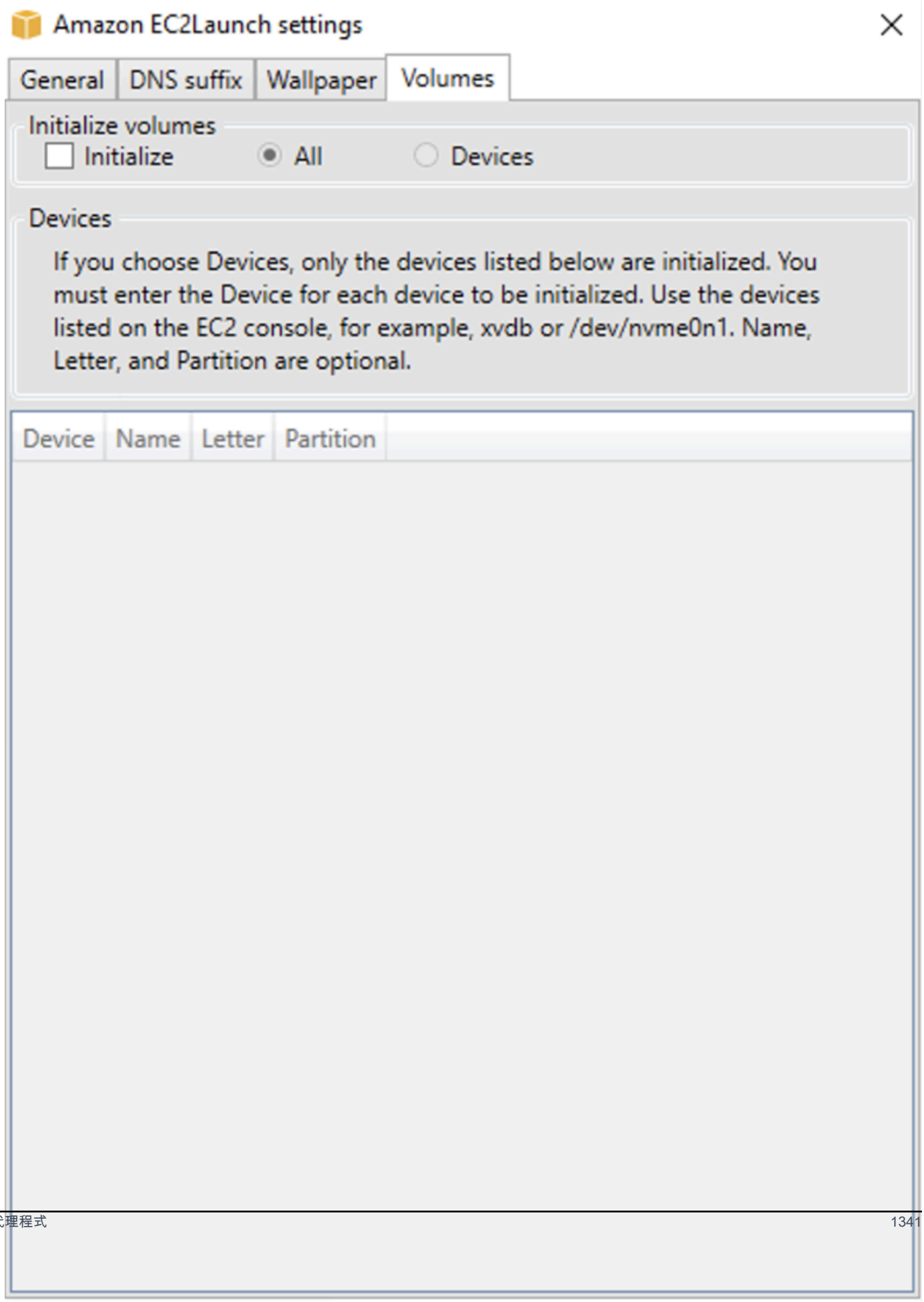
   - 在桌布上顯示執行個體詳細資訊 - 此核取方塊會啟用或停用桌布上的執行個體詳細資訊顯示。
   - 圖片路徑 (.jpg) - 指定要用作桌布背景的圖片路徑。
   - 選取要在桌布上顯示的屬性 - 選取要顯示在桌布上之執行個體詳細資訊的核取方塊。清除要從桌布中移除的先前選取的執行個體詳細資訊的核取方塊。
   - 在桌布上顯示執行個體標籤 - 選取下列其中一個設定以在桌布上顯示執行個體標籤：
     - 無 - 不在桌布上顯示任何執行個體標籤。
     - 顯示全部 - 在桌布上顯示所有執行個體標籤。
     - 顯示篩選結果 - 在桌布上顯示指定的執行個體標籤。當您選取此設定時，可以在執行個體標籤篩選方塊中新增您想要顯示在桌布上的執行個體標籤。

   > **Note**
   >
   > 您必須在中繼資料中啟用標籤，才能在桌布上顯示標籤。如需有關執行個體標籤和中繼資料的詳細資訊，請參閱 使用執行個體中繼資料檢視EC2執行個體的。



6. 在磁碟區標籤上，選取是否要初始化連線至執行個體的磁碟區。啟用會設定任何額外磁碟區的磁碟機代號，並將它們延伸到使用可用空間。如果您選取全部，則會初始化所有儲存磁碟區。如果您選取裝置，則只會初始化清單中指定的裝置。您必須為每個要初始化的裝置輸入裝置。使用EC2主控台上列出的裝置，例如，`xvdb`或`/dev/nvme0n1`。下拉式清單會顯示連接到執行個體的儲存磁碟區。若要輸入未連接到執行個體的裝置，請在文字欄位中輸入該裝置。

   名稱、字母和分割區是選填欄位。如果未指定分割區的值，則會以磁碟分割類型初始化大於 2 TB 的儲存磁碟`gpt`區，而小於 2 TB 的儲存磁碟區會以磁碟`mbr`分割類型初始化。如果已設定裝置，且非NTFS裝置包含分割區表格，或是磁碟的前 4 KB 包含資料，則會略過磁碟並記錄動作。


Amazon EC2Launch settings
General
DNS suffix
Wallpaper
Volumes
Initialize volumes
Initialize
All
Devices
Devices
If you choose Devices, only the devices listed below are initialized. You must enter the Device for each device to be initialized. Use the devices listed on the EC2 console, for example, xvdb or /dev/nvme0n1. Name, Letter, and Partition are optional.
Device
Name
Letter
Partition

以下是從EC2Launch對話方塊中輸入的設定建立的範例規劃YAML檔案。

```
version: 1.0
config:
  - stage: boot
 tasks:
      - task: extendRootPartition
  - stage: preReady
    tasks:
      - task: activateWindows
        inputs:
          activation:
            type: amazon
      - task: setDnsSuffix
        inputs:
          suffixes:
            - $REGION.ec2-utilities.amazonaws.com
      - task: setAdminAccount
        inputs:
          password:
            type: random
      - task: setWallpaper
        inputs:
          path: C:\ProgramData\Amazon\EC2Launch\wallpaper\Ec2Wallpaper.jpg
          attributes:
            - hostName
            - instanceId
            - privateIpAddress
            - publicIpAddress
            - instanceSize
            - availabilityZone
            - architecture
            - memory
            - network
  - stage: postReady
    tasks:
      - task: startSsm
```

EC2Launch使用 CLI

您可以使用指令行介面 (CLI) 來規劃您的EC2Launch設定和管理服務。下一節包含您可用來管理EC2Launch v2 之CLI命令的說明和使用資訊。

命令

- collect-logs
- get-agent-config
- list-volumes
- 重設
- run
- status
- sysprep
- validate
- version
- 桌布

collect-logs

收集的記錄檔EC2Launch、壓縮檔案，並將其放置在指定的目錄中。

範例

```
ec2launch collect-logs -o C:\Mylogs.zip
```

用途

`ec2launch collect-logs [flags]`

Flags

`-h, --help`

`collect-logs` 的說明

`-o, --output string`

壓縮輸出日誌的路徑

get-agent-config

以指定`agent-config.yml`的格式 (JSON或YAML) 列印。如果未指定任何格式，則會以先前指定的格式列印 `agent-config.yml`。

範例

```
ec2launch get-agent-config -f json
```

範例 2

下面的 PowerShell 命令顯示如何編輯和保存JSON格式的`agent-config`文件。

```
$config = & "$env:ProgramFiles/Amazon/EC2Launch/EC2Launch.exe" --format json |
 ConvertFrom-Json
$jumboFrame =@"
{
   "task": "enableJumboFrames"
}
"@
$config.config | %{if($_.stage -eq 'postReady'){$_.tasks += (ConvertFrom-Json -
InputObject $jumboFrame)}}
$config | ConvertTo-Json -Depth 6 | Out-File -encoding UTF8
$env:ProgramData/Amazon/EC2Launch/config/agent-config.yml
```

用途

`ec2launch get-agent-config [flags]`

Flags

`-h, --help`

`get-agent-config` 的說明

`-f, --format string`

`agent-config` 檔案的輸出格式：`json`、`yaml`

list-volumes

列出連接至執行個體的所有儲存磁碟區，包括暫時磁碟區和磁碟區。EBS

範例

```
ec2launch list-volumes
```

用途

`ec2launch list-volumes`

Flags

`-h, --help`

`list-volumes` 的說明

重設

此任務的主要目標是在下次代理程式執行時重設該代理程式。若要這麼做，命reset令會從本機`EC2Launch`目錄刪除 EC2Launch v2 的所有代理程式狀態資料 (請參閱EC2Launchv2 目錄結構)。選擇性重設會刪除服務和 Sysprep 日誌。

指令碼行為取決於代理程式執行指令碼的模式 (內嵌或分離)。

內嵌 (預設)

EC2Launchv2 代理程式一次執行一個指令碼 (`detach: false`)。這是預設設定。

> **Note**
>
> 當內嵌指令碼發出 reset 或 sysprep 命令時，它會立即執行並重設代理程式。目前任務完成後，代理程式就會關閉，而不執行任何後續任務。
> 例如，如果發出命令的任務後緊接著一個 `startSsm` 任務 (預設情況下會在使用者資料執行之後包含)，則任務不會執行，而且 Systems Manager 服務永遠不會啟動。

Detached

EC2Launchv2 代理程式會與其他工作同時執行指令碼 (`detach: true`)。

> **Note**
>
> 當分離的指令碼發出 reset 或 sysprep 命令時，這些命令會等待代理程式完成再執行。之後的工作仍 executeScript 會執行。

範例

```
ec2launch reset -c
```

用途

`ec2launch reset [flags]`

Flags

`-c`, `--clean`

在 `reset` 前清除執行個體日誌

`-h`, `--help`

`reset` 的說明

run

運行 EC2Launch V2。

範例

```
ec2launch run
```

用途

`ec2launch run [flags]`

Flags

`-h`, `--help`

`run` 的說明

status

取得 EC2Launch v2 代理程式的狀態。選擇性地封鎖處理程序，直到代理程式完成為止。處理程序結束程式碼會決定代理程式狀態：

- `0` – 代理程式已執行且成功。
- `1` – 代理程式已執行且失敗。
- `2` – 代理程式仍在執行中。

- 3 – 代理程式處於未知狀態。代理程式狀態為未執行或已停止。
- 4 – 嘗試擷取代理程式狀態時發生錯誤。
- 5 – 代理程式未執行，且上次已知執行的狀態未知。這可能表示以下其中一項：
  - `state.json` 和 `previous-state.json` 均被刪除。
  - `previous-state.json` 已毀損。

  這是執行 `reset` 命令後的代理程式狀態。

範例：

```
ec2launch status -b
```

用途

`ec2launch status [flags]`

Flags

`-b,--block`

封鎖處理程序，直到代理程式完成執行

`-h,--help`

`status` 的說明

sysprep

此任務的主要目標是在下次代理程式執行時重設該代理程式。為此，sysprep命令會重設代理程式狀態、更新`unattend.xml`檔案RDP、停用並執行 Sysprep。

指令碼行為取決於代理程式執行指令碼的模式 (內嵌或分離)。

內嵌 (預設)

EC2Launchv2 代理程式一次執行一個指令碼 (`detach: false`)。這是預設設定。

> **Note**
>
> 當內嵌指令碼發出 reset 或 sysprep 命令時，它會立即執行並重設代理程式。目前任務完成後，代理程式就會關閉，而不執行任何後續任務。

例如，如果發出命令的任務後緊接著一個 `startSsm` 任務 (預設情況下會在使用者資料執行之後包含)，則任務不會執行，而且 Systems Manager 服務永遠不會啟動。

Detached

EC2Launchv2 代理程式會與其他工作同時執行指令碼 (`detach: true`)。

> **Note**
> 當分離的指令碼發出 reset 或 sysprep 命令時，這些命令會等待代理程式完成再執行。之後的工作仍 executeScript 會執行。

範例：

```
ec2launch sysprep
```

用途

`ec2launch sysprep [flags]`

Flags

`-c,--clean`

在 `sysprep` 前清除執行個體日誌

`-h,--help`

Sysprep 說明

`-s,--shutdown`

在 `sysprep` 後關閉執行個體

validate

驗證 `agent-config` 檔案 `C:\ProgramData\Amazon\EC2Launch\config\agent-config.yml`。

範例

```
ec2launch validate
```

用途

`ec2launch validate [flags]`

Flags

-h ，`--help`

`validate` 的說明

version

取得可執行版本。

範例

```
ec2launch version
```

用途

`ec2launch version [flags]`

Flags

`-h, --help`

`version` 的說明

桌布

將新桌布設定為提供的桌布路徑 (.jpg 檔案)，並顯示選取的執行個體詳細資料。

語法

```
ec2launch wallpaper ^
--path="C:\ProgramData\Amazon\EC2Launch\wallpaper\Ec2Wallpaper.jpg" ^
--all-tags ^
--
attributes=hostName,instanceId,privateIpAddress,publicIpAddress,instanceSize,availabilityZone,a
```

輸入

參數

-允許-標籤 [*tag-name-1*, *tag-name-n*]

(選擇性) 要在桌布上顯示的實體標籤名稱的 Base64 編碼JSON陣列。您可以使用此標籤或 `--all-tags`，但不能同時使用兩者。

-屬性 *attribute-string-1*, *attribute-string-n*

(選用) 要將設定套用至桌布的 `wallpaper` 屬性字串清單 (以逗號分隔)。

[--路徑 |-p] *path-string*

(必要) 指定 `wallpaper` 背景圖片檔案路徑。

旗標

--all-tags

(選用) 在桌布上顯示所有執行個體標籤。您可以使用此標籤或 `--allowed-tags`，但不能同時使用兩者。

[--help | -h]

顯示 wallpaper 命令的說明。

EC2Launchv2 任務配置

本節包含 `agent-config.yml` 和使用者資料的組態結構描述任務、詳細資訊及範例。

任務與範例

- 結構描述：agent-config.yml
- 設定啟動或重新開機期間執行的 EC2Launch v2 使用者資料指令碼

結構描述：**`agent-config.yml`**

`agent-config.yml` 檔案的結構如下所示。請注意，不能在同一階段重複任務。有關任務屬性，請參閱後面的任務說明。

文件結構：agent-config.yml

JSON

```
{
 "version": "1.0",
 "config": [
  {
   "stage": "string",
   "tasks": [
    {
     "task": "string",
     "inputs": {
      ...
     }
    },
    ...
   ]
  },
  ...
 ]
}
```

YAML

```
version: 1.0
config:
- stage: string
  tasks:
  - task: string
 inputs:
   ...
  ...
...
```

範例：**agent-config.yml**

下列範例顯示 `agent-config.yml` 組態檔的設定。

```
version: 1.0
config:
- stage: boot
```

```
  tasks:
  - task: extendRootPartition
- stage: preReady
  tasks:
  - task: activateWindows
    inputs:
      activation:
        type: amazon
  - task: setDnsSuffix
    inputs:
      suffixes:
      - $REGION.ec2-utilities.amazonaws.com
  - task: setAdminAccount
    inputs:
      password:
        type: random
  - task: setWallpaper
    inputs:
      path: C:\ProgramData\Amazon\EC2Launch\wallpaper\Ec2Wallpaper.jpg
      attributes:
      - hostName
      - instanceId
      - privateIpAddress
      - publicIpAddress
      - instanceSize
      - availabilityZone
      - architecture
      - memory
      - network
- stage: postReady
  tasks:
  - task: startSsm
```

設定啟動或重新開機期間執行的 EC2Launch v2 使用者資料指令碼

下列YAML範例JSON和範例顯示使用者資料的文件結構。Amazon 會EC2剖析您在文件中指定的`tasks`陣列中指定的每個任務。每個任務都有自己的屬性集和要求集。如需詳細資訊，請參閱 EC2Launchv2 啟動工作的工作定義。

> **Note**
>
> 工作只能在使用者資料工作陣列中出現一次。

文件結構：使用者資料

JSON

```
{
 "version": "1.1",
 "tasks": [
  {
   "task": "string",
   "inputs": {
    ...
   },
  },
  ...
 ]
}
```

YAML

```
version: 1.1
tasks:
- task: string
  inputs:
    ...
...
```

範例：使用者資料

如需使用者資料的詳細資訊，請參閱 Amazon 如何EC2處理 Windows 實例的用戶數據。

下列YAML文件範例顯示 EC2Launch v2 以使用者資料的身分執行以建立檔案的 PowerShell 指令碼。

```
version: 1.1
tasks:
- task: executeScript
  inputs:
  - frequency: always
    type: powershell
    runAs: localSystem
    content: |-
      New-Item -Path 'C:\PowerShellTest.txt' -ItemType File
```

您可以使用與舊版啟動代理程XML式相容的使用者資料格式。EC2Launchv2 將腳本作為`UserData`階段中的`executeScript`任務運行。為了符合 EC2Launch v1 和EC2Config行為，使用者資料指令碼依預設會以附加/內嵌程序的方式執行。

您可以新增選用標籤來自訂指令碼的執行方式。例如，除了在執行個體啟動時執行一次使用者資料指令碼之外，若還想在執行個體重新啟動時執行，您可以使用下列標記：

`<persist>true</persist>`

範例：

```
<powershell>
  $file = $env:SystemRoot + "\Temp" + (Get-Date).ToString("MM-dd-yy-hh-mm")
  New-Item $file -ItemType file
</powershell>
<persist>true</persist>
```

您可以使用`<powershellArguments>`標籤指定一個或多個 PowerShell 引數。如果沒有傳遞任何引數，EC2Launchv2 預設會新增下列引數：`-ExecutionPolicy Unrestricted`

範例：

```
<powershell>
  $file = $env:SystemRoot + "\Temp" + (Get-Date).ToString("MM-dd-yy-hh-mm")
  New-Item $file -ItemType file
</powershell>
<powershellArguments>-ExecutionPolicy Unrestricted -NoProfile -NonInteractive</
powershellArguments>
```

若要將XML使用者資料指令碼做為分離處理程序執行，請將下列標記新增至您的使用者資料。

`<detach>true</detach>`

範例：

```
<powershell>
  $file = $env:SystemRoot + "\Temp" + (Get-Date).ToString("MM-dd-yy-hh-mm")
  New-Item $file -ItemType file
</powershell>
<detach>true</detach>
```

> **Note**
> 舊版啟動代理程式不支援分離標籤。

變更日誌：使用者資料

下表列出使用者資料的變更，並將其交叉參照至適用的 EC2Launch v2 代理程式版本。

使用者資料版本	詳細資訊	引入於
1.1	• 使用者資料任務會在代理程式組態檔案中的 `PostReady` 階段之前執行。 • 在啟動系統管理員代理程式之前執行使用者資料 (行為與 EC2Launch v1 和相同EC2Config) 。 *	EC2Launch V2 版本
1.0	• 將被棄用。 • 使用者資料任務會在代理程式組態檔案中的 `PostReady` 階段之後執行。這與 EC2Launch v1 不向後兼容。 • 受 Systems Manager Agent 啟動與使用者資料任務之間的競爭條件影響。	EC2Launch 第二版本 2.0.0

* 與預設 `agent-config.yml` 檔案搭配使用時。

EC2Launchv2 退出代碼並重新啟動

您可以使用 EC2Launch v2 來定義腳本如何處理退出代碼。根據預設，在指令碼中執行的最後一個命令的結束程式碼會報告為整個指令碼的結束程式碼。例如，如果指令碼包含三個命令，而第一個命令失敗，但後續命令成功，則會因最終命令成功而將執行狀態報告為 `success`。

如果您希望指令碼重新啟動執行個體，則必須`exit 3010`在指令碼中指定，即使重新啟動是指令碼中的最後一個步驟也是如此。 `exit 3010`指示 EC2Launch v2 重新啟動執行個體並再次呼叫指令碼，直到傳回不是的結束代碼`3010`，或是達到最大重新啟動計數為止。EC2Launchv2 允許每個任務最多 5

次重新啟動。如果您嘗試使用不同的機制 (例如 `Restart-Computer`) 從指令碼重新啟動執行個體，則指令碼執行狀態將不一致。例如，狀態可能會卡在重新啟動迴圈或不執行重新啟動。

如果您使用的是與舊版代理程式相容的使用XML者資料格式，則使用者資料的執行次數可能會比您預期的要多。如需詳細資訊，請參閱故障診斷一節中的 服務會多次執行使用者資料。

EC2LaunchV2 和系統

EC2Launchv2 服務會執行 Sysprep，這是一個 Microsoft 工具，可讓您建立可重複使用AMI的自訂視窗。當 EC2Launch v2 呼叫 Sysprep 時，它會使用中的檔案`%ProgramData%\Amazon\EC2Launch`來決定要執行哪些作業。您可以使用EC2Launch設定對話方塊或直接使用編輯YAML器或文字編輯器來間接編輯這些檔案。但是，設定對話方塊中有些進階EC2Launch設定無法使用，因此您必須直接編輯這些項目。

如果您在更新執行個體設定後AMI從執行個體建立，則新設定會套用至從新執行個體啟動的任何執行個體AMI。若要取得有關建立的資訊AMI，請參閱創建一個 Amazon 支EBS持 AMI。

EC2Launchv2 啟動工作的工作定義

EC2Launchv2 在啟動或 statup 期間執行的每項工作都有自己的一組屬性和需求。工作詳細資訊包括工作執行頻率的設定 — 執行一次或一律執行的代理程式開機程序階段、語法和YAML文件範例。如需詳細資訊，請檢閱此參考資料中顯示的作業詳細資訊。

EC2Launch第 2 版工作

- activateWindows
- enableJumboFrames
- enableOpenSsh
- executeProgram
- executeScript
- extendRootPartition
- initializeVolume
- optimizeEna
- setAdminAccount
- setDnsSuffix
- setHostName
- setWallpaper

- startSsm
- sysprep
- writeFile

activateWindows

針對一組 AWS KMS 伺服器啟動視窗。如果偵測到執行個體為 Bring-Your-Own-License (，則會略過啟動。BYOL

Frequency — 一次

AllowedStages — `[PreReady]`

Inputs —

`activation`：(映射)

`type`：(字串) 要使用的啟動類型，設定為 `amazon`

範例

```
task: activateWindows
  inputs:
    activation:
    type: amazon
```

enableJumboFrames

啟用「超大型框架」，這會增加網路介面卡的最大傳輸單位 (MTU)。如需詳細資訊，請參閱巨型框MTU架。

Frequency – 始終

AllowedStages — `[PostReady, UserData]`

Inputs — 無

範例

```
task: enableJumboFrames
```

enableOpenSsh

啟用 Windows 開啟，SSH並將執行個體的公開金鑰新增至授權金鑰資料夾。

Frequency — 一次

AllowedStages — `[PreReady, UserData]`

Inputs — 無

範例

下列範例顯示如何在執行個體SSH上啟用 Open，以及如何將執行個體的公開金鑰新增至授權金鑰資料夾。此組態僅適用於執行 Windows Server 2019 及更新版本的執行個體。

```
task: enableOpenSsh
```

executeProgram

執行具有可選引數和指定頻率的程式。

階段：您可以在 `PreReady`、`PostReady` 和 `UserData` 階段執行 `executeProgram` 任務。

頻率：可設定，請參閱輸入。

輸入

本節包含要執行的executeProgram工作 (輸入) 的一或多個程式。每個輸入都可以包括以下可配置的設置：

頻率 (字串)

(必要) 僅指定下列其中一個值：

- `once`
- `always`

路徑 (字串)

(必要) 要執行的可執行檔的檔案路徑。

引數 (字串清單)

(選用) 以逗號分隔的引數清單，作為輸入提供給程式。

runAs （字符串）

(必要) 必須設定為 `localSystem`

輸出

所有任務都會將日誌檔案條目寫入 `agent.log` 檔案。`executeProgram` 任務的其他輸出會分別儲存在動態命名的資料夾中，如下所示：

`%LocalAppData%\Temp\`*`EC2Launch########`*`\`*`outputfilename.tmp`*

輸出檔案的確切路徑包含在 `agent.log` 檔案中，例如：

```
Program file is created at: C:\Windows\system32\config\systemprofile\AppData\Local
\Temp\EC2Launch123456789\ExecuteProgramInputs.tmp
Output file is created at: C:\Windows\system32\config\systemprofile\AppData\Local
\Temp\EC2Launch123456789\Output.tmp
Error file is created at: C:\Windows\system32\config\systemprofile\AppData\Local
\Temp\EC2Launch123456789\Err.tmp
```

`executeProgram` 任務的輸出檔案

`ExecuteProgramInputs.tmp`

包含可執行檔的路徑，以及 `executeProgram` 任務在執行時傳送給其所有輸入參數。

`Output.tmp`

包含 `executeProgram` 任務執行之程式的執行期輸出。

`Err.tmp`

包含 `executeProgram` 任務執行之程式的執行期錯誤訊息。

範例

下列範例說明如何透過 `executeProgram` 任務，從執行個體上的本機目錄執行可執行檔。

範例 1：具有一個引數的設定可執行檔

此範例顯示在靜默模式下執行設定可執行檔的 `executeProgram` 任務。

```
task: executeProgram
```

```
  inputs:
    - frequency: always
      path: C:\Users\Administrator\Desktop\setup.exe
      arguments: ['-quiet']
```

範例 2：具有兩個引數的VLC可執行檔

此範例顯示執行VLC可執行檔的`executeProgram`工作，其中包含以輸入參數傳遞的兩個引數。

```
task: executeProgram
  inputs:
    - frequency: always
      path: C:\vlc-3.0.11-win64.exe
      arguments: ['/L=1033','/S']
      runAs: localSystem
```

executeScript

執行具有可選引數和指定頻率的指令碼。指令碼行為取決於代理程式執行指令碼的模式 (內嵌或分離)。

內嵌 (預設)

EC2Launchv2 代理程式一次執行一個指令碼 (`detach: false`)。這是預設設定。

> **Note**
>
> 當內嵌指令碼發出 reset 或 sysprep 命令時，它會立即執行並重設代理程式。目前任務完成後，代理程式就會關閉，而不執行任何後續任務。
> 例如，如果發出命令的任務後緊接著一個 `startSsm` 任務 (預設情況下會在使用者資料執行之後包含)，則任務不會執行，而且 Systems Manager 服務永遠不會啟動。

Detached

EC2Launchv2 代理程式會與其他工作同時執行指令碼 (`detach: true`)。

> **Note**
>
> 當分離的指令碼發出 reset 或 sysprep 命令時，這些命令會等待代理程式完成再執行。之後的工作仍 executeScript 會執行。

階段：您可以在 `PreReady`、`PostReady` 和 `UserData` 階段執行 `executeScript` 任務。

頻率：可設定，請參閱輸入。

輸入

本節包含要執行的executeScript工作 (輸入) 的一或多個指令碼。每個輸入都可以包括以下可配置的設置：

頻率 (字串)

(必要) 僅指定下列其中一個值：

- `once`
- `always`

類型 (字串)

(必要) 僅指定下列其中一個值：

- `batch`
- `powershell`

引數 (字串清單)

(選用) 傳遞給 shell 的字串引數清單。`type: batch` 不支援此參數。如果沒有傳遞任何引數，EC2Launchv2 預設會新增下列引數：`-ExecutionPolicy Unrestricted`

內容 (字串)

(必要) 指令碼內容。

runAs （字符串）

(必要) 僅指定下列其中一個值：

- `admin`
- `localSystem`

分離 (布林值)

(選擇性) EC2Launch v2 代理程式預設為一次執行一個指令碼 (`detach: false`)。若要與其他任務同時執行指令碼，請將值設定為 `true` (`detach: true`)。

> **Note**
>
> 當 `detach` 設定為 `true` 時，指令碼結束代碼 (包括 `3010`) 沒有任何作用。

輸出

所有任務都會將日誌檔案條目寫入 `agent.log` 檔案。`executeScript` 任務執行之指令碼的其他輸出會分別儲存在動態命名的資料夾中，如下所示：

`%LocalAppData%\Temp\`*`EC2Launch########`*`\`*`outputfilename.ext`*

輸出檔案的確切路徑包含在 `agent.log` 檔案中，例如：

```
Program file is created at: C:\Windows\system32\config\systemprofile\AppData\Local
\Temp\EC2Launch123456789\UserScript.ps1
Output file is created at: C:\Windows\system32\config\systemprofile\AppData\Local
\Temp\EC2Launch123456789\Output.tmp
Error file is created at: C:\Windows\system32\config\systemprofile\AppData\Local
\Temp\EC2Launch123456789\Err.tmp
```

`executeScript` 任務的輸出檔案

`UserScript.`*`ext`*

包含 `executeScript` 任務執行的指令碼。檔案的副檔名取決於您在 `executeScript` 任務的 `type` 參數中指定的指令碼類型，如下所示：

- 如果類型為 `batch`，則檔案副檔名為 `.bat`。
- 如果類型為 `powershell`，則檔案副檔名為 `.ps1`。

`Output.tmp`

包含 `executeScript` 任務執行之指令碼的執行期輸出。

`Err.tmp`

包含 `executeScript` 任務執行之指令碼的執行期錯誤訊息。

範例

下列範例示範如何使用 `executeScript` 任務執行內嵌指令碼。

範例 1：Hello World 輸出文字檔案

此範例顯示`executeScript`執行 PowerShell 指令碼以在`C:`磁碟機上建立「Hello world」文字檔的工作。

```
task: executeScript
```

```
  inputs:
    - frequency: always
      type: powershell
      runAs: admin
      content: |-
        New-Item -Path 'C:\PowerShellTest.txt' -ItemType File
        Set-Content 'C:\PowerShellTest.txt' "Hello world"
```

範例 2：執行兩個指令碼

此範例顯示 `executeScript` 任務可以執行多個指令碼，而指令碼類型不一定需要相符。

第一個指令碼 (`type: powershell`) 會將執行個體上目前執行的程序摘要寫入位於 `C:` 磁碟機上的文字檔案。

第二個指令碼 (`batch`) 會將系統資訊寫入 `Output.tmp` 檔案。

```
task: executeScript
  inputs:
    - frequency: always
      type: powershell
      content: |
        Get-Process | Out-File -FilePath C:\Process.txt
      runAs: localSystem
    - frequency: always
      type: batch
        content: |
        systeminfo
```

範例 3：具有重新啟動功能的等冪系統組態

此範例顯示 `executeScript` 任務，其執行等冪指令碼以執行下列系統組態，並在每個步驟之間重新啟動：

- 重新命名電腦。
- 將電腦加入網域。
- 啟用 Telnet。

該指令碼確保每個操作只執行一次。這可以防止重新啟動循環，並讓指令碼具有等冪性。

```
task: executeScript
  inputs:
    - frequency: always
```

```
      type: powershell
      runAs: localSystem
      content: |-
        $name = $env:ComputerName
        if ($name -ne $desiredName) {
          Rename-Computer -NewName $desiredName
          exit 3010
        }
        $domain = Get-ADDomain
        if ($domain -ne $desiredDomain)
        {
          Add-Computer -DomainName $desiredDomain
          exit 3010
        }
        $telnet = Get-WindowsFeature -Name Telnet-Client
        if (-not $telnet.Installed)
        {
          Install-WindowsFeature -Name "Telnet-Client"
          exit 3010
        }
```

extendRootPartition

延伸根磁碟區以使用磁碟上的所有可用空間。

Frequency — 一次

AllowedStages — `[Boot]`

Inputs — 無

範例

```
task: extendRootPartition
```

initializeVolume

初始化連接到執行個體的空白磁碟區，以便啟動和分割這些磁碟區。如果啟動代理程式偵測到磁碟區並非空白，就會略過初始化。如果磁碟區的前 4 KiB 空白，或磁碟區沒有 Windows 可辨識的磁碟機配置，磁碟區會視為空白。

不論磁碟機是否已經初始化，在此任務執行時永遠都會套用 `letter` 輸入參數。

initializeVolume 任務會執行下列動作。

- 將磁碟屬性 offline 和 readonly 設為 false。
- 建立分割區。如果未在 partition 輸入參數中指定分割區類型，會套用下列預設值：
  - 如果磁碟大小小於 2 TB，請將分割區類型設為 mbr。
  - 如果磁碟大小大於等於 2 TB，請將分割區類型設為 gpt。
- 將磁碟區格式化為NTFS。
- 將磁碟區標籤設定如下：
  - 如果有指定，請使用 name 輸入參數的值。
  - 如果磁碟區為暫時性，且未指定名稱，請將磁碟區標籤設為 Temporary Storage Z。
- 如果磁碟區是暫時的 (SSD或HDD不是 AmazonEBS)，請在磁碟區的根目錄建立Important.txt檔案，其中包含下列內容：

```
This is an 'Instance Store' disk and is provided at no additional charge.

*This disk offers increased performance since it is local to the host
*The number of Instance Store disks available to an instance vary by instance type
*DATA ON THIS DRIVE WILL BE LOST IN CASES OF IMPAIRMENT OR STOPPING THE INSTANCE.
 PLEASE ENSURE THAT ANY IMPORTANT DATA IS BACKED UP FREQUENTLY

For more information, please refer to: Amazon EC2 ###############.
```

- 將磁碟機代號設定為 letter 輸入參數中指定的值。

階段：您可以在 PostReady 和 UserData 階段執行 initializeVolume 任務。

頻率：始終。

輸入

您可以設定執行期參數，如下所示：

devices (映射清單)

(視情況而定) 啟動代理程式初始化之每個裝置的組態。initialize 輸入參數設為 devices 時，此為必要項目。

- device (字串，必要) – 在建立執行個體期間識別裝置。例如，xvdb、xvdf 或 \dev \nvme0n1。

- letter (字串，選用) – 一個字元。要指派的磁碟機代號。
- name (字串，選用) – 要指派的磁碟區名稱。
- partition (字串，選用) – 為要建立的分割區類型指定下列任一個值，或根據磁碟區大小讓啟動代理程式設定預設值：
  - MBR
  - gpt

initialize (字串)

(必要) 僅指定下列其中一個值：

- `all`
- `devices`

範例

下列範例說明 `initializeVolume` 任務的輸入組態範例。

範例 1：在執行個體上初始化兩個磁碟區

此範例說明在執行個體上初始化兩個次要磁碟區的 `initializeVolume` 任務。範例中的裝置 `DataVolume2` 為暫時性。

```
task: initializeVolume
inputs:
  initialize: devices
  devices:
  - device: xvdb
    name: DataVolume1
    letter: D
    partition: mbr
  - device: /dev/nvme0n1
    name: DataVolume2
    letter: E
    partition: gpt
```

範例 2：初始化連接至執行個體的EBS磁碟區

此範例顯示的`initializeVolume`工作會初始化所有連接至執行個體的空白EBS磁碟區。

```
task: initializeVolume
inputs:
  initialize: all
```

optimizeEna

根據目前的執行個體類型最佳化ENA設定；可能會重新啟動執行個體。

Frequency – 始終

AllowedStages — `[PostReady, UserData]`

Inputs — 無

範例

```
task: optimizeEna
```

setAdminAccount

為在本機電腦上建立的預設管理員帳戶設定屬性。

Frequency — 一次

AllowedStages — `[PreReady]`

Inputs —

`name`：(字串) 管理員帳戶的名稱

`password`：(映射)

`type`：(字符) 策略來設置密碼，無論是 `static`、`random` 或 `doNothing`

`data`：(字串) 儲存資料，如果 `type` 欄位是靜態的話

範例

```
task: setAdminAccount
inputs:
  name: Administrator
  password:
```

```
  type: random
```

setDnsSuffix

將DNS後綴添加到搜索後綴列表中。只有不存在的尾碼才會加入到清單中。如需啟動代理程式如何設定DNS尾碼的詳細資訊，請參閱。設定 EC2 Windows 啟動代理程式的DNS尾碼

Frequency – 始終

AllowedStages — `[PreReady]`

Inputs —

`suffixes`：（字符串列表）一個或多個有效DNS後綴列表; 有效的替代變量是和 `$REGION` `$AZ`

範例

```
task: setDnsSuffix
inputs:
  suffixes:
  - $REGION.ec2-utilities.amazonaws.com
```

setHostName

將電腦的主機名稱設定為自訂字串，如果未指定，`hostName`則設定為私人IPv4位址。

Frequency – 始終

AllowedStages — `[PostReady, UserData]`

Inputs —

`hostName`：(字串) 選擇性的主機名稱，必須格式化如下。

- 必須等於或少於 15 個字元
- 必須只包含英數字元 (a-z、A-Z、0-9) 和連字號 (-) 字元。
- 不得完全由數字字元組成。

`reboot`：(布林值) 表示當主機名被更改時是否允許重新啟動

範例

```
task: setHostName
inputs:
  reboot: true
```

setWallpaper

除了 Default User 之外，在每個現有使用者的啟動資料夾中建立 setwallpaper.lnk 捷徑檔案。使用者在執行個體啟動後第一次登入時，此捷徑檔案會執行。這會使用顯示執行個體屬性的自訂桌布，設定執行個體。

捷徑檔案路徑為：

```
$env:SystemDrive/Users/<user>/AppData/Roaming/Microsoft/Windows/Start Menu/Programs/
Startup/setwallpaper.lnk
```

> **Note**
>
> 移除 setWallpaper 任務時，不會刪除此捷徑檔案。如需詳細資訊，請參閱setWallpaper 任務未啟用，但桌布會在重新啟動時重設。

階段：您可以在 PreReady 和 UserData 階段設定桌布。

頻率：always

桌布組態

您可以使用下列設定來配置桌布。

輸入

您提供的輸入參數，以及可以設定桌布的屬性：

屬性 (字串清單)

(選用) 您可以將以下一個或多個屬性新增至桌布：

- architecture
- availabilityZone
- hostName

- instanceId
- instanceSize
- memory
- network
- privateIpAddress
- publicIpAddress

instanceTags

(選用) 您可以使用下列其中一個選項來進行此設定。

- AllTags(字串) — 將所有實體標籤新增至桌布。

```
instanceTags: AllTags
```

- instanceTags(字串清單) — 指定要新增至桌布的執行個體標籤名稱清單。例如：

```
instanceTags:
  - Tag 1
  - Tag 2
```

路徑 (字串)

(必要) 要用於桌布圖片的本機 .jpg 格式圖片檔案的檔案名稱路徑。

範例

下列範例顯示設定桌布背景圖片檔案路徑的桌布組態輸入，以及名為 Tag 1 和 Tag 2 的執行個體標籤，以及包含執行個體的主機名稱、執行個體 ID 及私有和公有 IP 地址的屬性。

```
task: setWallpaper
inputs:
  path: C:\ProgramData\Amazon\EC2Launch\wallpaper\Ec2Wallpaper.jpg
  attributes:
  - hostName
  - instanceId
  - privateIpAddress
  - publicIpAddress
instanceTags:
  - Tag 1
  - Tag 2
```

> **Note**
>
> 您必須在中繼資料中啟用標籤，才能在桌布上顯示標籤。如需有關執行個體標籤和中繼資料的詳細資訊，請參閱 使用執行個體中繼資料檢視EC2執行個體的。

startSsm

在 Sysprep 之後啟動「Systems Manager」(SSM) 服務。

Frequency – 始終

AllowedStages — `[PostReady, UserData]`

Inputs — 無

範例

```
task: startSsm
```

sysprep

重設服務狀態、更新`unattend.xml`RDP、停用和執行 Sysprep。只有在所有其他任務完成後才會執行此任務。

Frequency — 一次

AllowedStages — `[UserData]`

Inputs —

`clean`：(布林值）在執行 Sysprep 之前清除執行個體日誌

`shutdown`：(布林值）在執行 Sysprep 後關閉執行個體

範例

```
task: sysprep
inputs:
clean: true
shutdown: true
```

writeFile

將檔案寫入目的地。

Frequency — 請參閱 Inputs

AllowedStages — `[PostReady, UserData]`

Inputs —

`frequency`：(字符) 其中一個 `once` 或 `always`

`destination`：(字符) 要寫入內容的路徑

`content`：(字符) 要寫入文本的目的地

範例

```
task: writeFile
inputs:
  - frequency: once
  destination: C:\Users\Administrator\Desktop\booted.txt
  content: Windows Has Booted
```

疑難排解 EC2Launch v2 代理程式的問題

本節顯示 EC2Launch v2 的常見疑難排解案例、檢視 Windows 事件記錄檔的相關資訊，以及主控台記錄輸出和訊息。

故障診斷主題

- 常見的疑難排解案例：
- Windows 事件日誌
- EC2Launchv2 控制台日誌輸出

常見的疑難排解案例：

本節說明常見的疑難排解案例和解決步驟。

案例

- 服務無法設定桌布

- 服務無法執行使用者資料
- 服務只執行一次任務
- 服務無法執行任務
- 服務會多次執行使用者資料
- 從 EC2Launch v1 的計劃任務遷移到 EC2Launch v2 後無法運行
- 服務初始化不是EBS空的卷
- setWallpaper 任務未啟用，但桌布會在重新啟動時重設
- 服務卡在執行狀態
- 無效agent-config.yml防止開啟 EC2Launch v2 設定對話方塊
- task:executeScript should be unique and only invoked once

服務無法設定桌布

解析度

1. 檢查 `%AppData%\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\setwallpaper.lnk` 是否存在。
2. 檢查 `%ProgramData%\Amazon\EC2Launch\log\agent.log` 是否發生任何錯誤。

服務無法執行使用者資料

可能原因：服務可能在執行使用者資料之前失敗。

解析度

1. 檢查 `%ProgramData%\Amazon\EC2Launch\state\previous-state.json`。
2. 查看 `boot`、`network`、`preReady` 和 `postReadyLocalData` 是否已全部標記為成功。
3. 如果其中一個階段失敗，請檢查 `%ProgramData%\Amazon\EC2Launch\log\agent.log` 是否有特定的錯誤。

服務只執行一次任務

解析度

1. 檢查任務的頻率。

2. 如果服務已經在 Sysprep 之後執行，且任務頻率設定為 once，則任務將不會再次執行。
3. always如果您希望任務在每次 EC2Launch v2 運行時運行任務，請將任務的頻率設置為。

服務無法執行任務

解析度

1. 檢查 %ProgramData%\Amazon\EC2Launch\log\agent.log 中的最新項目。
2. 如果沒有發生錯誤，請嘗試從 "%ProgramFiles%\Amazon\EC2Launch\EC2Launch.exe" run 手動執行服務，以查看任務是否成功。

服務會多次執行使用者資料

解析度

EC2Launchv1 和 EC2Launch v2 之間的用戶數據處理方式不同。EC2Launch當設定為時，v1 會在執行個體上以排程工作的形式執行使persist用者資料true。如果將 persist 設定為 false，即使任務由於重新啟動而退出或在執行時中斷，也不會進行排程。

EC2Launchv2 將用戶數據作為代理任務運行並跟踪其運行狀態。如果使用者資料導致電腦重新啟動，或是使用者資料在執行時中斷，執行狀態將一直保持為 pending，且使用者資料將會在下次執行個體啟動時再次執行。如果您想防止使用者資料指令碼多次執行，請使指令碼等冪。

下列範例等冪指令碼會設定電腦名稱並加入網域。

```
<powershell>
  $name = $env:computername
  if ($name -ne $desiredName) {
 Rename-Computer -NewName $desiredName
  }
  $domain = Get-ADDomain
  if ($domain -ne $desiredDomain)
  {
 Add-Computer -DomainName $desiredDomain
  }
  $telnet = Get-WindowsFeature -Name Telnet-Client
  if (-not $telnet.Installed)
  {
 Install-WindowsFeature -Name "Telnet-Client"
  }
```

```
</powershell>
<persist>false</persist>
```

從 EC2Launch v1 的計劃任務遷移到 EC2Launch v2 後無法運行

解析度

移轉工具不會偵測任何連結至 EC2Launch v1 指令碼的排程工作，因此不會在 EC2Launch v2 中自動設定這些工作。若要設定這些工作，請編輯`agent-config.yml`檔案，或使用 EC2Launchv2 設定對話方塊。例如，如果執行個體有執行的排程工作`InitializeDisks.ps1`，則在執行移轉工具之後，您必須在 EC2Launch v2 設定對話方塊中指定要初始化的磁碟區。請參閱 使用 EC2Launch v2 設定對話方塊變更設定 程序的步驟 6。

服務初始化不是EBS空的卷

解析度

在初始化卷之前，EC2Launchv2 會嘗試檢測它是否為空。如果磁碟區不為空，則會略過初始化程序。偵測為非空白的任何磁碟區都不會初始化。如果磁碟區的前 4 KiB 為空，或磁碟區沒有 Windows 可辨識的磁碟機配置，磁碟區會被視為空。在 Linux 系統上初始化並格式化的磁碟區沒有 Windows 可辨識的磁碟機配置，例如或。MBR GPT因此，該磁碟區將視為空磁碟區且已初始化。如果要保留此數據，請不要依賴 EC2Launch v2 空驅動器檢測。請改為在 EC2Launchv2 設定對話方塊 (請參閱步驟 6) 或中指定您要初始化的磁碟區`agent-config.yml`。

`setWallpaper` 任務未啟用，但桌布會在重新啟動時重設

`setWallpaper` 任務會在 `Default User` 以外的每個現有使用者啟動資料夾中，建立 `setwallpaper.lnk` 捷徑檔案。使用者在執行個體啟動後第一次登入時，此捷徑檔案會執行。這會使用顯示執行個體屬性的自訂桌布，設定執行個體。移除 `setWallpaper` 任務不會刪除此捷徑檔案。您必須手動刪除此檔案，或使用指令碼將其刪除。

捷徑路徑為：

```
$env:SystemDrive/Users/<user>/AppData/Roaming/Microsoft/Windows/Start Menu/
Programs/Startup/setwallpaper.lnk
```

解析度

手動刪除此檔案，或使用指令碼將其刪除。

刪除捷徑檔案的範例 PowerShell 指令碼

```
foreach ($userDir in (Get-ChildItem "C:\Users" -Force -Directory).FullName)
{
 $startupPath = Join-Path $userDir -ChildPath "AppData\Roaming\Microsoft\Windows\Start
 Menu\Programs\Startup"
 if (Test-Path $startupPath)
 {
  $wallpaperSetupPath = Join-Path $startupPath -ChildPath "setwallpaper.lnk"
  if (Test-Path $wallpaperSetupPath)
  {
   Remove-Item $wallpaperSetupPath -Force -Confirm:$false
  }
 }
}
```

服務卡在執行狀態

Description

EC2Launchv2 被阻止，其日誌消息（`agent.log`）類似於以下內容：

```
2022-02-24 08:08:58 Info:
 ********************************************************************************
2022-02-24 08:08:58 Info: EC2Launch Service starting
2022-02-24 08:08:58 Info: Windows event custom log exists: Amazon EC2Launch
2022-02-24 08:08:58 Info: ACPI SPCR table not supported. Bailing Out
2022-02-24 08:08:58 Info: Serial port is in use. Waiting for Serial Port...
2022-02-24 08:09:00 Info: ACPI SPCR table not supported. Use default console port.
2022-02-24 08:09:02 Info: ACPI SPCR table not supported. Use default console port.
2022-02-24 08:09:04 Info: ACPI SPCR table not supported. Use default console port.
2022-02-24 08:09:06 Info: ACPI SPCR table not supported. Use default console port.
```

可能的原因

SAC已啟用並使用序列埠。如需詳細資訊，請參閱用SAC來排解 Windows 執行個體疑難排解。

解析度

請嘗試下列步驟來解決此問題：

- 停用正在使用序列連接埠的服務。
- 如果希望服務繼續使用序列連接埠，請編寫自訂指令碼來執行啟動代理程式任務，並將其做為排程任務呼叫。

無效**agent-config.yml**防止開啟 EC2Launch v2 設定對話方塊

描述

EC2Launchv2 設定會在開啟對話方塊之前嘗試剖析`agent-config.yml`檔案。如果YAML組態檔不遵循支援的結構描述，對話方塊會顯示下列錯誤：

```
Unable to parse configuration file agent-config.yml. Review configuration
file. Exiting application.
```

解析度

1. 驗證組態檔案是否遵循支援的結構描述。
2. 如果要從頭開始，請將預設組態檔案複製到 `agent-config.yml`。您可以使用「任務組態」區段中提供的範例 `agent-config.yml`。
3. 您也可以透過刪除 `agent-config.yml` 重新開始。EC2Launchv2 設置生成一個空的配置文件。

task:executeScript should be unique and only invoked once

描述

不能在同一階段重複任務。

解析度

某些工作必須以陣列的形式輸入，例如executeScript和executeProgram。如需如何將指令碼撰寫為陣列的範例，請參閱executeScript。

Windows 事件日誌

EC2Launchv2 會針對重要事件 (例如服務啟動、Windows 就緒以及工作成功和失敗) 發佈 Windows 事件記錄檔。事件 ID 會唯一識別特定事件。每個事件都包含階段、任務和層級資訊，以及說明。您可以使用事件 ID 來設定特定事件的觸發程序。

事件IDs提供有關事件的信息，並唯一標識一些事件。事件 ID 的最低有效位元顯示事件的嚴重性。

事件	最低有效位元
Success	. . .0
Informational	. . .1

事件	最低有效位元
Warning	. . .2
Error	. . .3

服務啟動或停止時產生的服務相關事件包含單一數字事件識別符。

事件	單一數字識別符
Success	0
Informational	1
Warning	2
Error	3

事件的 EC2LaunchService.exe 事件訊息以 Service: 為開頭。事件的 EC2Launch.exe 事件訊息不是以 Service: 為開頭。

四位數的事件IDs包括有關事件的階段、工作和嚴重性的資訊。

主題

- 事件 ID 格式
- 事件 ID 範例
- Windows 事件日誌結構描述

事件 ID 格式

下表顯示 EC2Launch v2 事件識別碼的格式。

3	2 1	0
S	T	L

3	2 1	0

表格中的字母和數字代表下列事件類型和定義。

Event type	定義
S (階段)	0 - 服務層級訊息 1 - 開機 2 - 網路 3- PreReady 5 - Windows 已準備就緒 6- PostReady 7 - 使用者資料
T (任務)	由對應的兩個值所代表的任務對每個階段都不同。若要檢視完整的事件清單，請參閱 Windows 事件日誌結構描述。
L (活動等級)	0 - 成功 1 - 資訊 2 - 警告 3 - 錯誤

事件 ID 範例

以下是示例事件IDs。

- `5000` - Windows 已做好使用準備

- 3010-在 PreReady 階段激活窗口任務成功
- 6013-在 PostReady 本機資料階段設定桌布工作發生錯誤

Windows 事件日誌結構描述

MessageId/事件識別碼	事件訊息
. . .0	Success
. . .1	Informational
. . .2	Warning
. . .3	Error
x	EC2Launch service-level logs
0	EC2Launch service exited successfully
1	EC2Launch service informational logs
2	EC2Launch service warning logs
3	EC2Launch service error logs
10	Replace state.json with previous-state.json
100	Serial Port
200	Sysprep
300	PrimaryNic
400	Metadata
x000	Stage (1 digit), Task (2 digits), Status (1 digit)

MessageId/事件識別碼	事件訊息
1000	Boot
1010	Boot - extend_root_partition
2000	Network
2010	Network - add_routes
3000	PreReady
3010	PreReady - activate_windows
3020	PreReady - install_egpu_manager
3030	PreReady - set_monitor_on
3040	PreReady - set_hibernation
3050	PreReady - set_admin_account
3060	PreReady - set_dns_suffix
3070	PreReady - set_wallpaper
3080	PreReady - set_update_schedule
3090	PreReady - output_log
3100	PreReady - enable_open_ssh
5000	Windows is Ready to use
6000	PostReadyLocalData
7000	PostReadyUserData
6010/7010	PostReadyLocal/UserData - set_wallpaper

MessageId/事件識別碼	事件訊息
6020/7020	PostReadyLocal/UserData - set_update_schedule
6030/7030	PostReadyLocal/UserData - set_hostname
6040/7040	PostReadyLocal/UserData - execute_program
6050/7050	PostReadyLocal/UserData - execute_script
6060/7060	PostReadyLocal/UserData - manage_package
6070/7070	PostReadyLocal/UserData - initialize_volume
6080/7080	PostReadyLocal/UserData - write_file
6090/7090	PostReadyLocal/UserData - start_ssm
7100	PostReadyUserData - enable_open_ssh
6110/7110	PostReadyLocal/UserData - enable_jumbo_frames

EC2Launchv2 控制台日誌輸出

本節包含 EC2Launch v2 的範例主控台記錄輸出，並列出所有 EC2Launch v2 主控台記錄錯誤訊息，以協助您疑難排解問題。如需執行個體主控台輸出及其存取方式的詳細資訊，請參閱the section called “執行個體主控台輸出”。

輸出

- EC2Launchv2 控制台日誌輸出
- EC2Launchv2 控制台日誌消息

EC2Launchv2 控制台日誌輸出

以下是 EC2Launch v2 的範例主控台記錄輸出。

```
2023/11/30 20:18:53Z: Windows sysprep configuration complete.
2023/11/30 20:18:57Z: Message: Waiting for access to metadata...
2023/11/30 20:18:57Z: Message: Meta-data is now available.
2023/11/30 20:18:57Z: AMI Origin Version: 2023.11.15
2023/11/30 20:18:57Z: AMI Origin Name: Windows_Server-2022-English-Full-Base
2023/11/30 20:18:58Z: OS: Microsoft Windows NT 10.0.20348
2023/11/30 20:18:58Z: OsVersion: 10.0
2023/11/30 20:18:58Z: OsProductName: Windows Server 2022 Datacenter
2023/11/30 20:18:58Z: OsBuildLabEx: 20348.1.amd64fre.fe_release.210507-1500
2023/11/30 20:18:58Z: OsCurrentBuild: 20348
2023/11/30 20:18:58Z: OsReleaseId: 2009
2023/11/30 20:18:58Z: Language: en-US
2023/11/30 20:18:58Z: TimeZone: UTC
2023/11/30 20:18:58Z: Offset: UTC +0000
2023/11/30 20:18:58Z: Launch: EC2 Launch v2.0.1643
2023/11/30 20:18:58Z: AMI-ID: ami-1234567890abcdef1
2023/11/30 20:18:58Z: Instance-ID: i-1234567890abcdef0
2023/11/30 20:18:58Z: Instance Type: c5.large
2023/11/30 20:19:00Z: Driver: AWS NVMe Driver v1.5.0.33
2023/11/30 20:19:00Z: SubComponent: AWS NVMe Driver v1.5.0.33;
 EnableSCSIPersistentReservations: 0
2023/11/30 20:19:00Z: Driver: AWS PV Driver Package v8.4.3
2023/11/30 20:19:01Z: Driver: Amazon Elastic Network Adapter v2.6.0.0
2023/11/30 20:19:01Z: RDPCERTIFICATE-SUBJECTNAME: EC2AMAZ-SO1T009
2023/11/30 20:19:01Z: RDPCERTIFICATE-THUMBPRINT:
 1234567890ABCDEF1234567890ABCDEF1234567890
2023/11/30 20:19:09Z: SSM: Amazon SSM Agent v3.2.1705.0
2023/11/30 20:19:13Z: Username: Administrator
2023/11/30 20:19:13Z: Password: <Password>
1234567890abcdef1EXAMPLEPASSWORD
</Password>
2023/11/30 20:19:14Z: User data format: no_user_data
2023/11/30 20:19:14Z: EC2LaunchTelemetry: IsTelemetryEnabled=true
2023/11/30 20:19:14Z: EC2LaunchTelemetry: AgentOsArch=windows_amd64
2023/11/30 20:19:14Z: EC2LaunchTelemetry: IsAgentScheduledPerBoot=true
```

```
2023/11/30 20:19:14Z: EC2LaunchTelemetry: AgentCommandErrorCode=0
2023/11/30 20:19:14Z: Message: Windows is Ready to use
```

EC2Launchv2 控制台日誌消息

以下是所有 EC2Launch v2 主控台記錄訊息的清單。

```
Message: Error EC2Launch service is stopping. {error message}
 Error setting up EC2Launch agent folders
 See instance logs for detail
 Error stopping service
 Error initializing service
Message: Windows sysprep configuration complete
Message: Invalid administrator username: {invalid username}
Message: Invalid administrator password
Username: {username}
Password: <Password>{encrypted password}</Password>
AMI Origin Version: {amiVersion}
AMI Origin Name: {amiName}
Microsoft Windows NT {currentVersion}.{currentBuildNumber}
OsVersion: {currentVersion}
OsProductName: {productName}
OsBuildLabEx: {buildLabEx}
OsCurrentBuild: {currentBuild}
OsReleaseId: {releaseId}
Language: {language}
TimeZone: {timeZone}
Offset: UTC {offset}
Launch agent: EC2Launch {BuildVersion}
AMI-ID: {amiId}
Instance-ID: {instanceId}
Instance Type: {instanceType}
RDPCERTIFICATE-SUBJECTNAME: {certificate subject name}
RDPCERTIFICATE-THUMBPRINT: {thumbprint hash}
SqlServerBilling: {sql billing}
SqlServerInstall: {sql patch leve, edition type}
Driver: AWS NVMe Driver {version}
Driver: Inbox NVMe Driver {version}
Driver: AWS PV Driver Package {version}
Microsoft-Hyper-V is installed.
Unable to get service status for vmms
Microsoft-Hyper-V is {status}
SSM: Amazon SSM Agent {version}
AWS VSS Version: {version}
```

```
Message: Windows sysprep configuration complete
Message: Windows is being configured. SysprepState is {state}
Windows is still being configured. SysprepState is {state}
Message: Windows is Ready to use
Message: Waiting for meta-data accessibility...
Message: Meta-data is now available.
Message: Still waiting for meta-data accessibility...
Message: Failed to find primary network interface...retrying...
User data format: {format}
```

EC2Launchv2 版本歷史記錄

版本歷程記錄

- EC2Launchv2 版本歷史記錄
- EC2Launchv2 遷移工具版本歷史記錄

EC2Launchv2 版本歷史記錄

下表說明 EC2Launch v2 的發行版本。

版本	詳細資訊	發行日期
2.0.1981	• 更新了非管理員使用者的`EC2Launch.exe` CLI命令錯誤訊息。	2024年7月6日
2.0.1948	• 添加了遙測以監視管理員密碼選項的使用情況。 • 已修改的EC2Launch權限。	2024年7月1日
2.0.1924	• 更新了設EC2Launch置用戶界面。 • 更新了桌布CLI指令。 • 更新了安EC2Launch裝程序。	2024年6月10日
2.0.1914	• 新增具有未指定閘道位址的路由 (`0.0.0.0`適用`::`於IPv4或用於IPv6)。	2024年6月5日

版本	詳細資訊	發行日期
	• 始終同時添加IPv4和IPv6路線。 • 修正未指定`Administrator` 使用者名稱時新增至`agent-config.yml` 檔案的問題。 • 已修改 EC2Launch v2 權限。	
2.0.1881	• 在任務中添加了一個加密密碼選`setAdminAccount` 項。 • 添加了在代理程序配置 .yml 中加密靜態密碼的CLI命令。 • 修正XML使用者資料在以系統管理員權限執行時不會新增 PowerShell 引數的問題。如需詳細資訊，請參閱Amazon 如何EC2處理 Windows 實例的用戶數據。 • 調整`executeScript` 工作和使用者資料指令碼使用`LocalSystem` 權限執行時的 PowerShell 引數。當引數為空時，代理程式會使用下列預設值：`-ExecutionPolicy Unrestricted` • 禁止將重複的驅動程序版本打印到控制台日誌。	2024年5月8日
2.0.1815	• 在 sysprep 之前調整了關鍵設置問題的錯誤處理失敗。 • 修正桌布和主機名稱工作在指派給主要網路介面的多個 IP 位址的執行個體上，可能會使用不正確 IP 位址的問題。 • 桌布和主機名稱工作已變更為IMDS先取得私有 IP，然後在停用時失敗回復。WMI IMDS • 修正`sc1`磁碟區因暫時性錯誤而無法初始化的`initializeVolume` 工作問題。	2024年3月6日

版本	詳細資訊	發行日期
2.0.1739	• 修正以 Windows 管理員使用者身分執行的`executeScript` 工作無法擷取結束代碼的問題。	2024年1月17日
2.0.1702	• 將 `Telemetry.log` 許可限制為 `read-execute` ，僅適用於標準使用者。 • 將 EC2Launch Windows 服務設定為在啟動失敗時重新啟動。 • 透過記錄 `route.exe` `stderr` 輸出，使 `add-routes` 故障可操作。 • 修正當路由指標超出 [1, 9999] 範圍時發生的問題。 • 為數種新的執行個體類型新增了桌布支援。 • 修正以 Windows 管理員使用者身分執行並將輸出傳送至 `stderr` 的使用者資料指令碼所造成的問題。	2024 年 1 月 4 日

版本	詳細資訊	發行日期
2.0.1643	• 將 `ebsnvme-id.exe` 工具更新至 1.1.0.7 版。 • 修正以 'metal-*' 開頭的金屬執行個體類型 (例如 metal-48x 1RSS) 的接收端調整 () 和接收佇列深度設定的問題。 • 已移除回報封鎖代理程式之 XML userdata 命令的遙測事件。 • 已更新 `setDnsSuffix` 任務，以根據登錄項目限制網域名稱轉移：`HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Dnscache\Parameters\DomainNameDevolutionLevel` 。 • 添加了一個公共任務CLI，並添加了網絡路由。 • 注意 – 這是正式支援 Windows Server 2012 的最新版本。 • 注意 – 這是正式支援 32 位元作業系統的最新版本。	2023 年 10 月 4 日
2.0.1580	• 修改日誌檔案許可時，已變更啟動代理程式處理錯誤的方式。 • 已新增連線至序列埠資源的逾時時間。如果序列埠正在使用中，則逾時允許啟動代理程式繼續執行。	2023 年 9 月 5 日

版本	詳細資訊	發行日期

2.0.1521	• 已棄用 `EC2Launch.exe` 、reset 和 sysprep 命令的 `–block` 旗標。
	• `EC2Launch.exe` 已更新，可偵測並處理內嵌 `executeScript` 任務中使用的 reset 和 sysprep 命令。這些命令會導致代理程式在 `executeScript` 任務執行後停止執行。
	• 更新XML用戶數據腳本默認情況下內聯運行。
	• 啟用用XML戶數據腳本以使用新`detach`標籤分離運行。如需詳細資訊，請參閱使用者資料指令碼。
	• 對代理程式日誌進行了下列變更。
	  • 更新代理程式日誌訊息。
	  • 從代理程式日誌中移除 `executeScript` 內容和輸出。
	  • 從代理程式日誌中移除 `executeProgram` 引數和輸出。
	• 對主控台日誌進行了下列變更。
	  • 為主控台日誌新增 `EnableSCSIPersistentReservations` 值。
		2023 年 7 月 3 日

版本	詳細資訊	發行日期
2.0.1303	• 新增網絡路由時新增了額外的錯誤處理和日誌行。 • 允許`executeScript` 和 PreReady 階段中的`executeProgram` 任務。 • 已更新`executeProgram` 工作，以產生類似於 executeScript 工作輸出的輸出檔案。如需詳細資訊，請參閱executeProgram。 • 新增遙測功能，以監控使用XML者資料中封鎖代理程式命令的使用	2023 年 5 月 3 日
2.0.1245	• 以純文字形式記錄當機呼叫堆疊，改善當機的可見性。 • 將 EventLog 服務作為啟動依賴項添加，以修復 Amazon EC2Launch 服務啟動速度比服務更快時崩潰的 EventLog 問題。 • 讓XML使用者資料在代理程式設定檔 (如 EC2Launch v1 和 EC2Config) PostReady 階段之前執行。 • 已新增YAML使用者資料 1.1 版，讓使用者資料在代理程式設定檔的 PostReady 階段前執行 (YAML使用者資料版本 1.0 在代理程式組態檔的 PostReady 階段之後執行)。	2023 年 3 月 8 日
2.0.1173	• 新增一個可選功能以在桌布上顯示執行個體標籤。如需詳細資訊，請參閱setWallpaper 。 • 未正確設定 Elastic Graphics 的安全群組時，會新增錯誤處理。 • 修復未啟用執行個體中繼資料服務時的逾時。	2023 年 2 月 6 日

版本	詳細資訊	發行日期
2.0.1121	• 修正未指派公開IPv4地址時，會在桌布上列印 404 錯誤的問題。 • 修復當裝置的磁碟機代號設為 D 時磁碟區檔案系統格式為 RAW 而非 NTFS 的問題。 • 修正NVMeSSD磁碟區錯誤識別為磁EBS碟區的問題。 • 在禁用時修復了激活 Windows 時IMDS的錯誤。	2023 年 1 月 4 日
2.0.1082	• 修正停用時`setWallpaper IMDS:privateIpAddress` 欄位為空白的問題。 • 修正停用時，將主機名稱設定為私人IPv4位址IMDS的問題。 • 修正在 Windows Server 2012 上初始化磁碟區的問題。 • 修正設定巨型訊框的問題。 • 修正執行個體啟動時未指定SSH索引鍵的錯誤。 • 修復 Windows 沒有 'ReleaseId' 註冊表項時 Windows 服務器 2012 上的錯誤。	2022 年 12 月 7 日
2.0.1011	• 修復了 P nPDevice ID 為空時查找網絡適配器的邏輯。	2022 年 11 月 11 日
2.0.1009	• 使用PCI區段資訊來選取主控台連接埠。	2022 年 11 月 8 日

版本	詳細資訊	發行日期
2.0.982	• 添加重試邏輯以獲取RDP信息。 • 修正 `d2.8xlarge` 執行個體的磁碟區初始化期間的錯誤。 • 修正在重新啟動後可能會選取錯誤網路轉接器的問題。 • 刪除不可用時ACPISPCR的錯誤警報錯誤消息。	2022 年 10 月 31 日
2.0.863	• 更新IMDS等待邏輯只發出IMDSv2請求。 • 新增邏輯，將磁碟機代號指派給已初始化但未掛載的磁碟區。 • 當 key pair type 不受支援時，印出更詳細的錯誤訊息。 • 修復 3010 重新啟動的程式碼錯誤。 • 新增檢查無效 base64 編碼的使用者資料功能。	2022 年 7 月 6 日
2.0.698	• 修復了執行指令碼時日誌輸出中的錯字。	2022 年 1 月 30 日
2.0.674	• 遙測會上傳已啟用/已停用的隱私權控制。 • 修正 `index out of bounds` 錯誤。 • 在 `sysprep` 期間移除桌布捷徑。	2021 年 11 月 15 日
2.0.651	• 添加邏輯以在 EC2Launch v2 安裝期間卸載舊版代理程序。 • `list-volume` CLI修復了根卷未列為卷 0 時的問題。	2021 年 10 月 7 日

版本	詳細資訊	發行日期
2.0.592	• 修正錯誤以正確報告階段狀態。 • 當日誌檔關閉時，移除錯誤的警示錯誤訊息。 • 新增遙測。	2021 年 8 月 31 日
2.0.548	• 為十六進位 IP 主機名稱新增開頭的零。 • 修正 `enableOpenSsh` 任務的檔案許可。 • 修正 sysprep 命令當機。	2021 年 8 月 4 日
2.0.470	• 修復了網絡階段中的錯誤，DHCP以等待將 IP 分配給實例。 • 修正 `SearchList` 登錄機碼不存在時的 `setDnsSuffix` 錯誤。 • 修復了中的DNS權力放積邏輯中的錯誤。`setDnsSuffix` • 在中繼重新開機後新增網路路由。 • 允許 `initializeVolume` 重新用字母標示現有磁碟區。 • 從版本子命令中移除額外的資訊。	2021 年 7 月 20 日
2.0.285	• 在分離的程序中新增了執行使用者指令碼的選項。 • 舊版 userdata（XMLuserdata）現在在分離的進程中運行，這與先前的啟動代理程序類似。 • 在`sysprep`和`reset`命令中加入CLI旗標，讓它們可以封鎖，直到服務停止為止。 • 限制組態資料夾許可。	2021 年 3 月 8 日

版本	詳細資訊	發行日期
2.0.207	• 將選擇性 hostName 欄位新增至 setHostName 工作。 • 修正重新開機錯誤。重新啟動任務 executeScript 並將 executeProgram 標記為執行中。 • 在狀態命令中新增更多傳回碼。 • 新增引導服務，以修正在執行 t2.nano 執行個體類型時的啟動問題。 • 修正全新安裝模式，以移除安裝程式未追蹤的檔案。	2021 年 2 月 2 日
2.0.160	• 修正 validate 命令，以偵測無效的階段名稱。 • 在 addroutes 任務中新增 w32tm resync 命令。 • 修復了更改DNS後綴搜索順序的問題。 • 新增檢查條件，以更詳細地報告無效的使用者資料。	2020 年 12 月 4 日
2.0.153	在中增加系統表示功能。 UserData	2020 年 11 月 3 日

版本	詳細資訊	發行日期
2.0.146	• 修復了非英語AMIs的 RootExtend 問題。 • 授與使用者群組寫入日誌檔案的權限。 • 為卷創建 MS 保留分GPT區。 • 在 Amazon EC2Launch 設置中添加列表卷命令和卷下拉菜單。 • 添加用於以 YAML 或 JSON 格式打印代理配置 .yml 文件的 get-agent-config 命令。 • 如果沒有偵測到公開金鑰，則清除靜態密碼。	2020 年 10 月 6 日
2.0.124	• 新增在桌布上顯示作業系統版本的選項。 • 初始化加密EBS磁碟區。 • 加入沒VPCs有本機DNS名稱的路由。	2020 年 9 月 10 日
2.0.104	• 創建DNS後綴搜索列表，如果它不存在。 • 如果沒有要求，則可略過休眠。	2020 年 8 月 12 日
2.0.0	初始版本。	2020 年 6 月 30 日

EC2Launchv2 遷移工具版本歷史記錄

下表說明 EC2Launch v2 移轉工具的發行版本。

當 EC2Launch v2 代理程式的新版本發行時，您可以收到通知。如需詳細資訊，請參閱訂閱 EC2 Windows 啟動代理程式通知。

版本	詳細資訊	發行日期
1.0.413	• 使用最新版本的 EC2Launch v2 代理程式更新移轉工具：2.0.1981。	2024年8月9日
1.0.412	• 使用最新版本的 EC2Launch v2 代理程式更新移轉工具：2.0.1948。	2024年8月7日
1.0.396	• 使用最新版本的 EC2Launch v2 代理程式更新移轉工具：2.0.1924。	2024年6月11日
1.0.394	• 使用最新版本的 EC2Launch v2 代理程式更新移轉工具：2.0.1914。	2024年6月6日
1.0.384	• 使用最新版本的 EC2Launch v2 代理程式更新移轉工具：2.0.1881。	2024年5月8日
1.0.358	• 使用最新版本的 EC2Launch v2 代理程式更新移轉工具：2.0.1815。	2024年3月8日
1.0.345	• 使用最新版本的 EC2Launch v2 代理程式更新移轉工具：2.0.1739。	2024年1月18日
1.0.342	• 使用最新版本的 EC2Launch v2 代理程式更新移轉工具：2.0.1702。	2024年1月5日
1.0.331	• 使用最新版本的 EC2Launch v2 代理程式更新移轉工具：2.0.1643 • 修復執行 `.Install.ps1 -DryRun` 時發生的錯誤。 • 修正移轉來源`random`期間密碼組態未正確設定為的問題EC2 Config。	2023 年 11 月 3 日

版本	詳細資訊	發行日期
	• 修正移轉來源期間設定`setWallpaper` 為`False`時會發生的錯誤EC2Launch。	
1.0.303	使用最新版本的 EC2Launch v2 代理程式更新移轉工具：2.0.1580。	2023 年 9 月 14 日
1.0.286	使用最新版本的 EC2Launch v2 代理程式更新移轉工具：2.0.1521。	2023 年 7 月 14 日
1.0.272	使用最新版本的 EC2Launch v2 代理程式更新移轉工具：2.0.1303。	2023 年 5 月 3 日
1.0.262	使用最新版本的 EC2Launch v2 代理程式更新移轉工具：2.0.1245。	2023 年 3 月 9 日
1.0.241	將 EC2Launch v2 代理程式的版本號碼遞增至 2.0.1011。	2022 年 12 月 7 日
1.0.218	• 驗證從執行個體中繼資料中擷取的區域值。 • 修復了語言套件中的遷移失敗錯誤。 • 將 EC2Launch v2 代理程式的版本號碼遞增至 2.0.863。	2022 年 9 月 3 日
1.0.162	• 將邏輯移除舊版代理程式至 EC2Launch v2 MSI。 • 將 EC2Launch v2 代理程式的版本號碼遞增至 2.0.698。	2022 年 3 月 18 日
1.0.136	將 EC2Launch v2 代理程式的版本號碼遞增至 2.0.651。	2021 年 10 月 13 日
1.0.130	將 EC2Launch v2 代理程式的版本號碼遞增至 2.0.548。	2021 年 8 月 5 日
1.0.113	使IMDSv2用代替IMDSv1.	2021 年 6 月 4 日
1.0.101	將 EC2Launch v2 代理程式的版本號碼遞增至 2.0.285。	2021 年 3 月 12 日

版本	詳細資訊	發行日期
1.0.86	將 EC2Launch v2 代理程式的版本號碼遞增至 2.0.207。	2021 年 2 月 3 日
1.0.76	將 EC2Launch v2 代理程式的版本號碼遞增至 2.0.160。	2020 年 12 月 4 日
1.0.69	將 EC2Launch v2 代理程式的版本號碼遞增至 2.0.153。	2020 年 11 月 5 日
1.0.65	將 EC2Launch v2 代理程式的版本號碼遞增至 2.0.146。	2020 年 10 月 9 日
1.0.60	將 EC2Launch v2 代理程式的版本號碼遞增至 2.0.124。	2020 年 9 月 10 日
1.0.54	• 如果未安裝代理程式，則安裝 EC2Launch v2。 • 將 EC2Launch v2 代理程式的版本號碼遞增至 2.0.104。 • 將SSM代理程式分離。	2020 年 8 月 12 日
1.0.50	移除 NuGet 相依性。	2020 年 8 月 10 日
1.0.0	初始版本。	2020 年 6 月 30 日

在 EC2 Windows 執行個體啟動期間，使用 EC2Launch v1 代理程式執行工作

EC2Launch是一組視窗 PowerShell 腳本，取代了視窗EC2Config服務器 2016 年和 2019 上的服務 AMIs。其中許多AMIs仍然可用。EC2Launchv2 是所有受支援 Windows 版本的最新啟動代理程式，它會取代EC2Config和EC2Launch。如需詳細資訊，請參閱使用 EC2Launch v2 代理程式在 EC2 Windows 執行個體啟動期間執行工作。

> **Note**
>
> 若要EC2Launch與配合使用IMDSv2，此版本必須是 1.3.2002730 或更新版本。

您可以使用下列 Windows PowerShell 命令來驗證的已安裝版本EC2Launch。

```
Test-ModuleManifest -Path "C:\ProgramData\Amazon\EC2-Windows\Launch\Module
\Ec2Launch.psd1" | Select Version
```

EC2Launch任務

EC2Launch在初始執行個體啟動期間，預設會執行下列工作：

- 設定呈現執行個體資訊的新桌面。
- 將電腦名稱設定為執行個體的私人IPv4位址。
- 將執行個體資訊傳送到 Amazon EC2 主控台。
- 將RDP憑證指紋傳送至EC2主控台。
- 為管理員帳戶設定隨機密碼。
- 添加DNS後綴。
- 動態擴充作業系統分割區，以包含任何未分割的空間。
- 執行使用者資料 (若指定)。如需指定使用者資料的詳細資訊，請參閱使用使用者資料輸入啟動執行個EC2體時執行指令。
- 設置持久靜態路由以訪問元數據服務和服務 AWS KMS 器。

> **Important**
>
> 如果從此執行個體建立自訂AMI，則會擷取這些路由做為作業系統組態的一部分，而從中啟動的任何新執行個體都AMI將保留相同的路由，而不論子網路位置為何。若要更新路由，請參閱在啟動自訂時，更新伺服器 2016 及更新版本的中繼資料/KMS路由 AMI。

下列工作有助於維護與EC2Config服務的回溯相容性。您也可以設定EC2Launch為在啟動期間執行下列工作：

- 初始化輔助EBS磁碟區。
- 將 Windows 事件記錄檔傳送至主EC2控台記錄檔。

- 將 Windows 已準備就緒可以使用的消息發送到EC2控制台。

如需 Windows Server 2019 的詳細資訊，請參閱 Microsoft.com 上的比較 Windows Server 版本功能。

EC2Launch 目錄結構

EC2Launch默認情況下安裝在視窗服務器 2016 和更高版本AMIs的根目錄中`C:\ProgramData\Amazon\EC2-Windows\Launch`。

> **Note**
>
> 根據預設，Windows 會將檔案和資料夾隱藏在 `C:\ProgramData` 下。若要檢視EC2Launch目錄和檔案，您必須在 Windows 檔案總管中輸入路徑，或變更資料夾內容以顯示隱藏的檔案和資料夾。

`Launch` 目錄包含下列子目錄。

- `Scripts`— 包含構成的 PowerShell 指令碼EC2Launch。
- `Module`— 包含用於建置與 Amazon 相關指令碼的模組EC2。
- `Config` – 含有您可以自訂的指令碼組態檔案。
- `Sysprep` – 含有 Sysprep 資源。
- `Settings` – 含有 Sysprep 圖形使用者介面的應用程式。
- `Library`— 包含EC2啟動代理程式的共用程式庫。
- `Logs` – 含有指令碼產生的日誌檔案。

遙測

遙測是其他資訊，可協 AWS 助您進一步瞭解您的需求、診斷問題，並提供可改善 AWS 服務體驗的功能。

EC2Launch版本`1.3.2003498`和更新版本會收集遙測資料，例如使用量度和錯誤。這些資料是從執EC2Launch行所在的 Amazon EC2 執行個體收集而來。這包括所有AMIs擁有的視窗 AWS。

下列類型的遙測收集方式EC2Launch：

- 用量資訊 – 代理程式命令、安裝方法和排程的執行頻率。
- 錯誤和診斷資訊 – 代理程式安裝和執行錯誤代碼。

收集資料的範例：

```
2021/07/15 21:44:12Z: EC2LaunchTelemetry: IsAgentScheduledPerBoot=true
2021/07/15 21:44:12Z: EC2LaunchTelemetry: IsUserDataScheduledPerBoot=true
2021/07/15 21:44:12Z: EC2LaunchTelemetry: AgentCommandCode=1
2021/07/15 21:44:12Z: EC2LaunchTelemetry: AgentCommandErrorCode=5
2021/07/15 21:44:12Z: EC2LaunchTelemetry: AgentInstallCode=2
2021/07/15 21:44:12Z: EC2LaunchTelemetry: AgentInstallErrorCode=0
```

預設為啟用遙測。您可隨時停用遙測收集。如果啟用遙測，則EC2Launch會傳送遙測資料，而不需要其他客戶通知。

您選擇啟用或停用遙測收集。

您可以選擇加入或退出遙測收集。您選取加入或退出遙測收集，以確保我們遵守您的遙測選項。

遙測可見性

啟用遙測後，它會顯示在 Amazon EC2 主控台輸出中，如下所示：

```
2021/07/15 21:44:12Z: Telemetry: <Data>
```

停用執行個體的遙測

若要藉由設定系統環境變數來停用遙測，請以管理員身分執行下列命令：

```
setx /M EC2LAUNCH_TELEMETRY 0
```

若要在安裝期間停用遙測，請執行 `install.ps1`，如下所示：

```
. .\install.ps1 -EnableTelemetry:$false
```

更多的主題 EC2Launch

- 安裝 EC2Launch 的最新版本
- 在您的 Windows 執行個體上設定 EC2Launch v1 代理程式
- EC2Launch 版本歷史記錄

安裝 EC2Launch 的最新版本

請遵循下列程序，在執行個體EC2Launch上下載並安裝最新版本的。

若要下載並安裝最新版本的 EC2Launch

1. 如果您已經在執行個體EC2Launch上安裝並設定，請備份EC2Launch組態檔案。安裝程序不會保留此檔案的變更內容。根據預設，檔案位於 `C:\ProgramData\Amazon\EC2-Windows\Launch\Config` 目錄。
2. 將 EC2-Windows-Launch.zip 下載至執行個體上的目錄。
3. 下載 install.ps1 到您下載 `EC2-Windows-Launch.zip` 的同一個目錄。
4. 執行 `install.ps1`
5. 如果您已備份組EC2Launch態檔案，請將其複製到目`C:\ProgramData\Amazon\EC2-Windows\Launch\Config`錄中。

若要下載並安裝最新版本的EC2Launch使用 PowerShell

如果您已經在執行個體EC2Launch上安裝並設定，請備份EC2Launch組態檔案。安裝程序不會保留此檔案的變更內容。根據預設，檔案位於 `C:\ProgramData\Amazon\EC2-Windows\Launch\Config` 目錄。

若要安裝EC2Launch使用的最新版本 PowerShell，請從 PowerShell 視窗執行下列命令

```
mkdir $env:USERPROFILE\Desktop\EC2Launch
$Url = "https://s3.amazonaws.com/ec2-downloads-windows/EC2Launch/latest/EC2-Windows-Launch.zip"
$DownloadZipFile = "$env:USERPROFILE\Desktop\EC2Launch\" + $(Split-Path -Path $Url -Leaf)
Invoke-WebRequest -Uri $Url -OutFile $DownloadZipFile
$Url = "https://s3.amazonaws.com/ec2-downloads-windows/EC2Launch/latest/install.ps1"
$DownloadZipFile = "$env:USERPROFILE\Desktop\EC2Launch\" + $(Split-Path -Path $Url -Leaf)
Invoke-WebRequest -Uri $Url -OutFile $DownloadZipFile
& $env:USERPROFILE\Desktop\EC2Launch\install.ps1
```

> **Note**
>
> 如果您在下載檔案時收到錯誤訊息，而且您使用的是 Windows 伺服器 2016，則可能需要為您的 PowerShell 終端機啟用 TLS 1.2。您可以使用以下命令為當前 PowerShell 會話啟用 TLS 1.2，然後再試一次：
>
> ```
> [Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
> ```

檢查啟動代理程式，如下所示，以確認安裝。

```
Import-Module C:\ProgramData\Amazon\EC2-Windows\Launch\Module\Ec2Launch.psm1
Import-LocalizedData -BaseDirectory C:\ProgramData\Amazon\EC2-Windows\Launch\Module\ -
FileName 'Ec2Launch.psd1' -BindingVariable moduleManifest
$moduleManifest.Get_Item('ModuleVersion')
```

在您的 Windows 執行個體上設定 EC2Launch v1 代理程式

第一次初始化執行個體之後，您可以設定EC2Launch為再次執行並執行不同的啟動工作。

任務

- 設定初始化任務
- 排程EC2Launch在每次開機時執行
- 初始化磁碟機和映射磁碟機代號
- 將 Windows 事件記錄檔傳送到主EC2控台
- 在成功開機後傳送「Windows 已就緒」訊息

設定初始化任務

在 `LaunchConfig.json` 檔案中指定設定以啟用或停用下列初始化任務：

- 將電腦名稱設定為執行個體私人IPv4位址。
- 將監視器設定為永遠保持開啟狀態。
- 設定新的桌面。
- 加入DNS字尾清單。

> **Note**
>
> 這會新增下列網域的DNS尾碼查詢，並設定其他標準尾碼。如需啟動代理程式如何設定DNS尾碼的詳細資訊，請參閱。設定 EC2 Windows 啟動代理程式的DNS尾碼
>
> ```
> region.ec2-utilities.amazonaws.com
> ```

- 擴充開機磁碟區大小。
- 設定管理員密碼。

進行初始化設定

1. 在要設定的執行個體上，於文字編輯器中開啟下列檔案：`C:\ProgramData\Amazon\EC2-Windows\Launch\Config\LaunchConfig.json`。
2. 視需要更新下列設定並儲存變更。只有在 `adminPassword` 是 `adminPasswordtype` 時，才要在 `Specify` 中提供密碼。

```
{
 "setComputerName": false,
 "setMonitorAlwaysOn": true,
 "setWallpaper": true,
 "addDnsSuffixList": true,
 "extendBootVolumeSize": true,
 "handleUserData": true,
 "adminPasswordType": "Random | Specify | DoNothing",
 "adminPassword": "password that adheres to your security policy (optional)"
}
```

密碼類型的定義如下：

`Random`

EC2Launch生成密碼並使用用戶的密鑰對其進行加密。系統會在執行個體啟動後停用此設定，以便執行個體重新開機或停止再啟動時，能持續使用這個密碼。

`Specify`

EC2Launch使用您在中指定的密碼`adminPassword`。如果密碼不符合系統需求，請改為EC2Launch產生隨機密碼。系統會將該密碼以純文字存放於 `LaunchConfig.json`，並在 Sysprep 設定管理員密碼後予以刪除。EC2Launch使用用戶的密鑰加密密碼。

`DoNothing`

EC2Launch使用您在`unattend.xml`檔案中指定的密碼。如果您未在 `unattend.xml` 中指定密碼，會停用管理員帳戶。

3. 在 Windows 中 PowerShell，執行下列命令來排程指令碼以 Windows 排程工作的形式執行。該指令碼會在下一次開機期間執行一次，然後停用這些任務，以避免再次執行。

```
C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 -Schedule
```

排程EC2Launch在每次開機時執行

您可以排程EC2Launch在每次開機時執行，而非僅在初始開機時執行。

若要啟EC2Launch用在每次開機時執行：

1. 開啟視窗 PowerShell 並執行下列命令：

```
C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 -
SchedulePerBoot
```

2. 或使用下列命令執行可執行檔：

```
C:\ProgramData\Amazon\EC2-Windows\Launch\Settings\Ec2LaunchSettings.exe
```

   然後選取 `Run EC2Launch on every boot`。您可以指定EC2執行個體`Shutdown without Sysprep`或`Shutdown with Sysprep`.

> **Note**
>
> 當您啟用EC2Launch在每次開機時執行時，下次EC2Launch執行時會發生下列情況：
>
> - 如果`AdminPasswordType`仍設定為`Random`，EC2Launch將在下次開機時產生新密碼。該啟動後，`AdminPasswordType`會自動設置`DoNothing`為防EC2Launch止在後續啟動時生成新密碼。若要EC2Launch避免在第一次開機時產生新密碼，請在重新開機`DoNothing`之前手動設定`AdminPasswordType`為。
> - `HandleUserData` 會設回 `false`，除非使用者資料將 `persist` 設成 `true`。如需詳細資訊，請參閱the section called “使用者資料指令碼”。

初始化磁碟機和映射磁碟機代號

在`DriveLetterMappingConfig.json`檔案中指定設定，將磁碟機代號對應至EC2執行個體上的磁碟區。指令碼會初始化尚未初始化和分割的磁碟機。如需有關在 Windows 中取得磁碟區詳細資訊的更多詳細資訊，請參閱 Microsoft 文件中的 Get-Volume。

將磁碟機代號映射到磁碟區

1. 在文字編輯器中開啟 `C:\ProgramData\Amazon\EC2-Windows\Launch\Config\DriveLetterMappingConfig.json` 檔案。
2. 指定下列磁碟區設定並儲存您的變更：

```
{
 "driveLetterMapping": [
  {
   "volumeName": "sample volume",
   "driveLetter": "H"
  }
 ]
}
```

3. 開啟 Windows PowerShell 並使用下列命令來執行初始化磁碟的指令EC2Launch碼：

```
C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeDisks.ps1
```

若要在每次執行個體開機時初始化磁碟，請新增 `-Schedule` 標記，如下所示：

```
C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeDisks.ps1 -Schedule
```

將 Windows 事件記錄檔傳送到主EC2控台

在`EventLogConfig.json`檔案中指定設定，以將 Windows 事件記錄檔傳送至EC2主控台記錄檔。

進行設定以傳送 Windows 事件日誌

1. 在執行個體上，於文字編輯器中開啟 `C:\ProgramData\Amazon\EC2-Windows\Launch\Config\EventLogConfig.json` 檔案。
2. 進行下列日誌設定並儲存您的變更：

```
{
 "events": [
  {
   "logName": "System",
   "source": "An event source (optional)",
   "level": "Error | Warning | Information",
   "numEntries": 3
```

```
    }
  ]
}
```

3. 在 Windows 中 PowerShell，執行下列命令，讓系統將指令碼排定為在每次執行個體開機時以 Windows 排程工作的形式執行。

```
C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\SendEventLogs.ps1 -Schedule
```

記錄檔可能需要三分鐘或更長時間才會出現在EC2主控台記錄檔中。

在成功開機後傳送「Windows 已就緒」訊息

每次啟動後，EC2Config服務都會向EC2控制台發送「Windows 已準備就緒」消息。EC2Launch僅在初始啟動後發送此消息。為了與EC2Config服務的向後兼容性，您可EC2Launch以安排在每次啟動後發送此消息。在執行個體上，開啟 Windows PowerShell 並執行下列命令。系統會排程指令碼做為 Windows 排程任務執行。

```
C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\SendWindowsIsReady.ps1 -Schedule
```

EC2Launch 版本歷史記錄

視窗AMIs開始與視窗伺服器 2016 年包括一組稱為EC2Launch視窗電源外殼指令碼。EC2Launch在初始執行個體啟動期間執行工作。如需 AWS Windows 中所含EC2Launch版本的相關資訊AMIs，請參閱 AWS Windows AMI 版本歷程記錄。

若要下載並安裝最新版本的EC2Launch，請參閱安裝 EC2Launch 的最新版本。

您可以在發行新版EC2Launch代理程式時收到通知。如需詳細資訊，請參閱訂閱 EC2 Windows 啟動代理程式通知。

下表說明的已發行版本EC2Launch。請注意，版本格式在 1.3.610 版後已變更。

版本	詳細資訊	發行日期
1.3.2005008	• 如果找不`Set-Wallpaper` 到預設桌布影像，則已更新為純色背景。	二零二四年八月六日

版本	詳細資訊	發行日期
1.3.2004959	• 已更新安裝程式邏輯，以防止在 Windows 伺服器 2025 或更新版本上不受支援	二零二四年七月二日
1.3.2004891	• 修正未按預期設定的false問題。HandleUserData • 已將Encrypted 密碼選項新增至LaunchConfig.json 。 • 已變更預設Settings UI為加密使用者指定密碼的行為。 • 添SetAdminPasswordConfig.ps1 加了將Specify密碼選項轉換為代理程序配置文件中的Encrypted 密碼選項。	2024 年五月三十一日
1.3.2004617	• 修正了設置壁紙時的錯誤。	2024 年一月十五日
1.3.2004592	• 已更新 install.ps1 為 %ProgramData%\Amazon\EC2-Windows\Launch 設定的存取許可。 • 限制EC2Launch資料夾/檔案存取權限，只能針對標準使用者帳戶讀取執行。 • 如IMDS果執行個體未啟用，則代理程式已變更為停止等待執行個體中繼資料服務 (IMDS) 初始化。 • 在等待初始化時添加了五分鐘的超IMDS時時間。 • 將代理程式變更為在 Windows is Ready 訊息之前 (而非之後) 將遙測寫入執行個體主控台日誌。 • 為數種新的執行個體類型新增了桌布支援。 如需有關EC2Launch目錄存取權限和使用者帳戶權限的詳細資訊，請參閱the section called “EC2Launch 目錄結構”。	2024 年 1 月 2 日

版本	詳細資訊	發行日期
1.3.2004491	• 新增遙測以監控指定管理員密碼選項的使用情況。	2023 年 11 月 9 日
1.3.2004462	• 在每次寫入序列主控台後新增排清。	2023 年 10 月 18 日
1.3.2004438	• 根據登錄項目限制網域名稱轉移：`HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Dnscache\Parameters\DomainNameDevolutionLevel` 。 • 僅將 `UserdataExecution.log` 許可限制給 `Administrators` 。 • 當日誌初始化失敗時，在 Windows Event Log 中新增錯誤訊息。	2023 年 10 月 4 日
1.3.2004256	• 為主控台日誌新增 `EnableSCSIPersistentReservations` 值。 • 為 Get-ConsolePort 新增重試功能。	2023 年 7 月 7 日
1.3.2004052	• 修正執行個體啟動時未指定SSH索引鍵時發生的錯誤。 • 已更新，可在失敗時重試啟動 mazonSSMAgent Windows 服務。 • 如果 SysprepInstance. BeforeSysprep cmd 因非零結束代碼而失敗，則更新為失敗 .ps1。	2023 年 3 月 8 日
1.3.2003975	• 修正影響`SysprepInstance.ps1` 傳回 1 的封隔器AMI組建 `$LastErrorCode` 的問題。	2022 年 12 月 24 日

版本	詳細資訊	發行日期
1.3.2003961	• 已修正快速啟動執行個體上明確指定的管理員密碼會被隨機密碼覆寫的問題。 • 修正SSM代理程式無法在較小的執行個體類型上啟動的問題。 • 修正執行個體主控台記錄檔包含`RDPCERTIFICATE-THUMBPRINT: 0000000000000000000000000`而非有效RDP憑證指紋值的問題。	2022 年 12 月 6 日
1.3.2003923	• 修復了 P nPDevice ID 為空時查找網絡適配器的邏輯。	2022 年 11 月 9 日
1.3.2003919	• 已更新「取得-」ConsolePort 以使用PCI區段資訊。 • 已修正在重新啟動後可能會選取錯誤網路轉接器的問題。 • 修正了啟動 SSM-代理程式逾時邏輯。 • 修正了 Send-AdminCredentials 函數別名的向後兼容性。	2022 年 11 月 8 日
1.3.2003857	• 選取主要網路轉接器時，應優先考慮具有預設閘道的轉接器。 • 擴充了記憶體內密碼加密。	2022 年 10 月 3 日
1.3.2003824	• 已修正 `setComputerName` 期間的錯誤。 • 新增邏輯，以便在偵測到BYOL帳單代碼時略過 Windows 啟用。 • 新增記憶體內密碼加密。 • 已修正 `m6id.4xlarge` 磁碟區初始化期間的錯誤。	2022 年 8 月 30 日
1.3.2003691	• 更新IMDS等待邏輯以僅IMDSv2提出請求。 • 修正了影響和GPU安裝的錯誤。	2022 年 6 月 21 日

版本	詳細資訊	發行日期
1.3.2003639	• 新增了網路轉接器等待邏輯，以防止在初始化之前使用。 • 已修正輕微的問題。	2022 年 5 月 10 日
1.3.2003498	• 新增的遙測。 • 新增了設定 UI 的捷徑。 • 格式化 PowerShell 腳本。 • 修復了 BeforeSysprep .cmd 完成之前發生關機的問題。	2022 年 1 月 31 日
1.3.2003411	已變更密碼產生邏輯，以排除複雜性低的密碼。	2021 年 8 月 4 日
1.3.2003364	更新安裝-EgpuManager 與IMDSv2支持。	2021 年 6 月 7 日
1.3.2003312	• 在 `setMonitorAlwaysOn` 設定前後新增日誌行。 • 將 AWS 硝基飛地包版本添加到控制台日誌中。	2021 年 5 月 4 日
1.3.2003284	透過更新將使用者資料存放至 `LocalAppData` 的位置，來改善許可模型。	2021 年 3 月 23 日
1.3.2003236	• 已更新設定使用者密碼的方法於 `Set-AdminAccount` 和 `Randomize-LocalAdminPassword` 中。 • 修正 `InitializeDisks` ，以在將磁碟設定為可寫入之前檢查是否已設定為唯讀。	2021 年 2 月 11 日
1.3.2003210	`install.ps1` 的本地化修正程式。	2021 年 1 月 7 日
1.3.2003205	更新 `%ProgramData%AmazonEC2-WindowsLaunch ModuleScripts` 目錄權限的 `install.ps1` 安全性修正程式。	2020 年 12 月 28 日
1.3.2003189	新增路由後新增 `w32tm resync`。	2020 年 12 月 4 日

版本	詳細資訊	發行日期
1.3.2003155	已更新執行個體類型資訊。	2020 年 8 月 25 日
1.3.2003150	新增 `OsCurrentBuild` 和 `OsReleaseId` 至主控台輸出。	2020 年 4 月 22 日
1.3.2003040	修正了 1 IMDS 版後備邏輯。	2020 年 4 月 7 日
1.3.2002730	增加了對 IMDS V2 的支持。	2020 年 3 月 3 日
1.3.2002240	已修正輕微的問題。	2019 年 10 月 31 日
1.3.2001660	修正在第一次執行 Sysprep 後未使用密碼的使用者自動登入的問題。	2019 年 7 月 2 日
1.3.2001360	已修正輕微的問題。	2019 年 3 月 27 日
1.3.2001220	所有 PowerShell 指令碼已簽署。	2019 年 2 月 28 日
1.3.2001200	已修正 InitializeDisks .ps1 的問題：在 Windows Server 容錯移轉叢集中的節點上執行指令碼，會將磁碟機代號與本機磁碟機代號相符的遠端節點上的磁碟機格式化。	2019 年 2 月 27 日
1.3.2001160	已修正 Windows 2019 中遺漏桌布的問題。	2019 年 2 月 22 日
1.3.2001040	• 添加插件，用於設置顯示器永遠不會關閉以解決ACPI問題。 • SQL服務器版本和版本寫入控制台。	2019 年 1 月 21 日
1.3.2000930	修復了在啟用 ipv ENIs 6 上將路由添加到元數據的問題。	2019 年 1 月 2 日

版本	詳細資訊	發行日期
1.3.2000760	• 為設備添加了默認配置RSS和接收隊列設ENA置。 • Sysprep 時停用休眠。	2018 年 12 月 5 日
1.3.2000630	• 為服務器添加了路由 169.254.169.253/32。DNS • 設定管理使用者的新增篩選條件 • 執行個體休眠改進。 • 添加了安排EC2Launch在每次啟動時運行的選項。	2018 年 11 月 9 日
1.3.2000430.0	• 將 169.254.169.123/32 路線新增至時間服務。AMZN • 已將路由 169.254.169.249/32 新增至授權服務。GRID • 嘗試啟動 Systems Manager 時新增了 25 秒的逾時。	2018 年 9 月 19 日
1.3.200039.0	• 修復了不正確的驅動器刻字EBSNVME卷。 • 為NVME驅動程序版本添加了額外的日誌	2018 年 8 月 15 日
1.3.2000080	已修正輕微的問題。	
1.3.610	已修正將輸出和錯誤重新導向到使用者資料之檔案的問題。	
1.3.590	• 在桌面中新增了缺少的執行個體類型。 • 已修正磁碟機代號映射和磁碟安裝的問題。	
1.3.580	• 已修正 `Get-Metadata` ，以對 Web 請求使用預設系統代理設定。 • NVMe在磁盤初始化中添加了一個特殊情況。 • 已修正輕微的問題。	
1.3.550	新增了 `-NoShutdown` 選項以啟用 Sysprep 而不關機。	
1.3.540	已修正輕微的問題。	
1.3.530	已修正輕微的問題。	

版本	詳細資訊	發行日期
1.3.521	已修正輕微的問題。	
1.3.0	• 已修正電腦名稱變更的十六進位長度問題。 • 已修正電腦名稱變更可能發生重新開機迴圈的問題。 • 已修正桌面設定中的問題。	
1.2.0	• 更新以在系統日誌中顯示有關已安裝操作系EC2統（OS）的信息。 • EC2系統記錄檔中顯示的更新EC2Launch和SSM代理程式版本。 • 已修正輕微的問題。	
1.1.2	• 更新以在EC2系統日誌中顯示ENA驅動程序信息。 • 更新以從主要NIC篩選器邏輯中排除 Hyper-V。 • 在註冊表項中添加了 AWS KMS 服務器和端口以進行KMS激活。 • 已改善多位使用者的桌面設定。 • 更新以清除來自持久性存放區的路由。 • 更新以從DNS尾碼清單中的可用區域移除 z。 • 更新以解決使用者資料中 < runAsLocal 系統 > 標籤的問題。	
1.1.1	初始版本.	

在EC2舊版 Windows 作業系統執行個體啟動期間，使用此EC2Config服務執行工作

> **Note**
>
> EC2Config文件僅供歷史參考之用。它執行的作業系統版本已不再受到 Microsoft 的支援。強烈建議您升級至最新的啟動代理程式。
> 最新的啟動代理程式為 EC2Launchv2，它取代了EC2Config和EC2Launch，並預先安裝在 AWS 視窗伺服器 2022 AMIs。您也可以遷移到第 EC2Launch 2 版使用遷移工具，或者您可以在 Windows 伺服器 2016 年和 2019 上手動安裝和設定代理程式。

視窗AMIs伺服器 2016 年之前的版本視窗包含選用的服務，EC2Config服務 (`EC2Config.exe`)。EC2Config在執行個體開機時啟動，並在啟動期間以及每次停止或啟動執行個體時執行工作。EC2Config也可以根據需要執行任務。其中部分作業為自動啟用，其他作業則必須手動啟用。雖然此項服務為選用，其中可存取專屬的進階功能。此服務會在LocalSystem 帳戶中執行。

此EC2Config服務會執行 Sysprep，這是一項 Microsoft 工具，可讓您建立可重複使用AMI 的自訂視窗。EC2Config呼叫 Sysprep 時，它會使用中的檔案`%ProgramFiles%\Amazon\EC2ConfigService\Settings`來決定要執行哪些作業。您可以使用 [EC2服務屬性] 系統對話方塊間接編輯這些檔案，或直接使用編輯XML器或文字編輯器編輯這些檔案。但是，有一些高級設置在 EC2 服務屬性系統對話框中不可用，因此您必須直接編輯這些條目。

如果您在更新執行個體設定後AMI從執行個體建立，則新設定會套用至從新執行個體啟動的任何執行個體AMI。若要取得有關建立的資訊AMI，請參閱創建一個 Amazon 支EBS持 AMI。

EC2Config使用設定檔來控制其操作。您可以使用圖形化工具或直接編輯檔案來更新這些設定XML檔案。服務的二進位檔案和其他檔案包含在 `%ProgramFiles%\Amazon\EC2ConfigService` 目錄中。

目錄

- EC2Config和 AWS Systems Manager
- EC2Config任務
- EC2Config設定檔案
- 安裝 EC2Config 的最新版本
- 配置。 NET服務的代理伺EC2Config服器設定
- 從 EC2 Windows 執行個體上的系統對話方塊設定EC2Config服務屬性
- 疑難排解EC2Config啟動代理程式的問題

- EC2Config 版本歷史記錄

EC2Config和 AWS Systems Manager

此EC2Config服務會AMIs針對在 2016 年 11 月之前發行的 Windows 伺服器 2016 之前的 Windows 伺服器版本建立的執行個體上，處 Systems Manager 統管理員要求。

在 2016 年 11 月之後發佈的 Windows 伺服器 2016 之前版本所建立的執行個體包括EC2Config服務和SSM代理程式。AMIsEC2Config執行先前描述的所有工作，而SSM代理程式會處理 Systems Manager 功能 (例如執行命令和狀態管理員) 的要求。

您可以使用執行命令將現有執行個體升級為最新版本的EC2Config服務和SSM代理程式。如需詳細資訊，請參閱《使用手冊》中的使用執行命令更新SSM代理程式。AWS Systems Manager

EC2Config任務

EC2Config在執行個體第一次啟動時執行初始啟動工作，然後停用它們。若要重新執行這些作業，您必須先明確啟用執行個體，然後再關閉執行個體，或手動執行 Sysprep。這些作業如下：

- 設定管理員帳戶的隨機加密密碼。
- 產生並安裝遠端桌面連線所用的主機憑證。
- 動態擴充作業系統的分割區，以包含任何尚未分割的空間。
- 執行指定的使用者資料 (和 Cloud-Init，如果尚未安裝)。如需指定使用者資料的詳細資訊，請參閱使用使用者資料輸入啟動執行個EC2體時執行指令。

EC2Config每次執行個體啟動時，都會執行下列工作：

- 變更主機名稱，以符合 Hex 表示法中的私有 IP 地址 (預設停用此作業，但必須啟用才能在執行個體啟動時執行)。
- 設定金鑰管理伺服器 (AWS KMS)，檢查 Windows 啟用狀態，並視需要啟動 Windows。
- 掛接所有 Amazon EBS 磁碟區和執行個體存放磁碟區，並將磁碟區名稱對應至磁碟機代號。
- 將事件日誌項目寫入主控台，以協助故障診斷 (預設停用此作業，但必須啟用才能在執行個體啟動時執行)。
- 將 Windows 已就緒的狀態寫入主控台。
- 將自訂路由新增至主要網路介面卡，以在連接單一NIC或多個 NICs IP 位址時啟用下列 IP 位址：`169.254.169.250``169.254.169.251`、和`169.254.169.254`。這些地址供 Windows 啟用和當您要存取執行個體中繼資料時使用。

> **Note**
>
> 如果 Windows 作業系統設定為使用IPv4，則可以使用這些IPv4連結本機位址。如果 Windows 作業系統已停用IPv4網路通訊協定堆疊並IPv6改為使用，請新增`[fd00:ec2::240]``169.254.169.250`和`169.254.169.251`。然後新增`[fd00:ec2::254]` 取代 `169.254.169.254`。

EC2Config每次使用者登入時，都會執行下列工作：

- 在桌面背景顯示桌布資訊。

執行個體在執行時，您可以要求視需要EC2Config執行下列工作：

- 執行 Sysprep 並關閉執行個體，以便您可以AMI從該執行個體建立執行個體。如需詳細資訊，請參閱EC2AMI使用視窗系統創建一個 Amazon。

EC2Config設定檔案

設定檔案會控制EC2Config服務的作業。這些檔案位在 `C:\Program Files\Amazon\Ec2ConfigService\Settings` 目錄：

- `ActivationSettings.xml` - 使用金鑰管理伺服器 (AWS KMS) 控制產品啟用。
- `AWS.EC2.Windows.CloudWatch.json`— 控制要傳送到哪些效能計數器，以 CloudWatch 及要傳送至記錄的 CloudWatch 記錄檔。
- `BundleConfig.xml`控制如何EC2Config準備執行個體以商店為建立的執行個體。AMI
- `Config.xml`—控制主要設定。
- `DriveLetterConfig.xml`—控制磁碟機代號映射。
- `EventLogConfig.xml`—控制執行個體開機時顯示在主控台上的事件日誌資訊。
- `WallpaperSettings.xml`—控制顯示在桌面背景上的資訊。

ActivationSettings.xml

此檔案包含控制產品啟用的設定。當 Windows 開機時，EC2Config服務會檢查視窗是否已經啟動。如果 Windows 尚未啟動，服務會搜尋指定的 AWS KMS 伺服器，嘗試啟動 Windows。

- `SetAutodiscover` - 表示是否自動偵測 AWS KMS 。
- `TargetKMSServer`儲存的私人 IP 位址。 AWS KMS AWS KMS 和您的執行個體必須在相同的區域。
- `DiscoverFromZone`從指DNS定的區域探索 AWS KMS 伺服器。
- `ReadFromUserData`從 UserData取得伺 AWS KMS 服器。
- `LegacySearchZones`從指DNS定的區域探索 AWS KMS 伺服器。
- `DoActivate`—使用區段中的指定設定嘗試啟用。此值可以為 `true` 或 `false`。
- `LogResultToConsole`—將結果顯示在主控台。

BundleConfig.xml

此檔案包含控制如何EC2Config準備執行個體以進行AMI建立的設定。

- `AutoSysprep`—指出是否自動使用 Sysprep。變更值為 `Yes` 將使用 Sysprep。
- `SetRDPCertificate`—設定遠端桌面伺服器的自我簽署憑證。這可讓您安全地RDP進入執行個體。如果新執行個體應取得該憑證，請變更值為 `Yes`。

  此設定不適用於 Windows Server 2016 以前版本的作業系統執行個體，因為它們可以產生自己的憑證。
- `SetPasswordAfterSysprep`—在新啟動的執行個體上設定隨機密碼，以使用者啟動金鑰加密，然後將加密的密碼輸出到主控台。如果新執行個體不應設定為隨機加密密碼，請變更此設定的值為 No。

Config.xml

外掛程式

- `Ec2SetPassword`—每次啟動執行個體時產生隨機加密密碼。此功能自初次啟動後將預設停用，因此重新啟動本執行個體不會變更使用者設定的密碼。變更此設定為 `Enabled`，將繼續在每次啟動執行個體時產生密碼。

  如果您打算AMI從執行個體建立執行個體，此設定非常重要。
- `Ec2SetComputerName`—依執行個體的 IP 地址將執行個體的主機名稱設為唯一名稱，並重新啟動執行個體。若要自行設定主機名稱，或避免現有的主機名稱遭到修改，您必須停用此設定。
- `Ec2InitializeDrives`—在啟動期間初始化及格式化所有磁碟區。此功能預設為啟用。

- `Ec2EventLog`—在主控台顯示事件日誌項目。根據預設，會顯示系統事件日誌中最近的三個錯誤項目。若要指定要顯示的事件日誌項目，請編輯 `EventLogConfig.xml` 目錄內的 `EC2ConfigService\Settings` 檔案。如需有關此檔案中設定的詳細資訊，請參閱元件庫中的 Eventlog 金鑰。MSDN
- `Ec2ConfigureRDP`—在執行個體上設定自我簽署憑證，讓使用者可使用遠端桌面安全存取執行個體。此設定不適用於 Windows Server 2016 以前版本的作業系統執行個體，因為它們可以產生自己的憑證。
- `Ec2OutputRDPCert`—顯示遠端桌面憑證資訊至主控台，讓使用者可對照指紋進行確認。
- `Ec2SetDriveLetter`—依使用者定義的設定來設定掛載磁碟區的磁碟機代號。依預設，將 Amazon EBS 磁碟區連接至執行個體時，可以使用執行個體上的磁碟機代號來裝載該磁碟區。若要指定磁碟機代號映射，請編輯 `DriveLetterConfig.xml` 目錄內的 `EC2ConfigService\Settings` 檔案。
- `Ec2WindowsActivate`—外掛程式將處理 Windows 啟用。外掛程式會檢查 Windows 是否啟動。如果沒有，它會更新用 AWS KMS 戶端設定，然後啟動 Windows。

  若要修改 AWS KMS 設定，請編輯位於`EC2ConfigService\Settings`目錄中的`ActivationSettings.xml`檔案。
- `Ec2DynamicBootVolumeSize`—擴充 Disk 0/Volume 0，以包含任何尚未分割的空間。
- `Ec2HandleUserData`—在 Sysprep 執行後建立並執行使用者於初次啟動執行個體時建立的指令碼。包裝在腳本標籤中的命令將保存到批處理文件中，並將包裝在 PowerShell 標籤中的命令保存到 .ps1 文件中（與 EC2 服務屬性系統對話方塊中的「用戶數據」核取方塊相對應）。
- `Ec2ElasticGpuSetup`如果實例與彈性物GPU件相關聯，則安裝彈性GPU軟體套件。
- `Ec2FeatureLogging`—將 Windows 功能安裝和對應的服務狀態傳送至主控台。僅支援 Microsoft Hyper-V 功能和對應的 vmms 服務。

全域設定

- `ManageShutdown`確保從執行個體存放區支援啟動的執行個體在執行 Sysprep 時AMIs不會終止。
- `SetDnsSuffixList`設置 Amazon EC2 網絡適配器的DNS後綴。這允許DNS解析在 Amazon 中運行的服務器，EC2而無需提供完整的域名。

> **Note**
>
> 這會新增下列網域的DNS尾碼查詢，並設定其他標準尾碼。如需啟動代理程式如何設定DNS尾碼的詳細資訊，請參閱。設定 EC2 Windows 啟動代理程式的DNS尾碼

```
region.ec2-utilities.amazonaws.com
```

- `WaitForMetaDataAvailable`確保EC2Config服務在繼續開機之前，將等待中繼資料可供存取且網路可用。此檢查EC2Config可確保可從中繼資料取得資訊，以供啟用及其他外掛程式使用。
- `ShouldAddRoutes`— 將自訂路由新增至主要網路介面卡，以便在連接多個NICs網路介面卡時啟用下列 IP 位址：169.254.169.250、169.254.169.251 和 169.254.169.254。這些地址供 Windows 啟用和當您要存取執行個體中繼資料時使用。
- `RemoveCredentialsfromSyspreponStartup`—下次服務啟動時從 `Sysprep.xml` 移除管理員密碼。若要確保此密碼保持一致，請編輯此設定。

DriveLetterConfig.xml

此檔案包含控制磁碟機代號映射的設定。根據預設，磁碟區可映射至任何可用的磁碟機代號。您可將磁碟區掛載至特定的磁碟機代號，如下所示。

```
<?xml version="1.0" standalone="yes"?>
<DriveLetterMapping>
  <Mapping>
    <VolumeName></VolumeName>
    <DriveLetter></DriveLetter>
  </Mapping>
  . . .
  <Mapping>
    <VolumeName></VolumeName>
    <DriveLetter></DriveLetter>
  </Mapping>
</DriveLetterMapping>
```

- `VolumeName`—磁碟區標籤。例如：*`My Volume`*。若要指定執行個體儲存體磁碟區的映射，請使用 `Temporary Storage X` 標籤，其中的 X 為 0 至 25 的數字。
- `DriveLetter`—磁碟機代號。例如：*`M:`*。假如磁碟機代號已在使用中，映射便會失敗。

EventLogConfig.xml

此檔案包含控制執行個體開機時顯示在主控台上之事件日誌資訊的設定。根據預設，會顯示系統事件日誌中最近的三個錯誤項目。

- Category—傳送至主控台的事件日誌金鑰。
- ErrorType——事件類型 (例如 Error、Warning、Information。)
- NumEntries—此類別存放的事件數量。
- LastMessageTime—為避免重複推送相同的訊息，服務會在每次推送訊息時更新此值。
- AppName—事件來源或記錄事件的應用程式。

WallpaperSettings.xml

此檔案包含控制顯示在桌面背景上之資訊的設定。預設顯示下列資訊。

- Hostname—顯示電腦名稱。
- Instance ID—顯示執行個體的 ID。
- Public IP Address—顯示執行個體的公有 IP 地址。
- Private IP Address—顯示執行個體的私有 IP 地址。
- Availability Zone—顯示執行個體正在執行的可用區域。
- Instance Size—顯示執行個體的類型。
- Architecture—顯示 PROCESSOR_ARCHITECTURE 環境變數的設定。

移除此項目，將移除預設顯示的任何資訊。您可新增額外要顯示的執行個體中繼資料，如下所示。

```
<WallpaperInformation>
  <name>display_name</name>
  <source>metadata</source>
  <identifier>meta-data/path</identifier>
</WallpaperInformation>
```

您可新增額外要顯示的系統環境變數，如下所示。

```
<WallpaperInformation>
  <name>display_name</name>
  <source>EnvironmentVariable</source>
  <identifier>variable-name</identifier>
</WallpaperInformation>
```

InitializeDrivesSettings.xml

此文件包含控制如何EC2Config初始化驅動器的設置。

依預設，請EC2Config初始化未與作業系統連線的磁碟機。您可自訂外掛程式，如下所示。

```
<InitializeDrivesSettings>
    <SettingsGroup>setting</SettingsGroup>
</InitializeDrivesSettings>
```

使用設定群組指定初始化磁碟機的方式：

FormatWithTRIM

格式化磁碟機時啟用TRIM指令。在磁碟機格式化和初始化之後，系統會還原TRIM組態。

從EC2Config版本 3.18 開始，依預設，會在磁碟格式化作業期間停用該TRIM指令。如此將可改善格式化的時間。使用此設定可TRIM在 3.18 版及更新EC2Config版本的磁碟格式化作業期間啟用。

FormatWithoutTRIM

格式化驅動器時禁用TRIM命令，並改善 Windows 中的格式化時間。在磁碟機格式化和初始化之後，系統會還原TRIM組態。

DisableInitializeDrives

停用新磁碟機的格式化。請用此設定手動初始化磁碟機。

安裝 EC2Config 的最新版本

默認情況下，該EC2Config服務包含在 Windows 服務器 2016 AMIs 之前。更新EC2Config服務時，新的 Windows AMIs 會 AWS 包含最新版本的服務。不過，您必須使用最新版本的更新您自己的 Windows AMIs 和執行個體EC2Config。

> **Note**
>
> EC2LaunchEC2Config在視窗伺服器 2016 年和 2019 年上取代。如需詳細資訊，請參閱在 EC2 Windows 執行個體啟動期間，使用 EC2Launch v1 代理程式執行工作。適用於所有受支援的 Windows 伺服器版本的最新啟動服務為 EC2Launchv2，這會取代EC2Config和 EC2Launch。

如需如何接收更新通知的相關資EC2Config訊，請參閱訂閱 EC2 Windows 啟動代理程式通知。如需各版本變更的詳細資訊，請參閱EC2Config 版本歷史記錄。

開始之前

- 請確認您擁有。 NET框架 3.5 SP1 或更高版本。
- 依預設，安裝程式會在安裝期間以預設設定檔案取代您的設定檔案，並在安裝完成時重新啟動EC2Config服務。如果您變更了EC2Config服務設定，請從`%Program Files%\Amazon\Ec2ConfigService\Settings`目錄複製`config.xml`檔案。更新EC2Config服務之後，您可以還原此檔案以保留您的組態變更。
- 如果您的版本早EC2Config於 2.1.19 版，而您要安裝 2.2.12 或更早版本，則必須先安裝 2.1.19 版。若要安裝 2.1.19 版，請下載 EC2Install_2.1.19.zip，解壓縮檔案，然後執行。`EC2Install.exe`

> **Note**
>
> 如果您的版本早EC2Config於 2.1.19 版，而您要安裝 2.3.313 或更新版本，則無需先安裝 2.1.19 版，則可以直接安裝該版本。

驗證 EC2Config 版本

請使用下列程序來驗證執行個體上安裝的EC2Config版本。

若要驗證已安裝的版本 EC2Config

1. 從您的執行個體啟動AMI並連線至執行個體。
2. 從 Control Panel (控制台) 中，選取 Programs and Features (程式和功能)。
3. 從已安裝程式清單中，找出 `Ec2ConfigService`。其版本編號將出現在 Version (版本) 欄中。

更新 EC2Config

請遵循下列程序，在執行個體EC2Config上下載並安裝最新版本的。

若要下載並安裝最新版本的 EC2Config

1. 下載並解壓縮EC2Config安裝程式。
2. 執行 `EC2Install.exe`。如需完整的選項清單，請執行 `EC2Install` 並加入 `/?` 選項。根據預設，設定程式會顯示提示。若要執行命令且不加入提示，請使用 `/quiet` 選項。

> **Important**
>
> 若要保留您儲存的`config.xml`檔案中的自訂設定，請`EC2Install`使用`/norestart`選項執行、還原設定，然後手動重新啟動EC2Config服務。

3. 如果您執行的是 4.0 或更新EC2Config版本，則必須從 Microsoft 服務嵌入式管SSM理單元重新啟動執行個體上的代理程式。

> **Note**
>
> 在您重新啟動或停止並啟動執行個體之前，更新的EC2Config版本資訊不會顯示在執行個體系統記錄檔或 Trusted Advisor 檢查中。

若要下載並安裝最新版本的EC2Config使用 PowerShell

若要下載、解壓縮並安裝最新版本EC2Config的 use PowerShell，請從 PowerShell 視窗執行下列命令：

```
$Url = "https://s3.amazonaws.com/ec2-downloads-windows/EC2Config/EC2Install.zip"
$DownloadZipFile = "$env:USERPROFILE\Desktop\" + $(Split-Path -Path $Url -Leaf)
$ExtractPath = "$env:USERPROFILE\Desktop\"
Invoke-WebRequest -Uri $Url -OutFile $DownloadZipFile
$ExtractShell = New-Object -ComObject Shell.Application
$ExtractFiles = $ExtractShell.Namespace($DownloadZipFile).Items()
$ExtractShell.NameSpace($ExtractPath).CopyHere($ExtractFiles)
Start-Process $ExtractPath
Start-Process `
    -FilePath $env:USERPROFILE\Desktop\EC2Install.exe `
    -ArgumentList "/S"
```

> **Note**
>
> 如果您在下載檔案時收到錯誤訊息，而且您使用的是 Windows 伺服器 2016 或更早版本，則可能需要為 PowerShell 終端機啟用 TLS 1.2。您可以使用以下命令為當前 PowerShell 會話啟用 TLS 1.2，然後再試一次：
>
> ```
> [Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
> ```

查看 `C:\Program Files\Amazon\` 是否有 `Ec2ConfigService` 目錄以驗證安裝。

配置。 NET服務的代理伺EC2Config服器設定

您可以使用下列其中一種方法將EC2Config服務設定為透過 Proxy 進行通訊：用 AWS SDK於。NET，`system.net`元素，或 Microsoft 組策略和互聯網瀏覽器。使用 AWS SDK的. NET是首選方法，因為您可以指定登錄憑據。

方法

- 使用 AWS SDK for .NET (偏好) 設定代理伺服器設定
- 使用 system.net 元素設定代理伺服器設定
- 使用 Microsoft 群組政策和 Microsoft Internet Explorer 設定代理伺服器設定

使用 AWS SDK for .NET (偏好) 設定代理伺服器設定

您可以透過指定`Ec2Config.exe.config`檔案中的`proxy`元素來設定EC2Config服務的 Proxy 設定。如需詳細資訊，請參閱的組態檔案 AWS SDK參考。 NET。

指定 Ec2Config.exe.config 中的代理伺服器元素

1. 在您希望EC2Config服務透過 Proxy 進行通訊的執行個體上編輯`Ec2Config.exe.config`檔案。根據預設，檔案位在下列目錄：`%ProgramFiles%\Amazon\Ec2ConfigService`。
2. 將以下 `aws` 元素新增到 `configSections`。請勿將此元素新增到任何現有的 `sectionGroups`。

   適用於 3.17 或EC2Config更早版本

   ```
   <configSections>
      <section name="aws" type="Amazon.AWSSection, AWSSDK"/>
   </configSections>
   ```

   對於EC2Config版本 3.18 或更高版本

   ```
   <configSections>
        <section name="aws" type="Amazon.AWSSection, AWSSDK.Core"/>
   </configSections>
   ```

3. 將以下 `aws` 元素新增到 `Ec2Config.exe.config` 檔案。

```
<aws>
   <proxy
     host="string value"
     port="string value"
     username="string value"
     password="string value" />
</aws>
```

4. 儲存您的變更。

使用 system.net 元素設定代理伺服器設定

您可在 `system.net` 檔案的 `Ec2Config.exe.config` 元素中指定 proxy 的設定。若要取得更多資訊，請參閱上的defaultProxy圖元（網路設定）MSDN。

指定 Ec2Config.exe.config 中的 system.net 元素

1. 在您希望EC2Config服務透過 Proxy 進行通訊的執行個體上編輯`Ec2Config.exe.config`檔案。根據預設，檔案位在下列目錄：`%ProgramFiles%\Amazon\Ec2ConfigService`。
2. 新增 `defaultProxy` 項目至 `system.net`。若要取得更多資訊，請參閱上的defaultProxy 圖元（網路設定）MSDN。

   例如，下列設定將路由所有流量，以使用 Internet Explorer 目前設定的 proxy，但不包含中繼資料和授權流量，這些流量將繞過代理伺服器。

```
<defaultProxy>
    <proxy usesystemdefault="true" />
    <bypasslist>
        <add address="169.254.169.250" />
        <add address="169.254.169.251" />
        <add address="169.254.169.254" />
        <add address="[fd00:ec2::250]" />
        <add address="[fd00:ec2::254]" />
    </bypasslist>
</defaultProxy>
```

3. 儲存您的變更。

使用 Microsoft 群組政策和 Microsoft Internet Explorer 設定代理伺服器設定

EC2Config服務會在本機系統使用者帳戶下執行。在執行個體上變更群組政策設定後，您可在 Internet Explorer 中為此帳戶指定執行個體範圍的代理伺服器設定。

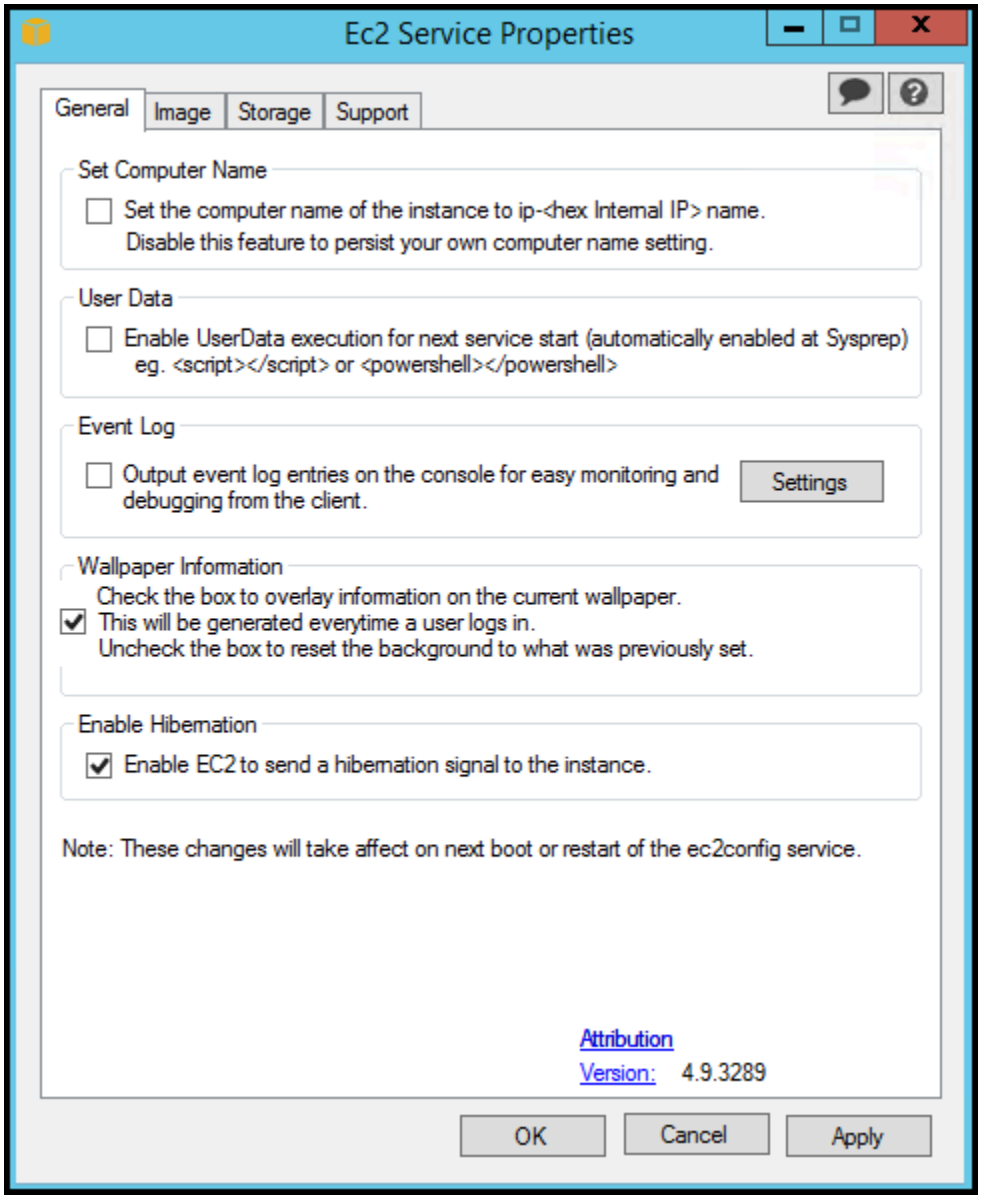
使用群組政策和 Internet Explorer 設定代理伺服器設定

1. 在您希望EC2Config服務透過 Proxy 進行通訊的執行個體上，以系統管理員身分開啟命令提示字元，輸入**`gpedit.msc`**，然後按 Enter 鍵。
2. 在本機群組政策編輯器中的 Local Computer Policy (本機電腦政策) 底下，選擇 Computer Configuration (電腦設定)、Administrative Templates (系統管理範本)、Windows Components (Windows 元件)、Internet Explorer。
3. 在右側窗格中，選擇 Make proxy settings per-machine (rather than per-user) (依每一部電腦設定 proxy 設定 (而不是依每一位使用者))，然後選擇 Edit policy setting (編輯政策設定)。
4. 選擇 Enabled (已啟用)，然後選擇 Apply (套用)。
5. 開啟 Internet Explorer，然後選擇 Tools (工具) 按鈕。
6. 選擇 Internet Option (網際網路選項)，然後選擇 Connections (連線) 索引標籤。
7. 選擇LAN設定。
8. 在 [Proxy 伺服器] 下方，選擇 [使用 Proxy 伺服器做為您的LAN選項]。
9. 指定地址和連接埠資訊，然後選擇 OK (確定)。

從 EC2 Windows 執行個體上的系統對話方塊設定EC2Config服務屬性

下列程序說明如何使用 [EC2服務屬性] 系統對話方塊來啟用或停用設定。

1. 啟動和連線至 Windows 執行個體。
2. 從 [開始] 功能表，按一下 [所有程式]，然後按一下 [EC2ConfigService設定]。



3. 在 [EC2服務屬性] 系統對話方塊的 [一般] 索引標籤上，您可以啟用或停用下列設定。

   Set Computer Name (設定電腦名稱)

   如果啟用此設定 (預設為停用)，每次開機時會將主機名稱與目前的內部 IP 地址比對；如果主機名稱與內部 IP 地址不符，將重設主機名稱以包含內部 IP 地址，接著系統會重新啟動，以選用新的主機名稱。若要自行設定主機名稱，或避免現有的主機名稱遭到修改，請勿啟用此設定。

   使用者資料

   使用者資料執行可讓您指定執行個體中繼資料中的指令碼。根據預設，這些指令碼會在初始啟動時執行。您也可將其設定為在下次重新啟動或啟動執行個體時，或每次重新啟動或啟動執行個體時執行。

如果指令碼較大，建議透過使用者資料下載指令碼，然後執行指令碼。

如需詳細資訊，請參閱使用者資料執行。

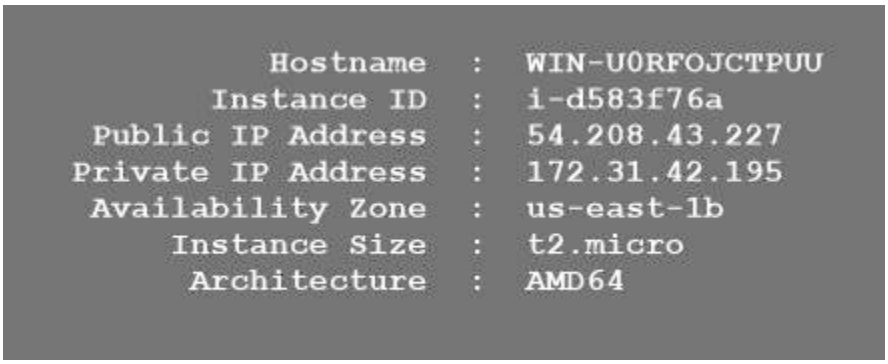
事件日誌

使用此設定可於開機時在主控台上顯示事件日誌項目，以方便進行監控和除錯。

按一下 Settings (設定)，指定要傳送至主控台之日誌項目的篩選條件。預設篩選條件會傳送系統事件日誌中最近的三個錯誤項目至主控台。

Wallpaper Information (桌布資訊)

使用此設定可在桌面背景顯示系統資訊。以下是顯示在桌面背景上的資訊範例。

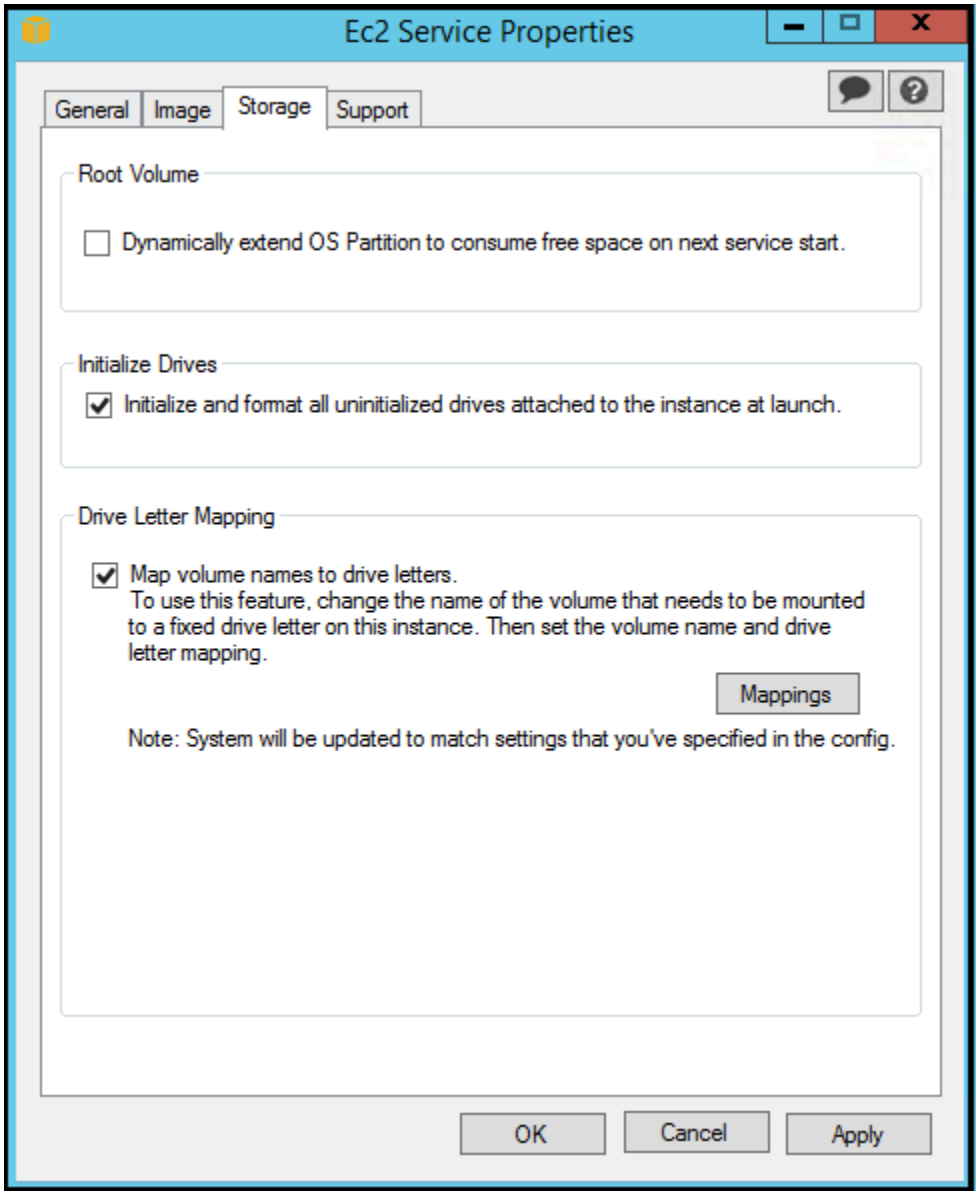


顯示在桌面背景上的資訊由設定檔 `EC2ConfigService\Settings\WallpaperSettings.xml` 控制。

啟用休眠

使用此設定可允許EC2向作業系統發出訊號以執行休眠。

4. 按一下 Storage (儲存) 索引標籤。您可以啟用或停用以下設定。



Root Volume (根磁碟區)

此設定將動態擴充 Disk 0/Volume 0，以包含任何尚未分割的空間。此設定在執行個體是從自訂大小的根裝置磁碟區啟動時很有用。

Initialize Drives (初始化磁碟機)

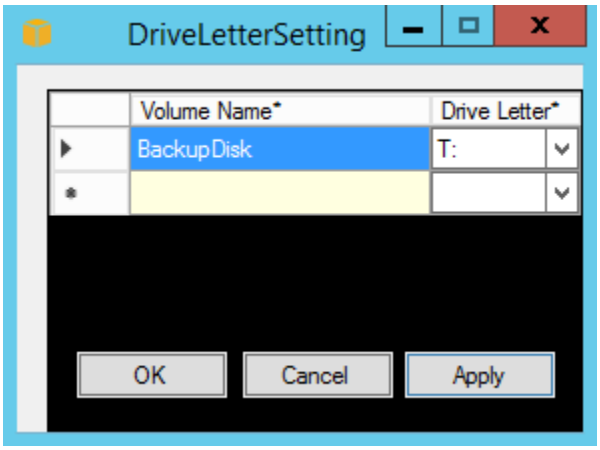
此設定會在啟動時格式化及掛載連接至執行個體的所有磁碟區。

Drive Letter Mapping (磁碟機代號映射)

系統會對應連接至執行個體的磁碟區與磁碟機代號。對於 Amazon EBS 磁碟區，預設值是指派從 D: 到 Z: 的磁碟機代號。對於實例存儲卷，默認值取決於驅動程序。 AWS PV 驅動程式和

Citrix PV 驅動程式會指派執行個體儲存磁碟區磁碟區代號，從 Z: 到 A:。Red Hat 驅動程式會為執行個體存放區磁碟區指派 D: 到 Z: 的磁碟機代號。

若要為磁碟區選擇磁碟機代號，請按一下 Mappings (映射)。在DriveLetterSetting對話方塊中，指定每個磁碟區的磁碟區名稱和磁碟機代號值，按一下 [套用]，然後按一下 [確定]。建議選取不會與可能正在使用中的磁碟機代號衝突的代號，例如選取較中間的英文字母做為磁碟機代號。



在您指定磁碟機代號對應，並附加磁碟區標籤與您指定的其中一個磁碟區名稱相同的磁碟區之後，EC2Config會自動將您指定的磁碟機代號指定給該磁碟區。但假如磁碟機代號已在使用中，磁碟機代號映射便會失敗。請注EC2Config意，當您指定磁碟機代號對應時，不會變更已掛接的磁碟區的磁碟機代號。

5. 若要儲存您的設定並稍後繼續處理，請按一下 [確定] 關閉 [EC2服務屬性] 系統對話方塊。如果您已完成自訂執行個體，並想要AMI從該執行個體建立執行個體，請參閱EC2AMI使用視窗系統創建一個 Amazon。

疑難排解EC2Config啟動代理程式的問題

下列資訊可協助您疑難排解EC2Config服務的問題。

無法存取EC2Config的執行個體上的更新

使用下列程序來更新使用遠端桌面無法存取的 Windows Server 執行個體上的EC2Config服務。

若要EC2Config在您無法連線到的 Amazon EBS 支援 Windows 執行個體上進行更新

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。

3. 尋找受影響的執行個體。選取執行個體，並依序選取 Instance state (執行個體狀態)、Stop instance (停止執行個體)。

> **Warning**
>
> 停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。若要保留執行個體存放區磁碟區的資料，請將資料備份至持久性儲存。

4. 選擇 Launch instances (啟動執行個體)，並在與受影響執行個體相同的可用區域中，建立暫時性的 `t2.micro` 執行個體。使用不同AMI於您用來啟動受影響執行個體的執行個體。

> **Important**
>
> 若您沒有在與受影響的執行個體相同的可用區域內建立執行個體，您將無法將受影響執行個體的根磁碟區連接到新的執行個體。

5. 在EC2主控台中，選擇 [磁碟區]。
6. 尋找受影響執行個體的根磁碟區。分開磁碟區，再將磁碟區連接至先前建立的暫時執行個體。使用預設裝置名稱 (xvdf) 連接它。
7. 使用遠端桌面連接至暫時性執行個體，然後使用 Disk Management 公用程式將磁碟區變成可用。
8. 下載最新版本的EC2Config服務。將 `.zip` 檔案中的檔案解壓縮到所連接之磁碟機上的 `Temp` 目錄中。
9. 在暫時性執行個體上，開啟 Run (執行) 對話方塊，輸入 **`regedit`**，然後按 Enter 鍵。
10. 選擇 `HKEY_LOCAL_MACHINE`。從 File (檔案) 功能表中，選擇 Load Hive (載入 Hive)。選擇磁碟機，然後導覽並開啟下列檔案：`Windows\System32\config\SOFTWARE`。出現提示時，指定機碼名稱。
11. 選取剛載入的機碼並導覽至 `Microsoft\Windows\CurrentVersion`。選擇 `RunOnce` 機碼。如果該機碼不存在，從內容 (按一下右鍵) 選單中選擇 `CurrentVersion`，選擇 New (新增)，然後選擇 Key (機碼)。將機碼命名為 `RunOnce`。
12. 從內容 (按一下右鍵) 選單中選擇 `RunOnce` 機碼選擇 New (新增)，然後選擇 String Value (字串值)。在名稱輸入 `Ec2Install`，在資料輸入 `C:\Temp\Ec2Install.exe /quiet`。
13. 選擇 `HKEY_LOCAL_MACHINE\`*`specified key name`*`\Microsoft\Windows NT\CurrentVersion\Winlogon` 機碼。從內容 (按一下右鍵) 選單中選擇 New (新增)，然後選擇 String Value (字串值)。在名稱輸入 **`AutoAdminLogon`**，在數值資料輸入 **`1`**。

14. 選擇 HKEY_LOCAL_MACHINE*specified key name*\Microsoft\Windows NT\CurrentVersion\Winlogon> 機碼。從內容 (按一下右鍵) 選單中選擇 New (新增)，然後選擇 String Value (字串值)。在名稱輸入 **DefaultUserName**，在數值資料輸入 **Administrator**。
15. 選擇 HKEY_LOCAL_MACHINE*specified key name*\Microsoft\Windows NT\CurrentVersion\Winlogon 機碼。從內容 (按一下右鍵) 選單中選擇 New (新增)，然後選擇 String Value (字串值)。在名稱輸入 **DefaultPassword**，在數值資料輸入密碼。
16. 在 Registry Editor (登錄編輯程式) 導覽窗格中，選擇一開始開啟登錄編輯程式時建立的暫時性機碼。
17. 從 File (檔案) 選單中，選擇 Unload Hive (解除載入 Hive)。
18. 在 Disk Management 公用程式中，選擇先前連接的磁碟機，開啟內容 (按一下滑鼠右鍵) 選單，然後選擇 Offline (離線)。
19. 在 Amazon EC2 主控台中，將受影響的磁碟區與暫存執行個體分離，然後使用裝置名稱/dev/sda1將其重新連接到執行個體。您必須指定此裝置名稱，將磁碟區指定為根磁碟區。
20. 停止並啟動 Amazon EC2 執行個體執行個體。
21. 執行個體啟動後，檢查系統日誌並確認出現 Windows is ready to use (Windows 已可開始使用) 的訊息。
22. 開啟登錄編輯程式，然後選擇 HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon。刪除您之前創建的字符串值鍵：AutoAdminLogonDefaultUserName，和DefaultPassword。
23. 刪除或停止您在此程序中建立的暫時性執行個體。

EC2Config 版本歷史記錄

視窗伺服器 2016 年AMIs之前的視窗包含稱為服務 (EC2Config.exe) 的選擇性EC2Config服務。EC2Config在執行個體開機時啟動，並在啟動期間以及每次停止或啟動執行個體時執行工作。

當EC2Config服務的新版本發布時，您可以收到通知。如需詳細資訊，請參閱訂閱 EC2 Windows 啟動代理程式通知。

下表說明的已發行版本EC2Config。如需SSM代理程式更新的相關資訊，請參閱系統管理員SSM代理程式版本說明。

版本	詳細資訊	發行日期
4.9.5777	• 修正部分執行個體類型設定不正確的問題。RSS	2024 年六月十七日

版本	詳細資訊	發行日期
	• 新版本的SSM代理程式3.3.484.0 。	
4.9.5554	• 根據登錄項目限制網域名稱轉移：HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Dnscache\Parameters\DomainNameDevolutionLevel 。 • 新版本的SSM代理程式3.2.1630.0 。	2023 年 10 月 4 日
4.9.5467	• 已新增重試功能，可探索主控台連接埠。 • 新版本的SSM代理程式3.1.2282.0 。	2023 年 8 月 1 日
4.9.5288	• 更新AWS核心SDK版本3.7.103.23 。 • 修復了僅在啟用的實例EC2Config 上更新AWS-UpdateEC2Config SSM文檔無法更新的問題IMDSv2。 • 新版本的SSM代理程式3.1.2144.0 。	2023 年 3 月 8 日
4.9.5231	• 新版本的SSM代理程式	2023 年 2 月 14 日
4.9.5103	• 已修正 r5d 和 i4i 執行個體系列上無法正確識別暫時性磁碟區的問題。 • SSM代理程式的新版本。	2022 年 12 月 5 日
4.9.5064	• 已更新，使用PCI區段資訊來選取主控台連接埠。 • 已簽署的 PowerShell 指令碼並新增版權標頭。 • 已修正主要網路轉接器選取邏輯。 • 新版本的SSM代理程式。	2022 年 11 月 16 日

版本	詳細資訊	發行日期
4.9.4588	• 更新IMDS等待邏輯以僅IMDSv2提出請求。 • 新增 libec2launch.dll 啟動代理程式共用庫。 • SSM代理程式的新版本。	2022 年 5 月 31 日
4.9.4556	• 添加了等待邏輯以確保NIC在使用前完全初始化。 • 新版本的 Log4Net 2.0.14.0 收取了安全修補程式。 • 新版的SSM代理程式 3.1.1045.0 取得安全性修補程式。	2022 年 3 月 1 日
4.9.4536	• 修復了 Temp 資料夾遺失時使用者資料損毀的問題。 • 新版本的SSM代理程式	2022 年 1 月 31 日
4.9.4508	• 修正正確運算 diskpart 指令碼路徑的問題。 • 新版本的SSM代理程式 3.1.338.0。	2021 年 10 月 6 日
4.9.4500	• 更新`Install-EgpuManagerConfig` 與 IMDS v2 的支持。 • 已更新 Web 連結以使用 https。 • SSM代理程式的新版本	2021 年 9 月 7 日
4.9.4419	• 修正了 1 IMDS 版後備邏輯 • 更新了 Windows 臨時目錄到EC2Config臨時目錄的所有用法 • SSM代理程式的新版本	2021 年 6 月 2 日
4.9.4381	• 增加了對SSM文檔架構 2.2 版本的支持 EC2ConfigUpdater • 將 AWS 硝基飛地包版本添加到控制台日誌 • 新版本的SSM代理程式	2021 年 5 月 4 日

版本	詳細資訊	發行日期
4.9.4326	• 移除了設定 UI 中的所有連結 • 這是支持視窗服務器 2008 的最後一個EC2Config版本。	2021 年 3 月 3 日
4.9.4279	• 已修正與 `Ec2ConfigMonitor` 排程任務相關的安全性問題 • 修正磁碟機代號對應問題和不正確的暫時磁碟計數 • 新增 `OsCurrentBuild` 和 `OsReleaseId` 至主控台輸出 • SSM代理程式的新版本	2020 年 12 月 11 日
4.9.4222	• 修正了 1 IMDS 版後備邏輯 • SSM代理程式的新版本	2020 年 4 月 7 日
4.9.4122	• 增加了對 IMDS V2 的支持 • 新版本的SSM代理程式	2020 年 3 月 4 日
4.9.3865	• 修復了金屬實例上的 Windows 服務器 2008 R2 檢測COM端口的問題 • SSM代理程式的新版本	2019 年 10 月 31 日
4.9.3519	• SSM代理程式的新版本	2019 年 6 月 18 日
4.9.3429	• SSM代理程式的新版本	2019 年 4 月 25 日
4.9.3289	• SSM代理程式的新版本	2019 年 2 月 11 日
4.9.3270	• 增加了插件，用於設置顯示器永遠不會關閉以解決ACPI問題 • SQL寫入控制台的服務器版本和版本 • 新版本的SSM代理程式	2019 年 1 月 22 日
4.9.3230	• 已更新磁碟機代號對應說明，以更切合功能。 • SSM代理程式的新版本	2019 年 1 月 10 日

版本	詳細資訊	發行日期
4.9.3160	• 增加了主要的等待時間 NIC • 為設備添加了默認配置RSS和接收隊列設ENA置 • Sysprep 時停用休眠 • SSM代理程式的新版本 • 升級 AWS SDK至 13 年 3 月 9 日	2018 年 12 月 15 日
4.9.3067	• 執行個體休眠改進 • SSM代理程式的新版本	2018 年 11 月 8 日
4.9.3034	• 為服務器添加了路由 169.254.169.253/32 DNS • SSM代理程式的新版本	2018 年 10 月 24 日
4.9.2986	• 為所有EC2Config相關二進製文件添加了簽名 • SSM代理程式的新版本	2018 年 10 月 11 日
4.9.2953	新版本的SSM代理程式 (2.3.117.0)	2018 年 10 月 2 日
4.9.2926	新版本的SSM代理程式 (2.3.68.0)	2018 年 9 月 18 日
4.9.2905	• 新版本的SSM代理程式 (2.3.50.0) • 將 169.254.169.123/32 路線新增至時間服務 AMZN • 在許可證服務中添加了路線 169.254.169.249/32 GRID • 修正EBSNVMe磁碟區標記為暫時性的問題	2018 年 9 月 17 日
4.9.2854	新版本的SSM代理程式 (2.3.13.0)	2018 年 8 月 17 日
4.9.2831	新版本的SSM代理程式 (2.2.916.0)	2018 年 8 月 7 日
4.9.2818	新版本的SSM代理程式 (2.2.902.0)	2018 年 7 月 31 日

版本	詳細資訊	發行日期
4.9.2756	新版本的SSM代理程式 (2.2.800.0)	2018 年 6 月 27 日
4.9.2688	新版本的SSM代理程式 (2.2.607.0)	2018 年 5 月 25 日
4.9.2660	新版本的SSM代理程式 (2.2.546.0)	2018 年 5 月 11 日
4.9.2644	新版本的SSM代理程式 (2.2.493.0)	2018 年 4 月 26 日
4.9.2586	新版本的SSM代理程式 (2.2.392.0)	2018 年 3 月 28 日
4.9.2565	• 新版本的SSM代理程式 (2.2.355.0) • 修正 M5 和 C5 執行個體的問題 (找不到 PV 驅動程式) • 新增執行個體類型、最新的 PV 驅動程式和NVMe驅動程式的主控台	2018 年 3 月 13 日
4.9.2549	新版本的SSM代理程式 (2.2.325.0)	2018 年 3 月 8 日
4.9.2461	新版本的SSM代理程式 (2.2.257.0)	2018 年 2 月 15 日
4.9.2439	新版本的SSM代理程式 (2.2.191.0)	2018 年 2 月 6 日
4.9.2400	新版本的SSM代理程式 (2.2.160.0)	2018 年 1 月 16 日

版本	詳細資訊	發行日期
4.9.2327	• 新版本的SSM代理程式 (2.2.120.0) • 在 Amazon EC2 裸機實例上添加了COM端口發現 • 在 Amazon EC2 裸機實例上添加了 Hyper-V 狀態日誌記錄	2018 年 1 月 2 日
4.9.2294	新版本的SSM代理程式 (2.2.103.0)	2017 年 12 月 4 日
4.9.2262	新版本的SSM代理程式 (2.2.93.0)	2017 年 11 月 15 日
4.9.2246	新版本的SSM代理程式 (2.2.82.0)	2017 年 11 月 11 日
4.9.2218	新版本的SSM代理程式 (2.2.64.0)	2017 年 10 月 29 日
4.9.2212	新版本的SSM代理程式 (2.2.58.0)	2017 年 10 月 23 日
4.9.2203	新版本的SSM代理程式 (2.2.45.0)	2017 年 10 月 19 日
4.9.2188	新版本的SSM代理程式 (2.2.30.0)	2017 年 10 月 10 日
4.9.2180	• 新版本的SSM代理程式 (2.2.24.0) • 增加了彈性GPU插件的GPU實例	2017 年 10 月 5 日
4.9.2143	新版本的SSM代理程式 (2.2.16.0)	2017 年 10 月 1 日
4.9.2140	新版本的SSM代理程式 (2.1.10.0)	
4.9.2130	新版本的SSM代理程式 (2.1.4.0)	
4.9.2106	新版本的SSM代理程式 (2.0.952.0)	

版本	詳細資訊	發行日期
4.9.2061	新版本的SSM代理程式 (2.0.922.0)	
4.9.2047	新版本的SSM代理程式 (2.0.913.0)	
4.9.2031	新版本的SSM代理程式 (2.0.902.0)	
4.9.2016	• 新版本的SSM代理程式 (2.0.879.0) • 修正了視窗服務器 2003 的 CloudWatch 日誌目錄路徑	
4.9.1981	• 新版本的SSM代理程式 (2.0.847.0) • 修正在EBS磁碟區`important.txt` 中產生的問題。	
4.9.1964	新版本的SSM代理程式 (2.0.842.0)	
4.9.1951	• 新版本的SSM代理程式 • 修正暫時性硬碟機之磁碟機代號 Z: 無法對應的問題。	
4.9.1925	• 新版本的SSM代理程式 (2.0.822.0) • [錯誤] 這個版本不是SSM代理程式 v4.9.1775 的有效更新目標。	
4.9.1900	新版本的SSM代理程式 (2.0.805.0)	
4.9.1876	• 新版本的SSM代理程式 (2.0.796.0) • 修正管理使用者資料執行的輸出/錯誤重新導向問題。	

版本	詳細資訊	發行日期
4.9.1863	• 新版本的SSM代理程式 (2.0.790.0) • 修復了將多個EBS卷附加到 Amazon EC2 實例的問題。 • 改進 CloudWatch 了採取配置路徑，保持向後兼容性。	
4.9.1791	新版本的SSM代理程式 (2.0.767.0)	
4.9.1775	新版本的SSM代理程式 (2.0.761.0)	
4.9.1752	新版本的SSM代理程式 (2.0.755.0)	
4.9.1711	新版本的SSM代理程式 (2.0.730.0)	
4.8.1676	新版本的SSM代理程式 (2.0.716.0)	
4.7.1631	新版本的SSM代理程式 (2.0.682.0)	
4.6.1579	• 新版本的SSM代理程式 (2.0.672.0) • 修正 v4.3、v4.4 和 v4.5 的代理程式更新問題	
4.5.1534	新版本的SSM代理程式 (2.0.645.1)	
4.4.1503	新版本的SSM代理程式 (2.0.633.0)	
4.3.1472	新版本的SSM代理程式 (2.0.617.1)	
4.2.1442	新版本的SSM代理程式	
4.1.1378	新版本的SSM代理程式 (2.0.558.0)	

版本	詳細資訊	發行日期
4.0.1343	• 執行命令、狀態管理員、 CloudWatch 代理程式和網域加入支援已移至另一個稱為代理程式的代理SSM程式。SSM代理程式將作為EC2Config升級的一部分進行安裝。如需詳細資訊，請參閱EC2Config和 AWS Systems Manager。 • 如果您在中設定了 ProxyEC2Config，則需要在升級之前更新代SSM理程式的 Proxy 伺服器設定。如果未更新代理伺服器設定，將無法使用執行命令管理執行個體。若要避免這種情況，請在更新至較新版本之前，先參閱下列資訊：在 Windows 執行個體上安裝和設定SSM代理程式 (位於AWS Systems Manager 使用手冊)。 • 如果您先前使用本機設定檔 (`AWS.EC2.Windows.CloudWatch.json`) 啟用執行個體的 CloudWatch 整合，則需要將檔案設定為與 SSM Agent 搭配使用。	
3.19.1153	• 對於具有舊 AWS KMS 配置的實例重新啟用激活插件。跳過BYOL用戶的激活。 • 更改默認TRIM行為在磁盤格式化操作期間被禁用，並添加了用FormatWithTRIM於覆蓋帶有 userdata 的InitializeDisks 插件。	
3.18.1118	• 修正以可靠地新增路由表至主要網路轉接器。 • 更新以改善對 AWS 服務的支援。	
3.17.1032	• 修正篩選條件設為相同類別時出現的重複系統日誌。 • 修正以避免磁碟初始化時發生停滯。	
3.16.930	新增在啟動時將「Window is Ready to use」(Window 已可開始使用) 事件記錄至 Windows 事件日誌的支援。	

版本	詳細資訊	發行日期
3.15.880	修正以允許將 Systems Manager 執行命令輸出上傳至名稱含「.」字元的 S3 儲存貯體。	
3.14.786	增加了覆蓋 InitializeDisks 插件設置的支持。例如：要加快SSD磁盤初始化速度，您可以TRIM通過在 userdata 中指定以臨時禁用： <InitializeDrivesSettings><SettingsGroup>FormatWithoutTRIM</SettingsGroup></InitializeDrivesSettings	
3.13.727	Systems Manager RunCommand - 修正以在 Windows 重新開機後可靠地處理命令。	
3.12.649	• 修正以在執行命令/指令碼時從容處理重新開機。 • 修正以從容取消執行命令。 • 透過 Systems Manager 執行命令安裝應用程式時，新增 (選擇性) 將MSI日誌上傳到 S3 的支援。	
3.11.521	• 修正程式以啟用RDP指紋產生的視窗伺服器 2003。 • 修正以在EC2Config記錄行中包含時區和UTC偏移量。 • Systems Manager 支援平行執行執行命令的命令。 • 還原先前的變更，將已分割的磁碟設為上線。	
3.10.442	• 修復安裝MSI應用程序時 Systems Manager 配置失敗。 • 修正以可靠地將儲存磁碟設為上線。 • 更新以改善對 AWS 服務的支援。	

版本	詳細資訊	發行日期
3.9.359	• 修正後期 Sysprep 指令碼，讓 Windows 更新設定保持為預設狀態。 • 修復密碼生成插件以提高獲取GPO密碼策略設置的可靠性。 • 將EC2Config/SSM記錄資料夾權限限制為本機系統管理員群組。 • 更新以改善對 AWS 服務的支援。	
3.8.294	• 修正當不在主要磁碟機上時，無法上傳記錄檔的問題。CloudWatch • 加入重試邏輯，改善磁碟初始化流程。 • 增加了改進了AMI創建過 SetPassword 程中插件偶爾失敗時的錯誤處理。 • 更新以改善對 AWS 服務的支援。	
3.7.308	• 改善 ec2config-cli 公用程式，以在執行個體內進行 config 測試與故障診斷。 • 避免在 O VPN pen 介面卡上為 AWS KMS 和中繼資料服務新增靜態路由。 • 修正使用者資料執行未執行「persist」標籤的問題。 • 改進了無法使用記錄到EC2控制台時的錯誤處理。 • 更新以改善對 AWS 服務的支援。	

版本	詳細資訊	發行日期
3.6.269	• 修正 Windows 啟用可靠性，在透過 AWS KMS啟動 Windows 時先使用連結本機地址 169.254.0.250/251 • 改善 Systems Manager、Windows 啟動和網域加入等狀況下的代理伺服器處理能力 • 修正重複的使用者帳戶行新增到 Sysprep 回答檔案的問題	
3.5.228	• 已解決 CloudWatch 外掛程式可能耗用過多CPU且記憶體讀取 Windows 事件記錄檔的案例 • 在EC2Config設置 UI 中添加了 CloudWatch 配置文檔的鏈接	
3.4.212	• 與 VM 匯入搭配使用EC2Config時的修正。 • 修正 WiX 安裝程式內的服務命名問題。	
3.3.174	• 改善 Systems Manager 和網域加入問題的例外處理能力。 • 變更以支援 Systems Manager SSM 架構版本控制。 • 修正 Win2K3 上的暫時性磁碟格式化問題。 • 支援設定超過 2TB 磁碟大小的變更。 • 設定 GC 模式為預設值，減少虛擬記憶體用量。 • Support 從UNC路徑`aws:psModule` 和`aws:application` 插件下載工件。 • 改善 Windows 啟用外掛程式的記錄功能。	

版本	詳細資訊	發行日期
3.2.97	• 延遲載入 Systems Manager SSM 組件的效能改善。 • 改善不正確之 sysprep2008.xml 的例外處理能力。 • Systems Manager「套用」設定的命令列支援。 • 支援有電腦重新命名等待時加入網域的變更。 • 支援 `aws:applications` 外掛程式中的選擇性參數。 • 支援 `aws:psModule` 外掛程式中的命令陣列。	
3.0.54	• 啟用 Systems Manager 的支援。 • 透過 Systems Manager 自動將 EC2 Windows 執行個體加入 AWS 目錄的網域。 • 透過 Systems Manager 設定及上傳 CloudWatch 記錄/指標。 • 通過 Systems Manager 安裝 PowerShell 模塊。 • 透過 Systems Manager 安裝MSI應用程式	
2.4.233	• 添加了計劃任務以EC2Config從服務啟動失敗中恢復。 • 改善主控台日誌錯誤訊息。 • 更新以改善對 AWS 服務的支援。	

版本	詳細資訊	發行日期
2.3.313	• 修正啟用記 CloudWatch 錄功能時，某些情況下會耗用大量記憶體的問題。 • 修正升級錯誤，允許 ec2config 2.1.19 以下版本現在可升級至最新版。 • 更新了COM端口打開異常，以便在日誌中更友好和有用。 • EC2 configServiceSettings UI 禁用了調整大小，並修復了 UI 中的歸因和版本顯示位置。	
2.2.12	• NullPointerException 在查詢用於確定偶爾返回空值的 Windows Sysprep 狀態的註冊表項時進行處理。 • 發佈 finally 區塊中的非受管資源。	
2.2.11	修正 CloudWatch 外掛程式中處理空白記錄行的問題。	
2.2.10	• 已移除透過 UI 設定 CloudWatch 記錄設定。 • 讓使用者能夠在`%ProgramFiles%\Amazon\Ec2ConfigService\Settings\AWS.EC2.Windows.CloudWatch.json` 檔案中定義 CloudWatch 記錄檔設定，以允許 future 的增強。	
2.2.9	修正未處理的例外狀況並新增記錄功能。	
2.2.8	• 修復EC2Config安裝程序中的 Windows 操作系統版本檢查以支持 Windows 服務器 2003 SP1 及更高版本。 • 修正為更新 Sysprep 組態檔而讀取登錄機碼時的 null 值處理能力。	

版本	詳細資訊	發行日期
2.2.7	• 增加了對在 Windows 2008 及EC2Config更高版本的系統執行期間運行的支持。 • 改善例外處理和記錄功能，以提升診斷效率	
2.2.6	• 減少上傳記錄事件時執行個體和 CloudWatch 記錄檔上的負載。 • 解決 CloudWatch 記錄外掛程式未始終保持啟用狀態的升級問題	
2.2.5	• 增加了將日誌上傳到日 CloudWatch 誌服務的支持。 • 修正了 Ec2 utputRDPCert O 外掛程式中的競爭狀態問題 • 將EC2Config服務恢復選項更改為重新啟動 TakeNoAction • 在EC2Config崩潰時添加了更多異常信息	
2.2.4	• 修正了 .cmd 中的一個錯字 PostSysprep • 修正了EC2Config不會將自己固定到 OS2 012+ 的開始菜單上的錯誤	

版本	詳細資訊	發行日期
2.2.3	• 增加了選項，EC2Config無需服務安裝後立即開始安裝。若要使用，請從命令提示執行「Ec2Install.exe start=false」 • 新增桌布外掛程式中用於控制新增/移除桌布的參數。要使用，請從命令提示符下運行「EC2 WallpaperInfo .exe 集」或「EC2 WallpaperInfo .exe 還原」 • 添加了對 RealTimeIsUniversal 鍵的檢查，將 RealTimeI sUniveral 註冊表項的不正確設置輸出到控制台 • 刪除了對 Windows 臨時文件夾的EC2Config依賴 • 移除了對 .Net 3.5 的 UserData 執行相依性	
2.2.2	• 新增對服務停止行為的檢查，以檢查資源是否發佈 • 修正加入網域時執行時間過長的問題	
2.2.1	• 更新安裝程式，以允許從舊版升級 • 修復了 .Net4.5 唯一環境中的 EC2 WallpaperInfo 錯誤 • 修正間歇性的驅動程式偵測錯誤 • 新增靜音安裝選項。執行 Ec2Install.exe 並加入「-q」選項。例如：「Ec2Install.exe -q」	
2.2.0	• 新增對僅限於 .Net4 和 .Net4.5 環境的支援 • 更新安裝程式	

版本	詳細資訊	發行日期
2.1.19	• 新增使用 Intel 網路驅動程式時的暫時性磁碟標籤支援 (例如 C3 執行個體類型)。如需詳細資訊，請參閱Amazon EC2 執行個體上的增強聯網。 • 在控制台輸出中添加了AMIAMI原始版本和原始名稱支持 • 變更主控台輸出，以執行一致的格式化/剖析 • 更新說明檔案	
2.1.18	• 添加了EC2ConfigWMI對象完成通知（-命名空間根\ Amazon-Class _EC2）ConfigService • 改善使用大型事件記錄的啟動WMI查詢效能；可能會CPU在初始執行期間造成長時間過高	
2.1.17	• 修復了標準輸出和標準錯誤緩衝填充的 UserData 執行問題 • 修復了 >= w2k8 操作系統的控制台輸出中有時出現不正確的 RDP指紋 • 控制台輸出現在包含 Windows 2008 + 的「RDPCERT IFICATE-SubjectName:」，其中包含機器名稱值 • 新增 D:\ 至磁碟機代號映射下拉式清單 • 移動說明按鈕至右上角並變更外觀/使用體驗 • 新增意見回饋問卷連結至右上角	

版本	詳細資訊	發行日期
2.1.16	• 常規選項卡包括鏈接到新版本的EC2Config下載頁面 • 桌面牆紙覆蓋現在存儲在用戶本地 Appdata 文件夾中，而不是我的文檔中，以支持 MyDoc重定向 • MSSQLServer系統後腳本中的名稱與系統同步（2008+） • 重新排序應用程式資料夾 (將檔案移至 Plugin 目錄並移除重複的檔案) • 變更系統日誌輸出 (主控台)： • *改為日期、名稱、值格式，以方便剖析 (請開始將相依性遷移至新格式) • * 增加了「EC2SetPassword」插件狀態 • *新增 Sysprep 開始和結束時間 • 修正非英文版作業系統上暫時性磁碟未標籤為「Temporary Storage」的問題 • 修復了運行 Sysprep 後EC2Config卸載失敗的問題	

版本	詳細資訊	發行日期
2.1.15	• 將中繼資料服務請求最佳化 • 中繼資料現在將繞過代理伺服器設定 • 暫時性磁碟已標籤為「Temporary Storage」，且找到的 Important.txt 將放入磁碟區 (僅限 Citrix PV 驅動程式)。如需詳細資訊，請參閱升級 EC2 Windows 執行個體上的光伏驅動。 • 暫時性磁碟將指派 Z 到 A 的代號 (僅限 Citrix PV 驅動程式) - 指派可用磁碟機代號映射外掛程式覆寫，覆寫為磁碟區標籤「Temporary Storage X」(其中 X 為 0-25 的數字) • UserData 現在立即在「窗口已準備就緒」之後運行	
2.1.14	桌面桌布修正	
2.1.13	• 桌面桌布預設將顯示主機名稱 • 移除對 Windows 時間服務的相依性 • 在將多個分配給單個接口IPs的情況下添加路由	
2.1.11	• 變更 Ec2Activation 外掛程式 • -每隔 30 天檢查一次啟用狀態 • -如果寬限期仍有 90 天 (共 180 天)，則重新嘗試啟用	

版本	詳細資訊	發行日期
2.1.10	• 桌面桌布疊加層不再與 Sysprep 或不使用 Sysprep 關機一致 • 在開頭為 <persist>true</persist> 的每個服務上執行的使用者資料選項 • 將/.cmd 的位置和名稱更改為/腳本/DisableWinUpdate.cmd PostSysprep • 在 /Script PostSysprep /.cmd 中，系統管理員密碼預設為不過期 • 解除安裝會從 c:\windows\setup\script\ CommandComplete .cmd 移除EC2Config PostSysprep 指令碼 • 新增路由表對自訂介面指標的支援	
2.1.9	UserData 執行不再限制為 3851 個字元	

版本	詳細資訊	發行日期
2.1.7	• OS 版本和語言識別符寫入至主控台 • EC2Config版本寫入控制台 • PV 驅動程式版本寫入至主控台 • 偵測錯誤檢查，並在找到錯誤後於下次開機時輸出至主控台 • 新增選項至 config.xml，以維持 Sysprep 登入資料 • 在開始時不可用的ENI情況下添加路由重試邏輯 • 用戶數據執行PID寫入控制台 • 從中擷取的最小產生密碼長度 GPO • 設定服務啟動重試 3 次 • 增加了 S3_ DownloadFile .ps1 和 S3_上傳文件.ps1的例子到/腳本文件夾	

版本	詳細資訊	發行日期
2.1.6	• 新增版本資訊至 General (一般) 索引標籤 • Bundle 索引標籤重新命名為 Image (影像) • 簡化指定密碼的流程，並將密碼相關 UI 從 General (一般) 索引標籤移至 Image (影像) 索引標籤 • Disk Settings (磁碟設定) 索引標籤重新命名為 Storage (儲存) • 新增 Support (支援) 索引標籤，並提供可執行故障排除的常用工具 • Windows Server 2003 `sysprep.ini` 設為預設擴充 OS 分割區 • 新增私有 IP 地址至桌布 • 私有 IP 地址將在桌布上顯示 • 新增主控台輸出的重試邏輯 • 修復了元數據可訪問性的 Com 端口異常-導EC2Config致在顯示控制台輸出之前終止 • 每次開機時檢查啟用狀態 -- 視需要啟動 • 修正相關路徑的問題 -- 發生在手動從啟動資料夾執行桌布捷徑時；指向管理員/日誌 • 修正 Windows Server 2003 使用者 (管理員除外) 的預設背景色彩	

版本	詳細資訊	發行日期
2.1.2	• (祖魯語UTC) 中的主控台時間戳記 • 移除 Sysprep 索引標籤上的超連結顯示 • 新增在 Windows 2008+ 第一次開機時動態擴充根磁碟區的功能 • 啟用設置密碼後，現在會自動啟EC2Config用設置密碼 • EC2Config在執行 Sysprep 之前檢查啟用狀態 (如果未啟動，則會顯示警告) • 視窗服務器 2003 `Sysprep.xml` 現在默認為UTC時區，而不是太平洋 • 隨機啟動伺服器 • Drive Mapping (磁碟機映射) 索引標籤重新命名為 Disk Settings (磁碟設定) • Initialize Drives UI (初始化磁碟機 UI) 項目從 General (一般) 索引標籤移至 Disk Settings (磁碟設定) 索引標籤 • 幫助按鈕現在指向HTML幫助文件 • 更新的HTML幫助文件與變化 • 更新磁碟機代號映射的「備註」文字 • 在 Sysprep 之前將 InstallUpdates .ps1 添加到 /Script 文件夾中，用於自動執行修補程序和清理	

版本	詳細資訊	發行日期
2.1.0	• 桌面桌布預設在第一次登入 (未中斷/重新連接) 後顯示執行個體資訊 • PowerShell 可以通過圍繞代碼來從用戶數據中運行<powershell></powershell>	

為您的 Windows 執行個體使用EC2快速啟動

當您設定適用AMI於EC2快速啟動的 Windows Server 時，Amazon EC2 會建立一組預先佈建的快照，以便加快啟動速度，如下所示。

1. Amazon 根據您的設置EC2啟動一組臨時 t3 實例。
2. 當每個臨時執行個體完成標準啟動步驟時，Amazon EC2 會建立執行個體的預先佈建快照。它會將快照存放在您的 Amazon S3 儲存貯體中。
3. 快照準備就緒後，Amazon EC2 會終止相關聯的 t3 執行個體，以盡可能降低資源成本。
4. Amazon 下次從EC2啟用的EC2快速啟動啟動啟動執行個體時AMI，會使用其中一個快照來大幅縮短啟動所需的時間。

Amazon EC2 會自動補充您現有的快照，因為它們使用它們從啟用的EC2快速啟動啟動啟AMI動執行個體。

任何可存取已啟用EC2快速啟動功能AMI的帳戶，都可以受益於縮短的啟動時間。當AMI擁有者授予您啟動執行個體的存取權時，預先佈建的快照會來自AMI擁有者的帳戶。

如果與您共AMI用支援「EC2快速啟動」的功能，您可以AMI自行啟用或停用共用的快速啟動。如果您AMI為EC2快速啟動啟用共用功能，Amazon EC2 會直接在您的帳戶中建立預先佈建的快照。如果您耗盡帳戶中的快照，您仍然可以使用AMI擁有者帳戶中的快照。

> **Note**
>
> EC2Fast Launch 會在啟動時立即刪除預先佈建的快照，以將儲存成本降至最低並防止重複使用。不過，如果刪除的快照符合保留規則，資源回收筒會自動保留它們。建議您檢閱資源回收筒保留規則的範圍，以免發生此類情況。如需詳細資訊，請參閱 Amazon EBS 使用者指南中的資源回收筒。

此功能與EBS快速快照還原不同。您必須以每個EBS快照為基礎明確啟用快速快照還原功能，而且它有自己的相關費用。

下列影片示範如何透過相關關鍵術語及其定義的快速概觀來設定 Windows AMI 以加快啟動速度：啟動 EC2 Windows 執行個體的速度提高 65% AWS。

資源成本

將 Windows 設定AMIs為「EC2快速啟動」無須支付任何服務費用。但是，標準定價適用於 Amazon EC2 使用的任何基礎 AWS 資源。若要進一步了解相關資源成本以及如何管理這些成本，請參閱 管理EC2快速啟動基礎資源的成本。

目錄

- 重要用語
- EC2視窗的快速啟動先決條件
- 為您的 Amazon EC2 視窗伺服器設定EC2快速啟動設定 AMI
- 啟AMIs用EC2快速啟動的檢視
- 管理EC2快速啟動基礎資源的成本
- 監控EC2快速啟動
- EC2快速啟動的服務連結角色

重要用語

EC2快速啟動功能使用下列主要術語：

已預先佈建的快照

從啟用「快速啟動」的 Windows 啟動執行個體AMI的EC2快照集，並且已完成下列 Windows 啟動步驟，視需要重新開機。

- Sysprep 特製化
- 窗戶開箱體驗 (OOBE)

完成這些步驟後，EC2Fast Launch 會停止執行個體，並建立一個快照AMI，以便稍後根據您的組態從快速啟動。

啟動頻率

控制 Amazon EC2 可在指定時間範圍內啟動的預先佈建快照數量。當您為您的啟用EC2快速啟動時AMI，Amazon EC2 會在背景中建立一組預先佈建的快照。例如，如果啟動頻率設定為每小時啟動五次 (預設值)，則EC2快速啟動會建立一組初始設定的五個預先佈建快照。

Amazon 從啟用EC2快速EC2啟動的執行個體啟動執行個體時，會使用其中一個預先佈建的快照來縮短啟動時間。AMI使用快照時，它們會自動補充，直到達到啟動頻率指定的數量。

如果您預期從您的執行個體啟動的數量 (例如，在特殊活動期間) 會激增，您AMI可以提前增加啟動頻率，以涵蓋您需要的其他執行個體。當您的啟動速率恢復正常時，則可以將頻率降低回原始值。

當您遇到啟動次數高於預期時，則可能會用到所有可用的預先佈建快照。這不會引發任何啟動失敗。但是，它可能會使某些執行個體經歷標準啟動過程，直到快照得到補充為止。

目標資源計數

已啟用快速啟動的 Amazon EC2 Windows 伺服器AMI可保留的預先佈建EC2快照數量。

最大平行啟動數

控制 Amazon EC2 可同時啟動多少個執行個體，以建立用於EC2快速啟動的預先佈建快照。如果您的目標資源計數高於您設定的 parallel 啟動上限，Amazon 會啟動 Max 並行EC2啟動指定的執行個體數量，以開始建立快照。當這些執行個體完成程序時，Amazon EC2 會擷取快照並停止執行個體。然後，它會繼續啟動更多執行個體，直到可用的快照總數達到目標資源計數為止。最大平行啟動數的值必須為 6 或更大。

EC2視窗的快速啟動先決條件

在設定EC2快速啟動之前，請確認您已符合下列必要條件，以便AMIs在中建立快照 AWS 帳戶：

- 如果您不使用啟動範本來設定設定，請確定已為您使用EC2快速啟動的區域設定預VPC設值。

> **Note**
>
> 如果您不小心VPC在計劃設定EC2快速啟動的「區域」中刪除了預設值，您可以VPC在該區域中建立新的預設值。若要進一步了解，請參閱 Amazon VPC 使用者指南VPC中的建立預設值。

- 若要指定非預設值VPC，您必須在設定 Windows 快速啟動時使用啟動範本。如需詳細資訊，請參閱設定EC2快速啟動時使用啟動範本。

- 如果您的帳戶包含IMDSv2針對 Amazon 執行個EC2體強制執行的政策，則必須建立啟動範本，以指定要強制執行IMDSv2的中繼資料組態。
- 私人EC2快速啟動AMIs必須支援使用者資料指令碼執行。
- 若要設定的EC2快速啟動AMI，您必須建立AMI使用Sysprep與關機選項。EC2快速啟動功能目前不支援從執AMIs行中執行個體建立的功能。

  若要建立AMI使用Sysprep，請參閱EC2AMI使用視窗系統創建一個 Amazon。
- 每個區域中最大 parallel 啟動AMIs的預設配額 AWS 帳戶 為 40。您可為自己的帳戶請求增加 Service Quotas，如下所示。

  1. 登入 AWS Management Console 並開啟 Service Quotas 主控台，位於https://console.aws.amazon.com/servicequotas/。
  2. 在導覽窗格中，選擇 AWS 服務。
  3. 在搜尋列中輸入 `EC2 Fast Launch`，然後選取結果。
  4. 選取 `Parallel instance launches` 的連結。這會將帶您到平行執行個體啟動數服務配額詳細資訊頁面。
  5. 選擇 Request quota increase (請求增加配額)。

  如需詳細資訊，請參閱《Service Quotas 使用者指南》中的請求提高配額。

為您的 Amazon EC2 視窗伺服器設定EC2快速啟動設定 AMI

您可以為您擁有的 Windows AMIs 設定EC2快速啟動AMIs，或從、 AWS Management Console、APISDKs、 CloudFormation或 AWS Command Line Interface (AWS CLI) 與您共用的快速啟動。設定EC2快速啟動之前，請確認您AMI符合建立預先佈建快照所需的所有必要條件。如需詳細資訊，請參閱 EC2視窗的快速啟動先決條件。

當您為 Windows 執行個體啟用更快的啟動時，Amazon 會EC2檢查以確保您具有從指定的啟動範本AMI和啟動範本 (如有提供) 啟動執行個體的必要許可，包括加密的許可AMIs。為了避免執行個體啟動程序期間發生錯誤，服務會在啟用EC2快速啟動之前驗證您的權限。如果您沒有必要的權限，服務會傳回錯誤，而且不會啟用EC2快速啟動。

EC2快速啟動與 EC2 Image Builder 整合，協助您在啟用EC2快速啟動的情況下建立自訂映像檔。如需詳細資訊，請參閱 EC2Image Builder 生器使AMI用指南中的為已啟用EC2快速啟動的 Windows 建立發佈設定 (AWS CLI)。

以下各節涵蓋 Amazon EC2 主控台和的設定步驟 AWS CLI。

啟用EC2快速啟動

若要啟用EC2快速啟動，請選擇符合您環境的索引標籤，然後按照步驟操作。

> **Note**
> 在更改這些設置之前，請確保您AMI的和您運行的區域符合所有內容EC2視窗的快速啟動先決條件。

Console

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格的 [影像] 下，選擇AMIs。
3. 選取「AMI名稱」旁邊的核取方塊，以選擇要更新的。
4. 從清單上方的「動作」功能表中AMIs，選擇「設定快速啟動」。這會開啟 [設定快速啟動] 頁面，您可以在其中設定 [EC2快速啟動] 的設定。
5. 若要開始使用預先佈建的快照以AMI更快速地從 Windows 啟動執行個體，請選取 [啟用 Windows 的快速啟動] 核取方塊。
6. 從 Set anticipated launch frequency (設定預期啟動頻率) 下拉式清單中，選擇一個值以指定為涵蓋預期執行個體啟動磁碟區而建立和維護的快照數量。
7. 修改完成後，請選擇 Save changes (儲存變更)。

> **Note**
> 如果您需要使用啟動範本來指定非預VPC設或設定的中繼資料設定IMDSv2，請參閱設定EC2快速啟動時使用啟動範本。

AWS CLI

該enable-fast-launch命令調用 Amazon EC2 EnableFastLaunchAPI操作。

語法:

```
aws ec2 enable-fast-launch \
```

```
  --image-id <value> \
  --resource-type <value> \ (optional)
  --snapshot-configuration <value> \ (optional)
  --launch-template <value> \ (optional)
  --max-parallel-launches <value> \ (optional)
  --dry-run | --no-dry-run \  (optional)
  --cli-input-json <value> \ (optional)
  --generate-cli-skeleton <value> \ (optional)
```

範例：

下列enable-fast-launch範例會針對指定的啟動六個 parallel 執行個體啟用EC2快速啟動AMI，以進行預先佈建。`ResourceType` 設定為 `snapshot`，這是預設值。

```
aws ec2 enable-fast-launch \
         --image-id ami-01234567890abcedf  \
         --max-parallel-launches 6 \
         --resource-type snapshot
```

輸出：

```
{
     "ImageId": "ami-01234567890abcedf",
     "ResourceType": "snapshot",
     "SnapshotConfiguration": {
         "TargetResourceCount": 10
     },
     "LaunchTemplate": {},
     "MaxParallelLaunches": 6,
     "OwnerId": "0123456789123",
     "State": "enabling",
     "StateTransitionReason": "Client.UserInitiated",
     "StateTransitionTime": "2022-01-27T22:16:03.199000+00:00"
 }
```

PowerShell

指Enable-EC2FastLaunch令程式會呼叫 Amazon EC2 EnableFastLaunchAPI作業，以在 Windows AMI 上啟用EC2快速啟動。

語法:

```
Enable-EC2FastLaunch
      -ImageId <String>
      -LaunchTemplate_LaunchTemplateId <String>
      -LaunchTemplate_LaunchTemplateName <String>
      -MaxParallelLaunch <Int32>
      -ResourceType <String>
      -SnapshotConfiguration_TargetResourceCount <Int32>
      -LaunchTemplate_Version <String>
      -Select <String>
      -PassThru <SwitchParameter>
      -Force <SwitchParameter>
```

範例：

下列Enable-EC2FastLaunch範例會針對指定的啟動六個 parallel 執行個體啟用EC2快速啟動AMI，以進行預先佈建。ResourceType 設定為 snapshot，這是預設值。

```
Enable-EC2FastLaunch `
 -ImageId ami-01234567890abcedf `
 -MaxParallelLaunch 6 `
 -Region us-west-2 `
 -ResourceType snapshot
```

輸出：

```
ImageId               : ami-01234567890abcedf
LaunchTemplate        :
MaxParallelLaunches   : 6
OwnerId               : 0123456789123
ResourceType          : snapshot
SnapshotConfiguration : Amazon.EC2.Model.FastLaunchSnapshotConfigurationResponse
State                 : enabling
StateTransitionReason : Client.UserInitiated
StateTransitionTime   : 2/25/2022 12:24:11 PM
```

停用EC2快速啟動

若要停用EC2快速啟動，請選擇符合您環境的索引標籤，然後依照下列步驟執行。

> **Note**
> 在更改這些設置之前，請確保您AMI的和您運行的區域符合所有內容EC2視窗的快速啟動先決條件。

Console

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格的 [影像] 下，選擇AMIs。
3. 選取「AMI名稱」旁邊的核取方塊，以選擇要更新的。
4. 從清單上方的「動作」功能表中AMIs，選擇「設定快速啟動」。這會開啟 [設定快速啟動] 頁面，您可以在其中設定 [EC2快速啟動] 的設定。
5. 清除 [啟用 Windows 快速啟動] 核取方塊可停用EC2快速啟動並移除預先佈建的快照。這會導致未AMI來為每個執行個體使用標準啟動程序。

   > **Note**
   > 停用 Windows 映像最佳化時，將自動刪除所有現存的預先佈建快照。必須完成此步驟，然後才能再次開始使用該功能。

6. 修改完成後，請選擇 Save changes (儲存變更)。

AWS CLI

該disable-fast-launch命令調用 Amazon EC2 DisableFastLaunchAPI操作。

語法:

```
aws ec2 disable-fast-launch \
  --image-id <value> \
  --force | --no-force \ (optional)
  --dry-run | --no-dry-run \ (optional)
  --cli-input-json <value> \ (optional)
  --generate-cli-skeleton <value> \ (optional)
```

範例：

下列disable-fast-launch範例會停用指定的EC2快速啟動AMI，並清除現有預先佈建的快照。

```
aws ec2 disable-fast-launch \
         --image-id ami-01234567890abcedf
```

輸出：

```
{
     "ImageId": "ami-01234567890abcedf",
     "ResourceType": "snapshot",
     "SnapshotConfiguration": {},
     "LaunchTemplate": {
         "LaunchTemplateId": "lt-01234567890abcedf",
         "LaunchTemplateName": "EC2FastLaunchDefaultResourceCreation-
a8c6215d-94e6-441b-9272-dbd1f87b07e2",
         "Version": "1"
     },
     "MaxParallelLaunches": 6,
     "OwnerId": "0123456789123",
     "State": "disabling",
     "StateTransitionReason": "Client.UserInitiated",
     "StateTransitionTime": "2022-01-27T22:47:29.265000+00:00"
 }
```

PowerShell

指Disable-EC2FastLaunch令程式會呼叫 Amazon EC2 DisableFastLaunchAPI作業。

語法:

```
Disable-EC2FastLaunch
      -ImageId <String>
      -ForceStop <Boolean>
      -Select <String>
      -PassThru <SwitchParameter>
      -Force <SwitchParameter>
```

範例：

下列Disable-EC2FastLaunch範例會停用指定的EC2快速啟動AMI，並清除現有預先佈建的快照。

```
Disable-EC2FastLaunch -ImageId ami-01234567890abcedf
```

輸出：

```
ImageId               : ami-01234567890abcedf
LaunchTemplate        :
Amazon.EC2.Model.FastLaunchLaunchTemplateSpecificationResponse
MaxParallelLaunches   : 6
OwnerId               : 0123456789123
ResourceType          : snapshot
SnapshotConfiguration :
State                 : disabling
StateTransitionReason : Client.UserInitiated
StateTransitionTime   : 2/25/2022 1:10:08 PM
```

設定EC2快速啟動時使用啟動範本

使用啟動範本，您可以設定 Amazon 每次從該範本啟動執行個體時EC2使用的一組啟動參數。您可以指定基本映像檔、執行個AMI體類型、儲存空間、網路設定等項目。

啟動範本是選擇性的，但下列特定情況除外，當您設定更快的啟動AMI時，您必須為 Windows 使用啟動範本：

- 您必須使用啟動範本來為 Windows AMI 指定非VPC預設值。
- 如果您的帳戶包含IMDSv2針對 Amazon 執行個EC2體強制執行的政策，則必須建立啟動範本，以指定要強制執行IMDSv2的中繼資料組態。

  使用包含EC2主控台中繼資料組態的啟動範本，或者當您在中執行enable-fast-launch指令或呼叫EnableFastLaunchAPI動作時。 AWS CLI

使用啟動範本時，Amazon EC2 EC2 快速啟動不支援下列組態。如果您使用「EC2快速啟動」的啟動範本，則不得指定下列任何項目：

- 使用者資料指令碼
- 終止保護
- 停用的詮釋
- 定點選項
- 終止執行處理的關閉行為
- 網路介面、彈性圖形或 Spot 執行個體請求的資源標籤

指定非預設值 VPC

步驟 1：建立啟動範本

建立指定 Windows 執行個體下列詳細資訊的啟動範本：

- 子VPC網路。
- `t3.xlarge` 的執行個體類型。

如需詳細資訊，請參閱 創建一個 Amazon EC2 啟動模板。

步驟 2：指定EC2快速啟動的啟動範本 AMI

選擇符合您程序的索引標籤：

Console

若要從指定EC2快速啟動的啟動範本 AWS Management Console，請依照下列步驟執行：

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格的 [影像] 下，選擇AMIs。
3. 選取「AMI名稱」旁邊的核取方塊，以選擇要更新的。
4. 從清單上方的「動作」功能表中AMIs，選擇「設定快速啟動」。這會開啟 [設定快速啟動] 頁面，您可以在其中設定 [EC2快速啟動] 的設定。
5. 此啟動範本方框會執行篩選搜尋，在您帳戶目前區域中搜尋與您輸入文字相符的啟動範本。在方框中指定所有或部分的啟動範本名稱或 ID，以比對出相符的啟動範本清單。例如，如果您在框`fast`中輸入，Amazon 會在當前區域中EC2找到您帳戶中名稱中具有「快速」的所有啟動模板。

   若要新建一個啟動範本，請選擇 Create launch template (建立啟動範本)。
6. 當您選擇啟動模板時，Amazon 會在「源模板版本」框中EC2顯示該模板的默認版本。若要指定其他版本，請反白要替換的預設版本，然後在方框中輸入所需的版本編號。
7. 修改完成後，請選擇 Save changes (儲存變更)。

AWS CLI, API

若要從中指定EC2快速啟動的啟動範本 AWS CLI，請在中執行enable-fast-launch命令時，在`--launch-template`參數中指定啟動範本名稱或 ID AWS CLI。

若要在API請求中指定「EC2快速啟動」的啟動範本，請在呼叫動作時在`LaunchTemplate`參數中指定啟EnableFastLaunchAPI動範本名稱或 ID。

如需EC2啟動範本的詳細資訊，請參閱在 Amazon 啟動範本中存放執行個體EC2啟動參數。

啟AMIs用EC2快速啟動的檢視

您可以使用中的describe-fast-launch-images命令或 PowerShell Cmdlet 的Get-EC2FastLaunchImage工具來取得已啟用EC2快速啟動AMIs的詳細資料。 AWS CLI

Amazon EC2 針對結果中傳回的每AMI個視窗提供下列詳細資訊：

- 啟用「EC2快速啟動」的影像 ID。AMI
- 用於預先佈建相關聯 Windows AMI 的資源類型。支援的值為：`snapshot`。
- 快照組態，這是一組參數，可AMI使用快照為相關聯的 Windows 設定預先佈建。
- 啟動範本資訊，包括相關聯從預先佈建的快照啟動 Window 執行個體時所AMI使用的啟動範本 ID、名稱和版本。
- 為建立資源而同時啟動的最大執行個體數。
- 相關聯的擁有者 ID AMI。這不會填入與AMIs您共用的項目。
- 關聯之「EC2快速啟動」的目前狀態AMI。支援的值包括：`enabling | enabling-failed | enabled | enabled-failed | disabling | disabling-failed`。

> **Note**
>
> 您也可以在EC2主控台的 [管理影像最佳化] 頁面中查看目前顯示的狀態為 [影像最佳化] 狀態。

- 關聯的「EC2快速啟動」AMI 變更為目前狀態的原因。
- 相關聯的「EC2快速啟動」AMI 變更為目前狀態的時間。

選擇與您的命令列環境相符的索引標籤：

AWS CLI

該describe-fast-launch-images命令調用 Amazon EC2 DescribeFastLaunchImagesAPI操作。

語法:

```
aws ec2 describe-fast-launch-images \
 --image-ids <value> \ (optional)
 --filters <value> \ (optional)
 --dry-run | --no-dry-run \ (optional)
 --cli-input-json <value> \ (optional)
 --starting-token <value> \ (optional)
 --page-size <value> \ (optional)
 --max-items <value> \ (optional)
 --generate-cli-skeleton <value> \ (optional)
```

範例：

下列describe-fast-launch-images範例說明設定為「EC2快速啟動」之帳戶AMIs中每個項目的詳細資料。在此範例中，帳戶AMI中只有一個設定為「EC2快速啟動」。

```
aws ec2 describe-fast-launch-images
```

輸出：

```
{
    "FastLaunchImages": [
        {
            "ImageId": "ami-01234567890abcedf",
            "ResourceType": "snapshot",
            "SnapshotConfiguration": {},
            "LaunchTemplate": {
                "LaunchTemplateId": "lt-01234567890abcedf",
                "LaunchTemplateName": "EC2FastLaunchDefaultResourceCreation-
a8c6215d-94e6-441b-9272-dbd1f87b07e2",
                "Version": "1"
            },
            "MaxParallelLaunches": 6,
            "OwnerId": "0123456789123",
            "State": "enabled",
            "StateTransitionReason": "Client.UserInitiated",
            "StateTransitionTime": "2022-01-27T22:20:06.552000+00:00"
        }
    ]
}
```

Tools for PowerShell

指Get-EC2FastLaunchImage令程式會呼叫 Amazon EC2 DescribeFastLaunchImagesAPI作業。

語法:

```
Get-EC2FastLaunchImage
-Filter <Filter[]>
-ImageId <String[]>
-MaxResult <Int32>
-NextToken <String>
-Select <String>
-NoAutoIteration <SwitchParameter>
```

範例：

下列Get-EC2FastLaunchImage範例說明設定為「EC2快速啟動」之帳戶AMIs中每個項目的詳細資料。在此範例中，帳戶AMI中只有一個設定為「EC2快速啟動」。

```
Get-EC2FastLaunchImage -ImageId ami-01234567890abcedf
```

輸出：

```
ImageId               : ami-01234567890abcedf
LaunchTemplate        :
 Amazon.EC2.Model.FastLaunchLaunchTemplateSpecificationResponse
MaxParallelLaunches   : 6
OwnerId               : 0123456789123
ResourceType          : snapshot
SnapshotConfiguration :
State                 : enabled
StateTransitionReason : Client.UserInitiated
StateTransitionTime   : 2/25/2022 12:54:43 PM
```

管理EC2快速啟動基礎資源的成本

將 Windows 設定AMIs為「EC2快速啟動」無須支付任何服務費用。但是，當您為 Amazon EC2 Windows 啟用EC2快速啟動時AMI，Amazon 用於準備和存放預先佈建快照的基礎 AWS 資源適EC2用標準定價。您可以設定成本配置標籤，以協助您追蹤和管理與EC2快速啟動資源相關的成本。如需如何設定成本分配標籤的詳細資訊，請參閱 追蹤帳單上的EC2快速啟動成本。

下列範例示範如何配置與EC2快速啟動快照成本相關的成本。

範例案例：AtoZ 範例公司有一個具有 50 GiB EBS 根磁碟區的視窗AMI。他們為他們啟用EC2快速啟動AMI，並將目標資源計數設置為五。在一個月的過程中，使用EC2快速啟動他們的AMI成本他們在 5.00 美元左右，成本細分如下：

1. 當 AtoZ 範例啟用EC2快速啟動時，Amazon EC2 會啟動五個小型執行個體。每個執行個體都會執行 Sysprep 和 OOBE Windows 啟動步驟，並視需要重新開機。每個執行個體需要幾分鐘的時間 (時間可能會有所不同，視該區域或可用區域 (AZ) 的忙碌程度以及的大小而定AMI)。

   成本

   - 執行個體執行時間成本 (或最低執行時間，如果適用)：五個執行個體
   - 磁碟區成本：五個EBS根磁碟區
2. 當預先佈建程序完成時，Amazon 會EC2拍攝執行個體的快照，並將其存放在 Amazon S3 中。快照通常會存放 4 到 8 小時，然後才會啟動使用。在這種情況下，每個快照的成本大約為 0.02 到 0.05 USD。

   成本

   - 快照儲存 (Amazon S3)：五個快照
3. Amazon EC2 拍攝快照後，它會停止實例。此時，執行個體將不再累計成本。然而，數EBS量成本繼續累積。

   成本

   - EBS磁碟區：相關EBS根磁碟區的成本持續。

> **Note**
>
> 此處顯示的費用僅供示範用途。您的費用會根據您的AMI組態和定價方案而有所不同。

追蹤帳單上的EC2快速啟動成本

成本配置標籤可協助您組織 AWS 帳單，以反映與「EC2快速啟動」相關的成本。當 Amazon 準備和存放預先佈建的快照以供EC2快速啟動時，您可以使用 Amazon EC2 新增至其建立的資源中的下列標籤：

標籤索引鍵：`CreatedBy`，值：`EC2 Fast Launch`

在帳單與成本管理主控台中啟用標籤並設定詳細的帳單報告後，`user:CreatedBy` 欄出現在報告中。此欄包含來自所有服務的值。但是，如果您下載CSV檔案，則可以將資料匯入試算表，並在值`EC2 Fast Launch`中篩選。啟動標籤 AWS Cost and Usage Report 時，此資訊也會顯示在中。

步驟 1：啟用使用者定義的成本分配標籤

若要在成本報告中包含資源標籤，您必須先在帳單與成本管理主控台中啟用該標籤。如需詳細資訊，請參閱《AWS Billing and Cost Management 使用者指南》中的啟用使用者定義的成本分配標籤。

> **Note**
>
> 啟用最多可能需要 24 小時。

步驟 2：設定成本報告

如果您已設定成本報告，則下次啟用完成後執行報告時，標籤的資料欄會出現。若是第一次設定成本報告，請選擇以下其中一項。

- 請參閱《AWS Billing and Cost Management 使用者指南》中的設定每月成本分配報告。
- 請參閱《AWS Cost and Usage Report 使用者指南》中的建立成本和用量報告。

> **Note**
>
> 最多可能需要 24 小時才能開始 AWS 將報告傳遞到 S3 儲存貯體。

您可以為您擁有的 Windows AMIs 設定EC2快速啟動AMIs，或從中的 Amazon EC2 主控台、、APISDKsCloudFormation、或ec2命令與您共用的快速啟動 AWS CLI。以下各節涵蓋 Amazon EC2 主控台和的設定步驟 AWS CLI。

您也可以建立自訂 WindowsAMIs，這些視窗已設定為使用 EC2 Image Builder EC2 快速啟動。如需詳細資訊，請參閱針對已啟用EC2快速啟動AMI的 Windows 建立發佈設定 (AWS CLI)。

監控EC2快速啟動

本節介紹如何監控已啟用EC2快速啟動的帳戶AMIs中的 Amazon EC2 Windows 伺服器。

監視EC2快速啟動狀態變更 EventBridge

當啟用了EC2快速啟動的 Windows AMI 的狀態發生變化時，Amazon EC2 會生成一個`EC2 Fast Launch State-change Notification`事件。然後 Amazon EC2 將狀態變化事件發送到 Amazon EventBridge（以前稱為 Amazon CloudWatch 活動）。

您可以建立 EventBridge 規則來觸發一或多個動作以回應狀態變更事件。例如，您可以建立 EventBridge 規則來偵測何時啟用EC2快速啟動，並執行下列動作：

- 向 Amazon SNS 主題傳送訊息，通知其訂閱者。
- 叫用執行某些動作的 Lambda 函數。
- 將狀態變更資料傳送至 Amazon 資料 Firehose 進行分析。

如需詳細資訊，請參閱 Amazon EventBridge 使用者指南中的建立針對事件做出回應的 Amazon EventBridge 規則。

狀態變更事件

EC2快速啟動功能會盡力發出JSON格式化狀態變更事件。Amazon 以近乎實時的方 EventBridge 式將事件EC2發送到。本節說明事件欄位，並顯示事件格式範例。

`EC2 Fast Launch State-change Notification`

imageId

識別EC2快速啟動狀態變更的。AMI

resourceType

用於預先佈建的資源類型。支援的值為：`snapshot`。預設值為 `snapshot`。

state

指定之EC2快速啟動功能的目前狀態AMI。有效值包括以下項目：

- 啟用 — 您已啟用的EC2快速啟動功能AMI，而 Amazon EC2 已開始為預先佈建程序建立快照。
- 啟用失敗 — 發生錯誤，造成預先佈建程序在您第一次啟用EC2. AMI 這可能會在預先佈建程序期間隨時發生。
- 啟用 — EC2 快速啟動功能已啟用。Amazon `enabled` 為新啟用的快速啟動EC2AMI建立第一個預先佈建的EC2快照後，狀態會變更為。如果AMI已啟用並再次進行預先佈建，狀態變更會立即發生。

- 啟用-失敗 — 只有在您的EC2快速啟動不是第一次進行預先佈建程序時，才AMI會套用此狀態。如果停用EC2快速啟動功能，稍後再次啟用，或者在第一次完成預先佈建後發生組態變更或其他錯誤，就可能會發生這種情況。
- 禁用 — 所有AMI者已關閉的EC2快速啟動功能AMI，Amazon EC2 已經開始清理過程。
- 停用 — [EC2快速啟動] 功能已停用。一旦 Amazon EC2 完成清理過程，狀態就會變為。`disabled`
- 禁用失敗 — 出現問題，導致清理程序失敗。這意味著某些預先佈建的快照可能仍會保留在帳戶中。

stateTransitionReason

EC2快速啟動狀態變更的原因AMI。

> **Note**
>
> 此事件訊息中的所有欄位都是必填欄位。

下列範例顯示新啟用的 EC2 Fast Launch，其中AMI已啟動第一個執行個體以啟動預先佈建程序。此時，狀態為 `enabling`。Amazon EC2 建立第一個預先佈建的快照後，狀態會變更為`enabled`。

```
{
 "version": "0",
 "id": "01234567-0123-0123-0123-012345678901",
 "detail-type": "EC2 Fast Launch State-change Notification",
 "source": "aws.ec2",
 "account": "123456789012",
 "time": "2022-08-31T20:30:12Z",
 "region": "us-east-1",
 "resources": [
  "arn:aws:ec2:us-east-1:123456789012:image/ami-123456789012"
 ],
 "detail": {
  "imageId": "ami-123456789012",
  "resourceType": "snapshot",
  "state": "enabling",
  "stateTransitionReason": "Client.UserInitiated"
 }
}
```

監控EC2快速啟動指標 CloudWatch

啟EC2AMIs用EC2快速啟動的 Amazon 將指標發送到 Amazon CloudWatch。您可以使用 AWS Management Console AWS CLI、或列出EC2快速啟動傳送的目標量度 CloudWatch。API`AWS/EC2`命名空間包含下列EC2快速啟動測量結果：

指標	描述
NumberOfAvailableFastLaunchSnapshots	每次啟用快速啟AMI動時可用的預先佈建EC2快照數目。
NumberOfInstancesFastLaunched	從預先佈建的EC2快照啟動每個啟用AMI的快照啟用的執行個體數目。
NumberOfInstancesNotFastLaunched	由於啟AMI動時缺少可用的預先佈建EC2快照，導致每次啟用快速啟動的執行個體數量導致冷開機。
FastLaunchSnapshotUsedToRefillStartTime	Amazon EC2 從快速啟動啟動啟AMI動新映像時的時間戳記，可在使用現有EC2快照後建立另一個快照。
FastLaunchSnapshotCreationTime	衡量 Amazon EC2 啟動執行個體並為啟用EC2快速啟動建立快照所花費的時間AMI。

EC2快速啟動的服務連結角色

Amazon EC2 針對代表您呼叫其他 AWS 服務 人所需的許可，使用服務連結角色。服務連結角色是一種獨特的IAM角色類型，直接連結至. AWS 服務服務連結角色提供委派權限的安全方式， AWS 服務 因為只有連結的服務可以擔任服務連結的角色。如需 Amazon 如何EC2使用IAM角色 (包括服務連結角色) 的詳細資訊，請參閱IAMAmazon 的角色 EC2。

Amazon EC2 使用名為的服務連結角色`AWSServiceRoleForEC2FastLaunch`來建立和管理一組預先佈建的快照，以減少從 Windows 啟動執行個體所需的時間。AMI

您不需要手動建立此服務連結角色。當您開始為您使用EC2快速啟動時AMI，Amazon EC2 會為您建立服務連結角色 (如果尚未存在)。

> **Note**
>
> 如果服務連結角色已從您的帳戶中刪除，您可以為其他 Windows 啟用EC2快速啟動，AMI以在您的帳戶中重新建立角色。或者，您可以禁用當前的EC2快速啟動AMI，然後再次啟用它。但是，停用此功能會導致您對所有新執行個體AMI使用標準啟動程序，而 Amazon EC2 會移除所有預先佈建的快照。所有預先佈建的快照都消失後，您可以AMI再次啟用EC2快速啟動。

Amazon EC2 不允許您編輯`AWSServiceRoleForEC2FastLaunch`服務鏈接的角色。因為有各種實體可能會參考服務連結角色，所以您無法在建立角色之後變更角色名稱。但是，您可以使用編輯角色的描述IAM。如需詳細資訊，請參閱IAM使用指南中的編輯服務連結角色。

您必須先刪除所有相關的資源，才能刪除服務連結角色。如此可在啟用EC2快速啟動的情況下保護與 Amazon EC2 Windows 伺服器AMI相關聯的 Amazon EC2 資源，因為您不會意外移除存取資源的權限。

Amazon 在提供 Amazon 服務的所有區域中EC2支援EC2快速啟動EC2服務連結角色。如需詳細資訊，請參閱區域。

`AWSServiceRoleForEC2FastLaunch` 授予的許可

Amazon EC2 使用受`EC2FastLaunchServiceRolePolicy`管政策完成下列動作：

- `cloudwatch:PutMetricData`— 將與EC2快速啟動相關聯的指標資料張貼至 Amazon EC2 命名空間。
- `ec2:CreateLaunchTemplate`— 為您的 Amazon EC2 Windows 服務器創建一個啟動模板，並啟AMI用了EC2快速啟動。
- `ec2:CreateSnapshot`— 在啟用快速啟動的情況下，為您的 Amazon EC2 Windows 伺服器AMI建立預先佈建的EC2快照。
- `ec2:CreateTags`— 為與啟用EC2快速啟動的 Amazon EC2 Windows 伺服器AMI啟動和預先佈建 Windows 執行個體相關聯的資源建立標籤。
- `ec2:DeleteSnapshots`— 如果先前已啟用的EC2快速啟AMI動已關閉，請刪除所有相關聯的預先佈建快照。
- `ec2:DescribeImages` – 描述所有資源的映像。
- `ec2:DescribeInstanceAttribute` – 描述所有資源的執行個體屬性。
- `ec2:DescribeInstanceStatus` – 描述所有資源的執行個體狀態。

- `ec2:DescribeInstances` – 描述所有資源的執行個體。
- `ec2:DescribeInstanceTypeOfferings` – 描述適用於所有資源的執行個體類型方案。
- `ec2:DescribeLaunchTemplates` – 描述所有資源的啟動範本。
- `ec2:DescribeLaunchTemplateVersions` – 描述所有資源的啟動範本版本。
- `ec2:DescribeSnapshots` – 描述所有資源的快照資源。
- `ec2:DescribeSubnets` – 描述所有資源的子網。
- `ec2:RunInstances`— 從已啟用EC2快速啟動的 Amazon EC2 Windows 伺服AMI器啟動執行個體，以執行佈建步驟。
- `ec2:StopInstances`— 停止從已啟用EC2快速啟動的 Amazon EC2 Windows 伺服器啟動AMI的執行個體，以建立預先佈建的快照。
- `ec2:TerminateInstances`— 從執行個體建立預先佈建的快照後，終止從已啟AMI用EC2快速啟動的 Amazon EC2 Windows 伺服器啟動的執行個體。
- `iam:PassRole` – 允許 `AWSServiceRoleForEC2FastLaunch` 服務連結角色使用啟動範本中的執行個體設定檔代表您啟動執行個體。

如需使用 Amazon 受管政策的詳細資訊EC2，請參閱AWS Amazon 的受管政策 EC2。

存取客戶管理的金鑰，以便搭配加密AMIs和EBS快照使用

先決條件

- 若要讓 Amazon EC2 代表您存AMI取加密，您必須擁有客戶受管金鑰中`createGrant`動作的許可。

當您為加密啟用EC2快速啟動時AMI，Amazon 會EC2確保授與
該`AWSServiceRoleForEC2FastLaunch`角色使用客戶受管金鑰存取您的權限AMI。代表您啟動執行個體和建立預先佈建的快照需要此許可。

變更您 Amazon EC2 執行個體的 Windows 管理員密碼

如果您從 AWS Windows 啟動執行個體AMI，預先安裝的啟動代理程式會依下列方式設定預設密碼：

- 對於視窗服務器 2022 及更高版本，EC2Launch v2生成默認密碼。
- 對於視窗伺服器 2016 和 2019，EC2Launch代理程式會產生預設密碼。
- 對於視窗伺服器 2012 R2 及更早版本，會EC2Config服務產生預設密碼。

> **Note**
>
> 對於 Windows 伺服器 2016 及更新版本AMIs，`Password never expires`本機系統管理員已停用。對於 Windows 伺服器 2016 之前的AMI版本，`Password never expires`會針對本機系統管理員啟用。

連線後變更管理員密碼

當您第一次連線至執行個體時，建議您變更預設的管理員密碼。使用下列程序變更 Windows 執行個體的管理員密碼。

> **Important**
>
> 將新密碼存放於安全處。您將無法使用 Amazon EC2 主控台擷取新密碼。主控台只能擷取預設的密碼。如果您在變更密碼後嘗試使用預設密碼連線執行個體，您會收到 "Your credentials did not work" (您的登入資料無法作用) 錯誤。

變更本機管理員密碼

1. 連線至執行個體並開啟命令提示。
2. 執行下列命令。如果您的新密碼包含特殊字元，請務必以雙引號括住密碼。

```
net user Administrator "new_password"
```

3. 將新密碼存放於安全處。

變更遺失或過期的密碼

如果您的密碼遺失或過期，您可產生新的密碼。如需密碼重設程序，請參閱重設 Amazon EC2 視窗執行個體的管理員密碼。

將選用的視窗伺服器元件新增至 Amazon EC2 視窗

若要存取並安裝選用元件，您必須找到適合您 Windows Server 版本的正確EBS快照集、從快照建立磁碟區，然後將磁碟區附加至執行個體。

開始之前

使用 AWS Management Console 或命令列工具取得執行個體的執行個體 ID 和可用區域。您必須在與執行個體相同的可用區域中建立EBS磁碟區。

請使用下列其中一個程序，將 Windows 伺服器元件新增至您的執行個體。

Console

使用主控台為您的執行個體新增 Windows 元件

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Snapshots (快照)。
3. 從 Filter (篩選條件) 列中，選擇 Public Snapshots (公有快照)。
4. 新增 Owner Alias (擁有者別名) 篩選條件，然後選擇 amazon。
5. 新增 Description (描述) 篩選條件，然後輸入 **`Windows`**。
6. 按 Enter
7. 選取符合您系統架構及語言偏好設定的快照。例如，若執行個體執行 Windows Server 2019，請選取 Windows 2019 English Installation Media (Windows 2019 英文版安裝媒體)。
8. 選擇 Actions (動作)、Create volume from snapshot (從快照建立磁碟區)。
9. 針對可用區域，選取符合您 Windows 執行個體的可用區域。選擇 Add tag (新增標籤)，然後為標籤索引鍵指定 **`Name`**，並為標籤值指定描述性名稱。選擇建立磁碟區。
10. 在 Volume Successfully Created (磁碟區建立成功) 訊息 (綠色橫幅) 中，選擇您剛剛建立的磁碟區。
11. 選擇 Actions (動作)、Attach Volume (連接磁碟區)。
12. 從 Instance (執行個體) 中，選取執行個體 ID。
13. 對於 Device name (裝置名稱)，輸入配件的裝置名稱。如需裝置名稱的說明，請參閱 Amazon EC2 執行個體上磁碟區的裝置名稱。
14. 選擇 Attach volume (連接磁碟區)。
15. 連線到您的執行個體，然後讓磁碟區可用。如需詳細資訊，請參閱 Amazon 使用EBS者指南中的讓 Amazon EBS 磁碟區可供使用。

> **Important**
>
> 請不要初始化磁碟區。

16. 開啟 Control Panel (控制面板)，Programs and Features (程式和功能)。選擇 Turn Windows features on or off (開啟或關閉 Windows 功能)。如果系統提示您輸入安裝媒體，請使用安裝媒體指定EBS磁碟區。
17. (選用) 完成媒體安裝後，您可以分離磁碟區。分離磁碟區後，即可將它刪除。

AWS CLI

若要使用 AWS CLI

1. 使用 describe-snapshots 命令，搭配 `owner-ids` 參數和 `description` 篩選條件，取得可用安裝媒體快照的清單。

```
aws ec2 describe-snapshots --owner-ids amazon --filters
 Name=description,Values=Windows*
```

2. 在輸出中，注意符合您系統架構和語言偏好設定的快照 ID。例如：

```
{
    "Snapshots": [
    ...
        {
            "OwnerAlias": "amazon",
            "Description": "Windows 2019 English Installation Media",
            "Encrypted": false,
            "VolumeId": "vol-be5eafcb",
            "State": "completed",
            "VolumeSize": 6,
            "Progress": "100%",
            "StartTime": "2019-10-25T20:00:47.000Z",
            "SnapshotId": "snap-22da283e",
            "OwnerId": "123456789012"
        },
    ...
   ]
}
```

3. 使用 create-volume 命令從快照建立磁碟區。指定與您執行個體相同的可用區域。

```
aws ec2 create-volume --snapshot-id snap-22da283e --volume-type gp2 --
availability-zone us-east-1a
```

4. 在輸出中，記下磁碟區 ID。

```
{
    "AvailabilityZone": "us-east-1a",
    "Encrypted": false,
    "VolumeType": "gp2",
    "VolumeId": "vol-0c98b37f30bcbc290",
    "State": "creating",
    "Iops": 100,
    "SnapshotId": "snap-22da283e",
    "CreateTime": "2017-04-18T10:33:10.940Z",
    "Size": 6
}
```

5. 使用 attach-volume 命令將磁碟區連接到您的執行個體。

```
aws ec2 attach-volume --volume-id vol-0c98b37f30bcbc290 --instance-
id i-01474ef662b89480 --device xvdg
```

6. 連線到您的執行個體，然後讓磁碟區可用。如需詳細資訊，請參閱 Amazon 使用EBS者指南中的讓 Amazon EBS 磁碟區可供使用。

> **Important**
>
> 請不要初始化磁碟區。

7. 開啟 Control Panel (控制面板)，Programs and Features (程式和功能)。選擇 Turn Windows features on or off (開啟或關閉 Windows 功能)。如果系統提示您輸入安裝媒體，請使用安裝媒體指定EBS磁碟區。
8. (選用) 完成媒體安裝後，請使用 detach-volume 命令將磁碟區與您的執行個體分離。分離磁碟區後，您可以使用 delete-volume 命令來刪除磁碟區。

Tools for Windows PowerShell

使用視窗適用的工具將 Windows 元件新增至您的執行個體 PowerShell

1. 搭配`Owner`和`description`篩選器使用Get-EC2Snapshot指令程式，以取得可用安裝媒體快照集的清單。

```
PS C:\> Get-EC2Snapshot -Owner amazon -Filter @{ Name="description";
 Values="Windows*" }
```

2. 在輸出中，注意符合您系統架構和語言偏好設定的快照 ID。例如：

```
...
DataEncryptionKeyId :
Description         : Windows 2019 English Installation Media
Encrypted           : False
KmsKeyId            :
OwnerAlias          : amazon
OwnerId             : 123456789012
Progress            : 100%
SnapshotId          : snap-22da283e
StartTime           : 10/25/2019 8:00:47 PM
State               : completed
StateMessage        :
Tags                : {}
VolumeId            : vol-be5eafcb
VolumeSize          : 6
...
```

3. 使用指New-EC2Volume令程式從快照建立磁碟區。指定與您執行個體相同的可用區域。

```
PS C:\> New-EC2Volume -AvailabilityZone us-east-1a -VolumeType gp2 -
SnapshotId snap-22da283e
```

4. 在輸出中，記下磁碟區 ID。

```
Attachments      : {}
AvailabilityZone : us-east-1a
CreateTime       : 4/18/2017 10:50:25 AM
Encrypted        : False
Iops             : 100
KmsKeyId         :
Size             : 6
SnapshotId       : snap-22da283e
State            : creating
Tags             : {}
VolumeId         : vol-06aa9e1fbf8b82ed1
VolumeType       : gp2
```

5. 使用指Add-EC2Volume令程式將磁碟區附加至您的執行個體。

```
PS C:\> Add-EC2Volume -InstanceId i-087711ddaf98f9489 -
VolumeId vol-06aa9e1fbf8b82ed1 -Device xvdh
```

6. 連線到您的執行個體，然後讓磁碟區可用。如需詳細資訊，請參閱 Amazon 使用EBS者指南中的讓 Amazon EBS 磁碟區可供使用。

> **Important**
>
> 請不要初始化磁碟區。

7. 開啟 Control Panel (控制面板)，Programs and Features (程式和功能)。選擇 Turn Windows features on or off (開啟或關閉 Windows 功能)。如果系統提示您輸入安裝媒體，請使用安裝媒體指定EBS磁碟區。
8. (選擇性) 當您完成安裝媒體時，請使用Dismount-EC2Volume指令程式將磁碟區與執行個體中斷連結。卸離磁碟區之後，您可以使用指Remove-EC2Volume令程式刪除磁碟區。

在 Windows 執行個體上安裝適用於 Linux 的EC2視窗子系統

您可以根據執行個體類型和執行個體作業系統安裝兩個版本的 Windows 子系統 (WSL)：WSL1 和 WSL 2。對於`.metal`執行個體類型，您可以安裝 WSL 1 或 WSL 2。對於所有其他執行個體類型，則適用下列需求：

- 對於虛擬化EC2執行個體，您必須安裝 WSL 1.
- 對於執行 Windows Server 的執行個體，作業系統版本必須是下列其中一項才能安裝WSL：
  - Windows Server 2019
  - Windows Server 2022

如需有關的詳細資訊WSL，請參閱 Microsoft 組建網站上的適用於 Linux 的 Windows 子系統文件。

安裝 WSL

下列指示會安裝WSL在EC2執行視窗伺服器 2022 的執行個體上。如需在EC2執行 Windows 伺服器 2019 的執行個體WSL上安裝的指示，請參閱 Microsoft 網站WSL上的舊版 Windows 伺服器上安裝。按照這些指示操作後，您可以使用以下說明中的步驟 3 來配置WSL為使用 WSL 1。

安裝 WSL 1

1. 若要安裝WSL，請在執行EC2個體上執行下列標準安裝命令，但請務必透過包含啟用 WSL 1 `--enable-wsl1`。依預設，會安裝 WSL 2。如果您的執行個體是使用虛擬化執行個體類型啟動的，您必須完成此程序中的步驟 3，才能將版本設定為 WSL 1。

```
wsl --install --enable-wsl1 --no-launch
```

2. 重新啟動EC2執行個體。

```
shutdown -r -t 20
```

3. 若要設定WSL為使用 WSL 1，請在執行個體上執行下列命令。如需有關設定WSL版本的詳細資訊，請參閱 Microsoft 組建網站WSL上舊版的手動安裝步驟。

```
wsl --set-default-version 1
```

4. 安裝預設發行版。

```
wsl --install
```

安裝WSL二

- 若要安裝WSL，請在執行EC2個體上執行下列標準安裝指令。依預設，會安裝 WSL 2。如果您要在執行個`.metal`體WSL上安裝，則這是唯一要執行的步驟。

```
wsl --install
```

  如需詳細資訊，請參閱使用 Microsoft 組建網站WSL上的在視窗上安裝 Linux。

將EC2視窗執行個體升級至較新版本的視窗伺服器

如果是時候從舊版升級 Windows 執行個體上的 EC2 Windows Server 作業系統，您可以使用下列其中一種方法。

就地升級

就地升級會在現有執行個體上運作。在此過程中，只有作業系統檔案會受到影響，而您的設定、伺服器角色和資料則保持不變。

移轉 (也稱為 side-by-side 升級)

遷移涉及擷取設定、組態和資料，並將這些設定移植到全新 EC2 Windows 執行個體上的較新作業系統。您可以從訂閱的公用或私人 AMI Windows 啟動執行個AMI體 AWS Marketplace，或與您共用的 Windows。您也可以AMI使用 EC2 Image Builder 建立自訂。如需詳細資訊，請參閱 Image Builder 使用指南。

> **Note**
>
> AWS 為在EC2執行個體上執行的 Windows 伺服器版本提供一組公開可用的 Amazon 機器映像 (AMIs)。這AMIs些資料會每月更新一次。如需有關最新視窗的資訊AMIs，請參閱「AWS 視窗AMI參考」。

Microsoft 傳統上建議遷移到 Windows 服務器的較新版本，而不是升級到位。遷移可能導致更少的升級錯誤或問題，但由於需要佈建新的執行個體、規劃和移植應用程式，以及調整新執行個體的組態設定，所以所需的時間可能會比就地升級還要長。就地升級的速度較快，但軟體不相容可能會產生錯誤。

目錄

- 在 EC2 Windows 執行個體上執行就地升級
- 使用自動化手冊升級 EC2 Windows 執行個體
- 將 EC2 Windows 執行個體移轉至目前一代的執行個體類型
- 疑難排解 EC2 Windows 執行個體上的作業系統升級

在 EC2 Windows 執行個體上執行就地升級

在您執行就地升級前，您必須判斷執行個體執行的網路驅動程式為何。PV 網路驅動程式可讓您使用遠端桌面存取您的執行個體。執行個體使用 AWS PV、Intel 網路介面卡或增強型網路驅動程式。如需詳細資訊，請參閱Windows 執行個體的全虛擬驅動程式。

在您開始就地升級之前

在您開始您的就地升級之前，請完成下列任務並記下以下重要詳細資訊。

- 閱讀 Microsoft 文件，了解升級需求、已知問題和限制。同時也請檢閱升級的正式說明。
  - Upgrade Options for Windows Server 2012
  - Upgrade Options for Windows Server 2012 R2
  - Upgrade and conversion options for Windows Server 2016
  - Upgrade and conversion options for Windows Server 2019
  - Windows Server 2022 的升級或轉換選項
  - Windows Server Upgrade Center
- 建議您在至少 2 vCPUs 和 4GB 的RAM執行個體上執行作業系統升級。必要時，您可以將執行個體變更為相同類型但更大的執行個體 (例如，將 t2.small 變更為 t2.large)、執行升級，然後再調整回原始大小。如果您需要保留執行個體大小，您可以使用執行個體主控台螢幕擷取畫面來監控進度。如需詳細資訊，請參閱Amazon EC2 實例類型更改。
- 確認您 Windows 執行個體的根磁碟區具有足夠的可用磁碟空間。Windows 安裝程序可能不會警告您的磁碟空間不足。如需升級至特定作業系統所需磁碟空間的資訊，請參閱 Microsoft 文件。若磁碟區沒有足夠的空間，您可以擴展它。如需詳細資訊，請參閱 Amazon EBS使用者指南中的 Amazon EBS 彈性磁碟區。
- 決定您的升級途徑。您必須將作業系統升級至相同的架構。例如，您必須將 32 位元系統升級至 32 位元系統。Windows Server 2008 R2 及更新版本為僅限 64 位元。
- 停用防毒軟體、反間諜軟體及防火牆。這些軟體類型可能會和升級程序衝突。請在升級完成之後再次啟用防毒軟體、反間諜軟體及防火牆。
- 如 將 EC2 Windows 執行個體移轉至目前一代的執行個體類型 所述更新至最新的驅動程式。
- 升級協助程式服務僅支援執行 Citrix PV 驅動程式的執行個體。若執行個體執行的是 Red Hat 驅動程式，您必須先手動升級那些驅動程式。

使用 AWS PV、Intel 網路介面卡或增強型網路驅動程式就地升級執行個體

透過下列程序，使用 AWS PV、Intel 網路介面卡或增強聯網網路驅動程式升級 Windows Server 執行個體。

執行就地升級

1. 針對備份或測試目AMI的，建立您計劃升級的系統。您接著便能在複本上執行升級，模擬測試環境。若升級完成，您便可以將流量切換至此執行個體，而無須經歷太多停機時間。若升級失敗，您可以回復至備份。如需詳細資訊，請參閱創建一個 Amazon 支EBS持 AMI。
2. 確認您的 Windows Server 執行個體使用最新的網路驅動程式。

a. 若要更新您的 AWS PV 驅動程式，請參閱升級 EC2 Windows 執行個體上的光伏驅動。

b. 若要更新ENA驅動程式，請參閱在 EC2 Windows 執行個體上安裝ENA驅動程式。

c. 若要更新您的 Intel 驅動程式，請參閱 使用英特爾 82599 VF 接口的增強型網絡

3. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。

4. 在導覽窗格中，選擇 Instances (執行個體)。尋找執行個體。記下執行個體的執行個體 ID 和可用區域。您在此程序的後續步驟中將需要此資訊。

5. 若您要從 Windows Server 2012 或 2012 R2 升級至 Windows Server 2016、2019 或 2022，請在您的執行個體上執行下列作業再繼續：

   a. 解除安裝EC2Config服務。如需詳細資訊，請參閱適用於 EC2Launch v2 和EC2Config代理程式的 Windows 服務管理。

   b. 安裝 EC2Launch v1 或 EC2Launch v2 代理程式。如需詳細資訊，請參閱在 EC2 Windows 執行個體啟動期間，使用 EC2Launch v1 代理程式執行工作和使用 EC2Launch v2 代理程式在 EC2 Windows 執行個體啟動期間執行工作。

   c. 安裝代 AWS Systems Manager SSM理程式。如需詳細資訊，請參閱《使用指南》中的〈AWS Systems Manager 使用SSM代理程式〉。

6. 從 Windows Server 安裝媒體快照建立新的磁碟區。

   a. 在左側導覽窗格中，於 Elastic Block Store 下方，選擇 Snapshots (快照)。

   b. 從篩選條件列中，選擇公有快照。

   c. 在搜尋列中，指定下列篩選條件：

      - 選擇所有者別名，然後選擇 =，最後選擇 amazon。
      - 選擇說明，然後開始輸入 **`Windows`**。選取與您要升級到的系統架構及語言偏好設定相符的 Windows 篩選條件。例如，選擇 Windows 2019 英文版安裝媒體以升級至 Windows Server 2019。

   d. 選取與您要升級到的系統架構和語言偏好設定相符的快照旁邊的核取方塊，然後選擇動作、從快照建立磁碟區。

   e. 在建立磁碟區頁面中，選擇符合 Windows 執行個體的可用區域，然後選擇建立磁碟區。

7. 在成功創建的卷卷中-***`1234567890example`***在頁面頂端的橫幅，選擇您剛剛建立的磁碟區的 ID。

8. 選擇 Actions (動作)、Attach Volume (連接磁碟區)。

9. 在附接磁碟區頁面中，對於執行個體，選取 Windows 執行個體的執行個體 ID，然後選擇附接磁碟區。

10. 請遵循讓 Amazon 磁碟區可供使用的步驟，讓新EBS磁碟區可供使用。

> **Important**
>
> 請勿初始化磁碟，因為這樣做會刪除現有資料。

11. 在 Windows 中 PowerShell，切換到新的磁碟區磁碟機。透過開啟連接到執行個體的安裝媒體磁碟區，開始升級。

    a. 若您要升級至 Windows Server 2016 或更新版本，請執行下列項目：

    ```
    .\setup.exe /auto upgrade /dynamicupdate disable
    ```

    > **Note**
    >
    > 在將 `/dynamicupdate` 選項設定為停用的情況下執行 setup.exe 會阻止 Windows 在 Windows Server 升級程序期間中安裝更新，因為在升級期間中安裝更新可能會導致失敗。升級完成後，您可以使用 Windows Update 安裝更新。

    若您要升級至較早版本的 Windows Server，請執行下列項目：

    ```
    Sources\setup.exe
    ```

    b. 針對 [選取要安裝的作業系統]，選取 Windows Server 執行個體SKU的完整安裝，然後選擇 [下一步]。

    c. 針對 Which type of installation do you want? (您希望的安裝類型為何？)，選擇 Upgrade (升級)。

    d. 完成協助程式。

Windows Server 安裝程式會複製並處理檔案。在幾分鐘之後，您的遠端桌面工作階段便會關閉。升級所需要花費的時間取決於在您 Windows Server 執行個體上執行之應用程式和伺服器角色的數目。升級程序最少可能需要花費 40 分鐘至數小時。執行個體的兩項狀態檢查中的其中一項會在升級程序過程中失敗。當升級完成後，兩項狀態檢查便都會通過。您可以檢查主控台輸出的系統日誌，或使用 Amazon CloudWatch 指標處理磁碟和CPU活動，以判斷升級是否正在進行中。

> **Note**
>
> 若升級至 Windows Server 2019，在升級完成之後，若需要的話，您可以手動變更桌面背景，移除先前作業系統的名稱。

若執行個體在數個小時之後仍無法通過兩項狀態檢查，請參閱疑難排解 EC2 Windows 執行個體上的作業系統升級。

升級後任務

1. 登入執行個體以啟動的升級。 NET框架並在出現提示時重新啟動系統。
2. 如果您尚未在先前步驟中這樣做，請安裝 EC2Launch v1 或 EC2Launch v2 代理程式。 如需詳細資訊，請參閱在 EC2 Windows 執行個體啟動期間，使用 EC2Launch v1 代理程式執行工作和使用 EC2Launch v2 代理程式在 EC2 Windows 執行個體啟動期間執行工作。
3. 如果您升級到 Windows 伺服器 2012 R2，我們建議您將光伏驅動程式升級為 AWS 光伏驅動程式。如果您在 Nitro-based 執行個體上進行升級，建議您安裝或升級NVME和ENA驅動程式。如需詳細資訊，請參閱 Windows Server 2012 R2、使用安裝或升級 AWS NVMe驅動程式 PowerShell 或 啟用 Windows 上的增強型網路。
4. 重新啟用防毒軟體、反間諜軟體及防火牆。

使用自動化手冊升級 EC2 Windows 執行個體

您可以 AWS 使用自動 AWS Systems Manager 化手冊在上執行 Windows 和SQL伺服器執行個體的自動升級。

目錄

- 相關服務
- 執行選項
- 升級 Windows Server
- 升級SQL伺服器

相關服務

自動升級程序會使用下列 AWS 服務：

- AWS Systems Manager。 AWS Systems Manager 是一個功能強大的統一介面，可集中管理您的 AWS 資源。如需詳細資訊，請參閱AWS Systems Manager《使用者指南》。
- AWS Systems Manager 代理程SSM式 (代理程式) 是可在 Amazon EC2 執行個體、現場部署伺服器或虛擬機器 (VM) 上安裝和設定的 Amazon 軟體。SSM代理程式可讓 Systems Manager 更新、管理及設定這些資源。此代理程式會處理 AWS Cloud 中來自 Systems Manager 服務的申請，然後依照要求中的指定執行。如需詳細資訊，請參閱《使用指南》中的〈AWS Systems Manager 使用SSM代理程式〉。
- AWS Systems Manager SSM手冊。R SSM unbook 會定義系統管理員在受管理的執行個體上執行的動作。SSMrunbook 使用 JavaScript 物件標記法 (JSON) 或YAML，它們包括您指定的步驟和參數。本主題使用兩個 Systems Manager 手SSM冊進行自動化。如需詳細資訊，請參閱 AWS Systems Manager 使用者指南中的 AWS Systems Manager Automation runbook 參考。

執行選項

當您在 Systems Manager 主控台選取 Automation (自動化) 時，請選取 Execute (執行)。在您選取自動化文件後，系統會提示您選取自動化執行選項。請從以下選項中選擇。在本主題稍後所提供途徑的步驟中，我們使用 Simple execution (簡易執行) 選項。

簡易執行

如果您想要更新單一執行個體，但不想經歷每個自動化步驟來稽核結果，請選擇此選項。稍後的升級步驟中會進一步詳述此選項。

Rate control (比率控制)

如果您要對多個執行個體套用升級，請選擇此選項。請定義以下設定。

- Parameter (參數)

  此設定 (也在 Multi-Account and Region (多帳戶和區域) 設定中) 定義自動化如何展開。
- 目標

  選取您要套用自動化的目標。此設定也在 Multi-Account and Region (多帳戶和區域) 設定中。
- Parameter Values (參數值)

  使用自動化文件參數中定義的值。
- Resource Group (資源群組)

在中 AWS，資源是您可以使用的實體。範例包括 Amazon EC2 執行個體、 AWS CloudFormation 堆疊或 Amazon S3 儲存貯體。如果您使用多個資源，則將它們作為一個組進行管理可能會很有用，而不是從一個服務移動到另一個 AWS 服務的每個任務。在某些情況下，您可能會想要管理大量相關資源，EC2例如組成應用程式層的執行個體。在此情況下，您在這些資源上可能需要一次執行大量動作。

- Tags (標籤)

  標籤可協助您以不同的方式對 AWS 資源進行分類，例如依用途、擁有者或環境。此分類在您擁有許多相同類型的資源時很有用。您可以使用指派的標籤快速識別特定資源。

- Rate Control (比率控制)

  Rate Control (比率控制) 也在 Multi-Account and Region (多帳戶和區域) 設定中。當您設定比率控制參數時，請以目標計數或機群的百分比，定義機群中有多少要套用自動化。

Multi-Account and Region (多帳戶和區域)

除了 Rate Control (比率控制) 下指定的參數 (也用在 Multi-Account and Region (多帳戶和區域) 設定中)，另外還有兩個設定：

- 帳戶和組織單位 (OUs)

  指定您要執行自動化的多個帳戶。

- AWS 區域

  指定您要 AWS 區域 在其中執行自動化操作的多個位置。

手動執行

此選項類似於 Simple execution (簡易執行)，但可讓您逐步完成每個自動化步驟並稽核結果。

升級 Windows Server

該手`AWSEC2-CloneInstanceAndUpgradeWindows`冊從您的帳戶中的 Windows 服務器實例創建一個 Amazon 機器映像（AMI），並將其升級AMI到您選擇的支持版本。此多步驟程序需要最多兩個小時才會完成。

自動升級程序AMIs包含兩種：

- 目前執行中的執行個體。第一個AMI是目前執行中的執行個體，未升級。這AMI是用來啟動另一個執行個體以執行就地升級。程序完成後，除非您明確要求保留原始執行個AMI體，否則系統會從您的帳戶中刪除此項目。此設定由參數處理 `KeepPreUpgradeImageBackUp` (預設值為`false`，表示AMI依預設會刪除)。
- 升級AMI。這AMI是自動化過程的結果。

最終結果是一個AMI，也就是的升級執行個體AMI。

升級完成後，您可以AMI在 Amazon 中啟動新功能來測試應用程式功能VPC。測試後，在執行另一次升級前，請先安排應用程式停機時間，再完全切換至已升級的執行個體。

必要條件

若要使用自動 AWS Systems Manager 化文件自動化您的 Windows 伺服器升級，您必須執行下列工作：

- 使用指定IAM政策建立IAM角色，以允許 Systems Manager 在 Amazon 執行個EC2體上執行自動化任務，並確認您符合使用 Systems Manager 的先決條件。如需詳細資訊，請參閱《AWS Identity and Access Management 使用指南》中的建立角色以將權限委派給 AWS 服務。
- 選取您希望如何執行自動化的選項。執行選項包括 Simple execution (簡易執行)、Rate control (比率控制)、Multi-account and Region (多帳戶和區域) 及 Manual execution (手動執行)。如需關於這些選項的詳細資訊，請參閱 執行選項。
- 確認您的執行個體上已安裝 SSM Agent。如需詳細資訊，請參閱在 Windows 伺服器的 Amazon EC2 執行個體上安裝和設定SSM代理程式
- 視窗 PowerShell 3.0 或更新版本必須安裝在您的執行個體上。
- 對於加入 Microsoft Active Directory 網域的執行個體，建議您指定沒有連線到您的網域控制站的 `SubnetId`，以協助避免主機名稱衝突。
- 執行個體子網路必須具有連線至網際網路的輸出連線，以提供 Amazon S3 AWS 服務 等存取權，以及從 Microsoft 下載修補程式的存取權。如果子網路是公用子網路且執行個體具有公用 IP 位址，或子網路是具有將網際網路流量傳送至公用NAT裝置的路由的私有子網路，則符合此需求。
- 此自動化適用於執行 Windows Server 2008 R2、Windows Server 2012 R2、Windows Server 2016 和 Windows Server 2019 的執行個體。
- 確認執行個體在開機磁碟中有 20 GB 的可用磁碟空間。
- 如果執行個體未使用由提供的 Windows 授權 AWS，請指定包含 Windows 伺服器 2012 R2 安裝媒體的 Amazon EBS 快照識別碼。若要執行此作業：

  1. 確認 Amazon EC2 執行個體正在執行 Windows 伺服器 2012 或更新版本。

2. 在執行執行個體的相同可用區域中建立 6 GB Amazon EBS 磁碟區。將磁碟區連結到執行個體。例如，將其掛載為 D 磁碟機。
3. 以滑鼠右鍵按一下ISO並將其掛接至執行個體，例如磁碟機 E。
4. 將驅動器 E：\ ISO 的內容複製到驅動器 D：\
5. 為上述步驟 2 中建立的 6 GB 磁碟區建立 Amazon EBS 快照。

Windows Server 升級限制

此自動化不支援升級 Windows 網域控制站、叢集或 Windows 桌面作業系統。此外，此自動化操作不支援已安裝下列角色的 Windows 伺服器的 Amazon EC2 執行個體：

- 遠端桌面工作階段主機 (RDSH)
- 遠端桌面連線代理人 (RDCB)
- 遠端桌面虛擬化主機 (RDVH)
- 遠端桌面 Web 存取 (RDWA)

執行 Windows Server 自動升級的步驟

請按照以下步驟使用 AWSEC2-CloneInstanceAndUpgradeWindows 自動化手冊升級您的 Windows 服務器實例。

1. 從 AWS Management Console (AWS 管理主控台) 開啟 Systems Manager。
2. 從左側導覽窗格中，在 Change Management (變更管理) 下選擇 Automation (自動化)。
3. 選擇 Execute automation (執行自動化)。
4. 搜尋稱為 `AWSEC2-CloneInstanceAndUpgradeWindows` 的自動化文件。
5. 當文件名稱出現時，請選取它。選取此文件時，將會出現文件詳細資訊。
6. 選擇 Execute automation (執行自動化) 以輸入此文件的參數。在頁面頂端，保持選取 Simple execution (簡易執行)。
7. 根據下列指導，輸入所要求的參數。

   - `InstanceID`

     類型：字串

     (必要) 執行安裝SSM代理程式的執行個體視窗伺服器 2008 年 R2、2016 年或 2019 年。

- `InstanceProfile`.

  類型：字串

  (必要) IAM 執行個體設定檔。這是用來針對 Amazon 執行個EC2體和執行 Systems Manager 自動化的IAM角色 AWS AMIs。如需詳細資訊，請參閱AWS Systems Manager 使用指南中的「為 Systems Manager 建立IAM執行個體設定檔」。

- `TargetWindowsVersion`

  類型：字串

  (必要) 選取目標 Windows 版本。

- `SubnetId`

  類型：字串

  (必要) 這是升級程序的子網路以及來源EC2執行個體所在的位置。確認子網路具有連線至 AWS 服務 (包括 Amazon S3) 以及 Microsoft 的輸出連線 (以便下載修補程式)。

- `KeepPreUpgradedBackUp`

  類型：字串

  (選用) 如果此參數設為 `true`，自動化會保留從執行個體建立的映像。預設設定為 `false`。

- `RebootInstanceBeforeTakingImage`

  類型：字串

  (選用) 預設為 `false` (不重新啟動)。如果將此參數設定為`true`，Systems Manager 會先重新啟動執行個體，然後再建立升級AMI的執行個體。

8. 輸入參數後，選擇 Execute (執行)。自動化開始時，您可以監控執行進度。
9. 自動化操作完成後，您將看到 AMI ID。您可以啟動AMI以確認 Windows 作業系統是否已升級。

> **Note**
>
> 自動化沒有必要執行所有步驟。這些步驟取決於自動化和執行個體的行為。Systems Manager 可能略過有些非必要的步驟。
>
> 此外，某些步驟可能會逾時。Systems Manager 會嘗試升級和安裝所有最新的修補程式。不過，根據特定步驟可定義的逾時設定，修補程式有時會逾時。發生此情況時，Systems

Manager 自動化會繼續下一個步驟，以確保內部作業系統升級到目標 Windows Server 版本。

10. 自動化完成後，您可以使用 AMI ID 啟動 Amazon EC2 執行個體以檢閱您的升級。如需如何從建立 Amazon EC2 執行個體的詳細資訊 AWS AMI，請參閱如何從自訂啟動EC2執行個體AMI？

升級SQL伺服器

AWSEC2-CloneInstanceAndUpgrade SQLServer 腳本AMI從您的帳戶中運行SQL服務器的 Amazon EC2 實例創建一個，然後升級AMI到更高版本的SQL服務器。此多步驟程序需要最多兩個小時才會完成。

在此工作流程中，自動化操作會AMI從執行個體建立，然後在您提供的子網路AMI中啟動新項目。然後，自動化會執行SQL伺服器的就地升級。升級完成後，自動化會AMI在終止升級的執行個體之前建立新的執行個體。

自動升級程序AMIs包含兩種：

- 目前執行中的執行個體。第一個AMI是目前執行中的執行個體，未升級。這AMI是用來啟動另一個執行個體以執行就地升級。程序完成後，除非您明確要求保留原始執行個AMI體，否則系統會從您的帳戶中刪除此項目。此設定由參數處理 `KeepPreUpgradeImageBackUp` (預設值為`false`，表示AMI依預設會刪除)。
- 升級AMI。這AMI是自動化過程的結果。

最終結果是一個AMI，也就是的升級執行個體AMI。

升級完成後，您可以AMI在 Amazon 中啟動新功能來測試應用程式功能VPC。測試後，在執行另一次升級前，請先安排應用程式停機時間，再完全切換至已升級的執行個體。

必要條件

若要使用自動 AWS Systems Manager 化文件自動化您的SQL伺服器升級，您必須執行下列工作：

- 使用指定IAM政策建立IAM角色，以允許 Systems Manager 在 Amazon 執行個EC2體上執行自動化任務，並確認您符合使用 Systems Manager 的先決條件。如需詳細資訊，請參閱《AWS Identity and Access Management 使用者指南》中的建立角色以委派許可給 AWS 服務。
- 選取您希望如何執行自動化的選項。執行選項包括 Simple execution (簡易執行)、Rate control (比率控制)、Multi-account and Region (多帳戶和區域) 及 Manual execution (手動執行)。如需關於這些選項的詳細資訊，請參閱 執行選項。

- Amazon EC2 執行個體必須使用視窗伺服器 2008 R2 或更新版本，以及SQL伺服器 2008 或更新版本。
- 確認您的執行個體上已安裝 SSM Agent。如需詳細資訊，請參閱在 Windows 伺服器的 Amazon EC2 執行個體上使用SSM代理程式。
- 確認執行個體具有足夠的可用磁碟空間：
  - 若您要從 Windows Server 2008 R2 升級至 2012 R2，或從 Windows Server 2012 R2 升級至更新版本的作業系統，請確認您的執行個體開機磁碟中有 20 GB 的可用磁碟空間。
  - 若您要從 Windows Server 2008 R2 升級至 2016 或更新版本，請確認執行個體開機磁碟中有 40 GB 的可用磁碟空間。
- 對於使用「自攜授權」(BYOL) SQL 伺服器版本的執行個體，則適用下列其他先決條件：
  - 提供包含目標SQL伺服器安裝媒體的 Amazon EBS 快照 ID。若要執行此作業：
    1. 確認 Amazon EC2 執行個體正在執行視窗伺服器 2008 R2 或更新版本。
    2. 在執行執行個體的相同可用區域中建立 6 GB Amazon EBS 磁碟區。將磁碟區連結到執行個體。例如，將其掛載為 D 磁碟機。
    3. 以滑鼠右鍵按一下ISO並將其掛接至執行個體，例如磁碟機 E。
    4. 將驅動器 E：\ ISO 的內容複製到驅動器 D：\
    5. 為步驟 2 中建立的 6 GB 磁碟區建立 Amazon EBS 快照。

SQL伺服器自動升級限制

使用 AWSEC2-CloneInstanceAndUpgrade SQLServer runbook 執行自動升級時，下列限制適用：

- 升級只能在使用 Windows 驗證的SQL伺服器上執行。
- 確認執行個體上沒有待定的安全性修補程式更新。開啟 Control Panel (控制面板)，接著選擇 Check for updates (檢查更新)。
- SQL不支援 HA 和鏡像模式下的伺服器部署。

執行SQL伺服器自動升級的步驟

請按照以下步驟使用升級SQL服務器 AWSEC2-CloneInstanceAndUpgrade SQLServer 自動化手冊。

1. 如果您還沒有，請下載SQL伺服器 2016 .iso 檔案並將其掛載到來源伺服器。
2. 掛載 .iso 檔案之後，將所有元件檔案複製到您選擇的任何磁碟區。

3. 拍攝磁碟區的 Amazon EBS 快照，然後將快照 ID 複製到剪貼簿以供日後使用。如需詳細資訊，請參閱 Amazon EBS使用者指南中的建立 Amazon EBS 快照。
4. 將執行個體設定檔附加到 Amazon EC2 來源執行個體。這可讓 Systems Manager 與執行個EC2體通訊，並在執行個體新增至 AWS Systems Manager 服務後執行指令。在此範例中，我們將角色命名為 `SSM-EC2-Profile-Role`，並將 `AmazonSSMManagedInstanceCore` 政策連接到角色。請參閱使AWS Systems Manager 用指南中的為 Systems Manager 建立IAM執行個體設定檔。
5. 在 AWS Systems Manager 主控台的左側導覽窗格中，選擇 [受管執行個體]。確認您的EC2執行個體位於代管執行個體清單中。如果幾分鐘後沒有看見您的執行個體，請參閱 AWS Systems Manager 使用者指南中的我的執行個體在哪裡？。
6. 在左側導覽窗格中，在 Change Management (變更管理) 下選擇 Automation (自動化)。
7. 選擇 Execute automation (執行自動化)。
8. 搜尋稱為 `AWSEC2-CloneInstanceAndUpgradeSQLServer` 的自動化文件。
9. 選擇`AWSEC2-CloneInstanceAndUpgradeSQLServerSSM`文件，然後選擇 [下一步]。
10. 確保選取 Simple execution (簡易執行) 選項。
11. 根據下列指導，輸入所要求的參數。

    - `InstanceId`

      類型：字串

      (必要) 執行SQL伺服器 2008 R2 (或更新版本) 的執行個體。
    - `IamInstanceProfile`

      類型：字串

      (必要) IAM 執行個體設定檔。
    - `SQLServerSnapshotId`

      類型：字串

      (必要) 目標SQL伺服器安裝媒體的快照 ID。包含SQL伺服器授權的執行個體不需要此參數。
    - `SubnetId`

      類型：字串

(必要) 這是升級程序的子網路以及來源EC2執行個體所在的位置。確認子網路具有連線至 AWS 服務 (包括 Amazon S3) 以及 Microsoft 的輸出連線 (以便下載修補程式)。

- `KeepPreUpgradedBackUp`

  類型：字串

  (選用) 如果此參數設為 `true`，自動化會保留從執行個體建立的映像。預設設定為 `false`。

- `RebootInstanceBeforeTakingImage`

  類型：字串

  (選用) 預設為 `false` (不重新啟動)。如果將此參數設定為`true`，Systems Manager 會先重新啟動執行個體，然後再建立升級AMI的執行個體。

- `TargetSQLVersion`

  類型：字串

  (選擇性) 目標SQL伺服器版本。預設值為 `2016`。

12. 輸入參數後，選擇 Execute (執行)。自動化開始時，您可以監控執行進度。
13. 當執行狀態顯示成功時，請展開輸出以檢視AMI資訊。您可以使用 AMI ID 來啟動您選擇VPC的SQL伺服器執行個體。
14. 打開 Amazon EC2 控制台。在左側導覽窗格中，選擇AMIs。您應該會看到新的AMI。
15. 若要確認已成功安裝新SQL伺服器版本，請選擇新AMI的伺服器版本，然後選擇 [啟動]。
16. 選擇您想要的執行個體類型AMI、要部署的VPC和子網路，以及您要使用的儲存區。由於您要從啟動新執行個體AMI，因此磁碟區會顯示為包含在您要啟動的新EC2執行個體中的選項。您可以移除任何這些磁碟區，或新增磁碟區。
17. 新增標籤以協助識別您的執行個體。
18. 將一或多個安全群組新增至執行個體。
19. 選擇 Launch Instance (啟動執行個體)。
20. 選取執行個體的標籤名稱，然後在 Actions (動作) 下拉式清單下，選取 Connect (連接)。
21. 確認新SQL伺服器版本是新執行個體上的資料庫引擎。

將 EC2 Windows 執行個體移轉至目前一代的執行個體類型

AWS Windows 使AMIs用 Microsoft 安裝媒體使用的默認設置進行配置，並具有一些自定義設置。這些自訂項目包括支援最新一代執行個體類型的驅動程式和設定，這些執行個體是建立在 AWS Nitro System 上的執行個體，例如 M5 或 C5。

在遷移至 Nitro 型執行個體 (包括裸機執行個體) 時，我們建議您在下列情況下遵循本主題中的步驟：

- 如果您要從自訂視窗啟動執行個體 AMIs
- 如果您要從 Amazon AMIs 提供的 Windows 啟動執行個體，且該執行個體是在 2018 年 8 月之前

如需詳細資訊，請參閱 Amazon EC2 Update — 其他執行個體類型、Nitro 系統和CPU選項。

> **Note**
>
> 下列遷移程序可以在 Windows Server 2008 R2 版和更新版本上執行。若要將 Linux 執行個體遷移至最新一代執行個體類型，請參閱the section called “實例類型變更”。

內容

- 第 1 部分：安裝和升級 AWS PV 驅動程序
- 第 2 部分：安裝和升級 ENA
- 第 3 部分：升級 AWS NVMe驅動程序
- 第 4 部分：更新EC2Config和 EC2Launch
- 第 5 部分：為 Bare Metal 執行個體安裝序列埠驅動程式
- 第 6 部分：更新電源管理設定
- 第 7 部分：更新新執行個體類型的 Intel 晶片組驅動程式
- （替代）使用升級 AWS PVENA，和NVMe驅動程序 AWS Systems Manager

> **Note**
>
> 或者，您可以使用 `AWSSupport-UpgradeWindowsAWSDrivers` 自動化文件，以自動執行第 1 部分、第 2 部分及第 3 部分中所述的程序。如果您選擇使用自動程序，請參閱（替代）使用升級 AWS PVENA，和NVMe驅動程序 AWS Systems Manager，然後繼續進行第 4 部分及第 5 部分。

開始之前

此程序假設您目前正在執行上一代 Xen 型執行個體類型 (例如 M4 或 C4)，而且您正在移轉至 Nitro System 上建置的執行個體。 AWS

您必須使用 3.0 或更新 PowerShell 版本才能成功執行升級。

> **Note**
> 移轉至最新一代執行個體時，現有的靜態 IP 或自訂DNS網路設定ENI可能會遺失，因為執行個體預設為新的增強型網路介面卡裝置。

在遵循此程序中的步驟之前，我們建議您建立該執行個體的備份。從EC2主控台選擇需要移轉的執行個體，開啟內容 (按一下滑鼠右鍵) 功能表，然後選擇「執行個體狀態」>「停止」。

> **Warning**
> 停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。若要保留執行個體存放區磁碟區上的任何資料，請確定您將資料備份至持久性儲存。

在EC2主控台中開啟執行個體的內容 (按一下滑鼠右鍵) 選單，選擇 [映像]，然後選擇 [建立映像]。

> **Note**
> 將執行個體類型移轉或變更為最新一代後，即可完成這些指示的第 4 和第 5 部分。不過，如果您要專門移轉至裸機執行個體類型，建議您在移轉前先完成它們。

第 1 部分：安裝和升級 AWS PV 驅動程序

雖然 Nitro 系統中未使用 AWS 光伏驅動程式，但如果您使用的是舊版 Citrix PV 或 AWS PV，您仍應該升級它們。最新的 AWS PV 驅動程式會解決之前版本驅動程式中的錯誤，這些錯誤可能會在您使用 Nitro 系統，或者需要遷移回 Xen 執行個體時出現。最佳作法是，我們建議您一律開啟 Windows 執行個體更新至最新的驅動程式 AWS。

請使用下列程序在視窗伺服器 2008 R2、視窗伺服器 2012 年 R2、視窗伺服器 2012 R2、視窗伺服器 2012 R2、視窗伺服器 2012 年 R2、視窗伺服器 2016 年或視窗伺服器 2019 上執行 AWS 光伏驅

動程式的就地升級，或從思杰光伏驅動程式升級至光伏驅動 AWS 如需詳細資訊，請參閱升級 EC2 Windows 執行個體上的光伏驅動。

若要升級網域控制站，請參閱升級網域控制站 (AWS PV 升級)。

若要執行 AWS 光伏驅動程式的升級或升級

1. 使用遠端桌面連線到執行個體並為執行個體準備升級。在執行升級前將所有非系統磁碟離線。如果您正在執行 AWS PV 驅動程式的就地更新，則不需要執行此步驟。在服務主控台中將非必要的服務設為手動啟動。
2. 將最新的驅動程式套件下載到執行個體。
3. 解壓縮資料夾的內容，然後執行 `AWSPVDriverSetup.msi`。

執行之後MSI，執行個體會自動重新啟動並升級驅動程式。執行個體最多可能會在 15 分鐘內無法使用。

升級完成且執行個體在 Amazon EC2 主控台中通過兩項運作狀態檢查後，請使用遠端桌面連線至執行個體，並驗證是否已安裝新驅動程式。在裝置管理員中，於 Storage Controllers (儲存控制器) 下，尋找 AWS PV Storage Host Adapter (AWS PV 儲存體主機控制器)。確認驅動程式版本與驅動程式版本歷史記錄表格中列出的最新版本相同。如需詳細資訊，請參閱AWS 光伏驅動器封裝歷史。

第 2 部分：安裝和升級 ENA

升級到最新的彈性網路轉接器驅動程式，以確保支援所有網路功能。如果您啟動執行個體，且其尚未啟用增強型聯網，您必須在執行個體上下載並安裝必要的網路轉接器驅動程式。然後，設定 enaSupport 執行個體屬性以啟動增強型聯網。您只能在支援的執行個體類型上啟用此屬性，且必須安裝ENA驅動程式才能啟用此屬性。如需詳細資訊，請參閱在EC2執行個體ENA上啟用增強型聯網。

1. 將最新的驅動程式下載到執行個體。如果您需要舊版驅動程式，請參閱ENA驅動程式版本紀錄。
2. 將 zip 封存檔解壓縮。
3. 通過從解壓縮的文件夾中運行`install.ps1` PowerShell腳本來安裝驅動程序。

> **Note**
>
> 請以管理員身分執行 `install.ps1` 指令碼，以避免發生安裝錯誤。

4. 檢查您是否AMI已 enaSupport 激活。若否，請遵循在EC2執行個體ENA上啟用增強型聯網中的文件繼續進行。

第 3 部分：升級 AWS NVMe驅動程序

AWS NVMe驅動程序用於與 Amazon EBS 和SSD實例存儲卷進行交互，這些卷在 Nitro 系統中作為NVMe塊設備公開，以獲得更好的性能。

> **Important**
>
> 針對您在上一代執行個體 AWS NVMe上安裝或升級時，針對將執行個體遷移至最新一代執行個體類型而修改下列指示。

1. 將最新的驅動程式套件下載到執行個體。

   如果您需要舊版驅動程式，請參閱以NVMe驅動程式版本取得支援的版本。
2. 將 zip 封存檔解壓縮。
3. 執行 `dpinst.exe` 命令以安裝驅動程式。
4. 開啟PowerShell工作階段並執行下列命令：

   ```
   PS C:\> start rundll32.exe sppnp.dll,Sysprep_Generalize_Pnp -wait
   ```

   > **Note**
   >
   > 若要套用命令，您必須以系統管理員身分執行 PowerShell 工作階段。 PowerShell（x86）版本將導致錯誤。
   > 此命令僅在裝置驅動程式上執行 sysprep。它不會執行完整的 sysprep 準備。
5. Windows Server 2008 R2 和 Windows Server 2012 請關閉執行個體，將執行個體類型變更為最新一代的執行個體並啟動它，然後繼續進行第 4 部分。若您在遷移至最新的執行個體類型前，在上一代的執行個體類型上再次啟動了該執行個體，執行個體將不會啟動。對於其他支援的WindowsAMIs，您可以在裝置 sysprep 之後隨時變更執行個體類型。

第 4 部分：更新EC2Config和 EC2Launch

對於 Windows 執行個體，最新的EC2Launch公用程式EC2Config和公用程式會在 Nitro 系統上執行時(包括EC2裸機) 提供額外的功能和資訊。默認情況下，該EC2Config服務包含在視窗服務器 2016 AMIs之前。EC2LaunchEC2Config在視窗伺服器 2016 及更新版本上取代AMIs。

當EC2Config和EC2Launch服務更新時，新的 Windows AMIs 會 AWS 包含最新版本的服務。不過，您必須使用最新版本的AMIs和來更新您自己的 Windows EC2Config 和執行個體EC2Launch。

安裝或更新 EC2Config

1. 下載並解壓縮EC2Config安裝程式。
2. 執行 `EC2Install.exe`。如需完整的選項清單，請執行 `EC2Install` 並加入 `/?` 選項。根據預設，設定程式會顯示提示。若要執行命令且不加入提示，請使用 `/quiet` 選項。

如需詳細資訊，請參閱安裝 EC2Config 的最新版本。

安裝或更新 EC2Launch

1. 如果您已經在執行個體EC2Launch上安裝並設定，請備份EC2Launch組態檔案。安裝程序不會保留此檔案的變更內容。根據預設，檔案位於 `C:\ProgramData\Amazon\EC2-Windows\Launch\Config` 目錄。
2. 將 EC2-Windows-Launch.zip 下載至執行個體上的目錄。
3. 下載 install.ps1 到您下載 `EC2-Windows-Launch.zip` 的同一個目錄。
4. 執行 `install.ps1`。

   > **Note**
   >
   > 請以管理員身分執行 `install.ps1` 指令碼，以避免發生安裝錯誤。

5. 如果您已備份組EC2Launch態檔案，請將其複製到目`C:\ProgramData\Amazon\EC2-Windows\Launch\Config`錄中。

如需詳細資訊，請參閱在 EC2 Windows 執行個體啟動期間，使用 EC2Launch v1 代理程式執行工作。

第 5 部分：為 Bare Metal 執行個體安裝序列埠驅動程式

`i3.metal`執行個體類型使用PCI基於序列裝置，而不是以 I/O 連接埠為基礎的序列裝置。最新的 Windows AMIs 會自動使用PCI基於串行設備並安裝串行端口驅動程序。如果您沒有使用從 Amazon 提供的 Windows AMI 2018.04.11 或更新版本啟動的執行個體，您必須安裝序列埠驅動程式，才能為序列裝置啟用密碼產生和主控台輸出等EC2功能。最新的EC2Config和EC2Launch實用程序還支持 i3.metal 並提供額外的功能。若您尚未這樣做，請遵循第 4 部分中的步驟。

安裝序列埠驅動程式

1. 將序列驅動程式套件下載到執行個體。
2. 擷取資料夾內容、針對 `aws_ser.INF` 開啟內容 (按一下右鍵) 功能表，然後選擇 install (安裝)。
3. 選擇 Okay (確定)。

第 6 部分：更新電源管理設定

電源管理設定的下列更新會設定永不關閉顯示器，讓 Nitro 系統上的作業系統能正常關機。截至 2018.11.28 Amazon AMIs 提供的所有窗口都已經具有此默認配置。

1. 開啟命令提示字元或 PowerShell 工作階段。
2. 執行下列命令：

```
powercfg /setacvalueindex 381b4222-f694-41f0-9685-ff5bb260df2e 7516b95f-
f776-4464-8c53-06167f40cc99 3c0bc021-c8a8-4e07-a973-6b14cbcb2b7e 0
powercfg /setacvalueindex 8c5e7fda-e8bf-4a96-9a85-a6e23a8c635c 7516b95f-
f776-4464-8c53-06167f40cc99 3c0bc021-c8a8-4e07-a973-6b14cbcb2b7e 0
powercfg /setacvalueindex a1841308-3541-4fab-bc81-f71556f20b4a 7516b95f-
f776-4464-8c53-06167f40cc99 3c0bc021-c8a8-4e07-a973-6b14cbcb2b7e 0
```

第 7 部分：更新新執行個體類型的 Intel 晶片組驅動程式

`u-6tb1.metalu-9tb1.metal`、和`u-12tb1.metal`執行個體類型使用的硬體需要先前未安裝在 Windows 上的晶片組驅動程式AMIs。如果您不是使用從 Amazon 提供的 Windows AMI 2018.11.19 或更新版本啟動的執行個體，您必須使用 Intel 晶片組公用程式來安裝驅動程式。INF

安裝晶片組驅動程式

1. 下載晶片組公用程式到執行個體。
2. 將檔案解壓縮。
3. 執行 `SetupChipset.exe`。
4. 接受 Intel 軟體授權合約，並安裝晶片組驅動程式。
5. 重新啟動執行個體。

（替代）使用升級 AWS PVENA，和NVMe驅動程序 AWS Systems Manager

`AWSSupport-UpgradeWindowsAWSDrivers` 自動化文件可以自動執行第 1 部分、第 2 部分及第 3 部分中所述的步驟。此方法也可以修復驅動程式升級失敗的執行個體。

`AWSSupport-UpgradeWindowsAWSDrivers`自動化文件會升級或修復指定EC2執行個體上的儲存裝置和網路 AWS 驅動程式。此文件會嘗試透過呼叫代理程 AWS Systems Manager 式 (代理程式) 線上安裝最新版本的 AWS 驅動程SSM式。如果SSM代理程式無法連絡，文件可以在明確要求時執行 AWS 驅動程式的離線安裝。

> **Note**
>
> 此程序將無法在網域控制器上進行。若要更新網域控制器上的驅動程式，請參閱升級網域控制站 (AWS PV 升級)。

若要使用自動升級 AWS PV ENA、和NVMe驅動程式 AWS Systems Manager

1. 在 https://console.aws.amazon.com/系 Systems Manager 中打開系統管理器控制台。
2. 選擇 Automation (自動化)、Execute Automation (執行自動化)。
3. 搜尋並選取 AWSSupport-UpgradeWindows AWSDrivers 自動化文件，然後選擇 [執行自動化]。
4. 在「輸入參數」區段中，設定下列選項：

   執行個體 ID

   輸入要升級之執行個體的唯一 ID。

   AllowOffline

   (選用) 選擇下列其中一個選項：

   - `True` – 選擇此選項來執行離線安裝。升級過程中執行個體會停止並重新啟動。

     > **Warning**
     >
     > 停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。若要保留執行個體存放區磁碟區上的任何資料，請確定您將資料備份至持久性儲存。

   - `False` – (預設值) 若要執行線上安裝，請將此選項保持選取狀態。升級過程中執行個體會重新啟動。

> **Important**
>
> 線上和離線升級會在嘗試升級作業AMI之前建立一個。自動化完成後AMI仍然存在。保護您對的存取權限AMI，或在不再需要時將其刪除。

SubnetId

(選用) 輸入下列其中一個值：

- `SelectedInstanceSubnet` – (預設值)升級程序會將 helper 執行個體啟動至與要升級之執行個體相同的子網路。子網必須允許與 Systems Manager 端點 (`ssm.*`) 進行通訊。
- `CreateNewVPC`— 升級程序會將協助程式執行個體啟動至新的執行個體VPC。如果您不確定目標執行個體的子網是否允許與 `ssm.*` 端點進行通訊，請使用此選項。您的使用者必須具有建立VPC.
- 特定子網路 ID – 指定要將 helper 執行個體啟動至其中之特定子網路的 ID。子網必須與要升級的執行個體位於相同的可用區域，而且它必須允許與 `ssm.*` 端點進行通訊。

5. 選擇 Execute (執行)。
6. 允許升級以完成。完成線上升級最多可能需要 10 分鐘，而完成離線升級最需要多 25 分鐘。

疑難排解 EC2 Windows 執行個體上的作業系統升級

AWS 針對升級協助程式服務的問題或問題提供升級支援，此 AWS 公用程式可協助您執行涉及 Citrix PV 驅動程式的就地升級。

升級之後，執行個體可能會暫時遇到高於平均CPU使用率，而. NET執行階段最佳化服務會最佳化. NET框架。這是預期的行為。

若執行個體在數個小時之後仍無法通過兩項狀態檢查，請檢查以下項目。

- 若您升級到 Windows Server 2008，兩項狀態檢查在數個小時之後仍然失敗，升級可能已失敗，並會顯示提示，讓您 Click OK (按一下確認) 以確認還原。因為無法在此狀態存取主控台，沒有任何方式能按一下按鈕。為了解決這個問題，請通過 Amazon EC2 控制台或執行重新啟動API。初始化重新開機可能需要十分鐘以上。執行個體可能會在 25 分鐘可用。
- 從伺服器移除應用程式或伺服器角色並再試一次。

若執行個體在從伺服器移除應用程式或伺服器角色後仍然無法通過兩項狀態檢查，請執行下列作業。

- 停止執行個體，並將根磁碟區連接到另一個執行個體。如需詳細資訊，請參閱等待中繼資料服務中如何停止並將根磁碟區連接到另一個執行個體的說明。
- 分析 Windows 安裝日誌檔案及事件日誌，查看故障。

針對其他作業系統升級或遷移的問題，我們建議檢閱在您開始就地升級之前中列出的文章。

教學：將 Amazon EC2 執行個體 Connect 到 Amazon RDS 資料庫

教學課程目標

本教學的目標是學習如何使用設定 Amazon EC2 執行個體和 Amazon RDS 資料庫之間的安全連線 AWS Management Console。

設定連線有不同的選項。在本教學課程中，我們將探討以下三個選項：

- 選項 1：使用EC2主控台自動將執行個體連線至資RDS料庫

  使用EC2主控台中的自動連線功能，自動設定EC2執行個體與RDS資料庫之間的連線，以允許EC2執行個體和RDS資料庫之間的流量。
- 選項 2：使用RDS主控台自動將執行個體連線至資RDS料庫

  使用RDS主控台中的自動連線功能，自動設定EC2執行個體與RDS資料庫之間的連線，以允許EC2執行個體和RDS資料庫之間的流量。
- 選項 3：建立安全群組，以手動方式將執行處理連線至RDS資料庫

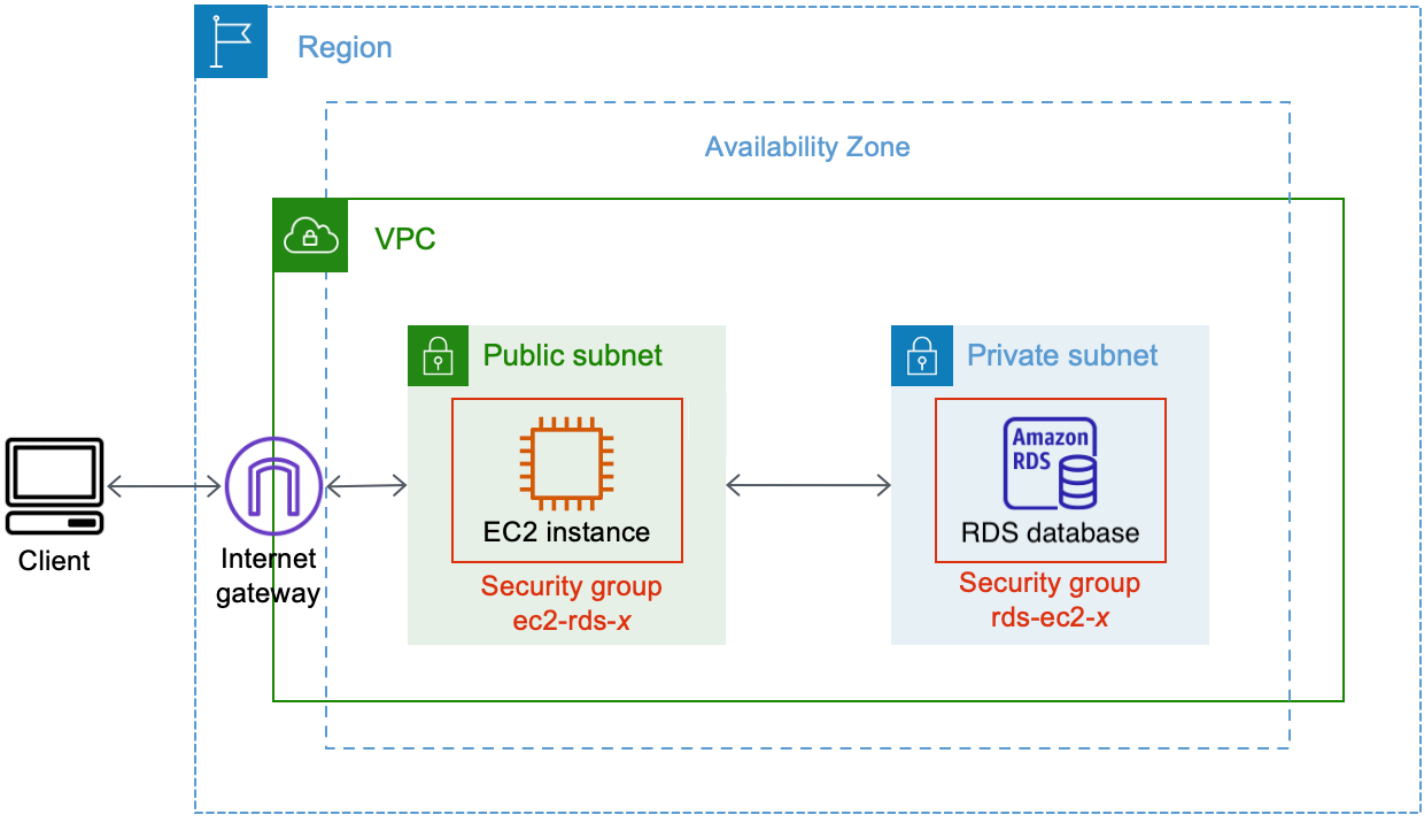
  透過手動設定並指派安全群組，以重現選項 1 和選項 2 中自動連線功能自動建立的組態，設定EC2執行個體與RDS資料庫之間的連線。

Context

為什麼您想要設定EC2執行個體和RDS資料庫之間的連線，讓我們考慮下列情況：您的網站提供表單給使用者填寫。您需要在資料庫中擷取表單資料。您可以在設定為 Web 伺服器的EC2執行個體上託管您的網站，也可以擷取資料RDS庫中的表單資料。EC2實例和數據RDS庫需要相互連接，以便表單數據可以從EC2實例到數據RDS庫。本教學課程會說明如何設定該連線。請注意，這只是連接EC2執行個體和RDS資料庫的使用案例的其中一個範例。

架構

下圖展示了所建立的資源，以及完成本教學課程中的所有步驟後產生的架構組態。



此圖展示了您將建立的下列資源：

- 您將在相同的 AWS 區域、VPC和可用區域中建立EC2執行個體和RDS資料庫。
- 您將在公有子網路中建立EC2執行個體。
- 您將在私有子網路中建立RDS資料庫。

  當您使用RDS主控台建立RDS資料庫並自動連線EC2執行個體時，會自動選取資料庫的資料庫子網路群組和公用存取設定。VPC資RDS料庫會自動建立在與EC2執行個體相同VPC的私有子網路中。
- 網際網路使用者可以使用SSH或HTTP/HTTPS透過網際網路閘道連線至EC2執行個體。
- 網際網路使用者無法直接連線至RDS資料庫；只有EC2執行個體會連線至資RDS料庫。
- 當您使用自動連線功能允許EC2執行個體和RDS資料庫之間的流量時，系統會自動建立並新增下列安全群組：
  - 安全性群組 ec2-RDS-*x*已建立並新增至EC2執行個體。它有一個引用 rds-ec2-的出站規則*x*安全群組做為其目的地。這允許來自EC2實例的流量使用 rds-ec2-到達RDS數據庫*x*安全性群組。

- 安全性群組 rds-ec2-*x*已建立並新增至資RDS料庫。它有一個引用 ec2-rds-的入站規則*x*安全群組做為其來源。這允許來自帶有 ec2-EC2 rds-執行個體的流量*x*安全組到達數RDS據庫。

透過使用不同的安全性群組 (一個用於EC2執行個體，另一個用於RDS資料庫)，您可以更好地控制執行個體和資料庫的安全性。如果您要在執行個體和資料庫上使用相同的安全群組，然後修改安全群組以適應資料庫，則修改會同時影響執行個體和資料庫。換句話說，如果您使用一個安全群組，您可能會無意中修改資源 (執行個體或資料庫) 的安全性，因為您忘記了安全群組已連接至該資源。

自動建立的安全群組也會遵守最低權限，因為其只允許透過建立特定於工作負載的安全群組對，在資料庫連接埠上進行此工作負載的相互連線。

考量事項

完成本教學課程中的任務後，請考量以下內容：

- 兩個主控台 - 在本教學課程中，您將使用下列兩個主控台：
  - Amazon EC2 主控台 — 您將使用EC2主控台啟動執行個體、自動將EC2執行個體連線到RDS資料庫，以及使用手動選項，透過建立安全群組來設定連線。
  - Amazon RDS 主控台 — 您將使用RDS主控台建立RDS資料庫，並自動將EC2執行個體連線到RDS資料庫。
- One VPC — 若要使用自動連線功能，您的EC2執行個體和RDS資料庫必須相同VPC。

  如果您要手動設定EC2執行個體與RDS資料庫之間的連線，您可以在其中一個執行個體啟動EC2執行個VPC體，而在另一個執行個體啟動RDS資料庫VPC；不過，您需要設定額外的路由和設VPC定。這是本教學課程中涵蓋的情況。
- One AWS 區域 — EC2 執行個體和RDS資料庫必須位於相同的區域。
- 兩個安全群組 — EC2 執行個體與RDS資料庫之間的連線是由兩個安全性群組設定的：EC2執行個體的安全性群組，以及資料庫的安全性群組。RDS

  當您使用EC2主控台或RDS主控台中的自動連線功能設定連線時 (本教學課程的選項 1 和選項 2)，安全性群組會自動建立並指派給EC2執行個體和RDS資料庫。

  如果您不使用自動連線功能，則需要手動建立並指派安全群組。您可以在本教學課程的選項 3 中執行此操作。

教學課程所需時間

30 分鐘

您可以一次性完成整個教學課程，也可以一次完成一個任務。

成本

完成此教學課程後，您可能會為所建立的 AWS 資源產生費用。

您可以在免費方案EC2下使用 Amazon，前提是您的 AWS 帳戶少於 12 個月，並且您可以根據免費方案要求設定資源。

如果您的EC2執行個體和資料RDS庫位於不同的可用區域，則會產生資料傳輸費用。為避免產生這些費用，EC2執行個體和RDS資料庫必須位於相同的可用區域中。如需資料傳輸費用的相關資訊，請參閱 Amazon EC2 隨需定價頁面上的資料傳輸。

若要避免在完成教學課程後產生費用，請務必刪除不再需要的資源。如需刪除資源的步驟，請參閱 工作 4 (選擇性)：清除。

選項 1：使用EC2主控台自動將執行個體連線至資RDS料庫

選項 1 的目標是探索EC2主控台中的自動連線功能，該功能會自動設定EC2執行個體和RDS資料庫之間的連線，以允許從執EC2行個體到資料庫的流量。RDS在選項 3 中，您將了解如何手動設定連線。

任務

- 開始之前
- 工作 1 (選擇性)：建立RDS資料庫
- 工作 2 (選擇性)：啟動EC2執行個體
- 工作 3：自動將EC2執行個體連線至資RDS料庫
- 任務 4：驗證連線組態
- 工作 5 (選擇性)：清除

開始之前

完成此教學課程需要以下項目：

- 與執行個體位於相同VPC的資RDS料EC2庫。您可以使用現有的RDS資料庫，或遵循工作 1 中的步驟來建立新的RDS資料庫。
- 與RDS資料庫位於相同的EC2執VPC行個體。您可以使用現有的EC2執行個體，也可以依照工作 2 中的步驟建立新的EC2執行個體。
- 呼叫以下操作的許可：
  - `ec2:AssociateRouteTable`
  - `ec2:AuthorizeSecurityGroupEgress`
  - `ec2:CreateRouteTable`
  - `ec2:CreateSecurityGroup`
  - `ec2:CreateSubnet`
  - `ec2:DescribeInstances`
  - `ec2:DescribeNetworkInterfaces`
  - `ec2:DescribeRouteTables`
  - `ec2:DescribeSecurityGroups`
  - `ec2:DescribeSubnets`
  - `ec2:ModifyNetworkInterfaceAttribute`
  - `ec2:RevokeSecurityGroupEgress`

工作 1 (選擇性)：建立RDS資料庫

> **Note**
>
> 創建 Amazon RDS 數據庫不是本教程的重點。如果你已經有一個RDS數據庫，並希望在本教程中使用它，你可以跳過這個任務。
> 如果您使用現有的RDS資料庫，請確定它與您的EC2執行個體位於VPC相同的資料庫中，以便您可以使用自動連線功能。

此任務的目標是創建一個RDS數據庫，以便您可以完成任務 3，您可以在其中配置EC2實例和RDS數據庫之間的連接。此工作中的步驟設定RDS資料庫，如下所示：

- 發動機類型：我的 SQL
- 範本：免費方案
- 資料庫執行個體識別符：**`tutorial-database-1`**

- 資料庫執行個體類別：`db.t3.micro`

> **Important**
>
> 在生產環境中，您應設定資料庫以滿足您的特定需求。

若要建立我的SQLRDS資料庫

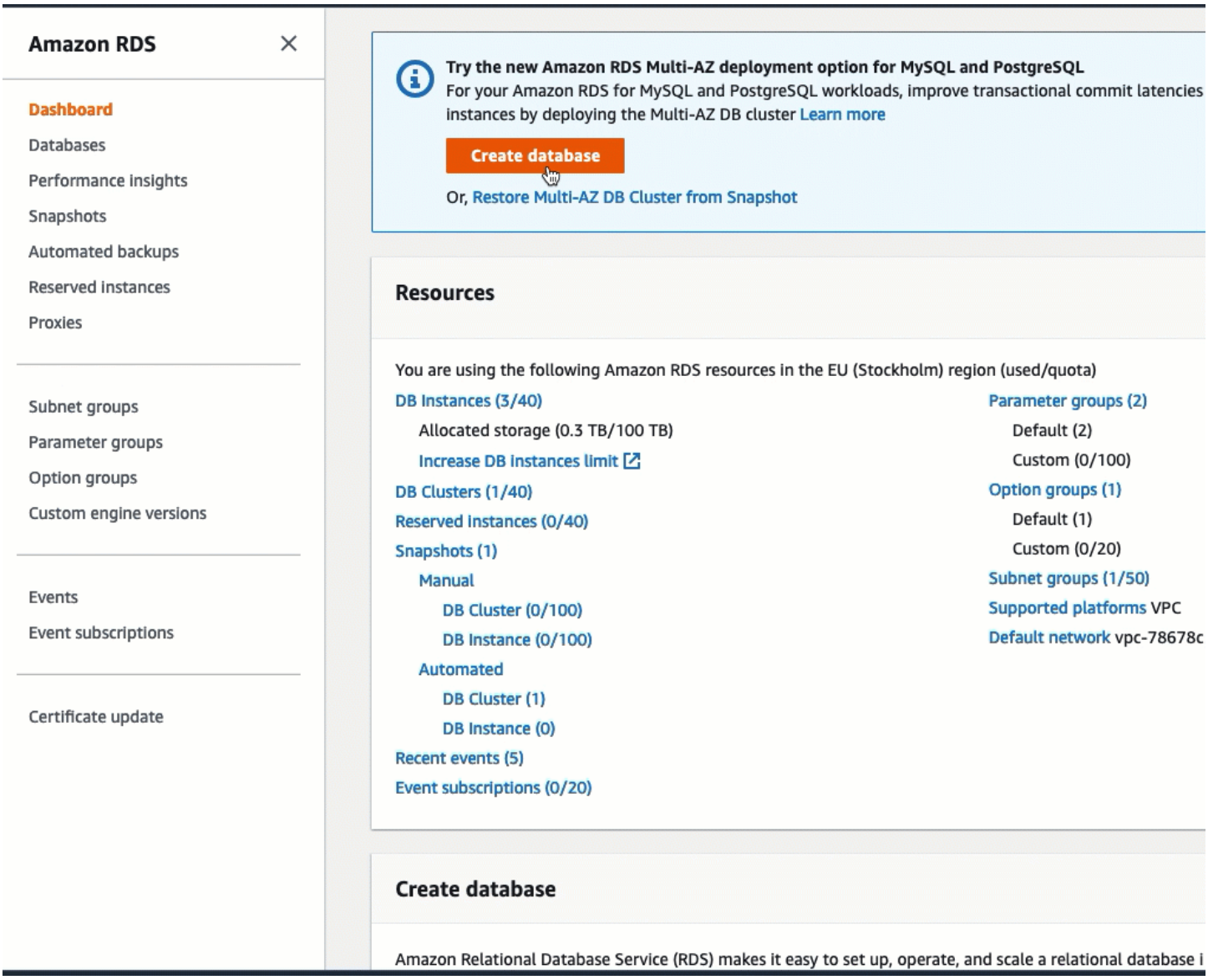
1. 在打開 Amazon RDS 控制台https://console.aws.amazon.com/rds/。
2. 從區域選取器 (位於右上角) 中選取 AWS 區域。資料庫和EC2執行個體必須位於相同的區域，才能使用EC2主控台中的自動連線功能。
3. 在儀表板中，選擇 Create database (建立資料庫)。
4. 在 Choose a database creation method (選取資料庫建立方法) 中，確認選取 Standard Create (標準建立)。如果您選擇「輕鬆建立」，則無法使用選VPC取器。您必須確定資料庫與EC2執行個體位於VPC相同的位置，才能使用EC2主控台中的自動連線功能。
5. 在 [引擎選項] 下，針對 [引擎類型] 選擇 [我] SQL。
6. 在 Templates (範本) 中，選擇符合您需求的範例範本。在本教學課程中，選擇免費層以免費建立RDS資料庫。但請注意，只有在您的帳戶使用未滿 12 個月時，才能使用免費方案。其他限制適用。在 Free tier (免費方案) 方塊中選擇 Info (資訊) 連結，可閱讀更多資訊。
7. 在 Settings (設定) 下，執行下列動作：
   a. 在 DB instance identifier (資料庫執行個體識別符) 中，輸入資料庫的名稱。針對本教學，輸入 **`tutorial-database-1`**。
   b. 對於 Master username (主要使用者名稱)，請保留預設名稱，即 **`admin`**。
   c. 對於 Master password (主要密碼)，請輸入您在本教學課程中可以記住的密碼，然後在 Confirm password (確認密碼) 中再次輸入密碼。
8. 在「執行個體組態」下，對於資料庫執行個體類別，保留預設值，即 db. t3.micro。 如果您的帳戶少於 12 個月，則可以免費使用此數據庫類。其他限制適用。如需詳細資訊，請參閱AWS 免費方案。
9. 在 [連線] 下方，針對 [計算資源]，選擇 [不連線到EC2計算資源]，因為稍後會在工作 3 中連線EC2執行個體和資RDS料庫。

   稍後，在本教學課程的選項 2 中，您將選擇 [連線到EC2計算資源]，在RDS主控台中試用自動連線功能。)

10. 對於虛擬私有雲 (VPC)，請選擇一個VPC. VPC必須具有資料庫子網路群組。若要使用自動連線功能，您的EC2執行個體和RDS資料庫必須相同VPC。
11. 請保留本頁面中其他欄位的所有預設值。
12. 選擇建立資料庫。

    在 Databases (資料庫) 畫面中，新資料庫的 Status (狀態) 為 Creating (正在建立)，直到資料庫準備就緒可供使用為止。狀態變更為 Available (可用) 時，您便可連線至資料庫。視資料庫類別和儲存體數量而定，可能需要最多 20 分鐘的時間，新資料庫才會可用。

檢視動畫：建立RDS資料庫



工作 2 (選擇性)：啟動EC2執行個體

> **Note**
>
> 啟動執行個體並不是本教學課程的重點。如果您已經擁有 Amazon EC2 執行個體，並且想要在本教學中使用它，則可以略過此任務。
> 如果您使用現有的EC2執行個體，請確定它與RDS資料庫位於VPC相同的執行個體，以便您可以使用自動連線功能。

此任務的目標是啟動EC2執行個體，以便您可以完成任務 3，在其中設定EC2執行個體和 Amazon RDS 資料庫之間的連線。此工作中的步驟設定執EC2行個體，如下所示：

- 執行個體名稱：**`tutorial-instance-1`**
- AMI: Amazon 2
- 執行個體類型：`t2.micro`
- 自動指派公有 IP：已啟用
- 具有以下三個規則的安全群組：
  - SSH從您的 IP 位址允許
  - 允許任何地方的HTTPS流量
  - 允許任何地方的HTTP流量

> **Important**
>
> 在生產環境中，您應設定執行個體以滿足您的特定需求。

啟動 EC2 執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 從區域選取器 (位於右上角) 中選取 AWS 區域。執行個體和資RDS料庫必須位於相同的區域，才能使用EC2主控台中的自動連線功能。
3. 在EC2儀表板上，選擇 [啟動執行個體]。

4. 在 Name and tags (名稱和標籤) 下，針對 Name (名稱)，輸入用於識別執行個體的名稱。針對本教學課程，將執行個體命名為 **`tutorial-instance-1`**。雖然執行個體名稱不是必要的，但是當您在EC2主控台中選取執行個體時，名稱可協助您輕鬆識別它。
5. 在「應用程式和作業系統映像」下，選擇AMI符合您 Web 伺服器需求的影像。本教學課程使用 Amazon Linux 2。
6. 在 Instance type (執行個體類型) 中，針對 Instance type (執行個體類型)，選取符合您 Web 伺服器需求的執行個體類型。本教學課程使用的是 `t2.micro`。

   > **Note**
   >
   > 您可以在免費方`t2.micro`案EC2下使用 Amazon，前提是您的 AWS 帳戶年齡少於 12 個月，並且您可以選擇`t2.micro`執行個體類型（或`t3.micro`在無法使用的區域）。

7. 在 key pair (login) (金鑰對 (登入)) 中，針對 Key pair name (金鑰對名稱) 選擇您的金鑰對。
8. 在Network settings (網路設定) 中，執行下列操作：

   a. 對於網路和子網路，如果您尚未變更預設VPC或子網路，則可以保留預設設定。

      如果您已變更預設VPC或子網路，請檢查下列項目：

      i. 執行個體必須與RDS資料庫位於VPC相同，才能使用自動連線功能。默認情況下，您只有一個VPC。

      ii. 啟動執行VPC個體時，必須連接網際網路閘道，才能從網際網路存取 Web 伺服器。您的預設VPC值會透過網際網路閘道自動設定。

      iii. 若要確保您的執行個體接收公有 IP 地址，針對 Auto-assign public IP (自動指派公有 IP)，檢查是否已選取 Enable (啟用)。如果選取 Disable (停用)，則請選取 Edit (編輯) (位於 Network Settings (網路設定) 右側)，然後針對 Auto-assign public IP (自動指派公有 IP) 選取 Enable (啟用)。

   b. 若要使用連線至執行個體SSH，您需要一個安全群組規則來授權 SSH (Linux) 或 RDP (Windows) 流量來自您電腦的公共IPv4位址。根據預設，當您啟動執行個體時，會建立新的安全群組，其中包含允許來自任何地方的輸入SSH流量的規則。

      若要確保只有您的 IP 位址可以連線到您的執行個體，請在 [防火牆 (安全群組)] 下，從 [允許SSH傳輸來源] 核取方塊旁的下拉式清單中選擇 [我的 IP]。

   c. 若要允許流量從網際網路傳輸到您的執行個體，請選取下列核取方塊：

      - 允許來自HTTPs網際網路的流量

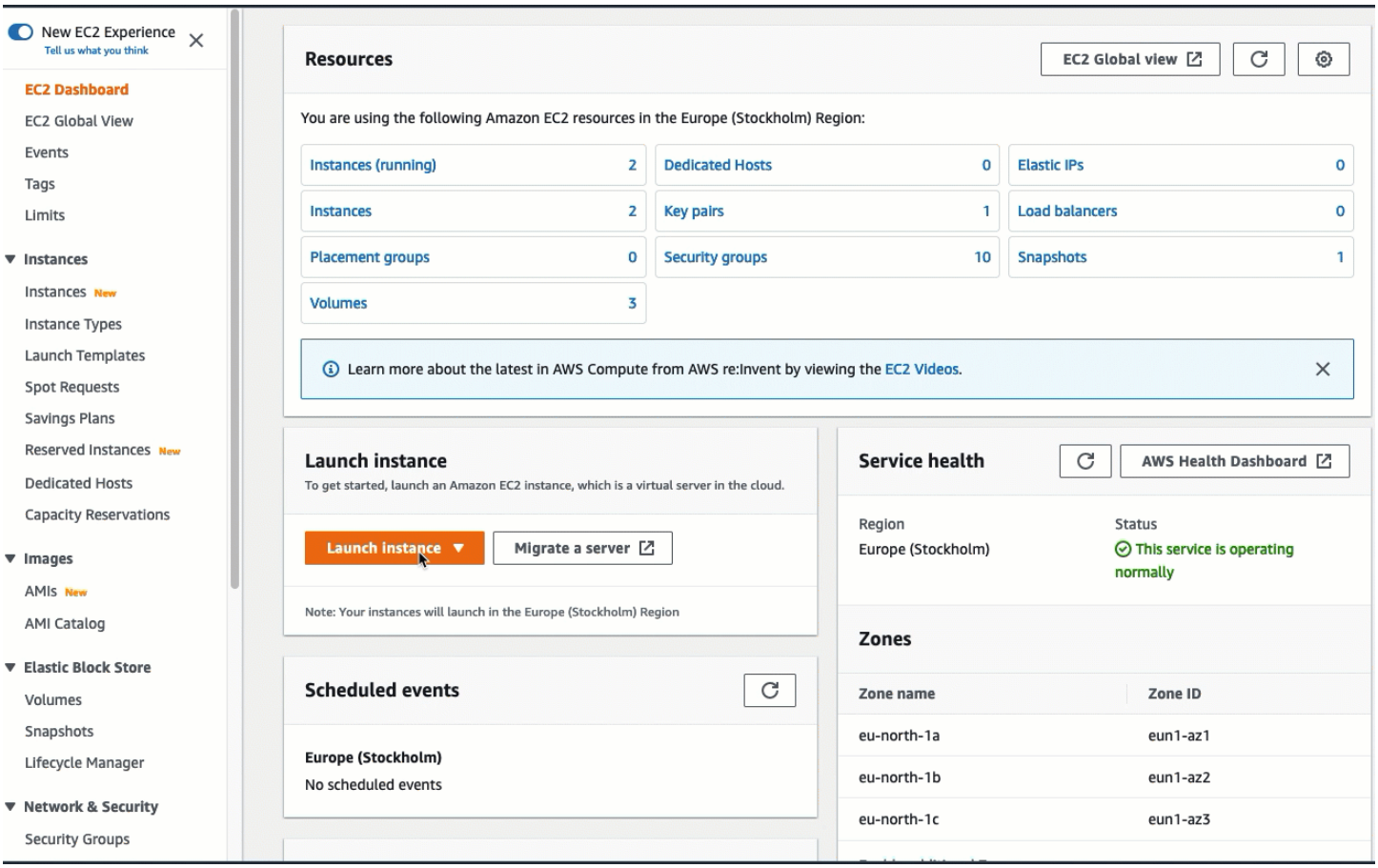
- 允許來自HTTP網際網路的流量

9. 在 Summary (摘要) 面板中，檢閱您的執行個體組態，然後選擇 Launch instance (啟動執行個體)。
10. 保持確認頁面開啟。當您自動將執行個體連線至資料庫時，下一個任務會需要它。

    如果執行個體無法啟動或狀態立即進入 `terminated` 而非 `running`，請參閱解決 Amazon EC2 執行個體啟動問題。

如需有關啟動執行個體的詳細資訊，請參閱 使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體。

檢視動畫：啟動EC2實體



工作 3：自動將EC2執行個體連線至資RDS料庫

這項工作的目標是使用EC2主控台中的自動連線功能，自動設定EC2執行個體與RDS資料庫之間的連線。

使用EC2主控台自動將EC2執行處理連線至資RDS料庫

1. 在執行個體啟動確認頁面 (應該從先前的工作開啟) 上，選擇 [Connect RDS 資料庫]。

   如果已關閉確認頁面，則請按照下列步驟操作：

   a. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
   b. 在導覽窗格中，選擇 Instances (執行個體)。
   c. 選取您剛建立的EC2執行個體，然後選擇 [動作]、[網路連線]、[Connect RDS 資料庫]。

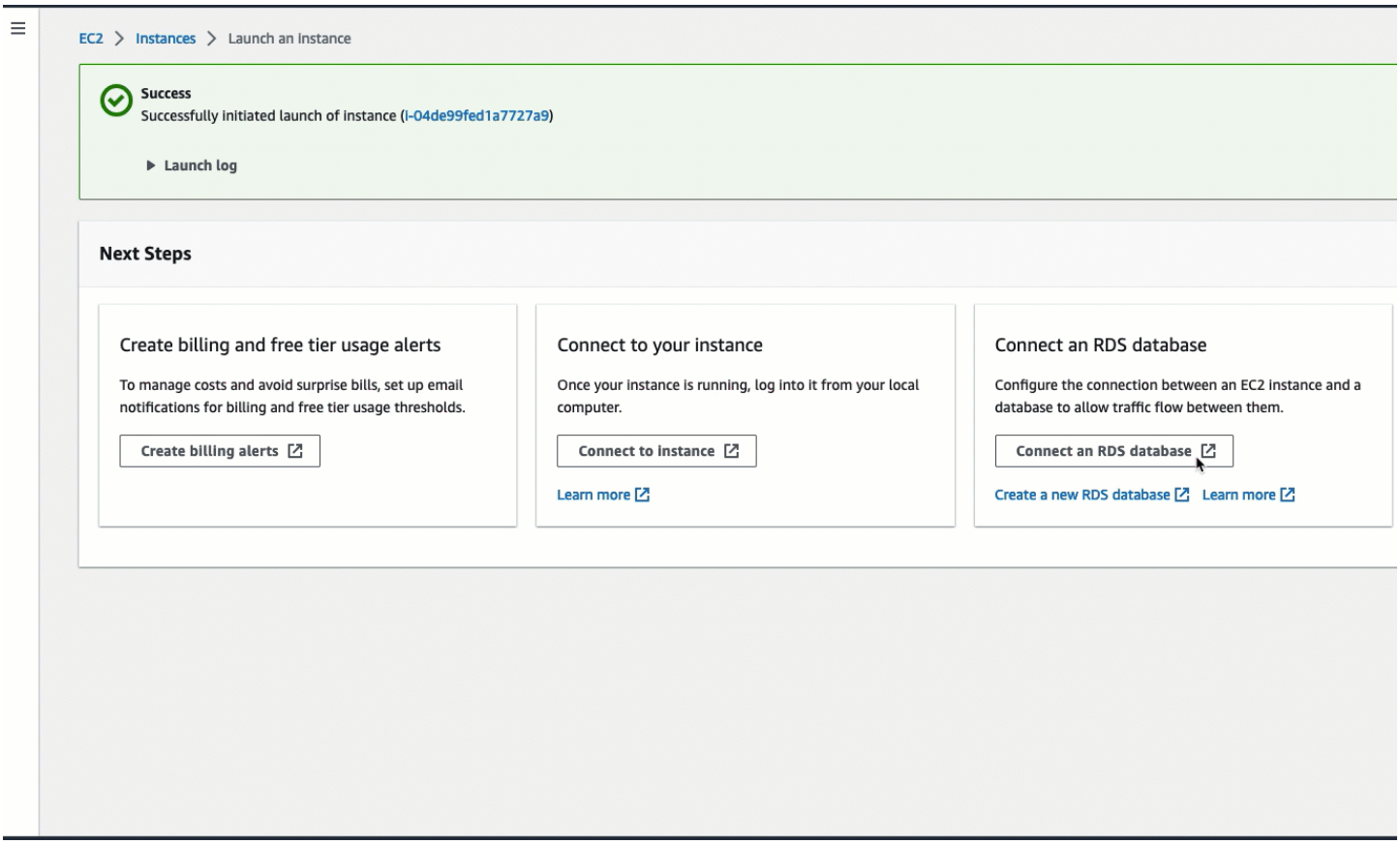
      如果「Connect RDS 資料庫」無法使用，請檢查EC2執行處理是否處於「執行中」狀態。

2. 對於 Database role (資料庫角色)，請選擇 Instance (執行個體)。在這種情況下，Instance (執行個體) 是指資料庫執行個體。
3. 對於RDS資料庫，請選擇您在工作 1 中建立的RDS資料庫。

   > **Note**
   > EC2執行個體和RDS資料庫必須相同VPC，才能彼此連線。

4. 選擇連線。

檢視動畫：自動將新啟動的執行個EC2體連線至資料庫 RDS



任務 4：驗證連線組態

此任務的目標是確認已建立兩個安全群組並指派給執行個體和資料庫。

當您使用主控台中的自動連線功能來設定連線時，會自動建立安全群組並將其指派給執行個體和資料庫，如下所示：

- 安全性群組 rds-ec2-*x*已建立並新增至資RDS料庫。它有一個引用 ec2-rds-的入站規則*x*安全群組做為其來源。這允許來自帶有 ec2-EC2 rds-執行個體的流量*x*安全組到達數RDS據庫。
- 安全性群組 ec2-RDS-*x*已建立並新增至EC2執行個體。它有一個引用 rds-ec2-的出站規則*x*安全群組做為其目的地。這允許來自EC2實例的流量使用 rds-ec2-到達RDS數據庫*x*安全性群組。

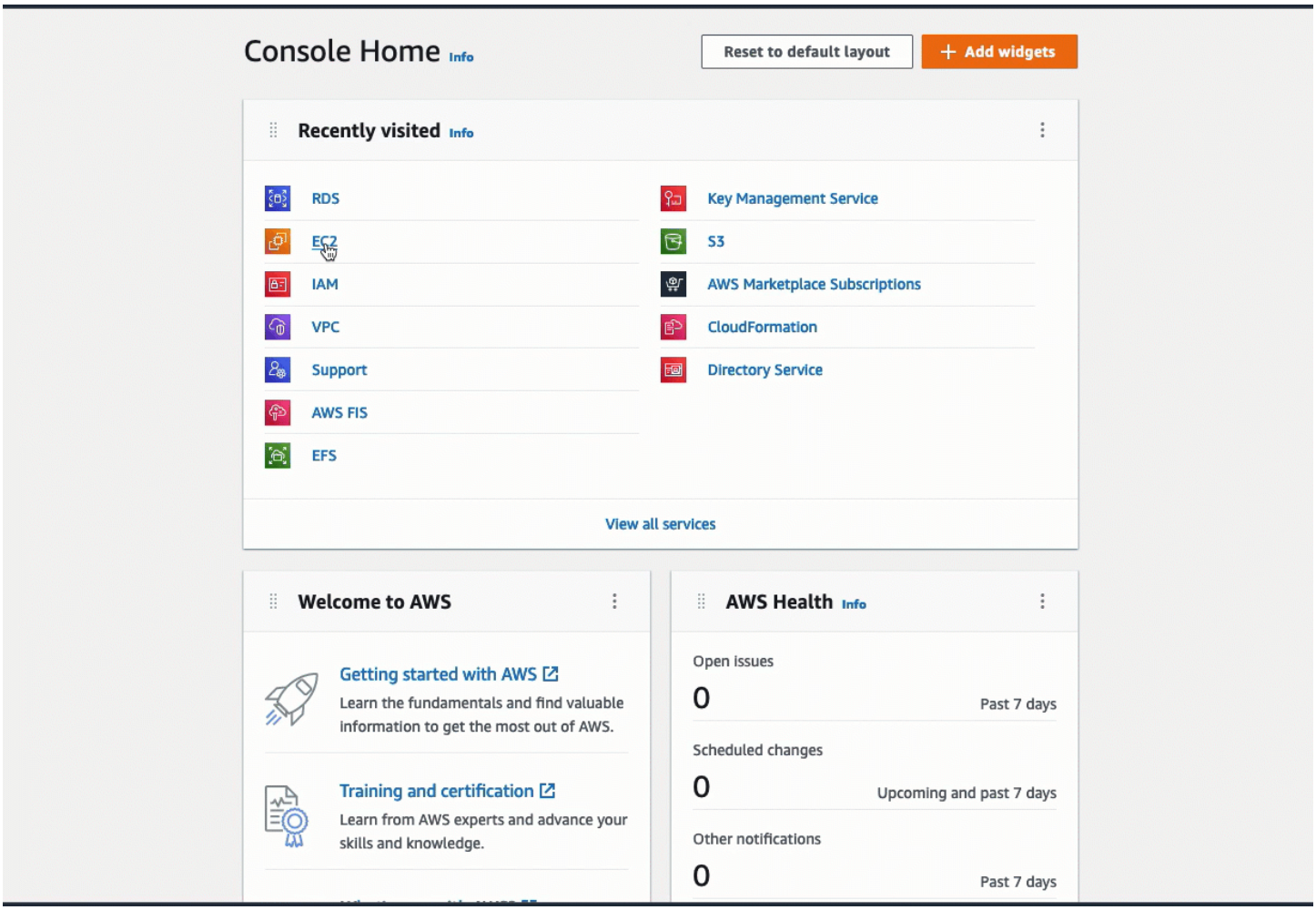
若要使用主控台驗證連線組態

1. 在打開 Amazon RDS 控制台https://console.aws.amazon.com/rds/。
2. 在導覽頁面中，選擇 Databases (資料庫)。
3. 選擇您為此教學課程建立的資RDS料庫。

4. 在 [連線與安全性] 索引標籤的 [安VPC全性] 下，確認名為 rds-ec2-的安全群組***x***顯示。
5. 選擇正確的方法***x***安全性群組。主控台中的「安全群組」畫EC2面隨即開啟。
6. 選擇正確的方法***x***安全組打開它。
7. 選擇 Inbound Rules (傳入規則) 索引標籤。
8. 確認下列安全群組規則存在，如下所示：
   - 類型：MYSQL/Aurora
   - 連接埠範圍：3306
   - 來源：***sg-0987654321example*** /EC2-***x***— 這是指派給您在上述步驟中驗證之EC2執行個體的安全性群組。
   - 說明：允許EC2執行個體連線的規則 ***sg-1234567890example*** 附
9. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
10. 在導覽窗格中，選擇 Instances (執行個體)。
11. 選擇您在上一EC2個作業中選取要連線至RDS資料庫的執行處理，然後選擇「安全」頁籤。
12. 在 [安全性詳細資料]、[安全性群組] 下，確認名為 ec2-rds-的安全性群組***x***在列表中。*x* 是一個數字。
13. 選擇 EC2-***x***安全組打開它。
14. 選擇 Outbound rules (傳出規則) 索引標籤。
15. 確認下列安全群組規則存在，如下所示：
    - 類型：MYSQL/Aurora
    - 連接埠範圍：3306
    - 目標：***sg-1234567890example*** /rds-***x***
    - 說明：允許從此安全群組連接的任何執行個體到 **`database-tutorial`** 之連線的規則

透過確認這些安全群組和安全群組規則是否存在，以及這些規則是否已按照本程序中所述將它們指派給RDS資料庫和EC2執行處理，您可以使用自動連線功能驗證連線是否已自動設定。

檢視動畫：驗證連線組態



您已完成本教學課程的選項 1。您現在可以完成選項 2，其中會教導您如何使用RDS主控台將EC2執行個體自動連線到RDS資料庫，或者您可以完成選項 3，教您如何手動設定在選項 1 中自動建立的安全性群組。

工作 5 (選擇性)：清除

現在您已經完成教學課程，最好清理 (刪除) 您不再想要使用的任何資源。清理 AWS 資源可防止您的帳戶產生任何進一步的費用。

如果您特別針對本教學課程啟動EC2執行個體，您可以終止執行個體，以停止產生與其相關的任何費用。

欲使用主控台來終止執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。

2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取您為此教學課程建立的執行個體，然後選取 Instance state (執行個體狀態)、Terminate instance (終止執行個體)。
4. 出現確認提示時，請選擇 Terminate (終止)。

如果您專門為本教學課程建立了RDS資料庫，您可以刪除該資料庫，以停止產生與其相關的任何費用。

使用主控台刪除RDS資料庫

1. 在打開 Amazon RDS 控制台https://console.aws.amazon.com/rds/。
2. 在導覽窗格中，選擇 Databases (資料庫)。
3. 選取您為此教學課程建立的RDS資料庫，然後選擇動作，刪除。
4. 在方塊中輸入 **`delete me`**，然後選擇 Delete (刪除)。

選項 2：使用RDS主控台自動將執行個體連線至資RDS料庫

選項 2 的目標是探索RDS主控台中的自動連線功能，該功能會自動設定EC2執行個體和RDS資料庫之間的連線，以允許從執EC2行個體到資料庫的流量。RDS在選項 3 中，您將了解如何手動設定連線。

任務

- 開始之前
- 工作 1 (選用)：啟動EC2執行個體
- 工作 2：建立RDS資料庫並自動將其連線至執行個EC2體
- 任務 3：驗證連線組態
- 工作 4 (選擇性)：清除

開始之前

完成此教學課程需要以下項目：

- 與RDS資料庫位於相同的EC2執VPC行個體。您可以使用現有的EC2執行個體，也可以依照工作 1 中的步驟建立新的執行個體。
- 呼叫以下操作的許可：
  - `ec2:AssociateRouteTable`

- `ec2:AuthorizeSecurityGroupEgress`
- `ec2:CreateRouteTable`
- `ec2:CreateSecurityGroup`
- `ec2:CreateSubnet`
- `ec2:DescribeInstances`
- `ec2:DescribeNetworkInterfaces`
- `ec2:DescribeRouteTables`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeSubnets`
- `ec2:ModifyNetworkInterfaceAttribute`
- `ec2:RevokeSecurityGroupEgress`

工作 1 (選用)：啟動EC2執行個體

> **Note**
>
> 啟動執行個體並不是本教學課程的重點。如果您已經擁有 Amazon EC2 執行個體，並且想要在本教學中使用它，則可以略過此任務。

此任務的目標是啟動EC2執行個體，以便您可以完成任務 2，在其中設定EC2執行個體和 Amazon RDS 資料庫之間的連線。此工作中的步驟設定執EC2行個體，如下所示：

- 執行個體名稱：**`tutorial-instance-2`**
- AMI: Amazon 2
- 執行個體類型：`t2.micro`
- 自動指派公有 IP：已啟用
- 具有以下三個規則的安全群組：
  - SSH從您的 IP 位址允許
  - 允許任何地方的HTTPS流量
  - 允許任何地方的HTTP流量

> **Important**
>
> 在生產環境中，您應設定執行個體以滿足您的特定需求。

啟動 EC2 執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在EC2儀表板上，選擇 [啟動執行個體]。
3. 在 Name and tags (名稱和標籤) 下，針對 Name (名稱)，輸入用於識別執行個體的名稱。針對本教學課程，將執行個體命名為 **`tutorial-instance-2`**。雖然執行個體名稱不是必要的，但是當您在RDS主控台中選取執行個體時，名稱可協助您輕鬆識別它。
4. 在「應用程式和作業系統映像」下，選擇AMI符合您 Web 伺服器需求的影像。本教學課程使用 Amazon Linux。
5. 在 Instance type (執行個體類型) 中，針對 Instance type (執行個體類型)，選取符合您 Web 伺服器需求的執行個體類型。本教學課程使用的是 `t2.micro`。

   > **Note**
   >
   > 您可以在免費方`t2.micro`案EC2下使用 Amazon，前提是您的 AWS 帳戶年齡少於 12 個月，並且您可以選擇`t2.micro`執行個體類型（或`t3.micro`在無法使用的區域）。

6. 在 key pair (login) (金鑰對 (登入)) 中，針對 Key pair name (金鑰對名稱) 選擇您的金鑰對。
7. 在Network settings (網路設定) 中，執行下列操作：

   a. 對於網路和子網路，如果您尚未變更預設VPC或子網路，則可以保留預設設定。

      如果您已變更預設VPC或子網路，請檢查下列項目：

      i. 執行個體必須與資料庫位於VPC同一個資RDS料庫中，才能使用自動連線組態。默認情況下，您只有一個VPC。

      ii. 啟動執行VPC個體時，必須連接網際網路閘道，才能從網際網路存取 Web 伺服器。您的預設VPC值會透過網際網路閘道自動設定。

      iii. 若要確保您的執行個體接收公有 IP 地址，針對 Auto-assign public IP (自動指派公有 IP)，檢查是否已選取 Enable (啟用)。如果選取 Disable (停用)，則請選取 Edit (編輯) (位於 Network Settings (網路設定) 右側)，然後針對 Auto-assign public IP (自動指派公有 IP) 選取 Enable (啟用)。

b. 若要使用連線至執行個體SSH，您需要一個安全群組規則來授權 SSH (Linux) 或 RDP (Windows) 流量來自您電腦的公共IPv4位址。根據預設，當您啟動執行個體時，會建立新的安全群組，其中包含允許來自任何地方的輸入SSH流量的規則。

   若要確保只有您的 IP 位址可以連線到您的執行個體，請在 [防火牆 (安全群組)] 下，從 [允許SSH傳輸來源] 核取方塊旁的下拉式清單中選擇 [我的 IP]。

c. 若要允許流量從網際網路傳輸到您的執行個體，請選取下列核取方塊：

   - 允許來自HTTPs網際網路的流量
   - 允許來自HTTP網際網路的流量

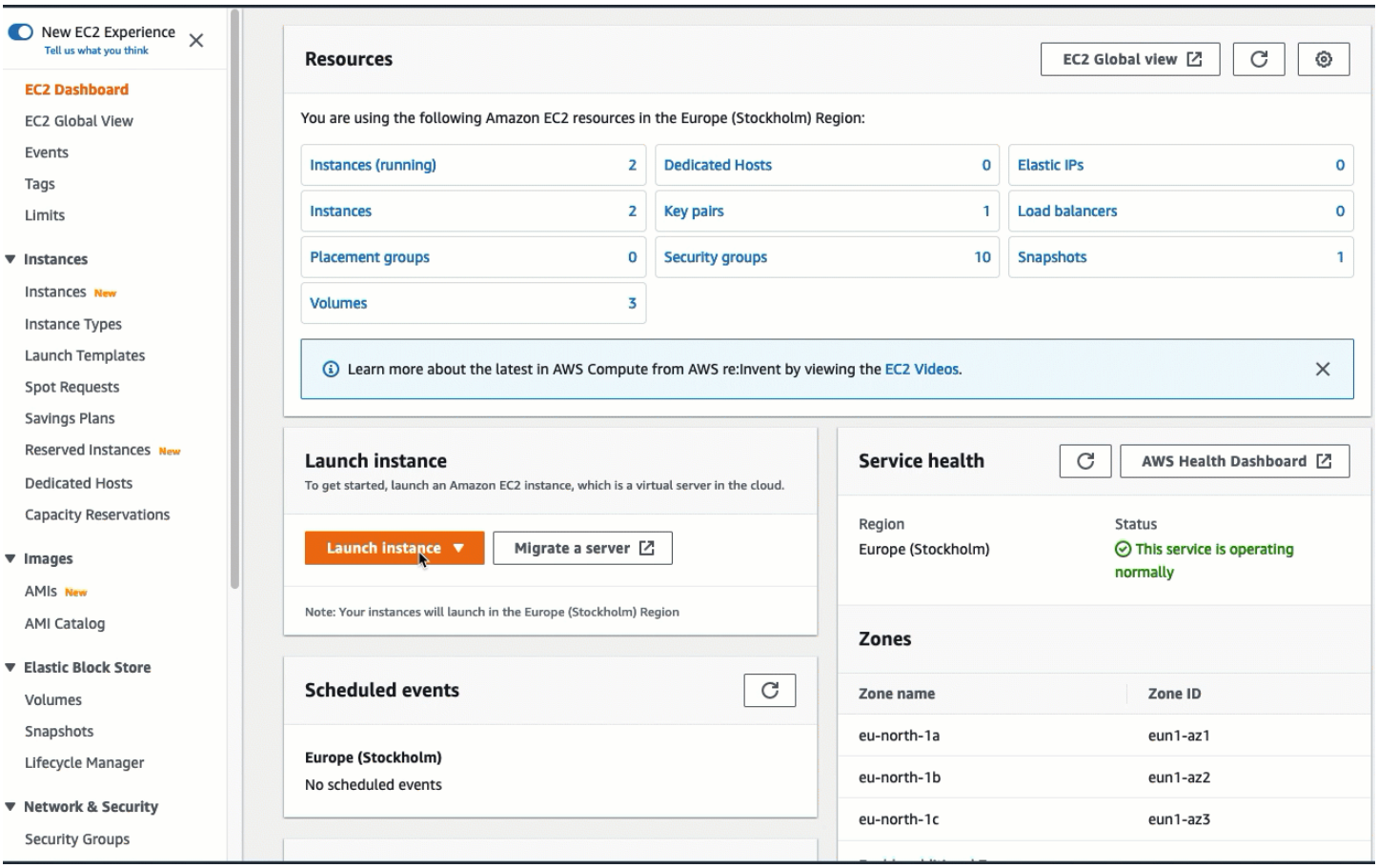
8. 在 Summary (摘要) 面板中，檢閱您的執行個體組態，然後選擇 Launch instance (啟動執行個體)。

9. 選擇 View all instances (檢視所有執行個體)，以關閉確認頁面並返回主控台。您的執行個體首先會處於 `pending` 狀態，然後進入 `running` 狀態。

   如果執行個體無法啟動或狀態立即進入 `terminated` 而非 `running`，請參閱解決 Amazon EC2 執行個體啟動問題。

如需有關啟動執行個體的詳細資訊，請參閱 使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體。

檢視動畫：啟動EC2實體



工作 2：建立RDS資料庫並自動將其連線至執行個EC2體

此任務的目的是創建一個RDS數據庫，並使用RDS控制台中的自動連接功能來自動配置EC2實例和RDS數據庫之間的連接。此任務中的步驟會設定資料庫執行個體，如下所示：

- 發動機類型：我的 SQL
- 範本：免費方案
- 資料庫執行個體識別符：**`tutorial-database`**
- 資料庫執行個體類別：`db.t3.micro`

> **Important**
>
> 在生產環境中，您應設定執行個體以滿足您的特定需求。

若要建立RDS資料庫並自動將其連線至EC2執行處理

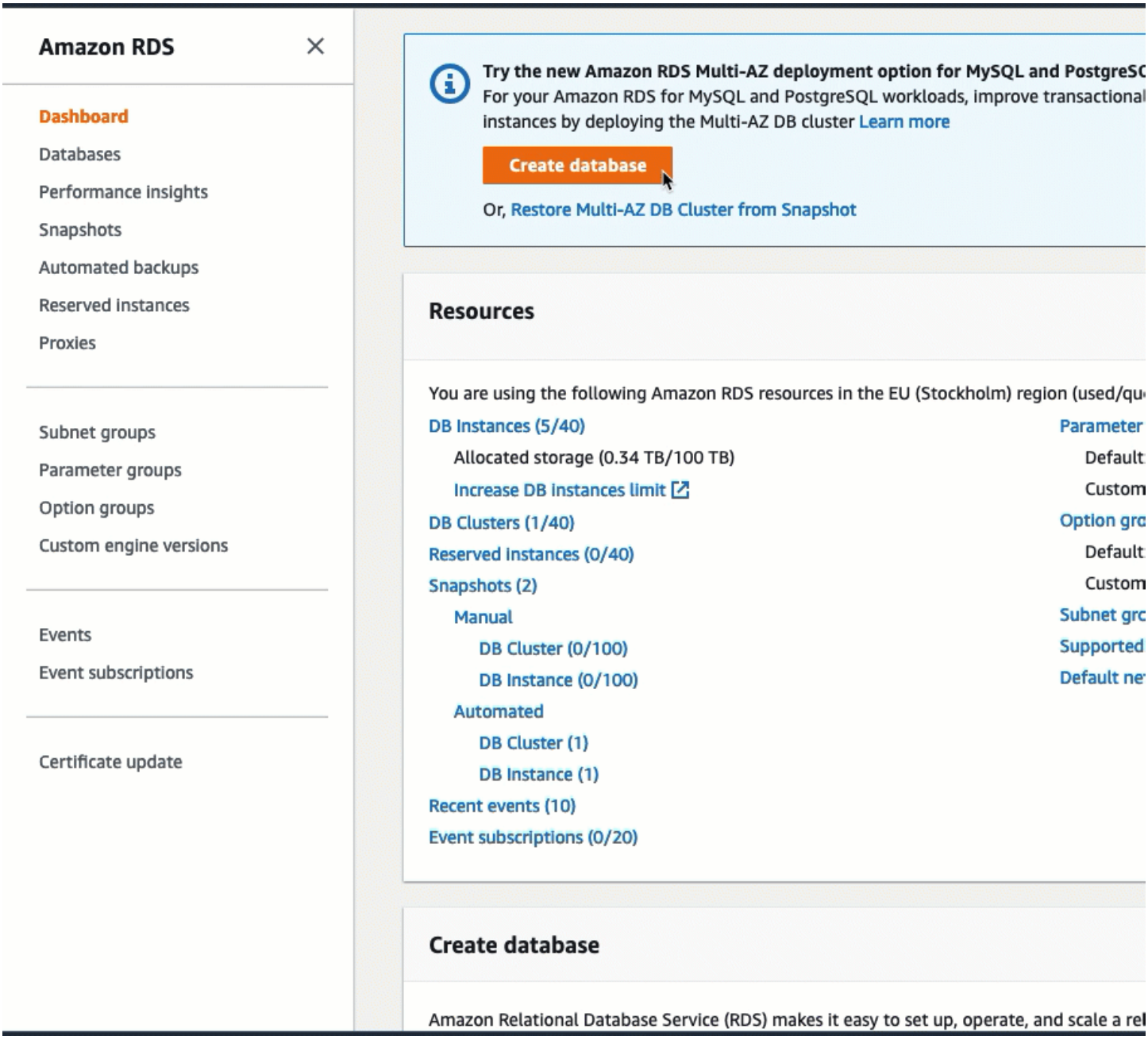
1. 在打開 Amazon RDS 控制台https://console.aws.amazon.com/rds/。
2. 從「區域」選取器 (位於右上角) 中，選擇您 AWS 區域 在其中建立EC2執行個體的項目。EC2執行個體和資RDS料庫必須位於相同的區域中。
3. 在儀表板中，選擇 Create database (建立資料庫)。
4. 在 Choose a database creation method (選取資料庫建立方法) 中，確認選取 Standard Create (標準建立)。如果您選擇 Easy create (輕鬆建立)，則無法使用自動連線功能。
5. 在 [引擎選項] 下，針對 [引擎類型] 選擇 [我] SQL。
6. 在 Templates (範本) 中，選擇符合您需求的範例範本。在本教學課程中，選擇免費層以免費建立RDS資料庫。但請注意，只有在您的帳戶使用未滿 12 個月時，才能使用免費方案。其他限制適用。在 Free tier (免費方案) 方塊中選擇 Info (資訊) 連結，可閱讀更多資訊。
7. 在 Settings (設定) 下，執行下列動作：
   a. 在 DB instance identifier (資料庫執行個體識別符) 中，輸入資料庫的名稱。針對本教學，輸入 **`tutorial-database`**。
   b. 對於 Master username (主要使用者名稱)，請保留預設名稱，即 **`admin`**。
   c. 對於 Master password (主要密碼)，請輸入您在本教學課程中可以記住的密碼，然後在 Confirm password (確認密碼) 中再次輸入密碼。
8. 在「執行個體組態」下，對於資料庫執行個體類別，保留預設值，即 db. t3.micro。 如果您的帳戶少於 12 個月，則可以免費使用此執行個體。其他限制適用。如需詳細資訊，請參閱AWS 免費方案。
9. 在 [連線] 下，對於 [計算資源]，選擇 [Connect 線到EC2計算資源]。這是RDS控制台中的自動連接功能。
10. EC2例如，選擇您要連線的EC2執行個體。在本教學課程中，您可以選擇在上一個任務中建立的執行個體 (您已命名為 **`tutorial-instance`**)，或選擇另一個現有執行個體。如果在清單中沒有看到您的執行個體，則請選擇 Connectivity (連線) 右側的重新整理圖示。

    當您使用自動連線功能時，會將安全性群組新增至此EC2執行個體，並將另一個安全群組新增至資RDS料庫。安全群組會自動設定為允許EC2執行個體和RDS資料庫之間的流量。在下一個工作中，您將確認安全群組是否已建立並指派給EC2執行個體和RDS資料庫。
11. 選擇建立資料庫。

在 Databases (資料庫) 畫面中，新資料庫的 Status (狀態) 為 Creating (正在建立)，直到資料庫準備就緒可供使用為止。狀態變更為 Available (可用) 時，您便可連線至資料庫。視資料庫類別和儲存體數量而定，可能需要最多 20 分鐘的時間，新資料庫才會可用。

若要進一步了解，請參閱 Amazon RDS 使用者指南中的設定與EC2執行個體的自動網路連線。

檢視動畫：建立RDS資料庫並自動將其連線至EC2執行個體



任務 3：驗證連線組態

此任務的目標是確認已建立兩個安全群組並指派給執行個體和資料庫。

當您使用主控台中的自動連線功能來設定連線時，會自動建立安全群組並將其指派給執行個體和資料庫，如下所示：

- 安全性群組 rds-ec2-*x*已建立並新增至資RDS料庫。它有一個引用 ec2-rds-的入站規則*x*安全群組做為其來源。這允許來自帶有 ec2-EC2 rds-執行個體的流量*x*安全組到達數RDS據庫。
- 安全性群組 ec2-RDS-*x*已建立並新增至EC2執行個體。它有一個引用 rds-ec2-的出站規則*x*安全群組做為其目的地。這允許來自EC2實例的流量使用 rds-ec2-到達RDS數據庫*x*安全性群組。

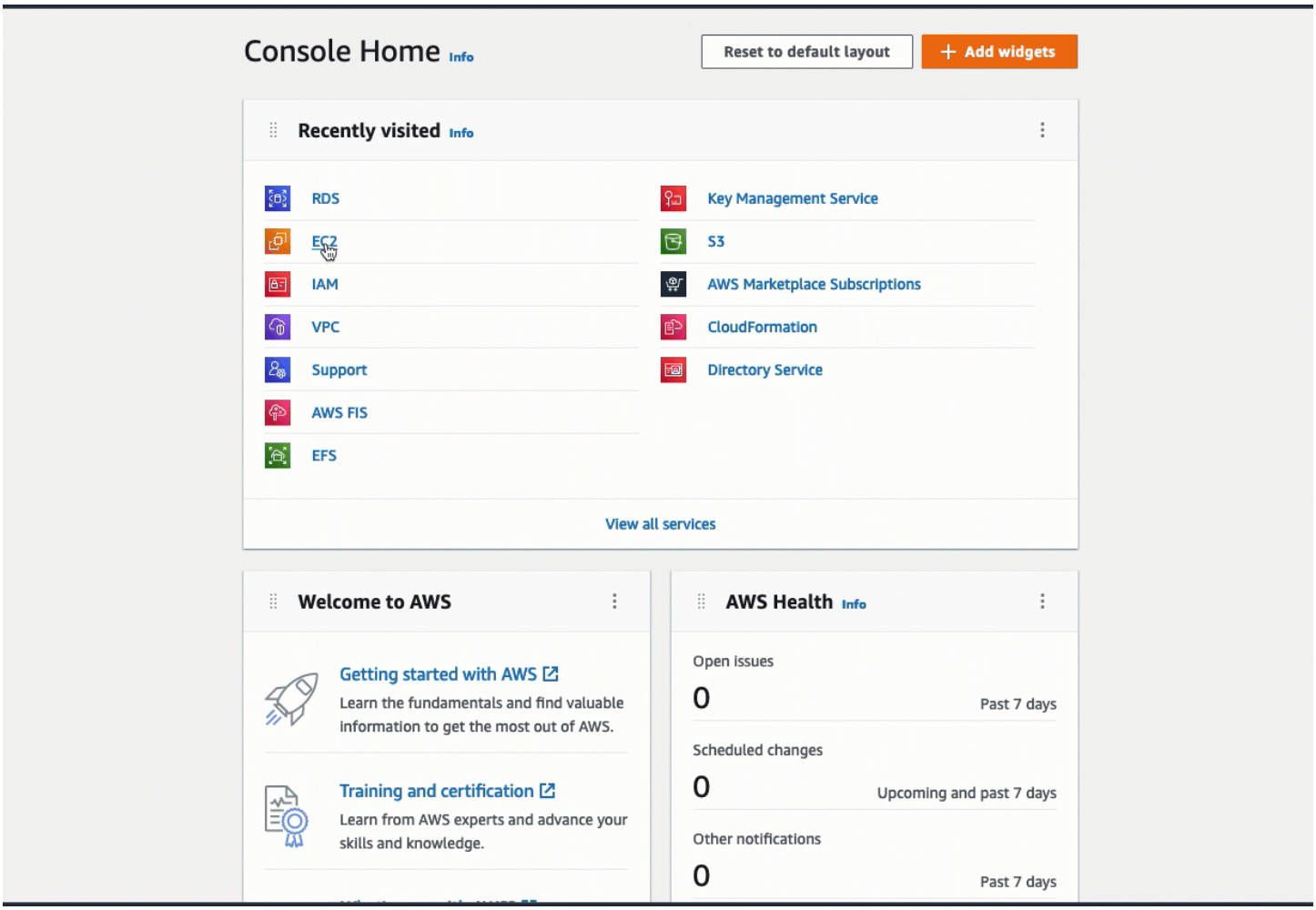
若要使用主控台驗證連線組態

1. 在打開 Amazon RDS 控制台https://console.aws.amazon.com/rds/。
2. 在導覽頁面中，選擇 Databases (資料庫)。
3. 選擇您為此教學課程建立的資RDS料庫。
4. 在 [連線與安全性] 索引標籤的 [安VPC全性] 下，確認名為 rds-ec2-的安全群組*x*顯示。
5. 選擇正確的方法*x*安全性群組。主控台中的「安全群組」畫EC2面隨即開啟。
6. 選擇正確的方法*x*安全組打開它。
7. 選擇 Inbound Rules (傳入規則) 索引標籤。
8. 確認下列安全群組規則存在，如下所示：
   - 類型：MYSQL/Aurora
   - 連接埠範圍：3306
   - 來源：***sg-0987654321example*** /EC2-*x*— 這是指派給您在上述步驟中驗證之EC2執行個體的安全性群組。
   - 說明：允許EC2執行個體連線的規則 ***sg-1234567890example*** 附
9. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
10. 在導覽窗格中，選擇 Instances (執行個體)。
11. 選擇您在上一EC2個作業中選取要連線至RDS資料庫的執行處理，然後選擇「安全」頁籤。
12. 在 [安全性詳細資料]、[安全性群組] 下，確認名為 ec2-rds-的安全性群組*x*在列表中。*x* 是一個數字。
13. 選擇 EC2-*x*安全組打開它。

14. 選擇 Outbound rules (傳出規則) 索引標籤。
15. 確認下列安全群組規則存在，如下所示：
    - 類型：MYSQL/Aurora
    - 連接埠範圍：3306
    - 目標：***sg-1234567890example*** /rds-***x***
    - 說明：允許從此安全群組連接的任何執行個體到 **`database-tutorial`** 之連線的規則

透過確認這些安全群組和安全群組規則是否存在，以及這些規則是否已按照本程序中所述將它們指派給RDS資料庫和EC2執行處理，您可以使用自動連線功能驗證連線是否已自動設定。

檢視動畫：驗證連線組態



您已完成本教學課程的選項 2。您現在可以完成選項 3，其會教導您如何手動設定在選項 2 中自動建立的安全群組。

工作 4 (選擇性)：清除

現在您已經完成教學課程，最好清理 (刪除) 您不再想要使用的任何資源。清理 AWS 資源可防止您的帳戶產生任何進一步的費用。

如果您特別針對本教學課程啟動EC2執行個體，您可以終止執行個體，以停止產生與其相關的任何費用。

欲使用主控台來終止執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取您為此教學課程建立的執行個體，然後選取 Instance state (執行個體狀態)、Terminate instance (終止執行個體)。
4. 出現確認提示時，請選擇 Terminate (終止)。

如果您專門為本教學課程建立了RDS資料庫，您可以刪除該資料庫，以停止產生與其相關的任何費用。

使用主控台刪除RDS資料庫

1. 在打開 Amazon RDS 控制台https://console.aws.amazon.com/rds/。
2. 在導覽窗格中，選擇 Databases (資料庫)。
3. 選取您為此教學課程建立的RDS資料庫，然後選擇動作，刪除。
4. 在方塊中輸入 **`delete me`**，然後選擇 Delete (刪除)。

選項 3：建立安全群組，以手動方式將執行處理連線至RDS資料庫

Option 3 的目標是學習如何透過手動複製自動連線功能的組態來手動設定EC2執行個體與RDS資料庫之間的連線。

任務

- 開始之前
- 工作 1 (選用)：啟動EC2執行個體
- 工作 2 (選擇性)：建立RDS資料庫
- 工作 3：建立安全群組並將其指派給EC2執行個體，以手動方式將執行個體連線至資RDS料庫

- 工作 4 (選擇性)：清除

開始之前

完成此教學課程需要以下項目：

- 與RDS資料庫位於相同的EC2執VPC行個體。您可以使用現有的EC2執行個體，也可以依照工作 1 中的步驟建立新的執行個體。
- 與執行個體位於相同VPC的資RDS料EC2庫。您可以使用現有的RDS資料庫，或遵循工作 2 中的步驟來建立新的資料庫。
- 呼叫以下操作的許可：
  - `ec2:AssociateRouteTable`
  - `ec2:AuthorizeSecurityGroupEgress`
  - `ec2:CreateRouteTable`
  - `ec2:CreateSecurityGroup`
  - `ec2:CreateSubnet`
  - `ec2:DescribeInstances`
  - `ec2:DescribeNetworkInterfaces`
  - `ec2:DescribeRouteTables`
  - `ec2:DescribeSecurityGroups`
  - `ec2:DescribeSubnets`
  - `ec2:ModifyNetworkInterfaceAttribute`
  - `ec2:RevokeSecurityGroupEgress`

工作 1 (選用)：啟動EC2執行個體

> **Note**
>
> 啟動執行個體並不是本教學課程的重點。如果您已經擁有 Amazon EC2 執行個體，並且想要在本教學中使用它，則可以略過此任務。

此任務的目標是啟動EC2執行個體，以便您可以完成任務 3，在其中設定EC2執行個體和 Amazon RDS 資料庫之間的連線。此工作中的步驟設定執EC2行個體，如下所示：

- 執行個體名稱：**`tutorial-instance`**
- AMI: Amazon 2
- 執行個體類型：`t2.micro`
- 自動指派公有 IP：已啟用
- 具有以下三個規則的安全群組：
  - SSH從您的 IP 位址允許
  - 允許任何地方的HTTPS流量
  - 允許任何地方的HTTP流量

> **Important**
>
> 在生產環境中，您應設定執行個體以滿足您的特定需求。

啟動 EC2 執行個體

1. 登錄 AWS Management Console 並在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在EC2儀表板上，選擇 [啟動執行個體]。
3. 在 Name and tags (名稱和標籤) 下，針對 Name (名稱)，輸入用於識別執行個體的名稱。針對本教學課程，將執行個體命名為 **`tutorial-instance-manual-1`**。雖然執行個體名稱不是強制性的，但名稱可協助您輕鬆識別其。
4. 在「應用程式和作業系統映像」下，選擇AMI符合您 Web 伺服器需求的影像。本教學課程使用 Amazon Linux。
5. 在 Instance type (執行個體類型) 中，針對 Instance type (執行個體類型)，選取符合您 Web 伺服器需求的執行個體類型。本教學課程使用的是 `t2.micro`。

   > **Note**
   >
   > 您可以在免費方`t2.micro`案EC2下使用 Amazon，前提是您的 AWS 帳戶年齡少於 12 個月，並且您可以選擇`t2.micro`執行個體類型（或`t3.micro`在無法使用的區域）。

6. 在 key pair (login) (金鑰對 (登入)) 中，針對 Key pair name (金鑰對名稱) 選擇您的金鑰對。
7. 在Network settings (網路設定) 中，執行下列操作：

a. 對於網路和子網路，如果您尚未變更預設VPC或子網路，則可以保留預設設定。

   如果您已變更預設VPC或子網路，請檢查下列項目：

   i. 執行個體必須與RDS資料庫位於VPC相同的位置。默認情況下，您只有一個VPC。

   ii. 啟動執行VPC個體時，必須連接網際網路閘道，才能從網際網路存取 Web 伺服器。您的預設VPC值會透過網際網路閘道自動設定。

   iii. 若要確保您的執行個體接收公有 IP 地址，針對 Auto-assign public IP (自動指派公有 IP)，檢查是否已選取 Enable (啟用)。如果選取 Disable (停用)，則請選取 Edit (編輯) (位於 Network Settings (網路設定) 右側)，然後針對 Auto-assign public IP (自動指派公有 IP) 選取 Enable (啟用)。

b. 若要使用連線至執行個體SSH，您需要一個安全群組規則來授權 SSH (Linux) 或 RDP (Windows) 流量來自您電腦的公共IPv4位址。根據預設，當您啟動執行個體時，會建立新的安全群組，其中包含允許來自任何地方的輸入SSH流量的規則。

   若要確保只有您的 IP 位址可以連線到您的執行個體，請在 [防火牆 (安全群組)] 下，從 [允許SSH傳輸來源] 核取方塊旁的下拉式清單中選擇 [我的 IP]。

c. 若要允許流量從網際網路傳輸到您的執行個體，請選取下列核取方塊：

   - 允許來自HTTPs網際網路的流量
   - 允許來自HTTP網際網路的流量

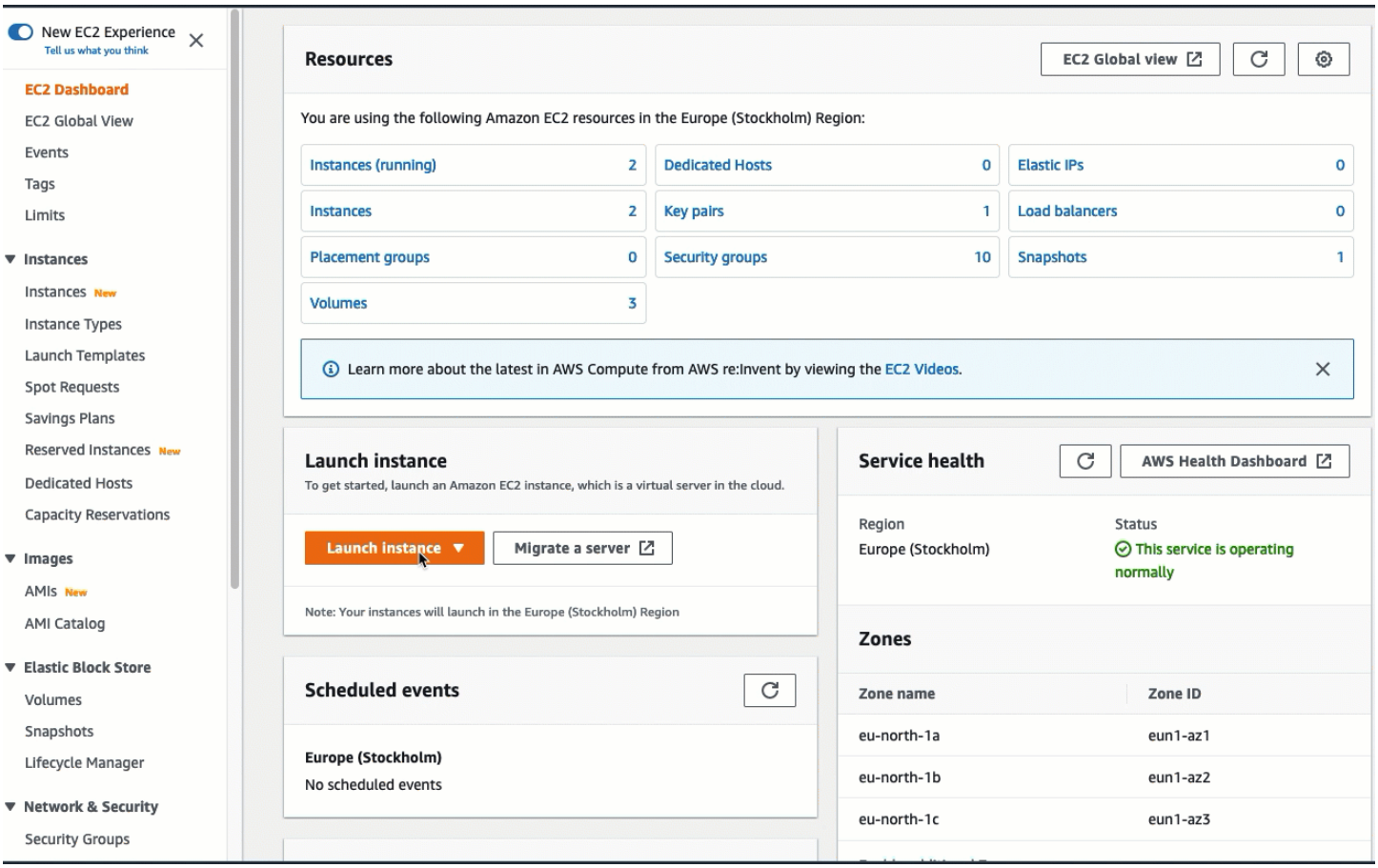
8. 在 Summary (摘要) 面板中，檢閱您的執行個體組態，然後選擇 Launch instance (啟動執行個體)。

9. 選擇 View all instances (檢視所有執行個體)，以關閉確認頁面並返回主控台。您的執行個體首先會處於 `pending` 狀態，然後進入 `running` 狀態。

   如果執行個體無法啟動或狀態立即進入 `terminated` 而非 `running`，請參閱解決 Amazon EC2 執行個體啟動問題。

如需有關啟動執行個體的詳細資訊，請參閱 使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體。

檢視動畫：啟動EC2實體



工作 2 (選擇性)：建立RDS資料庫

> **Note**
>
> 建立RDS資料庫並不是本教學課程本部分的重點。如果您已經有一個RDS數據庫，並希望在本教程中使用它，則可以跳過此任務。

此任務的目的是創建一個RDS數據庫。當您將這個執行個體連線至執行個體時，您會在工作 3 中使用此EC2執行個體。此工作中的步驟設定RDS資料庫，如下所示：

- 發動機類型：我的 SQL
- 範本：免費方案
- 資料庫執行個體識別符：**`tutorial-database-manual`**
- 資料庫執行個體類別：`db.t3.micro`

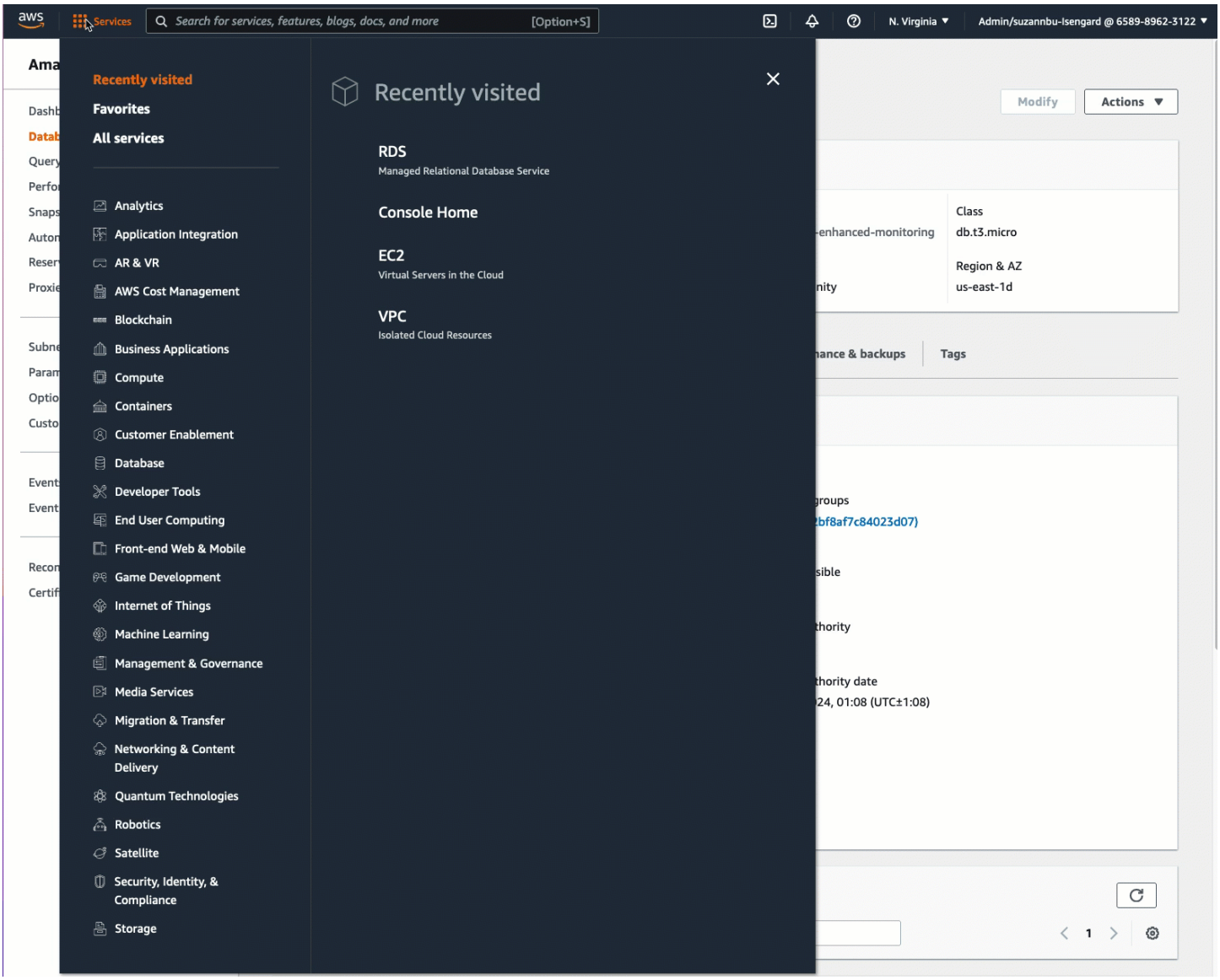
> **Important**
>
> 在生產環境中，您應設定執行個體以滿足您的特定需求。

若要建立我的SQL資料庫執行個體

1. 在打開 Amazon RDS 控制台https://console.aws.amazon.com/rds/。
2. 從「區域」選取器 (位於右上角) 中，選擇您 AWS 區域 在其中建立EC2執行個體的項目。EC2執行個體和資料庫執行個體必須位於相同的區域。
3. 在儀表板中，選擇 Create database (建立資料庫)。
4. 在 Choose a database creation method (選擇資料庫建立方法) 中，選擇 Easy create (輕鬆建立)。當您選擇此選項時，無法使用自動設定連線的自動連線功能。
5. 在 [引擎選項] 下，針對 [引擎類型] 選擇 [我] SQL。
6. 在 DB instance size (資料庫執行個體大小) 中，選擇 Free tier (免費方案)。
7. 對於資料庫執行個體識別碼，請輸入RDS資料庫的名稱。針對本教學，輸入 **`tutorial-database-manual`**。
8. 對於 Master username (主要使用者名稱)，請保留預設名稱，即 **`admin`**。
9. 對於 Master password (主要密碼)，請輸入您在本教學課程中可以記住的密碼，然後在 Confirm password (確認密碼) 中再次輸入密碼。
10. 選擇建立資料庫。

    在 Databases (資料庫) 畫面中，新資料庫執行個體的 Status (狀態) 為 Creating (正在建立)，直到資料庫執行個體準備就緒可供使用為止。狀態變更為 Available (可用) 時，您便能連線至資料庫執行個體。視資料庫執行個體類別和儲存體數量而定，可能需要最多 20 分鐘的時間，新執行個體才會可用。

檢視動畫：建立資料庫執行個體



工作 3：建立安全群組並將其指派給EC2執行個體，以手動方式將執行個體連線至資RDS料庫

此工作的目標是透過手動執行下列動作來重現自動連線功能的連線組態：您建立兩個新的安全性群組，然後將安全性群組分別新增至執行個EC2體和RDS資料庫。

建立兩個新的安全群組，並將一個群組分別指派給EC2執行處理和RDS資料庫

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 首先建立要新增至EC2執行個體的安全性群組，如下所示：

a. 在導覽窗格中，選擇 Security Groups (安全群組)。

b. 選擇 Create Security Group (建立安全群組)。

c. 對於 Security group name (安全群組名稱)，輸入安全群組的描述性名稱。針對本教學，輸入 **`ec2-rds-manual-configuration`**。

d. 對於 Description (描述)，輸入簡短描述。針對本教學，輸入 **`EC2 instance security group to allow EC2 instance to securely connect to RDS database`**。

e. 選擇建立安全群組。建立RDS資料庫安全性群組之後，您會回到這個安全性群組，以新增輸出規則。

3. 現在，建立要新增至RDS資料庫的安全性群組，如下所示：

a. 在導覽窗格中，選擇 Security Groups (安全群組)。

b. 選擇 Create Security Group (建立安全群組)。

c. 對於 Security group name (安全群組名稱)，輸入安全群組的描述性名稱。針對本教學，輸入 **`rds-ec2-manual-configuration`**。

d. 對於 Description (描述)，輸入簡短描述。針對本教學，輸入 **`RDS database security group to allow EC2 instance to securely connect to RDS database`**。

e. 在 Inbound rules (傳入規則) 中，選擇 Add rule (新增規則)，然後執行下列動作：

i. 選擇「MYSQL/Aurora」做為「類型」。

ii. 針對來源，選擇您在此程序的步驟 2 中rds-manual-configuration建立的EC2執行個體安全性群組 ec2-。

f. 選擇建立安全群組。

4. 編輯EC2執行個體安全性群組以新增輸出規則，如下所示：

a. 在導覽窗格中，選擇 Security Groups (安全群組)。

b. 選取EC2執行個體安全性群組 (您已命名為此群組**`ec2-rds-manual-configuration`**)，然後選擇 [輸出規則] 索引標籤。

c. 選擇 Edit outbound rules (編輯傳出規則)。

d. 選擇 Add rule (新增規則) 並執行下列動作：

i. 選擇「MYSQL/Aurora」做為「類型」。

ii. 針對來源，選擇您在此程序的步驟 3 中建立的RDS資料庫安全群組 rds-ec2-手動組態。

iii. 選擇儲存規則。

5. 將EC2執行個體安全性群組新增至EC2執行個體，如下所示：
   a. 在導覽窗格中，選擇 Instances (執行個體)。
   b. 選取EC2執行個體，然後選擇 [動作]、[安全性]、[變更安全群組
   c. 在 [關聯的安全群組] 下，選擇 [選取安全群組] 欄位，選擇您先前建立的 ec2-rds-manual-configuration，然後選擇 [新增安全群組]。
   d. 選擇 Save (儲存)。
6. 將RDS資料庫安全性群組新增至資RDS料庫，如下所示：
   a. 在打開 Amazon RDS 控制台https://console.aws.amazon.com/rds/。
   b. 在導覽窗格中，選取 Databases (資料庫)，然後選取您的資料庫。
   c. 選擇 Modify (修改)。
   d. 在 Connectivity (連線) 中，針對 Security group (安全群組) 選擇您先前建立的 rds-ec2-manual-configuration，然後選擇 Continue (繼續)。
   e. 在 Scheduling of modifications (修改排程) 中，選擇 Apply immediately (立即套用)。
   f. 選擇 Modify DB instance (修改資料庫執行個體)。

   您現在已完成手動步驟，這些步驟會模擬當使用自動連線功能時所發生的自動步驟。

您已完成本教學課程的選項 3。如果您已完成選項 1、2 和 3，且不再需要本教學課程中建立的資源，您應該將其刪除，以避免產生不必要的費用。如需詳細資訊，請參閱工作 4 (選擇性)：清除。

工作 4 (選擇性)：清除

現在您已經完成教學課程，最好清理 (刪除) 您不再想要使用的任何資源。清理 AWS 資源可防止您的帳戶產生任何進一步的費用。

如果您特別針對本教學課程啟動EC2執行個體，您可以終止執行個體，以停止產生與其相關的任何費用。

欲使用主控台來終止執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取您為此教學課程建立的執行個體，然後選取 Instance state (執行個體狀態)、Terminate instance (終止執行個體)。

4. 出現確認提示時，請選擇 Terminate (終止)。

如果您專門為本教學課程建立了RDS資料庫，您可以刪除該資料庫，以停止產生與其相關的任何費用。

使用主控台刪除RDS資料庫

1. 在打開 Amazon RDS 控制台https://console.aws.amazon.com/rds/。
2. 在導覽窗格中，選擇 Databases (資料庫)。
3. 選取您為此教學課程建立的RDS資料庫，然後選擇動作，刪除。
4. 在方塊中輸入 **`delete me`**，然後選擇 Delete (刪除)。

EC2艦隊和現貨艦隊

EC2叢集和 Spot Fleet 的設計目的是在單一作業中啟動數十、數百或數千個 Amazon EC2 執行個體的有用方式。叢集中的每個執行個體均由啟動範本設定，或是您在啟動時手動設定的一組啟動參數來設定。

主題

- 功能和優勢
- 哪種是最好的車隊方法？
- 您的EC2車隊或 Spot 叢集的組態選項
- 與EC2艦隊合作
- 使用現貨艦隊
- 監控您的EC2車隊或 Spot 車隊
- EC2艦隊教程
- EC2叢集的範例CLI組態
- 範例CLI組態競價型叢集
- EC2車隊和競價型艦隊的配額

功能和優勢

叢集提供下列功能和優點，可讓您在多個執行個體上EC2執行應用程式時，最大限度地節省成本並最佳化可用性和效能。

多種實例類型

叢集可以啟動多個執行個體類型，確保它不依賴於任何單一執行個體類型的可用性。這會增加叢集中執行個體的整體可用性。

跨可用區域分佈執行個體

機群會自動嘗試在多個可用區域內平均分佈執行個體以利達成高可用性。這會在可用區域變得無法使用時提供復原能力。

多種購買選項

一個叢集可以啟動多個購買選項 (Spot 和隨需執行個體)，讓您透過 Spot 執行個體使用情況將成本最佳化。您也可以將預留執行個體和 Savings Plan 折扣與機群中的隨需執行個體搭配使用，善加利用這些折扣。

自動替換 Spot 執行個體

如果您的叢集包含 Spot 執行個體，當 Spot 執行個體中斷時，它可以自動要求替換 Spot 容量。透過容量重新平衡，叢集還可以監控並主動更換面臨中斷風險較高的 Spot 執行個體。

保留隨需容量

叢集可以使用隨需容量保留來保留隨需容量。叢集還可以包含 ML 的容量區塊，讓您可以在 future 的日期預留GPU執行個體，以支援您的短期機器學習 (ML) 工作負載。

哪種是最好的車隊方法？

一般而言，我們建議您使用 Amazon EC2 Auto Scaling 啟動 Spot 和隨需執行個體叢集，因為它提供了可用來管理叢集的其他功能。其他功能清單包括 Spot 和隨需執行個體的自動運作狀態檢查替換、以應用程式為基礎的運作狀態檢查，以及與 Elastic Load Balancing 的整合，可確保應用程式流量均勻分佈到運作良好的執行個體。您也可以在使用 Amazon ECS、Amazon EKS (自我管理節點群組) 和 Amazon VPC 萊迪斯等 AWS 服務時使用自動擴展群組。如需詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南。

如果您無法使用 Amazon EC2 Auto Scaling，則可以考慮使用EC2叢集或 Spot 叢集。EC2車隊和競價型艦隊提供相同的核心功能。不過，EC2叢集只能透過命令列使用，不提供主控台支援。Spot Fleet 提供主控台支援，但是以舊版API為基礎，無需計劃投資。

請使用下表來決定要使用的叢集方法。

艦隊方法	何時使用？	使用案例
Amazon EC2 Auto Scaling	• 您需要具有單一組態或混合組態的多個執行個體。 • 您想要自動化執行個體的生命週期管理。	建立 Auto Scaling 群組，管理執行個體的生命週期，同時維護所需的執行個體數量。支援在指定的最小和最大限制之間水平擴展 (新增更多執行個體)。

艦隊方法	何時使用？	使用案例
EC2艦隊	• 您需要具有單一組態或混合組態的多個執行個體。 • 您希望自行管理您的執行個體生命週期。 • 如果您不需要 auto 擴展，我們建議您使用 EC2 Fleet `instant` 類型。	在單一作業中建立隨需執行個體和 Spot 執行個體的`instant`叢集，並具有多個啟動規格，這些規格會因執行個體類型AMI、可用區域或子網路而異。競價型執行個體配置策略預設為`lowest-price` 每個單位，但我們建議將其變更為`price-capacity-optimized` 。
Spot 機群	• 我們強烈不建議使用 Spot Fleet，因為它是基於沒API有計劃投資的遺產。 • 如果您想要管理執行個體生命週期，請使用EC2叢集。 • 如果您不想管理執行個體生命週期，請使用 Auto Scaling 群組。	只有在您需要主控台支援使用案例的時機使用叢集時，才使用 Spot EC2 叢集。

您的EC2車隊或 Spot 叢集的組態選項

規劃EC2叢集或 Spot 叢集時，建議您在決定如何設定叢集時考慮下列選項。

組態選項	問題	文件
機隊請求類型	您想要針對所需目標容量提交一次性要求的叢集，還是要隨時間維持目標容量的叢集？	EC2叢集和 Spot 叢集請求類型
Spot 執行個體	您是否打算在叢集中包含 Spot 執行個體？ 檢閱 Spot 最佳做法，並在規劃叢集時使用它們，以便以最低的價格佈建執行個體。	Amazon EC2 現貨的最佳實踐

組態選項	問題	文件
您車隊的消費上限	您想限制每小時為您的車隊支付多少費用嗎？	為您的EC2車隊或 Spot 艦隊設定消費上限
例證類型和以屬性為基礎的例證類型選擇	您要在叢集中指定執行個體類型，還是讓 Amazon EC2 選取符合您應用程式需求的執行個體類型？	指定EC2叢集或 Spot 叢集的執行個體類型選取屬性
執行個體權重	您是否要為每個執行個體類型指派權重來表示其運算容量和效能，以便 Amazon EC2 可以選擇任何可用執行個體類型的組合來實現所需的目標容量？	使用執行個體加權來管理EC2叢集或 Spot 叢集的成本和效能
分配策略	您是否要決定是否針對叢集中的 Spot 執行個體和隨需執行個體使用的可用容量、價格或執行個體類型進行最佳化？	使用配置策略來確定EC2叢集或 Spot 叢集如何滿足 Spot 和隨需容量
容量重新平衡	您是否希望叢集自動取代有風險的 Spot 執行個體？	使用叢集和 S EC2 pot 叢集中的容量重新平衡來取代有風險的 Spot 執行個體
隨需容量保留	您是否要為叢集中的隨需執行個體保留容量？	使用容量保留在EC2叢集中保留隨需容量

EC2叢集和 Spot 叢集請求類型

EC2叢集或 Spot Fleet 的請求類型會決定請求是同步還是非同步，以及是對所需目標容量的一次性請求，還是持續維護容量的持續努力。設定叢集時，您必須指定要求類型。

EC2叢集和 Spot 叢集都提供兩種請求類型：`request`和`maintain`。此外，EC2艦隊還提供了第三種請求類型稱為`instant`.

叢集請求類型

`instant`（僅限EC2車隊）

如果您將要求類型設定為`instant`，EC2叢集會針對您想要的容量提出同步一次性要求。在回API應中，它會傳回啟動的執行個體，並為無法啟動的執行個體提供錯誤。如需詳細資訊，請參閱設定類型的EC2叢集 instant。

`request`

如果您將要求類型設定為`request`，叢集會針對您想要的容量提出非同步一次性要求。如果容量因 Spot 中斷而減少，叢集不會嘗試補充 Spot 執行個體，如果容量不可用，也不會在替代 Spot 容量集區中提交請求。`request`使用主控台建立 Spot 叢集類型時，請清除 [維護目標容量] 核取方塊。

`maintain` (default)

如果將請求類型設定為`maintain`，叢集會針對您想要的容量發出非同步請求，並透過自動補充任何中斷的 Spot 執行個體來維護它。`maintain`使用主控台建立 Spot 叢集類型時，請選取維護目標容量核取方塊

設定類型的EC2叢集 instant

類型即時的EC2叢集是同步的一次性要求，只會嘗試一次啟動您想要的容量。API回應會列出已啟動的執行處理，以及無法啟動之執行處理的錯誤。使用類型瞬間EC2艦隊有幾個好處，這在本文中描述。本文末尾處會提供範例組態。

對於需要僅啟動才能啟動EC2執行個體API的工作負載，您可以使用. RunInstances API 但是，使用時 RunInstances，您只能啟動隨需執行個體或 Spot 執行個體，但不能在同一個請求中啟動兩者。此外，當您使用 RunInstances 啟動 Spot 執行個體時，您的競價型執行個體請求僅限於一種執行個體類型和一個可用區域。此操作針對單個 Spot 容量集區 (一組具有相同執行個體類型和可用區域的未使用執行個體)。如果 Spot 容量集區沒有足夠的 Spot 執行個體容量供您的請求使用，則 RunInstances 呼叫會失敗。

建議您不 RunInstances 要使用啟動 Spot 執行個體，而是將`type`參數設定`instant`為使用，以獲得下列優點： CreateFleet API

- 在一個請求中啟動隨需執行個體和 Spot 執行個體。EC2叢集可以啟動隨需執行個體、Spot 執行個體或兩者。如果有可用容量且您的請求中每個小時的價格上限超過 Spot 價格，則會滿足 Spot 執行個體的請求。
- 增加 Spot 執行個體的可用性。透過使用類型的EC2叢集`instant`，您可以按照 Spot 最佳做法啟動 Spot 執行個體，並獲得所帶來的好處：

- Spot 最佳實務：對執行個體類型和可用區域具有彈性。

  好處：透過指定數個執行個體類型和可用區域，您可以增加 Spot 容量集區的數目。這可讓 Spot 服務更有機會尋找並配置您所需的 Spot 運算容量。一個好的經驗法則是針對每個工作負載至少 10 個執行個體類型靈活彈性，並確保所有可用區域都設定為在VPC.

- 發現最佳做法：使用price-capacity-optimized配置策略。

  優點：`price-capacity-optimized`配置策略可從最可用的 Spot 容量集區識別執行個體，然後從這些集區中最低價格自動佈建執行個體。由於您的競價型執行個體容量來自具有最佳容量的集區，因此可降低當 Amazon EC2 需要恢復容量時，Spot 執行個體中斷的可能性。

- 取得更廣泛的功能組合。對於需要僅啟動的工作負載API，以及您希望管理執行個體生命週期而不是讓EC2叢集為您管理它的工作負載，請使用EC2叢集類型`instant`而非. RunInstancesAPI EC2艦隊提供了更廣泛的功能比 RunInstances，如下面的例子所示。對於所有其他工作負載，您應該使用 Amazon EC2 Auto Scaling，因為它為各種工作負載提供了更全面的功能集，例如ELB支援的應用程式、容器化工作負載和佇列處理任務。

您可以使用即時類型的EC2叢集將執行個體啟動到容量區塊中。如需詳細資訊，請參閱教學課程：設定EC2叢集以將執行個體啟動至容量區塊。

AWS Amazon EC2 自 Auto Scaling 和 Amazon 等服務EMR使用類型即時的EC2叢集來啟動EC2執行個體。

類型即時EC2艦隊的先決條件

如需建立EC2叢集的先決條件，請參閱EC2機隊先決條。

即時EC2艦隊如何運作

使用類型的EC2艦隊時`instant`，事件的順序如下：

1. 將CreateFleet要求類型設定為`instant`。如需詳細資訊，請參閱建立EC2艦隊。請注意，API撥打電話後，就無法修改。
2. 當您撥API打電話時，EC2叢集會針對您想要的容量提出同步一次性要求。
3. API回應會列出已啟動的執行處理，以及無法啟動之執行處理的錯誤。
4. 您可以描述您的EC2叢集、列出與您的EC2叢集相關聯的執行個體，以及檢視EC2叢集的歷史記錄。
5. 執行個體啟動後，您可以刪除叢集請求。刪除機群請求時，您也可以選擇終止相關聯的執行個體，或讓它們繼續執行。

6. 您隨時可終止執行個體。

範例

下列範例顯示如何`instant`針對不同的使用案例使用類型的EC2叢集。如需有關使用EC2 CreateFleet API參數的詳細資訊，請參閱 Amazon 參EC2API考CreateFleet中的。

範例

- 範例 1：使用容量最佳化分配策略啟動 Spot 執行個體
- 範例 2：使用容量最佳化分配策略啟動單一 Spot 執行個體
- 範例 3：使用執行個體權重啟動 Spot Fleet
- 範例 4：在單一可用區域內啟動 Spot 執行個體
- 範例 5：在單一可用區域內啟動單一執行個體類型的 Spot 執行個體
- 範例 6：僅在可啟動最小目標容量時才啟動 Spot 執行個體
- 範例 7：僅當可在單個可用區域中啟動相同執行個體類型的最低目標容量時，才能啟動 Spot 執行個體
- 範例 8：使用多個啟動範本來啟動執行個體
- 範例 9：啟動基於隨需執行個體的 Spot 執行個體
- 範例 10：基於使用容量預留和優先順序分配策略的隨需執行個體，使用容量最佳化分配策略啟動 Spot 執行個體
- 範例 11：使用 capacity-optimized-prioritized 配置策略啟動競價型執行個體
- 範例 12：指定 Systems Manager 參數而非 AMI ID

範例 1：使用容量最佳化分配策略啟動 Spot 執行個體

下列範例會指定「EC2叢集」類型中所需的參數`instant`：啟動樣版、目標產能、預設採購選項，以及啟動樣版覆寫。

- 啟動範本由其啟動範本名稱和版本編號識別。
- 12 個啟動範本覆寫會指定 4 個不同執行個體類型和 3 個不同的子網，每個都位於不同的可用區域。每個執行個體類型和子網組合都定義了一個 Spot 容量集區，從而產生 12 個 Spot 容量集區。
- 機群的目標容量為 20 個執行個體。
- 預設購買選項為 `spot`，這會導致機群嘗試將 20 個 Spot 執行個體啟動到 Spot 容量集區中，且擁有啟動中執行個體數目之最佳容量。

```
{
    "SpotOptions": {
        "AllocationStrategy": "capacity-optimized"
    },
    "LaunchTemplateConfigs": [
      {
         "LaunchTemplateSpecification":{
            "LaunchTemplateName":"ec2-fleet-lt1",
            "Version":"$Latest"
         },
         "Overrides":[
            {
               "InstanceType":"c5.large",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"c5.large",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"c5.large",
               "SubnetId":"subnet-49e41922"
            },
            {
               "InstanceType":"c5d.large",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"c5d.large",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"c5d.large",
               "SubnetId":"subnet-49e41922"
            },
            {
               "InstanceType":"m5.large",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"m5.large",
               "SubnetId":"subnet-e7188bab"
            },
```

```
            {
               "InstanceType":"m5.large",
               "SubnetId":"subnet-49e41922"
            },
            {
               "InstanceType":"m5d.large",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"m5d.large",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"m5d.large",
               "SubnetId":"subnet-49e41922"
            }
         ]
      }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 20,
        "DefaultTargetCapacityType": "spot"
    },
    "Type": "instant"
}
```

範例 2：使用容量最佳化分配策略啟動單一 Spot 執行個體

您可以 TotalTargetCapacity 將設定為 1，以最佳方式一次啟動一個 Spot 執行個體`instant`，方法是進行多個EC2叢集API呼叫類型。

下列範例會指定「即時EC2叢集」類型中所需的參數：啟動樣板、目標產能、預設採購選項，以及啟動範本覆寫。啟動範本由其啟動範本名稱和版本編號識別。12 個啟動範本覆寫會擁有 4 個不同執行個體類型和 3 個不同的子網，每個都位於不同的可用區域。機群的目標容量為 1 個執行個體，並且預設購買選項為 Spot，這會導致機群嘗試根據容量最佳化的分配策略從 12 個 Spot 容量集區之一啟動 Spot 執行個體，以從最可用的容量集區啟動 Spot 執行個體。

```
{
    "SpotOptions": {
        "AllocationStrategy": "capacity-optimized"
    },
    "LaunchTemplateConfigs": [
      {
```

```
        "LaunchTemplateSpecification":{
           "LaunchTemplateName":"ec2-fleet-lt1",
           "Version":"$Latest"
        },
        "Overrides":[
           {
              "InstanceType":"c5.large",
              "SubnetId":"subnet-fae8c380"
           },
           {
              "InstanceType":"c5.large",
              "SubnetId":"subnet-e7188bab"
           },
           {
              "InstanceType":"c5.large",
              "SubnetId":"subnet-49e41922"
           },
           {
              "InstanceType":"c5d.large",
              "SubnetId":"subnet-fae8c380"
           },
           {
              "InstanceType":"c5d.large",
              "SubnetId":"subnet-e7188bab"
           },
           {
              "InstanceType":"c5d.large",
              "SubnetId":"subnet-49e41922"
           },
           {
              "InstanceType":"m5.large",
              "SubnetId":"subnet-fae8c380"
           },
           {
              "InstanceType":"m5.large",
              "SubnetId":"subnet-e7188bab"
           },
           {
              "InstanceType":"m5.large",
              "SubnetId":"subnet-49e41922"
           },
           {
              "InstanceType":"m5d.large",
              "SubnetId":"subnet-fae8c380"
```

```
            },
            {
               "InstanceType":"m5d.large",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"m5d.large",
               "SubnetId":"subnet-49e41922"
            }
         ]
      }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 1,
        "DefaultTargetCapacityType": "spot"
    },
    "Type": "instant"
}
```

範例 3：使用執行個體權重啟動 Spot Fleet

下列範例使用執行個體權重，這表示價格為每單位小時而非每執行個體小時的價格。假設工作負載的單位需要 15 GB 記憶體和 4，每個啟動組態都會列出不同的執行個體類型和不同的權重，根據執行個體可在執行個體上執行的工作負載單位數量而定vCPUs。例如，一個 m5.xlarge（4 vCPUs 和 16 GB 的內存）可以運行一個單元，並且加權為 1，m5.2xlarge（8 vCPUs 和 32 GB 的內存）可以運行 2 個單位，並加權為 2，依此類推。目標總容量設定為 40 個單位。預設購買選項為 Spot，且分配策略容量最佳化，這會產生 40 個 m5.xlarge (40 除以 1)、20 個 m5.2xlarge (40 除以 2)、10 個 m5.4xlarge (40 除以 4)、5 個 m5.8xlarge (40 除以 8)，或基於容量最佳化分配策略，權重相加達到所需容量的執行個體類型的混合。

如需詳細資訊，請參閱 使用執行個體加權來管理EC2叢集或 Spot 叢集的成本和效能。

```
{
   "SpotOptions":{
      "AllocationStrategy":"capacity-optimized"
   },
   "LaunchTemplateConfigs":[
      {
         "LaunchTemplateSpecification":{
            "LaunchTemplateName":"ec2-fleet-lt1",
            "Version":"$Latest"
         },
```

```
        "Overrides":[
           {
              "InstanceType":"m5.xlarge",
              "SubnetId":"subnet-fae8c380",
              "WeightedCapacity":1
           },
           {
              "InstanceType":"m5.xlarge",
              "SubnetId":"subnet-e7188bab",
              "WeightedCapacity":1
           },
           {
              "InstanceType":"m5.xlarge",
              "SubnetId":"subnet-49e41922",
              "WeightedCapacity":1
           },
           {
              "InstanceType":"m5.2xlarge",
              "SubnetId":"subnet-fae8c380",
              "WeightedCapacity":2
           },
           {
              "InstanceType":"m5.2xlarge",
              "SubnetId":"subnet-e7188bab",
              "WeightedCapacity":2
           },
           {
              "InstanceType":"m5.2xlarge",
              "SubnetId":"subnet-49e41922",
              "WeightedCapacity":2
           },
           {
              "InstanceType":"m5.4xlarge",
              "SubnetId":"subnet-fae8c380",
              "WeightedCapacity":4
           },
           {
              "InstanceType":"m5.4xlarge",
              "SubnetId":"subnet-e7188bab",
              "WeightedCapacity":4
           },
           {
              "InstanceType":"m5.4xlarge",
              "SubnetId":"subnet-49e41922",
```

```
                "WeightedCapacity":4
            },
            {
                "InstanceType":"m5.8xlarge",
                "SubnetId":"subnet-fae8c380",
                "WeightedCapacity":8
            },
            {
                "InstanceType":"m5.8xlarge",
                "SubnetId":"subnet-e7188bab",
                "WeightedCapacity":8
            },
            {
                "InstanceType":"m5.8xlarge",
                "SubnetId":"subnet-49e41922",
                "WeightedCapacity":8
            }
        ]
    }
  ],
  "TargetCapacitySpecification":{
    "TotalTargetCapacity":40,
    "DefaultTargetCapacityType":"spot"
  },
  "Type":"instant"
}
```

範例 4：在單一可用區域內啟動 Spot 執行個體

您可以將 Spot 選項 SingleAvailabilityZone 設定為 true，將叢集設定為在單一可用區域中啟動所有執行個體。

12 個啟動範本覆寫會擁有不同執行個體類型和子網 (每個都位於不同的可用區域)，但權重容量相同。總目標容量為 20 個執行個體，預設購買選項為 Spot，Spot 分配策略為容量最佳化。EC2叢集會使用啟動規格，從具有最佳容量的 Spot 容量集區啟動 20 個 Spot 執行個體，全部在單一 AZ 中啟動。

```
{
    "SpotOptions": {
        "AllocationStrategy": "capacity-optimized",
        "SingleAvailabilityZone": true
    },
    "LaunchTemplateConfigs": [
      {
```

```
        "LaunchTemplateSpecification":{
           "LaunchTemplateName":"ec2-fleet-lt1",
           "Version":"$Latest"
        },
        "Overrides":[
           {
              "InstanceType":"c5.4xlarge",
              "SubnetId":"subnet-fae8c380"
           },
           {
              "InstanceType":"c5.4xlarge",
              "SubnetId":"subnet-e7188bab"
           },
           {
              "InstanceType":"c5.4xlarge",
              "SubnetId":"subnet-49e41922"
           },
           {
              "InstanceType":"c5d.4xlarge",
              "SubnetId":"subnet-fae8c380"
           },
           {
              "InstanceType":"c5d.4xlarge",
              "SubnetId":"subnet-e7188bab"
           },
           {
              "InstanceType":"c5d.4xlarge",
              "SubnetId":"subnet-49e41922"
           },
           {
              "InstanceType":"m5.4xlarge",
              "SubnetId":"subnet-fae8c380"
           },
           {
              "InstanceType":"m5.4xlarge",
              "SubnetId":"subnet-e7188bab"
           },
           {
              "InstanceType":"m5.4xlarge",
              "SubnetId":"subnet-49e41922"
           },
           {
              "InstanceType":"m5d.4xlarge",
              "SubnetId":"subnet-fae8c380"
```

```
            },
            {
               "InstanceType":"m5d.4xlarge",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"m5d.4xlarge",
               "SubnetId":"subnet-49e41922"
            }
         ]
      }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 20,
        "DefaultTargetCapacityType": "spot"
    },
    "Type": "instant"
}
```

範例 5：在單一可用區域內啟動單一執行個體類型的 Spot 執行個體

您可以將叢集設定為 true 和 true，以啟動相同執行個體類型的所有執行個體，並 SingleAvailabilityZone 在單一可用區域中啟動。 SpotOptions SingleInstanceType

12 個啟動範本覆寫會擁有不同執行個體類型和子網 (每個都位於不同的可用區域)，但權重容量相同。總目標容量為 20 個執行個體，預設購買選項為 Spot，Spot 分配策略為容量最佳化。EC2叢集會從 Spot 執行個體集區的單一 AZ 中啟動 20 個相同執行個體類型的 Spot 執行個體，並使用啟動規格提供最佳容量。

```
{
    "SpotOptions": {
        "AllocationStrategy": "capacity-optimized",
        "SingleInstanceType": true,
        "SingleAvailabilityZone": true
    },
    "LaunchTemplateConfigs": [
      {
         "LaunchTemplateSpecification":{
            "LaunchTemplateName":"ec2-fleet-lt1",
            "Version":"$Latest"
         },
         "Overrides":[
            {
```

```
                "InstanceType":"c5.4xlarge",
                "SubnetId":"subnet-fae8c380"
            },
            {
                "InstanceType":"c5.4xlarge",
                "SubnetId":"subnet-e7188bab"
            },
            {
                "InstanceType":"c5.4xlarge",
                "SubnetId":"subnet-49e41922"
            },
            {
                "InstanceType":"c5d.4xlarge",
                "SubnetId":"subnet-fae8c380"
            },
            {
                "InstanceType":"c5d.4xlarge",
                "SubnetId":"subnet-e7188bab"
            },
            {
                "InstanceType":"c5d.4xlarge",
                "SubnetId":"subnet-49e41922"
            },
            {
                "InstanceType":"m5.4xlarge",
                "SubnetId":"subnet-fae8c380"
            },
            {
                "InstanceType":"m5.4xlarge",
                "SubnetId":"subnet-e7188bab"
            },
            {
                "InstanceType":"m5.4xlarge",
                "SubnetId":"subnet-49e41922"
            },
            {
                "InstanceType":"m5d.4xlarge",
                "SubnetId":"subnet-fae8c380"
            },
            {
                "InstanceType":"m5d.4xlarge",
                "SubnetId":"subnet-e7188bab"
            },
            {
```

```
                "InstanceType":"m5d.4xlarge",
                "SubnetId":"subnet-49e41922"
            }
        ]
      }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 20,
        "DefaultTargetCapacityType": "spot"
    },
    "Type": "instant"
}
```

範例 6：僅在可啟動最小目標容量時才啟動 Spot 執行個體

您可以透過將 Spot 選項設定為要一起啟動的最小目標容量，將叢集設定 MinTargetCapacity 為僅在可啟動最小目標容量時啟動執行個體。

12 個啟動範本覆寫會擁有不同執行個體類型和子網 (每個都位於不同的可用區域)，但權重容量相同。總目標容量和最小目標容量設定為 20 個執行個體，預設購買選項為 Spot，Spot 分配策略為容量最佳化。只有在能夠同時啟動全部 20 個執行個體的情況下，EC2叢集才會從 Spot 容量集區啟動 20 個 Spot 執行個體，並使用啟動範本覆寫的最佳容量。

```
{
    "SpotOptions": {
        "AllocationStrategy": "capacity-optimized",
        "MinTargetCapacity": 20
    },
    "LaunchTemplateConfigs": [
      {
        "LaunchTemplateSpecification":{
            "LaunchTemplateName":"ec2-fleet-lt1",
            "Version":"$Latest"
        },
        "Overrides":[
            {
                "InstanceType":"c5.4xlarge",
                "SubnetId":"subnet-fae8c380"
            },
            {
                "InstanceType":"c5.4xlarge",
                "SubnetId":"subnet-e7188bab"
            },
```

```
            {
               "InstanceType":"c5.4xlarge",
               "SubnetId":"subnet-49e41922"
            },
            {
               "InstanceType":"c5d.4xlarge",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"c5d.4xlarge",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"c5d.4xlarge",
               "SubnetId":"subnet-49e41922"
            },
            {
               "InstanceType":"m5.4xlarge",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"m5.4xlarge",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"m5.4xlarge",
               "SubnetId":"subnet-49e41922"
            },
            {
               "InstanceType":"m5d.4xlarge",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"m5d.4xlarge",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"m5d.4xlarge",
               "SubnetId":"subnet-49e41922"
            }
         ]
      }
   ],
   "TargetCapacitySpecification": {
```

```
        "TotalTargetCapacity": 20,
        "DefaultTargetCapacityType": "spot"
    },
    "Type": "instant"
}
```

範例 7：僅當可在單個可用區域中啟動相同執行個體類型的最低目標容量時，才能啟動 Spot 執行個體

只有當最小目標容量可以在單一可用區域中以單一執行個體類型啟動時，您才可以設定叢集啟動執行個體，方法是將 Spot 選項 MinTargetCapacity 設定為要與 SingleInstanceType 和選 SingleAvailabilityZone 項一起啟動的最小目標容量。

覆寫啟動範本的 12 個啟動規格擁有不同執行個體類型和子網 (每個都位於不同的可用區域)，但權重容量相同。目標容量總計和最小目標容量都設定為 20 個執行個體，預設的購買選項為 Spot，Spot 配置策略為容量最佳化、真實且 SingleInstanceType SingleAvailabilityZone 為真。只有在能夠同時啟動全部 20 個執行個體的情況下，EC2叢集才會從 Spot 容量集區啟動 20 個相同執行個體類型的 Spot 執行個體，並使用啟動規格提供最佳容量。

```
{
    "SpotOptions": {
        "AllocationStrategy": "capacity-optimized",
        "SingleInstanceType": true,
        "SingleAvailabilityZone": true,
        "MinTargetCapacity": 20
    },
    "LaunchTemplateConfigs": [
      {
         "LaunchTemplateSpecification":{
            "LaunchTemplateName":"ec2-fleet-lt1",
            "Version":"$Latest"
         },
         "Overrides":[
            {
               "InstanceType":"c5.4xlarge",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"c5.4xlarge",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"c5.4xlarge",
               "SubnetId":"subnet-49e41922"
```

```
            },
            {
               "InstanceType":"c5d.4xlarge",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"c5d.4xlarge",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"c5d.4xlarge",
               "SubnetId":"subnet-49e41922"
            },
            {
               "InstanceType":"m5.4xlarge",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"m5.4xlarge",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"m5.4xlarge",
               "SubnetId":"subnet-49e41922"
            },
            {
               "InstanceType":"m5d.4xlarge",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"m5d.4xlarge",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"m5d.4xlarge",
               "SubnetId":"subnet-49e41922"
            }
         ]
      }
   ],
   "TargetCapacitySpecification": {
       "TotalTargetCapacity": 20,
       "DefaultTargetCapacityType": "spot"
   },
```

```
    "Type": "instant"
}
```

範例 8：使用多個啟動範本來啟動執行個體

您可以透過指定多個啟動範本，將機群設定為針對不同執行個體類型或執行個體類型群組，啟動具有不同啟動規格的執行個體。在這個範例中，我們想要針對不同的執行個體類型設定不同的EBS磁碟區大小，而且我們已經在啟動範本中進行了設定。

在此範例中，我們根據大小，針對 3 個執行個體類型使用 3 種不同的啟動範本。所有啟動範本的啟動規格覆寫會根據例證類型使用例證權重。 vCPUs 總目標容量為 144 個單位，預設購買選項為 Spot，Spot 分配策略為容量最佳化。EC2機隊可以使用啟動範本啟動 9 c5n.4 倍大 (144 除以 16)，或使用啟動範本 ec2-Fleet-9xl，或使用啟動範本 ec2-Fleet-18xl (144 除以 72)，或者使用啟動範本 ec2-Fleet-18xl 的 2 c5n.18xlarge (144 除以 72)，或是將所需的重量混合為基礎的執行個體類型容量最佳化配置策略。

```
{
    "SpotOptions": {
        "AllocationStrategy": "capacity-optimized"
    },
    "LaunchTemplateConfigs": [
      {
          "LaunchTemplateSpecification":{
             "LaunchTemplateName":"ec2-fleet-lt-18xl",
             "Version":"$Latest"
          },
          "Overrides":[
             {
                "InstanceType":"c5n.18xlarge",
                "SubnetId":"subnet-fae8c380",
                "WeightedCapacity":72
             },
             {
                "InstanceType":"c5n.18xlarge",
                "SubnetId":"subnet-e7188bab",
                "WeightedCapacity":72
             },
             {
                "InstanceType":"c5n.18xlarge",
                "SubnetId":"subnet-49e41922",
                "WeightedCapacity":72
             }
```

```
            ]
        },
        {
            "LaunchTemplateSpecification":{
                "LaunchTemplateName":"ec2-fleet-lt-9xl",
                "Version":"$Latest"
            },
            "Overrides":[
                {
                    "InstanceType":"c5n.9xlarge",
                    "SubnetId":"subnet-fae8c380",
                    "WeightedCapacity":36
                },
                {
                    "InstanceType":"c5n.9xlarge",
                    "SubnetId":"subnet-e7188bab",
                    "WeightedCapacity":36
                },
                {
                    "InstanceType":"c5n.9xlarge",
                    "SubnetId":"subnet-49e41922",
                    "WeightedCapacity":36
                }
            ]
        },
        {
            "LaunchTemplateSpecification":{
                "LaunchTemplateName":"ec2-fleet-lt-4xl",
                "Version":"$Latest"
            },
            "Overrides":[
                {
                    "InstanceType":"c5n.4xlarge",
                    "SubnetId":"subnet-fae8c380",
                    "WeightedCapacity":16
                },
                {
                    "InstanceType":"c5n.4xlarge",
                    "SubnetId":"subnet-e7188bab",
                    "WeightedCapacity":16
                },
                {
                    "InstanceType":"c5n.4xlarge",
                    "SubnetId":"subnet-49e41922",
```

```
                "WeightedCapacity":16
            }
        ]
      }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 144,
        "DefaultTargetCapacityType": "spot"
    },
    "Type": "instant"
}
```

範例 9：啟動基於隨需執行個體的 Spot 執行個體

下列範例指定機群 20 個執行個體的總目標容量，以及 5 個隨需執行個體的目標容量。預設購買選項為 Spot。機群依照指定啟動 5 個隨需執行個體，但需要再啟動 15 個執行個體才能實現總目標容量。差異的購買選項計算方式為 TotalTargetCapacity — OnDemandTargetCapacity = DefaultTargetCapacityType，這會導致叢集啟動 15 個 Spot 執行個體，根據容量最佳化的配置策略，形成 12 個 Spot 容量集區之一。

```
{
    "SpotOptions": {
        "AllocationStrategy": "capacity-optimized"
    },
    "LaunchTemplateConfigs": [
      {
         "LaunchTemplateSpecification":{
            "LaunchTemplateName":"ec2-fleet-lt1",
            "Version":"$Latest"
         },
         "Overrides":[
            {
               "InstanceType":"c5.large",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"c5.large",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"c5.large",
               "SubnetId":"subnet-49e41922"
            },
```

```
            {
               "InstanceType":"c5d.large",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"c5d.large",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"c5d.large",
               "SubnetId":"subnet-49e41922"
            },
            {
               "InstanceType":"m5.large",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"m5.large",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"m5.large",
               "SubnetId":"subnet-49e41922"
            },
            {
               "InstanceType":"m5d.large",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"m5d.large",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"m5d.large",
               "SubnetId":"subnet-49e41922"
            }
         ]
      }
   ],
   "TargetCapacitySpecification": {
       "TotalTargetCapacity": 20,
       "OnDemandTargetCapacity": 5,
       "DefaultTargetCapacityType": "spot"
   },
```

```
    "Type": "instant"
}
```

範例 10：基於使用容量預留和優先順序分配策略的隨需執行個體，使用容量最佳化分配策略啟動 Spot 執行個體

您可以將容量保留的使用策略設定為，將叢集設定為在啟動具有預設目標容量類型為 Spot 的隨需執行個體基礎時，先使用隨需容量保留 use-capacity-reservations-first。因此，如果多個執行個體集區有未使用的 容量預留，則會套用所選擇的隨需分配策略。在此範例中，隨需分配策略為優先順序。

在此範例中，有 6 個未使用的容量預留可供使用。這小於機群的目標隨需容量 10 個隨需執行個體。

帳戶在 2 個集區中有以下 6 個未使用的容量預留。每個集區中的容量保留數會以表示 AvailableInstanceCount。

```
{
    "CapacityReservationId": "cr-111",
    "InstanceType": "m5.large",
    "InstancePlatform": "Linux/UNIX",
    "AvailabilityZone": "us-east-1a",
    "AvailableInstanceCount": 3,
    "InstanceMatchCriteria": "open",
    "State": "active"
}

{
    "CapacityReservationId": "cr-222",
    "InstanceType": "c5.large",
    "InstancePlatform": "Linux/UNIX",
    "AvailabilityZone": "us-east-1a",
    "AvailableInstanceCount": 3,
    "InstanceMatchCriteria": "open",
    "State": "active"
}
```

下列機群組態僅顯示此範例的相關組態。隨需配置策略的優先順序，容量保留的使用策略為。 use-capacity-reservations-firstSpot 分配策略是容量最佳化。目標總容量為 20、隨需目標容量為 10、預設目標容量類型為 spot。

```
{
    "SpotOptions": {
        "AllocationStrategy": "capacity-optimized"
```

```
    },
    "OnDemandOptions":{
       "CapacityReservationOptions": {
         "UsageStrategy": "use-capacity-reservations-first"
       },
       "AllocationStrategy":"prioritized"
    },
    "LaunchTemplateConfigs": [
      {
         "LaunchTemplateSpecification":{
            "LaunchTemplateName":"ec2-fleet-lt1",
            "Version":"$Latest"
         },
         "Overrides":[
            {
               "InstanceType":"c5.large",
               "SubnetId":"subnet-fae8c380",
               "Priority": 1.0
            },
            {
               "InstanceType":"c5.large",
               "SubnetId":"subnet-e7188bab",
               "Priority": 2.0
            },
            {
               "InstanceType":"c5.large",
               "SubnetId":"subnet-49e41922",
               "Priority": 3.0
            },
            {
               "InstanceType":"c5d.large",
               "SubnetId":"subnet-fae8c380",
               "Priority": 4.0
            },
            {
               "InstanceType":"c5d.large",
               "SubnetId":"subnet-e7188bab",
               "Priority": 5.0
            },
            {
               "InstanceType":"c5d.large",
               "SubnetId":"subnet-49e41922",
               "Priority": 6.0
            },
```

```
            {
               "InstanceType":"m5.large",
               "SubnetId":"subnet-fae8c380",
               "Priority": 7.0
            },
            {
               "InstanceType":"m5.large",
               "SubnetId":"subnet-e7188bab",
               "Priority": 8.0
            },
            {
               "InstanceType":"m5.large",
               "SubnetId":"subnet-49e41922",
               "Priority": 9.0
            },
            {
               "InstanceType":"m5d.large",
               "SubnetId":"subnet-fae8c380",
               "Priority": 10.0
            },
            {
               "InstanceType":"m5d.large",
               "SubnetId":"subnet-e7188bab",
               "Priority": 11.0
            },
            {
               "InstanceType":"m5d.large",
               "SubnetId":"subnet-49e41922",
               "Priority": 12.0
            }
         ]
      }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 20,
        "OnDemandTargetCapacity": 10,
        "DefaultTargetCapacityType": "spot"
    },
    "Type": "instant"
}
```

使用上述組態建立 instant 類型機群之後，會啟動以下 20 個執行個體以符合目標容量：

- us-east-1a 中的 7 個 c5.large 隨需執行個體 – us-east-1a 中的 c5.large 是第一優先順序，並有 3 個未使用 c5.large 容量預留可供使用。容量預留會先用來啟動 3 個隨需執行個體，再根據隨需分配策略啟動額外的 4 個隨需執行個體，即此範例中的優先順序。
- us-east-1a 中的 3 個 m5.large 隨需執行個體 – us-east-1a 中的 m5.large 是第二優先順序，並有 3 個未使用 c3.large 容量預留可供使用。
- 來自 12 個 Spot 容量集區中的 10 個 Spot 執行個體，該集區根據容量最佳化的分配策略具有最佳容量。

啟動叢集之後，您可以執行describe-capacity-reservations以查看剩餘的未使用容量保留數量。在此範例中，您應該會看到下列回應，其中顯示了已使用的所有 c5.large 和 m5.large 容量預留。

```
{
    "CapacityReservationId": "cr-111",
    "InstanceType": "m5.large",
    "AvailableInstanceCount": 0
}

{
    "CapacityReservationId": "cr-222",
    "InstanceType": "c5.large",
    "AvailableInstanceCount": 0
}
```

範例 11：使用 capacity-optimized-prioritized 配置策略啟動競價型執行個體

下列範例會指定「即時EC2叢集」類型中所需的參數：啟動樣板、目標產能、預設採購選項，以及啟動範本覆寫。啟動範本由其啟動範本名稱和版本編號識別。覆寫啟動範本的 12 個啟動規格擁有已指派優先順序的 4 個不同執行個體類型和 3 個不同的子網，每個都位於不同的可用區域。叢集的目標容量為 20 個執行個體，而預設的購買選項為 Spot，這會導致叢集嘗試根據 capacity-optimized-prioritized 配置策略從 12 個 Spot 容量集區中的一個啟動 20 個 Spot 執行個體，這些執行個體會盡最大努力實作優先順序，但會先針對容量進行最佳化。

```
{
    "SpotOptions": {
        "AllocationStrategy": "capacity-optimized-prioritized"
    },
    "LaunchTemplateConfigs": [
      {
         "LaunchTemplateSpecification":{
            "LaunchTemplateName":"ec2-fleet-lt1",
```

```
            "Version":"$Latest"
         },
         "Overrides":[
            {
               "InstanceType":"c5.large",
               "SubnetId":"subnet-fae8c380",
               "Priority": 1.0
            },
            {
               "InstanceType":"c5.large",
               "SubnetId":"subnet-e7188bab",
               "Priority": 1.0
            },
            {
               "InstanceType":"c5.large",
               "SubnetId":"subnet-49e41922",
               "Priority": 1.0
            },
            {
               "InstanceType":"c5d.large",
               "SubnetId":"subnet-fae8c380",
               "Priority": 2.0
            },
            {
               "InstanceType":"c5d.large",
               "SubnetId":"subnet-e7188bab",
               "Priority": 2.0
            },
            {
               "InstanceType":"c5d.large",
               "SubnetId":"subnet-49e41922",
               "Priority": 2.0
            },
            {
               "InstanceType":"m5.large",
               "SubnetId":"subnet-fae8c380",
               "Priority": 3.0
            },
            {
               "InstanceType":"m5.large",
               "SubnetId":"subnet-e7188bab",
               "Priority": 3.0
            },
            {
```

```
                "InstanceType":"m5.large",
                "SubnetId":"subnet-49e41922",
                "Priority": 3.0
            },
            {
                "InstanceType":"m5d.large",
                "SubnetId":"subnet-fae8c380",
                "Priority": 4.0
            },
            {
                "InstanceType":"m5d.large",
                "SubnetId":"subnet-e7188bab",
                "Priority": 4.0
            },
            {
                "InstanceType":"m5d.large",
                "SubnetId":"subnet-49e41922",
                "Priority": 4.0
            }
        ]
    }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 20,
        "DefaultTargetCapacityType": "spot"
    },
    "Type": "instant"
}
```

範例 12：指定 Systems Manager 參數而非 AMI ID

下列範例會使用啟動範本來指定叢集中執行個體的組態。在此範例中`ImageId`，針對使用系統管理員參數來參照，而不AMI是指定 AMI ID。執行個體啟動時，Systems Manager 參數將解析為 AMI ID。

在此範例中，系 Systems Manager 參數以有效的格式指定：`resolve:ssm:golden-ami`。Systems Manager 參數還有其他有效格式。如需詳細資訊，請參閱使用「Systems Manager」參數而不是 AMI ID。

> **Note**
>
> 機群類型必須為 `instant`。其他叢集類型不支援指定系統管理員參數而非AMI識別碼。

```
{
    "LaunchTemplateData": {
        "ImageId": "resolve:ssm:golden-ami",
        "InstanceType": "m5.4xlarge",
        "TagSpecifications": [{
            "ResourceType": "instance",
            "Tags": [{
                "Key": "Name",
                "Value": "webserver"
            }]
        }]
    }
}
```

為您的EC2車隊或 Spot 艦隊設定消費上限

您可以設定每小時願意花費在EC2艦隊或 Spot 艦隊上的金額上限。達到您的支出限制時，即使尚未達到目標容量，叢集也會停止啟動執行個體。

隨需執行個體和 Spot 執行個體有個別的支出限制。

設定EC2叢集中隨需執行個體和 Spot 執行個體的支出限制

使用創建艦隊（AWS CLI）命令和下列參數：

- 針對隨需執行個體：在`OnDemandOptions`結構中，在`MaxTotalPrice`欄位中指定您的支出限制。
- 對於競價型執行個體：在`SpotOptions`結構中，在`MaxTotalPrice`欄位中指定您的支出限制。

設定 Spot 叢集中隨需執行個體和 Spot 執行個體的支出限制

您可以使用 Amazon EC2 主控台或設 AWS CLI 定您的支出限制。

(主控台) 建立 Spot 叢集時，請選取 [設定 Spot 執行個體的最大成本] 核取方塊，然後在 [設定最高費用 (每小時)] 中輸入值。如需詳細資訊，請參閱步驟 6. 中使用已定義的參數建立 Spot Fleet 請求 (主控台)的。

(AWS CLI) 使用指request-spot-fleet令及下列參數：

- 針對隨需執行個體：在`OnDemandMaxTotalPrice`欄位中指定您的支出限制。
- 對於競價型執行個體：在`SpotMaxTotalPrice`欄位中指定您的支出限制。

範例

以下範例顯示兩個不同的案例。在第一個範例中，叢集在達到針對隨需執行個體設定的目標容量 (OnDemandTargetCapacity) 時停止啟動隨需執行個體。在第二個範例中，叢集在達到您願意為隨需執行個體每小時支付的最高金額時停止啟動隨需執行個體 (MaxTotalPrice)。

範例：達到目標容量時停止啟動隨需執行個體

假設有 m4.large 隨需執行個體 的請求，其中：

- 隨需價格：每小時 0.10 USD
- OnDemandTargetCapacity：10
- MaxTotalPrice：1.50 USD

叢集會啟動 10 個隨需執行個體，因為隨需執行個體的總計 $1.00 (10 個執行個體 x 0.10 美元) 不超過 $1.50 美元。MaxTotalPrice

範例：達到最高總價時停止啟動隨需執行個體

假設有 m4.large 隨需執行個體 的請求，其中：

- 隨需價格：每小時 0.10 USD
- OnDemandTargetCapacity：10
- MaxTotalPrice：0.80 USD

如果叢集啟動隨需目標容量 (10 個隨需執行個體)，則每小時的總成本為 $1.00。這超出為隨需執行個體指定的 MaxTotalPrice 的金額 (0.80 USD)。為了避免支出超過您願意支付的費MaxTotalPrice用，叢集僅啟動 8 個隨需執行個體 (低於隨需目標容量)，因為啟動次數會超過隨需執行個體的執行個體。

爆量效能執行個體

如果您使用高載效能執行個體類型啟動 Spot 執行個體，並且計劃立即使用高載效能 Spot 執行個體，而且沒有閒置時間來累積積CPU分，建議您以標準模式啟動它們，以避免支付更高的費用。如果您以無限制模式啟動高載效能 Spot 執行個體並CPU立即爆發，您將花費剩餘的點數進行爆量。如果您在短時間內使用執行個體，則執行個體沒有時間累積積CPU分來支付剩餘積分，而且在您終止執行個體時，系統會向您收取剩餘積分的費用。

無限制模式僅適用於高載效能 Spot 執行個體的執行時間足以累積積CPU分以進行爆發的 Spot 執行個體。否則，支付剩餘額度會使爆量效能 競價型執行個體比使用其他執行個體更為昂貴。如需詳細資訊，請參閱何時使用無限模式與固定模式 CPU。

啟動額度旨在透過提供足夠的運算資源來設定執行個體，創造 T2 執行個體具有生產力的初始啟動體驗。不允許重複啟動 T2 執行個體來獲得新的啟動額度。如果您需要持續CPU，您可以賺取積分 (透過閒置一段時間)、對 T2 Spot 執行個體使用無限制模式，或使用具有專CPU用的執行個體類型。

指定EC2叢集或 Spot 叢集的執行個體類型選取屬性

建立EC2叢集或 Spot 叢集時，您必須指定一或多個執行個體類型，以便在叢集中設定隨需執行個體和 Spot 執行個體。作為手動指定執行個體類型的替代方法，您可以指定執行個體必須具有的屬性，Amazon EC2 將使用這些屬性識別所有執行個體類型。這就是所謂的屬性型執行個體類型選項。例如，您可以指定執行個體 vCPUs 所需的最小和最大數量，叢集會使用任何符合這些 v 需CPU求的可用執行個體類型啟動執行個體。

屬性型執行個體類型選擇非常適合可對其使用哪些執行個體類型具有彈性的工作負載和架構，例如執行容器或 Web 機群、處理大數據，以及實作持續整合和部署 (CI/CD) 工具時。

優勢

屬性型執行個體類型選擇具有下列優勢：

- 輕鬆使用正確的執行個體類型 — 有這麼多可用的執行個體類型，因此尋找適合您工作負載的執行個體類型可能非常耗時。當您指定執行個體類型屬性時，執行個體類型會自動具有工作負載所需的屬性。
- 簡化組態 — 若要手動指定叢集的多個執行個體類型，您必須為每個執行個體類型建立個別的啟動範本覆寫。但是，有了屬性型執行個體類型選擇，若要提供多個執行個體類型，您只需要在啟動範本或在啟動範本覆寫中指定執行個體屬性。
- 自動使用新的執行個體類型 — 當您指定執行個體屬性而非執行個體類型時，您的叢集可以在發行時使用新一代執行個體類型，「future 校對」叢集的配置。
- 執行個體類型彈性 — 當您指定執行個體屬性而非執行個體類型時，叢集可以從各種執行個體類型中進行選取，以啟動競價型執行個體類型，這符合 Spot 執行個體類型彈性的最佳實務。

主題

- 屬性型執行個體類型選擇的運作方式
- 價格保護
- 考量事項

- 使用以屬性為基礎的執行個體類型選擇建立EC2叢集
- 建立具有屬性型執行個體類型選擇的 Spot Fleet
- 有效且無效的EC2叢集配置範例
- 有效且無效的競價型叢集組態範例
- 預覽具有所指定屬性的執行個體類型

屬性型執行個體類型選擇的運作方式

若要在機群組態中使用屬性型執行個體類型選擇，請將執行個體類型清單取代為執行個體所需的執行個體屬性清單。EC2叢集或 Spot 叢集會在任何具有指定執行個體屬性的可用執行個體類型上啟動執行個體。

主題

- 執行個體屬性的類型
- 要在何處設定屬性型執行個體類型選擇
- EC2叢集或 Spot 叢集在佈建叢集時如何使用以屬性為基礎的執行個體類型選取

執行個體屬性的類型

您可以指定數個執行個體屬性來表示運算需求，例如：

- v CPU 計數 — 每個執行個體的最小和最大 vCPUs 數目。
- 記憶體 — 每個執行個體 GiBs 的記憶體下限和最大記憶體。
- 本地存儲 — 是否使用EBS或實例存儲卷用於本地存儲。
- 高載效能 — 是否使用 T 執行個體系列，包括 T4G、T3A、T3 和 T2 類型。

如需每個屬性和預設值的說明，請參閱 Amazon 參EC2API考InstanceRequirements中的。

要在何處設定屬性型執行個體類型選擇

根據您使用的是主控台還是 AWS CLI，您可以指定以屬性為基礎之執行個體類型選取的執行個體屬性，如下所示：

在主控台中，您可以在下列叢集設定元件中指定執行個體屬性：

- 在啟動範本中，則參考機群請求中的啟動範本

- (僅限 Spot 叢集) 在叢集請求中

在中 AWS CLI，您可以在下列一或所有叢集組態元件中指定執行個體屬性：

- 在啟動範本中，則參考機群請求中的啟動範本
- 在啟動範本覆寫中

  如果您想要混合使用不同的執行個體AMIs，可以在多個啟動範本覆寫中指定執行個體屬性。例如，不同的執行個體類型可以使用 x86 和 ARM 型處理器。
- (僅限 Spot 艦隊) 在啟動規格中

EC2叢集或 Spot 叢集在佈建叢集時如何使用以屬性為基礎的執行個體類型選取

EC2艦隊或 Spot 艦隊以下列方式佈建艦隊：

- 它會識別具有指定屬性的執行個體類型。
- 它使用價格保護來確定要排除哪些執行個體類型。
- 它決定要考慮根據具有相符執行個體類型的區域或可用區域啟動執行個體的容量集區。 AWS
- 它會套用指定的配置策略，以決定要從哪個容量集區啟動執行個體。

  請注意，以屬性為基礎的執行個體類型選取不會挑選要佈建叢集的容量集區；這就是配置策略的工作。

  如果您指定配置策略，叢集將會根據指定的配置策略啟動執行個體。
  - 對於競價型執行個體，屬性型執行個體類型選擇支援價格容量最佳化、容量最佳化和最低價格配置策略。請注意，我們不建議使用最低價 Spot 配置策略，因為 Spot 執行個體的中斷風險最高。
  - 對於隨需執行個體，屬性型執行個體類型選取支援最低價格配置策略。
- 如果具有所指定執行個體屬性的執行個體類型沒有容量，則無法啟動任何執行個體，而且機群會傳回錯誤。

價格保護

價格保護是一項功能，可防止您的EC2叢集或 Spot Fleet 使用執行個體類型，即使它們恰好符合您指定的屬性，您也會認為這些執行個體類型過於昂貴。要使用價格保護，請設置價格閾值。然後，當 Amazon EC2 選取具有屬性的執行個體類型時，就會排除定價高於閾值的執行個體類型。

Amazon EC2 計算價格閾值的方式如下：

- Amazon EC2 首先從符合屬性的執行個體類型中識別出價格最低的執行個體類型。
- EC2然後，Amazon 會採用您為價格保護參數指定的值 (以百分比表示)，並將其與識別執行個體類型的價格相乘。結果是用作價格閾值的價格。

隨需執行個體和 Spot 執行個體有不同的價格門檻。

當您使用以屬性為基礎的執行個體類型選項建立叢集時，預設會啟用價格保護。您可以保留預設值，也可以指定自己的值。

您也可以關閉價格保護。若要指出沒有價格保護閾值，請指定較高的百分比值，例如999999。

主題

- 如何識別最低價執行個體類型
- 隨需執行個體價格保
- 競價型實例價格保護
- 指定價格保護門檻

如何識別最低價執行個體類型

Amazon 透過識別與指EC2定屬性相符的執行個體類型中最低價格的執行個體類型，來決定價格閾值的價格。它通過以下方式執行此操作：

- 它首先查看與您的屬性匹配的當前一代 C，M 或 R 實例類型。如果找到任何相符項目，就會識別價格最低的執行個體類型。
- 如果沒有相符項目，則會查看符合您屬性的任何目前產生的執行個體類型。如果找到任何相符項目，就會識別價格最低的執行個體類型。
- 如果沒有相符項目，就會查看符合屬性的任何上一代執行個體類型，並識別價格最低的執行個體類型。

隨需執行個體價格保

隨需執行個體類型的價格保護閾值是以比識別的最低價隨需執行個體類型 (`OnDemandMaxPricePercentageOverLowestPrice`) 高的百分比來計算。您可以指定您願意支付的更高百分比。如果您未指定此參數，則系統會使用預設值來計算比識別價格高出 20% 的價格保護閾值。`20`

例如，如果識別的隨需執行個體價格為且您指定25，則價格閾值會高出 25% 0.4271。0.4271它的計算方式如下：0.4271 * 1.25 = 0.533875計算的價格是您願意為隨需執行個體支付的最高價格，在此範例中，Amazon EC2 將排除任何成本超過的隨需執行個體類型0.533875。

競價型實例價格保護

根據預設，Amazon EC2 會自動套用最佳競價型執行個體價格保護，以持續從各種執行個體類型中進行選取。您也可以自行手動設定價格保護。但是，讓 Amazon 為您EC2做到這一點可以提高您的 Spot 容量滿足的可能性。

您可以使用下列其中一個選項手動指定價格保護。如果您手動設定價格保護，我們建議您使用第一個選項。

- 已識別最低價格隨需執行個體類型的百分比 [MaxSpotPriceAsPercentageOfOptimalOnDemandPrice]

  例如，如果識別的隨需執行個體類型價格為且您指定60，則價格閾值為 60% 0.4271。0.4271它的計算方式如下：0.4271 * 0.60 = 0.25626計算的價格是您願意為競價型執行個體支付的最高價格，在此範例中，Amazon EC2 將排除任何成本超過的 Spot 執行個體類型0.25626。
- 高於識別的最低價 Spot 執行個體類型的百分比 [SpotMaxPricePercentageOverLowestPrice]

  例如，如果識別的競價型執行個體類型價格為且您指定25，則價格閾值將高出 25% 0.1808。0.1808它的計算方式如下：0.1808 * 1.25 = 0.226計算的價格是您願意為競價型執行個體支付的最高價格，在此範例中，Amazon EC2 將排除任何成本超過的 Spot 執行個體類型0.266。我們不建議使用此參數，因為現貨價格可能會波動，因此您的價格保護閾值也可能會波動。

指定價格保護門檻

若要使用指定價格保護閾值 AWS CLI

使用建立EC2叢集或 Spot 叢集時 AWS CLI，針對以屬性為基礎的執行個體類型選取設定叢集，然後執行下列動作：

- 若要指定隨需執行個體價格保護閾值，請在JSON組態檔案的InstanceRequirements結構中OnDemandMaxPricePercentageOverLowestPrice，以百分比輸入價格保護閾值。
- 若要指定競價型執行個體價格保護閾值，請在JSON組態檔案的InstanceRequirements結構中指定下列其中一個參數：

- 對於`MaxSpotPriceAsPercentageOfOptimalOnDemandPrice`，以百分比輸入價格保護閾值。
- 對於`SpotMaxPricePercentageOverLowestPrice`，以百分比輸入價格保護閾值。

如需詳細資訊，請參閱 使用以屬性為基礎的執行個體類型選擇建立EC2叢集 或 建立具有屬性型執行個體類型選擇的 Spot Fleet。

(僅限競價型叢集) 使用主控台指定價格保護閾值

在主控台中建立 Spot 叢集時，針對以屬性為基礎的執行個體類型選擇設定叢集，然後執行下列動作：

- 若要指定隨需執行個體價格保護閾值，請在 [其他執行個體屬性] 下選擇 [隨需價格保護]，選擇 [新增屬性]，然後以百分比輸入價格保護閾值。
- 若要指定競價型執行個體價格保護閾值、其他執行個體屬性、選擇競價型價格保護、選擇 [新增屬性]、選擇價格基礎的基礎值，然後以百分比輸入價格保護閾值。

> **Note**
>
> 建立叢集時，如果您設定`TargetCapacityUnitType`為`vcpu`或`memory-mib`，則會根據每 V CPU 或每個記憶體的價格而非每個執行個體價格套用價格保護閾值。

考量事項

- 您可以在EC2叢集或 Spot 叢集中指定執行個體類型或執行個體屬性，但不能同時指定兩者。

  使用時CLI，啟動範本覆寫會覆寫啟動範本。例如，如果啟動範本包含執行個體類型，且啟動範本覆寫包含執行個體屬性，則執行個體屬性所識別的執行個體將覆寫啟動範本中的執行個體類型。
- 使用時CLI，當您將實體屬性指定為覆寫時，您也無法指定權重或優先順序。
- 您最多可以在請求組態中指定四個 `InstanceRequirements` 結構。

使用以屬性為基礎的執行個體類型選擇建立EC2叢集

您可以使用將EC2叢集設定為使用屬性型執行個體類型選取。 AWS CLI

使用以屬性為基礎的執行個體類型選取建立EC2叢集 ()AWS CLI

使用建立叢集 (AWS CLI) 指令來建立叢集。EC2在檔案中指定叢集配JSON置。

```
aws ec2 create-fleet \
    --region us-east-1 \
    --cli-input-json file://file_name.json
```

範例 *file_name*.json 檔案

下列範例包含將EC2叢集設定為使用以屬性為基礎的執行個體類型選取項目的參數，並在後面接著文字說明。

```
{
 "SpotOptions": {
  "AllocationStrategy": "price-capacity-optimized"
 },
 "LaunchTemplateConfigs": [{
  "LaunchTemplateSpecification": {
   "LaunchTemplateName": "my-launch-template",
   "Version": "1"
  },
  "Overrides": [{
   "InstanceRequirements": {
    "VCpuCount": {
     "Min": 2
    },
    "MemoryMiB": {
     "Min": 4
    }
   }
  }]
 }],
 "TargetCapacitySpecification": {
  "TotalTargetCapacity": 20,
  "DefaultTargetCapacityType": "spot"
 },
 "Type": "instant"
}
```

屬性型執行個體類型選取的屬性在 `InstanceRequirements` 結構中予以指定。在此範例中，會指定兩個屬性：

- `VCpuCount`— 至少指 vCPUs 定 2。由於未指定最大值，因此沒有最大限制。
- `MemoryMiB` – 至少指定 4 MiB 的記憶體。由於未指定最大值，因此沒有最大限制。

任何具有 2 或更多記憶體 vCPUs 和 4 MiB 或更多記憶體的執行個體類型都會被識別。不過，當EC2叢集佈建叢集時，價格保護和配置策略可能會排除某些執行個體類型。

如需您可以指定的所有可能屬性的清單和說明，請參閱 Amazon EC2 API 參考InstanceRequirements中的。

> **Note**
>
> 當 `InstanceRequirements` 併入機群組態中時，必須排除 `InstanceType` 和 `WeightedCapacity`；它們無法與執行個體屬性同時決定機群組態。

JSON也包含下列叢集配置：

- `"AllocationStrategy": "price-capacity-optimized"` – 機群中 Spot 執行個體的分配策略。
- `"LaunchTemplateName": "my-launch-template", "Version": "1"` – 啟動範本包含一些執行個體組態資訊，但如果指定了任何執行個體類型，則會由 `InstanceRequirements` 中指定的屬性進行覆寫。
- `"TotalTargetCapacity": 20` – 目標容量為 20 個執行個體。
- `"DefaultTargetCapacityType": "spot"` – 預設容量為 Spot 執行個體。
- `"Type": "instant"` – 機群的要求類型為 `instant`。

建立具有屬性型執行個體類型選擇的 Spot Fleet

您可以使用 Amazon EC2 主控台或將叢集設定為使用屬性型執行個體類型選取。 AWS CLI

主題

- 使用主控台建立 Spot Fleet
- 使用 AWS CLI建立 Spot Fleet

使用主控台建立 Spot Fleet

為屬性型執行個體類型選擇配置 Spot Fleet (主控台)

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。

2. 在導覽窗格中，選擇 Spot Requests (Spot 請求)，然後選擇 Request Spot Instances (請求 Spot 執行個體)。
3. 遵循步驟來建立 Spot Fleet。如需詳細資訊，請參閱 使用已定義的參數建立 Spot Fleet 請求 (主控台)。

   在建立 Spot Fleet 時，為屬性型執行個體類型選擇設定機群，如下所示：

   a. 對於 Instance type requirements (執行個體類型請求)，選擇 Specify instance attributes that match your compute requirements (指定符合運算需求的執行個體屬性)。
   b. 對於 vCPUs，輸入所需的最小和最大數目vCPUs。若要指定無限制，請選取 No minimum (無下限)、No maximum (無上限)，或兩者。
   c. 對於 Memory (GiB) (記憶體 (GiB))，輸入所需記憶體數量的下限和上限。若要指定無限制，請選取 No minimum (無下限)、No maximum (無上限)，或兩者。
   d. (選用) 對於 Additional instance attributes (其他執行個體屬性)，您可以選擇性地指定一或多個屬性，以更詳細地表達您的運算需求。每個額外屬性都會將進一步的限制新增至您的請求。
   e. (選用) 展開 Preview matching instance types (預覽相符的執行個體類型)，以檢視具有所指定屬性的執行個體類型。

使用 AWS CLI建立 Spot Fleet

若要為屬性型執行個體類型選取設定 Spot Fleet (AWS CLI)

使用 request-spot-fleet(AWS CLI) 指令建立競價型叢集。在檔案中指定叢集配JSON置。

```
aws ec2 request-spot-fleet \
    --region us-east-1 \
    --spot-fleet-request-config file://file_name.json
```

範例 *file_name*.json 檔案

下列範例包含將 Spot 機群設定為使用屬性型執行個體類型選取的參數，其後是文字說明。

```
{
 "AllocationStrategy": "priceCapacityOptimized",
 "TargetCapacity": 20,
 "Type": "request",
 "LaunchTemplateConfigs": [{
  "LaunchTemplateSpecification": {
   "LaunchTemplateName": "my-launch-template",
```

```
    "Version": "1"
   },
   "Overrides": [{
    "InstanceRequirements": {
     "VCpuCount": {
      "Min": 2
     },
     "MemoryMiB": {
      "Min": 4
     }
    }
   }]
  }]
 }
```

屬性型執行個體類型選取的屬性在 `InstanceRequirements` 結構中予以指定。在此範例中，會指定兩個屬性：

- `VCpuCount`— 至少指 vCPUs 定 2。由於未指定最大值，因此沒有最大限制。
- `MemoryMiB` – 至少指定 4 MiB 的記憶體。由於未指定最大值，因此沒有最大限制。

任何具有 2 或更多記憶體 vCPUs 和 4 MiB 或更多記憶體的執行個體類型都會被識別。但是，當 Spot 機群佈建機群時，價格保護和分配策略可能會排除某些執行個體類型。

如需您可以指定的所有可能屬性的清單和說明，請參閱 Amazon EC2 API 參考InstanceRequirements中的。

> **Note**
>
> 當 `InstanceRequirements` 併入機群組態中時，必須排除 `InstanceType` 和 `WeightedCapacity`；它們無法與執行個體屬性同時決定機群組態。

JSON也包含下列叢集配置：

- `"AllocationStrategy": "priceCapacityOptimized"` – 機群中 Spot 執行個體的分配策略。
- `"LaunchTemplateName": "my-launch-template", "Version": "1"` – 啟動範本包含一些執行個體組態資訊，但如果指定了任何執行個體類型，則會由 `InstanceRequirements` 中指定的屬性進行覆寫。

- "TargetCapacity": *20* – 目標容量為 20 個執行個體。
- "Type": "*request*" – 機群的要求類型為 request。

有效且無效的EC2叢集配置範例

如果您使用 AWS CLI 建立EC2叢集，則必須確定您的叢集組態有效。下列範例顯示有效和無效的組態。

當組態包含下列項目時，視為無效：

- 同時具有 Overrides 和 InstanceRequirements 的單一 InstanceType
- 兩個 Overrides 結構, 一個具有 InstanceRequirements，而另一個具有 InstanceType
- 兩個在同一 InstanceRequirements 內具有重疊屬性值的 LaunchTemplateSpecification 結構

範例組態

- 有效組態：具有覆寫的單一啟動範本
- 有效組態：具有多個的單一啟動範本 InstanceRequirements
- 有效組態：兩個啟動範本，每個都具有覆寫
- 有效組態：僅限指定的 InstanceRequirements，沒有重疊的屬性值
- 組態無效：Overrides 包含 InstanceRequirements 和 InstanceType
- 組態無效：兩個 Overrides 包含 InstanceRequirements 和 InstanceType
- 組態無效：重疊屬性值

有效組態：具有覆寫的單一啟動範本

下列組態有效。其包含一個啟動範本和一個 Overrides 結構 (其中包含一個 InstanceRequirements 結構)。範例組態的文字說明如下。

```
{
        "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateName": "My-launch-template",
                "Version": "1"
                },
```

```
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 2,
                            "Max": 8
                        },
                        "MemoryMib": {
                            "Min": 0,
                            "Max": 10240
                        },
                        "MemoryGiBPerVCpu": {
                            "Max": 10000
                        },
                        "RequireHibernateSupport": true
                    }
                }
            ]
        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 5000,
        "DefaultTargetCapacityType": "spot",
        "TargetCapacityUnitType": "vcpu"
        }
    }
}
```

InstanceRequirements

若要使用屬性型執行個體選擇，您必須在組態中包含 `InstanceRequirements` 結構，並為機群中的執行個體指定所需的屬性。

在上述範例中，指定了下列執行個體屬性：

- `VCpuCount`— 執行個體類型必須至少為 2 且最多 8 個vCPUs。
- `MemoryMiB` – 執行個體類型必須具有最多 10240 MiB 的記憶體。最小值若為 0，表示沒有最小限制。
- `MemoryGiBPerVCpu`— 執行個體類型每 v CPU 最多必須有 10,000 GiB 的記憶體。`Min` 為選用參數。若省略它，表示沒有最小限制。

TargetCapacityUnitType

TargetCapacityUnitType 參數會指定目標容量的單位。在此範例中，目標容量為5000且目標容量單位類型為vcpu，它們共同指定所需的目標容量 5,000 vCPUs。EC2叢集將啟動足夠的執行個體，因此叢集 vCPUs 中的總數為 5,000 個vCPUs。

有效組態：具有多個的單一啟動範本 InstanceRequirements

下列組態有效。其包含一個啟動範本和一個 Overrides 結構 (其中包含兩個 InstanceRequirements 結構)。中指定VCpuCount的屬性InstanceRequirements有效，因為值不會重疊 — 第一個InstanceRequirements結構指定 a 為 0-2vCPUs，而第二個InstanceRequirements結構則指定 4-8。vCPUs

```
{
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 2
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                },
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 4,
                            "Max": 8
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                }
```

```
                ]
            }
        ],
        "TargetCapacitySpecification": {
            "TotalTargetCapacity": 1,
            "DefaultTargetCapacityType": "spot"
        }
    }
}
```

有效組態：兩個啟動範本，每個都具有覆寫

下列組態有效。其包含兩個啟動範本，每個都具有一個 `Overrides` 結構 (其中包含一個 `InstanceRequirements` 結構)。這個組態有助於同一個機群中的 `arm` 和 `x86` 架構支援。

```
{
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "armLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 2
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                },
                {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "x86LaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
```

```
                            "Min": 0,
                            "Max": 2
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                }
              ]
            }
        ],
         "TargetCapacitySpecification": {
            "TotalTargetCapacity": 1,
            "DefaultTargetCapacityType": "spot"
        }
    }
}
```

有效組態：僅限指定的 **InstanceRequirements**，沒有重疊的屬性值

下列組態有效。其包含兩個 LaunchTemplateSpecification 結構，每個都有一個啟動範本和一個 Overrides 結構 (其中包含一個 InstanceRequirements 結構)。中指定VCpuCount的屬性InstanceRequirements有效，因為值不會重疊 — 第一個InstanceRequirements結構指定 a 為 0-2vCPUs，而第二個InstanceRequirements結構則指定 4-8。vCPUs

```
{
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 2
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
```

```
                }
              ]
            },
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyOtherLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 4,
                            "Max": 8
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                }
              ]
            }
        ],
        "TargetCapacitySpecification": {
            "TotalTargetCapacity": 1,
            "DefaultTargetCapacityType": "spot"
        }
    }
}
```

組態無效：**Overrides** 包含 **InstanceRequirements** 和 **InstanceType**

下列組態無效。Overrides 結構同時包含 InstanceRequirements 和 InstanceType。針對 Overrides，您可以指定 InstanceRequirements 或 InstanceType，但不能同時指定兩者。

```
{
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
```

```
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 2
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                },
                {
                    "InstanceType": "m5.large"
                }
              ]
            }
        ],
        "TargetCapacitySpecification": {
            "TotalTargetCapacity": 1,
            "DefaultTargetCapacityType": "spot"
        }
    }
}
```

組態無效：兩個 **Overrides** 包含 **InstanceRequirements** 和 **InstanceType**

下列組態無效。Overrides 結構同時包含 InstanceRequirements 和 InstanceType。您可以指定 InstanceRequirements 或 InstanceType，但不能同時指定兩者，即使它們處於不同的 Overrides 結構也一樣。

```
{
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 2
```

```
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                }
              ]
            },
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyOtherLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceType": "m5.large"
                }
              ]
            }
        ],
         "TargetCapacitySpecification": {
            "TotalTargetCapacity": 1,
            "DefaultTargetCapacityType": "spot"
        }
    }
}
```

組態無效：重疊屬性值

下列組態無效。兩個 `InstanceRequirements` 結構每個都包含 `"VCpuCount": {"Min": 0, "Max": 2}`。這些屬性的值重疊，這會導致容量集區重複。

```
{
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
```

```
                            "Min": 0,
                            "Max": 2
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    },
                    {
                      "InstanceRequirements": {
                          "VCpuCount": {
                              "Min": 0,
                              "Max": 2
                          },
                          "MemoryMiB": {
                              "Min": 0
                          }
                      }
                  }
                }
              ]
            }
        ],
         "TargetCapacitySpecification": {
            "TotalTargetCapacity": 1,
            "DefaultTargetCapacityType": "spot"
        }
    }
}
```

有效且無效的競價型叢集組態範例

如果您使用 AWS CLI 建立 Spot 叢集，則必須確定您的叢集組態有效。下列範例顯示有效和無效的組態。

當組態包含下列項目時，視為無效：

- 同時具有 `Overrides` 和 `InstanceRequirements` 的單一 `InstanceType`
- 兩個 `Overrides` 結構, 一個具有 `InstanceRequirements`，而另一個具有 `InstanceType`
- 兩個在同一 `InstanceRequirements` 內具有重疊屬性值的 `LaunchTemplateSpecification` 結構

範例組態

- 有效組態：具有覆寫的單一啟動範本
- 有效組態：具有多個的單一啟動範本 InstanceRequirements
- 有效組態：兩個啟動範本，每個都具有覆寫
- 有效組態：僅限指定的 InstanceRequirements，沒有重疊的屬性值
- 組態無效：Overrides 包含 InstanceRequirements 和 InstanceType
- 組態無效：兩個 Overrides 包含 InstanceRequirements 和 InstanceType
- 組態無效：重疊屬性值

有效組態：具有覆寫的單一啟動範本

下列組態有效。其包含一個啟動範本和一個 `Overrides` 結構 (其中包含一個 `InstanceRequirements` 結構)。範例組態的文字說明如下。

```
{
    "SpotFleetRequestConfig": {
        "AllocationStrategy": "priceCapacityOptimized",
        "ExcessCapacityTerminationPolicy": "default",
        "IamFleetRole": "arn:aws:iam::000000000000:role/aws-ec2-spot-fleet-tagging-
role",
        "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateName": "My-launch-template",
                "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 2,
                            "Max": 8
                        },
                        "MemoryMib": {
                            "Min": 0,
                            "Max": 10240
                        },
                        "MemoryGiBPerVCpu": {
                            "Max": 10000
                        },
```

```
                        "RequireHibernateSupport": true
                    }
                }
            ]
        }
    ],
        "TargetCapacity": 5000,
            "OnDemandTargetCapacity": 0,
            "TargetCapacityUnitType": "vcpu"
    }
}
```

InstanceRequirements

若要使用屬性型執行個體選擇，您必須在組態中包含 `InstanceRequirements` 結構，並為機群中的執行個體指定所需的屬性。

在上述範例中，指定了下列執行個體屬性：

- `VCpuCount`— 執行個體類型必須至少為 2 且最多 8 個vCPUs。
- `MemoryMiB` – 執行個體類型必須具有最多 10240 MiB 的記憶體。最小值若為 0，表示沒有最小限制。
- `MemoryGiBPerVCpu`— 執行個體類型每 v CPU 最多必須有 10,000 GiB 的記憶體。`Min` 為選用參數。若省略它，表示沒有最小限制。

TargetCapacityUnitType

`TargetCapacityUnitType` 參數會指定目標容量的單位。在此範例中，目標容量為`5000`且目標容量單位類型為`vcpu`，它們共同指定所需的目標容量 5,000 vCPUs。Spot 叢集將啟動足夠的執行個體，因此叢集 vCPUs 中的總數為 5,000 個vCPUs。

有效組態：具有多個的單一啟動範本 InstanceRequirements

下列組態有效。其包含一個啟動範本和一個 `Overrides` 結構 (其中包含兩個 `InstanceRequirements` 結構)。中指定`VCpuCount`的屬性`InstanceRequirements`有效，因為值不會重疊 — 第一個`InstanceRequirements`結構指定 a 為 0-2vCPUs，而第二個`InstanceRequirements`結構則指定 4-8。vCPUs

```
{
    "SpotFleetRequestConfig": {
        "AllocationStrategy": "priceCapacityOptimized",
```

```
        "ExcessCapacityTerminationPolicy": "default",
        "IamFleetRole": "arn:aws:iam::000000000000:role/aws-ec2-spot-fleet-tagging-
role",
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 2
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                },
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 4,
                            "Max": 8
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                }
              ]
            }
        ],
        "TargetCapacity": 1,
        "OnDemandTargetCapacity": 0,
        "Type": "maintain"
    }
}
```

有效組態：兩個啟動範本，每個都具有覆寫

下列組態有效。其包含兩個啟動範本，每個都具有一個 Overrides 結構 (其中包含一個 InstanceRequirements 結構)。這個組態有助於同一個機群中的 arm 和 x86 架構支援。

```
{
    "SpotFleetRequestConfig": {
        "AllocationStrategy": "priceCapacityOptimized",
        "ExcessCapacityTerminationPolicy": "default",
        "IamFleetRole": "arn:aws:iam::000000000000:role/aws-ec2-spot-fleet-tagging-
role",
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "armLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 2
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                },
                {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "x86LaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 2
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
```

```
                    }
                }
              ]
            }
        ],
        "TargetCapacity": 1,
        "OnDemandTargetCapacity": 0,
        "Type": "maintain"
    }
}
```

有效組態：僅限指定的 **InstanceRequirements**，沒有重疊的屬性值

下列組態有效。其包含兩個 LaunchTemplateSpecification 結構，每個都有一個啟動範本和一個 Overrides 結構 (其中包含一個 InstanceRequirements 結構)。中指定VCpuCount的屬性InstanceRequirements有效，因為值不會重疊 — 第一個InstanceRequirements結構指定 a 為 0-2vCPUs，而第二個InstanceRequirements結構則指定 4-8。vCPUs

```
{
    "SpotFleetRequestConfig": {
        "AllocationStrategy": "priceCapacityOptimized",
        "ExcessCapacityTerminationPolicy": "default",
        "IamFleetRole": "arn:aws:iam::000000000000:role/aws-ec2-spot-fleet-tagging-
role",
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 2
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                }
              ]
```

```
            },
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyOtherLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 4,
                            "Max": 8
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                }
              ]
            }
        ],
        "TargetCapacity": 1,
        "OnDemandTargetCapacity": 0,
        "Type": "maintain"
    }
}
```

組態無效：**Overrides** 包含 **InstanceRequirements** 和 **InstanceType**

下列組態無效。Overrides 結構同時包含 InstanceRequirements 和 InstanceType。針對 Overrides，您可以指定 InstanceRequirements 或 InstanceType，但不能同時指定兩者。

```
{
    "SpotFleetRequestConfig": {
        "AllocationStrategy": "priceCapacityOptimized",
        "ExcessCapacityTerminationPolicy": "default",
        "IamFleetRole": "arn:aws:iam::000000000000:role/aws-ec2-spot-fleet-tagging-
role",
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyLaunchTemplate",
                    "Version": "1"
```

```
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 2
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                },
                {
                    "InstanceType": "m5.large"
                }
              ]
            }
        ],
        "TargetCapacity": 1,
        "OnDemandTargetCapacity": 0,
        "Type": "maintain"
    }
}
```

組態無效：兩個 **Overrides** 包含 **InstanceRequirements** 和 **InstanceType**

下列組態無效。Overrides 結構同時包含 InstanceRequirements 和 InstanceType。您可以指定 InstanceRequirements 或 InstanceType，但不能同時指定兩者，即使它們處於不同的 Overrides 結構也一樣。

```
{
    "SpotFleetRequestConfig": {
        "AllocationStrategy": "priceCapacityOptimized",
        "ExcessCapacityTerminationPolicy": "default",
        "IamFleetRole": "arn:aws:iam::000000000000:role/aws-ec2-spot-fleet-tagging-
role",
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyLaunchTemplate",
                    "Version": "1"
                },
```

```
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 2
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                }
              ]
            },
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyOtherLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceType": "m5.large"
                }
              ]
            }
        ],
        "TargetCapacity": 1,
        "OnDemandTargetCapacity": 0,
        "Type": "maintain"
    }
}
```

組態無效：重疊屬性值

下列組態無效。兩個 `InstanceRequirements` 結構每個都包含 `"VCpuCount": {"Min": 0, "Max": 2}`。這些屬性的值重疊，這會導致容量集區重複。

```
{
    "SpotFleetRequestConfig": {
        "AllocationStrategy": "priceCapacityOptimized",
        "ExcessCapacityTerminationPolicy": "default",
        "IamFleetRole": "arn:aws:iam::000000000000:role/aws-ec2-spot-fleet-tagging-
role",
```

```
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 2
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    },
                    {
                      "InstanceRequirements": {
                          "VCpuCount": {
                              "Min": 0,
                              "Max": 2
                          },
                          "MemoryMiB": {
                              "Min": 0
                          }
                      }
                  }
                }
              ]
            }
        ],
        "TargetCapacity": 1,
        "OnDemandTargetCapacity": 0,
        "Type": "maintain"
    }
}
```

預覽具有所指定屬性的執行個體類型

您可以使用 get-instance-types-from-instance-要求 AWS CLI 指令來預覽符合您指定屬性的執行個體類型。這特別有助於解決在不啟動任何執行個體的情況下要在請求組態中指定哪些屬性。請注意，該命令不會考慮可用容量。

若要透過指定屬性來預覽執行個體類型的清單，請使用 AWS CLI

1. (選擇性) 若要產生所有可指定的可能屬性，請使用 get-instance-types-from-instance-需求指令和參數。`--generate-cli-skeleton`您可以選擇性地將輸出導向至檔案來儲存它，方法是使用 `input > attributes.json`。

```
aws ec2 get-instance-types-from-instance-requirements \
    --region us-east-1 \
    --generate-cli-skeleton input > attributes.json
```

預期的輸出結果

```
{
    "DryRun": true,
    "ArchitectureTypes": [
        "i386"
    ],
    "VirtualizationTypes": [
        "hvm"
    ],
    "InstanceRequirements": {
        "VCpuCount": {
            "Min": 0,
            "Max": 0
        },
        "MemoryMiB": {
            "Min": 0,
            "Max": 0
        },
        "CpuManufacturers": [
            "intel"
        ],
        "MemoryGiBPerVCpu": {
            "Min": 0.0,
            "Max": 0.0
        },
        "ExcludedInstanceTypes": [
            ""
        ],
        "InstanceGenerations": [
            "current"
        ],
```

```
        "SpotMaxPricePercentageOverLowestPrice": 0,
        "OnDemandMaxPricePercentageOverLowestPrice": 0,
        "BareMetal": "included",
        "BurstablePerformance": "included",
        "RequireHibernateSupport": true,
        "NetworkInterfaceCount": {
            "Min": 0,
            "Max": 0
        },
        "LocalStorage": "included",
        "LocalStorageTypes": [
            "hdd"
        ],
        "TotalLocalStorageGB": {
            "Min": 0.0,
            "Max": 0.0
        },
        "BaselineEbsBandwidthMbps": {
            "Min": 0,
            "Max": 0
        },
        "AcceleratorTypes": [
            "gpu"
        ],
        "AcceleratorCount": {
            "Min": 0,
            "Max": 0
        },
        "AcceleratorManufacturers": [
            "nvidia"
        ],
        "AcceleratorNames": [
            "a100"
        ],
        "AcceleratorTotalMemoryMiB": {
            "Min": 0,
            "Max": 0
        },
        "NetworkBandwidthGbps": {
            "Min": 0.0,
            "Max": 0.0
        },
        "AllowedInstanceTypes": [
            ""
```

```
        ]
    },
    "MaxResults": 0,
    "NextToken": ""
}
```

2. 使用上一個步驟的輸出建立JSON組態檔案，並依照下列方式進行配置：

> **Note**
>
> 您必須提供 ArchitectureTypes、VirtualizationTypes、VCpuCount 和 MemoryMiB 的值。您可以省略其他屬性；省略時，就會使用預設值。
> 如需每個屬性及其預設值的說明，請參閱get-instance-types-from執行個體需求。

   a. 針對 ArchitectureTypes，指定一或多種類型的處理器架構。
   b. 針對 VirtualizationTypes，指定一或多種類型的虛擬化。
   c. 對於VCpuCount，指定的最小和最大數目vCPUs。若要指定沒有下限，請針對 Min，指定 0。若要指定沒有上限，請省略 Max 參數。
   d. 針對 MemoryMiB，指定記憶體的數量下限和上限 (以 MiB 為單位)。若要指定沒有下限，請針對 Min，指定 0。若要指定沒有上限，請省略 Max 參數。
   e. 您可以選擇性地指定一或多個其他屬性，以進一步限制傳回的執行個體類型清單。

3. 若要預覽具有您在JSON檔案中指定之屬性的例證類型，請使用 get-instance-types-from-instance-required 指令，並使用參數指定JSON檔案的名稱和路徑。--cli-input-json您可以選擇性地格式化要以表格格式出現的輸出。

```
aws ec2 get-instance-types-from-instance-requirements \
    --cli-input-json file://attributes.json \
    --output table
```

範例 *attributes.json* file

在此範例中，必要的屬性包含在JSON檔案中。它們是 ArchitectureTypes、VirtualizationTypes、VCpuCount 和 MemoryMiB。此外，選用 InstanceGenerations 屬性也包含在內。請注意，對於 MemoryMiB，可以省略 Max，以表示沒有限制。

```
{
```

```
    "ArchitectureTypes": [
        "x86_64"
    ],
    "VirtualizationTypes": [
        "hvm"
    ],
    "InstanceRequirements": {
        "VCpuCount": {
            "Min": 4,
            "Max": 6
        },
        "MemoryMiB": {
            "Min": 2048
        },
        "InstanceGenerations": [
            "current"
        ]
    }
}
```

範例輸出

```
-----------------------------------------
|GetInstanceTypesFromInstanceRequirements|
+----------------------------------------+
||             InstanceTypes            ||
|+--------------------------------------+|
||             InstanceType             ||
|+--------------------------------------+|
||  c4.xlarge                           ||
||  c5.xlarge                           ||
||  c5a.xlarge                          ||
||  c5ad.xlarge                         ||
||  c5d.xlarge                          ||
||  c5n.xlarge                          ||
||  d2.xlarge                           ||
...
```

4. 在識別符合您需求的執行個體類型之後，請記下您使用的執行個體屬性，以便在設定機群請求時可以使用它們。

使用執行個體加權來管理EC2叢集或 Spot 叢集的成本和效能

透過執行個體權重，您可以為EC2叢集或 Spot 叢集中的每個執行個體類型指派加權，以表示其相對於彼此的運算容量和效能。根據權重，叢集可以使用指定執行個體類型的任意組合，只要它能夠滿足所需的目標容量即可。這可以幫助您管理車隊的成本和性能。

重量代表執行個體類型貢獻總目標容量的容量單位。

範例：使用執行個體加權進行績效管理

假設您的叢集有兩種執行個體類型，而且您為每個執行個體類型指派不同的權重，以反映每種執行個體類型需要多少才能達到相同效能，如下所示：

- `m5.large`— 重量：1
- `m5.2xlarge`-重量：4

通過分配這些權重，你說你需要 4 `m5.large` 個實例來實現與 1 相同的性能`m5.2xlarge`。

若要計算指定目標容量需要每個執行個體類型的執行個體數目，請使用下列公式：

```
target capacity / weight = number of instances
```

如果您的目標容量為 8 個單位，則艦隊可以使用`m5.large`或或`m5.2xlarge`兩者混合來滿足目標容量，如下所示：

- 8 `m5.large` 個執行個體 (8 個容量/重量為 1 = 8 個執行個體)
- 2 `m5.2xlarge` 個執行個體 (8 個容量/重量為 4 = 2 個執行個體)
- 四`m5.large`和一 `m5.2xlarge`

範例：使用執行環境加權進行成本管理

根據預設，您指定的價格是每執行個體小時。使用執行個體權重功能時，您指定的價格是每單位小時。您可以透過將執行個體類型價格除以它所代表的單位數，來計算每單位小時的價格。叢集會計算要啟動的執行個體數目，方法是將目標容量除以執行個體權重。如果結果不是整數，則機群會將其捨入到下一個整數，以便您的機群大小不低於其目標容量。即使所啟動的執行個體容量超過所請求的目標容量，該機群也可以選取您在啟動規格中指定的任何集區。

下表包含計算範例，以判斷目標容量為 10 之車隊的每單位價格。

執行個體類型	執行個體權重	目標容量	已啟動的執行個體數目	每個執行個體小時的價格	每個單位小時的價格
`r3.xlarge`	2	10	5 (10 除以 2)	0.05 USD	$0.025 (.05 除以 2)
`r3.8xlarge`	8	10	2 (10 除以 8，將結果四捨五入)	0.10 USD	$0.0125 (.10 除以 8)

請依照下列方式使用叢集執行個體加權，以履行時每單位最低價格在集區中佈建您想要的目標容量：

1. 以執行個體 (預設值) 或您選擇的單位 (例如 v CPU、記憶體、儲存體或輸送量) 設定叢集的目標容量。
2. 設定每單位價格
3. 為每個啟動規格指定權重，即執行個體類型代表目標容量的單位數。

執行個體權重範例

請考慮使用下列組態的叢集要求：

- 目標容量為 24
- 執行個體類型為 `r3.2xlarge` 且權重為 6 的啟動規格
- 執行個體類型為 `c3.xlarge` 且權重為 5 的啟動規格

權重表示執行個體類型代表目標容量的單位數。如果第一次啟動規格提供每單位最低價格 (`r3.2xlarge`每執行個體小時的價格除以 6)，叢集就會啟動其中四個執行個體 (24 除以 6)。

如果第二次啟動規格提供每單位最低價格 (`c3.xlarge`每執行個體小時的價格除以 5)，叢集就會啟動其中五個執行個體 (24 除以 5，結果四捨五入)。

執行個體權重與分配策略

請考慮使用下列組態的叢集要求：

- 目標容量為 30 個 競價型執行個體
- 執行個體類型為 `c3.2xlarge` 且權重為 8 的啟動規格
- 執行個體類型為 `m3.xlarge` 且權重為 8 的啟動規格
- 執行個體類型為 `r3.xlarge` 且權重為 8 的啟動規格

艦隊將啟動四個實例（30 除以 8，結果四捨五入）。採用 `diversified` 策略，機群會在三個集區的每一個集區啟動一個執行個體，並且三個集區任一個中的第四個執行個體會提供最低的每單位價格。

使用配置策略來確定EC2叢集或 Spot 叢集如何滿足 Spot 和隨需容量

當您在EC2叢集或 Spot 叢集中使用多個容量集區 (每個都包含執行個體類型和可用區域) 時，您可以使用配置策略來管理 Amazon 如何從這些集區EC2履行 Spot 和隨需容量。配置策略可針對可用容量、價格和要使用的執行個體類型進行最佳化。Spot 執行個體和隨需執行個體有不同的配置策略。

主題

- 競價型執行個體 的分配策略
- 隨需執行個體的配置策略
- 選擇適當的 Spot 配置策略
- 維持 Spot 執行個體的目標容量
- 針對隨需容量排定執行個體類型的優先順序

競價型執行個體 的分配策略

您的啟動組態會決定EC2叢集或 Spot 叢集可從中啟動 Spot 執行個體的所有可能 Spot 容量集區 (執行個體類型和可用區域)。不過，在啟動執行個體時，叢集會使用您指定的配置策略，從所有可能的集區中挑選特定的集區。

> **Note**
>
> (僅限 Linux 執行個體) 如果將競價型執行個體設定為啟用 AMDSEV-SNP，則需支付額外的小時使用費，相當於所選執行個體類型隨需小時費率的 10%。如果配置策略使用價格作為輸入，則車隊不會包含此額外費用；僅使用 Spot 價格。

您可以為 Spot 執行個體指定下列其中一個配置策略：

價格容量最佳化 (建議使用)

叢集會針對正在啟動的執行個體數量，識別具有最高容量可用性的集區。這表示我們將從我們認為在短期內中斷機會最低的集區請求 Spot 執行個體。然後，叢集會從這些集區的最低價格請求 Spot 執行個體。

價格容量最佳化配置策略是大多數 Spot 工作負載的最佳選擇，例如無狀態容器化應用程式、微型服務、Web 應用程式、資料和分析任務以及批次處理。

如果您使用的是 AWS CLI，則參數名稱適用`price-capacity-optimized`於EC2叢集和`priceCapacityOptimized` Spot 叢集。

容量最佳化

叢集會針對正在啟動的執行個體數量，識別具有最高容量可用性的集區。這表示我們將從我們認為在短期內中斷機會最低的集區請求 Spot 執行個體。您可以選擇性地為叢集中的每個執行個體類型設定優先順序，其中叢集會先針對容量進行最佳化，但會盡最大努力遵守執行個體類型優先順序。

使用 競價型執行個體，定價會依據長期的供需趨勢隨時間慢慢改變，但是容量會即時波動。容量最佳化策略會透過查看即時容量資料並預測哪些可用資料，自動將 Spot 執行個體啟動到最多可用的集區。這適用於可能具有較高中斷成本與重新啟動工作相關的工作負載，例如持續整合 (CI)、影像和媒體轉譯、深度學習和高效能運算 (HPC) 工作負載，這些工作負載可能與重新啟動工作相關的中斷成本較高。容量最佳化策略可減少中斷的可能性，可降低工作負載的整體成本。

或者，您也可以使用容量最佳化的優先順序配置策略搭配優先順序參數，將執行個體類型從最高到最低優先順序排序。您可以為不同的執行個體類型設定相同的優先順序。叢集會先針對容量進行最佳化，但會盡最大努力遵循執行個體類型優先順序 (例如，如果遵守優先順序不會對叢集佈建最佳容量的能力造成重大影響)。對於必須將中斷可能性降至最低的工作負載來說，這是一個很好的選擇，而且某些執行個體類型的偏好也很重要。請注意，當您為 Spot 容量設定執行個體類型的優先順序時，如果隨需配置策略設定為優先順序，您的隨需執行個體也會套用相同的優先順序。對於競價型叢集，只有當您的叢集使用啟動範本時，才支援使用優先順序。

如果您使用的是 AWS CLI，參數名稱`capacity-optimized-prioritized`為 `capacity-optimized`「EC2叢集」`capacityOptimized` 和 `capacityOptimizedPrioritized`「競價型叢集」。

多元化

競價型執行個體 會分佈在所有 Spot 容量集區。如果您使用的是 AWS CLI，則參數名稱適用`diversified`於「EC2叢集」和「競價型叢集」。

最低價格（不建議使用）

> **Warning**
>
> 我們不建議使用最低價格配置策略，因為它對您的競價型執行個體具有最高的中斷風險。

Spot 執行個體來自價格最低且還有可用容量的集區。使用時 AWS CLI，這是預設策略。不過，我們建議您透過指定價格容量最佳化配置策略來覆寫預設值。

使用最低價格策略時，如果價格最低的集區沒有可用容量，Spot 執行個體將來自下一個具有可用容量的最低價集區。如果集區在滿足您想要的容量之前耗盡容量，叢集將繼續從下一個最低價的集區繪製來滿足您的要求。為確保達到所需的容量，您可能會從數個集區中接收 Spot 執行個體。

由於此策略僅考慮執行個體價格，而不考慮容量可用性，因此可能會導致高中斷率。

最低價格配置策略僅在使用時可用 AWS CLI。參數名稱`lowest-price`適用於「EC2叢集」和`lowestPrice`「競價型叢集」。

要使用的池數

要配置目標 Spot 容量的 Spot 集區數量。只有在配置策略設定為最低價格時才有效。叢集會選取價格最低的競價型集區，並將目標 Spot 容量平均分配到您指定的 Spot 集區數目。

請注意，叢集會嘗試從您以最佳方式指定的集區數量繪製 Spot 執行個體。如果集區在滿足您的目標容量之前用完 Spot 容量，叢集將繼續從下一個最低價的集區繪製來滿足您的請求。為確保達到目標容量，您可能會從超過您指定的集區數量中接收 Spot 執行個體。同樣地，如果大多數集區沒有 Spot 容量，您可能會從少於您指定的集區數量中接收完整目標容量。

只有在指定最低價格配置策略時才能使用此參數，且僅在使用時才能使用 AWS CLI。參數名稱`InstancePoolsToUseCount`適用於「EC2艦隊」和「競價型叢集」。

隨需執行個體的配置策略

您的啟動設定會決定EC2叢集或 Spot 叢集可從中啟動隨需執行個體的所有可能容量集區 (執行個體類型和可用區域)。不過，在啟動執行個體時，叢集會使用您指定的配置策略，從所有可能的集區中挑選特定的集區。

您可以為隨需執行個體指定下列其中一種配置策略：

最低價

隨需執行個體來自價格最低且具有可用容量的集區。這是預設策略。

如果價格最低的集區沒有可用容量，則隨需執行個體來自具有可用容量的下一個價格最低的集區。

如果集區在滿足您想要的容量之前耗盡容量，叢集將繼續從下一個最低價的集區繪製來滿足您的要求。為確保符合您所需的容量，您可能會收到來自多個集區的隨需執行個體。

優先

叢集會使用您指派給每個啟動範本覆寫的優先順序，以最高優先順序啟動執行個體類型。此策略無法用於以屬性為基礎的執行個體類型選取。如需如何使用此配置策略的範例，請參閱針對隨需容量排定執行個體類型的優先順序。

選擇適當的 Spot 配置策略

您可以選擇適當的 Spot 分配策略，進而針對您的使用案例最佳化機群。

平衡最低價格與容量可用性

為了平衡最低價 Spot 容量集區與具有最高容量可用性的競價型容量集區之間的權衡，我們建議您使用價格容量最佳化配置策略。此策略會根據集區的價格和這些集區中 Spot 執行個體的容量可用性，決定要從哪些集區請求 Spot 執行個體的集區。這表示我們將從我們認為在短期內中斷機會最低的集區請求 Spot 執行個體，同時我們仍會將價格納入考慮。

如果您的叢集執行彈性且無狀態的工作負載，包括容器化應用程式、微型服務、Web 應用程式、資料和分析工作以及批次處理，請使用價格容量最佳化配置策略，以達到最佳的成本節省和容量可用性。

如果您的機群執行的工作負載有較高的重新啟動工作中斷的成本，則您應該實作檢查點作業，以便應用程式可以在中斷時從該點重新啟動。通過使用檢查點，您可以使價格容量優化配置策略非常適合這些工作負載，因為它可以從價格最低的集區中分配容量，這些集區也提供較低的競價型執行個體中斷率。

如需使用價格容量最佳化配置策略的JSON組態範例，請參閱下列內容：

- EC2艦隊 — 範例 10：在 price-capacity-optimized 叢集中啟動 Spot 執行個體
- 現貨艦隊 — 範例 11：在 priceCapacityOptimized 叢集中啟動 Spot 執行個體

當工作負載具有較高的中斷成本時

如果您執行的工作負載使用價格相似的執行個體類型，或是中斷成本非常重要，以致於與中斷的邊際增加相比，節省成本不足，您可以選擇性地使用容量最佳化策略。該策略可從最可用的 Spot 容量集區分配容量，而這些集區會提供較少中斷的可能性，進而降低您工作負載的整體成本。

如果必須將中斷的可能性降至最低，但對特定執行個體類型的偏好設定很重要時，您可以使用容量最佳化的優先順序配置策略來表示集區優先順序，然後將執行個體類型的使用順序從最高到最低優先順序來表示集區優先順序。

請注意，當您設定容量最佳化優先順序時，如果將隨需配置策略設定為優先順序，也會將相同的優先順序套用至您的隨需執行個體。另請注意，對於競價型叢集，只有當您的叢集使用啟動範本時，才支援使用優先順序。

如需使用容量最佳化配置策略的範例JSON組態，請參閱下列內容：

- EC2艦隊 — 範例 8：在容量最佳化叢集中啟動 Spot 執行個體
- 現貨艦隊 — 範例 9：在容量最佳化機群中啟動 Spot 執行個體

如需使用容量最佳化優先順序配置策略的JSON組態範例，請參閱下列內容：

- EC2艦隊 — 範例 9：在具有優先順序的容量最佳化叢集中啟動 Spot 執行個體
- 現貨艦隊 — 範例 10：使用優先順序在容量最佳化機群中啟動 Spot 執行個體

當您的工作負載具有時間彈性且容量可用性並非相關因素時

如果您的機群很小或運作時間很短，您可以使用最佳化的價格容量，以最大限度地節省成本，同時仍考慮容量可用性。

當您的機群很龐大或執行時間較長時

如果您的叢集規模龐大或長時間執行，您可以使用多元化策略將 Spot 執行個體分配到多個集區，以提高叢集的可用性。例如，如果您的叢集指定 10 個集區，而目標容量為 100 個執行個體，則叢集會在每個集區中啟動 10 個 Spot 執行個體。如果某個集區的 Spot 價格超過該集區的最高價格，則只有 10% 的機群會受到影響。隨著時間，使用這種策略也會使您的機群對任何一個集區 Spot 價格上漲較不敏感。透過多元化策略，叢集不會將 Spot 執行個體啟動到任何競價型價格等於或高於隨需價格的集區中。

維持 Spot 執行個體的目標容量

在 Spot 執行個體因 Spot 價格變更或 Spot 容量集區的可用容量而終止後，類型的叢集會maintain啟動替代 Spot 執行個體。分配策略會確定從中啟動替代執行個體的集區，如下所示：

- 如果配置策略經過最佳化的價格容量，則叢集會在具有最多 Spot 執行個體容量可用性的集區中啟動替換執行個體，同時考慮價格並識別具有高容量可用性的最低價格集區。
- 如果配置策略已最佳化容量，叢集會在具有最多 Spot 執行個體容量可用性的集區中啟動替換執行個體。
- 如果配置策略是多元化的，則叢集會將替換的 Spot 執行個體分配到剩餘的集區中。

針對隨需容量排定執行個體類型的優先順序

當EC2叢集或 Spot 叢集嘗試滿足您的隨需容量時，預設會先啟動價格最低的執行個體類型。如果將隨需配置策略設定為優先順序，叢集會使用優先順序來決定滿足隨需容量時要先使用的執行個體類型。優先順序會指派至啟動範本覆寫，優先順序最高的將會先啟動。

範例：排定執行個體類型的優先順序

在本例中，您設定三個啟動範本覆寫，每個範本均具有不同的執行個體類型。

執行個體類型的隨需價格在價格範圍內。以下是本範例中使用的執行個體類型，依價格順序列出，從最便宜的執行個體類型開始：

- `m4.large` – 最便宜
- `m5.large`
- `m5a.large`

若您不使用優先順序來決定順序，則機群會以最便宜的執行個體類型開始來實現隨需容量。

但是，假設您有要首先使用且未使用的 `m5.large` 預留執行個體。您可設定啟動範本覆寫，以便依優先順序使用執行個體類型，如下所示：

- `m5.large` – 優先順序 1
- `m4.large` – 優先順序 2
- `m5a.large` – 優先順序 3

使用叢集和 S EC2 pot 叢集中的容量重新平衡來取代有風險的 Spot 執行個體

透過容量重新平衡，您的EC2叢集或 Spot 叢集可以透過主動更換存在中斷風險的 Spot 執行個體來維持所需的 Spot 容量。當 Spot 執行個體面臨中斷風險較高時，Amazon EC2 會傳送重新平衡建議。如果啟用了容量重新平衡，則重新平衡建議會在有風險的執行個體中斷之前觸發新的 Spot 執行個體的啟動。

容量重新平衡可協助您維持工作負載的可用性，方法是在 Amazon 中斷執行個體之前，使用新的 Spot 執行個體主動擴充叢集。EC2

將EC2叢集設定為使用容量重新平衡來啟動替代 Spot 執行個體

使用創建艦隊（AWS CLI）命令和結構中的相關參數。`MaintenanceStrategies`如需組JSON態範例，請參閱範例 7：設定容量重新平衡以啟動替代 Spot 執行個體。

將 Spot 叢集設定為使用容量重新平衡來啟動替代 Spot 執行個體

您可以使用 Amazon EC2 主控台或設定 AWS CLI 容量重新平衡。

(主控台) 建立 Spot 叢集時，請選取 [容量重新平衡] 核取方塊。如需詳細資訊，請參閱 使用已定義的參數建立 Spot Fleet 請求 (主控台) 中的步驟 6.d.。

（AWS CLI）使用命request-spot-fleet令和結`SpotMaintenanceStrategies`構中的相關參數。如需組JSON態範例，請參閱範例 8：設定容量重新平衡以啟動替代 競價型執行個體。

主題

- 限制
- 組態選項
- 考量事項

限制

- 容量重新平衡僅適用於類型為 `maintain` 的機群。
- 機群執行時，您無法修改容量重新平衡設定。若要變更容量重新平衡設定，您必須刪除機群並建立新的機群。

組態選項

F `ReplacementStrategy` or EC2 叢集和競價型叢集支援下列兩個值：

`launch-before-terminate`

Amazon EC2 終止在新的替代 Spot 執行個體啟動後收到重新平衡通知的競價型執行個體。當指定 `launch-before-terminate` 時，您亦須指定 `termination-delay` 的值。新的替代執行個體啟動後，Amazon 會EC2等待的持續時間`termination-delay`，然後終止舊執行個體。對於 `termination-delay`，下限為 120 秒 (2 分鐘)，上限則為 7200 秒 (2 小時)。

建議僅在您可以預測執行個體關閉程序將需要多長時間才能完成時，才使用 `launch-before-terminate`。這將確保只在關機程序完成之後，才會終止舊執行個體。請注意，Amazon EC2 可以在發出前兩分鐘警告中斷舊執行個體`termination-delay`。

我們強烈建議您不`launch-before-terminate`要將 `lowest-price` (EC2叢集) 或 `lowestPrice` (Spot Fleet) 配置策略與結合使用，以避免更換 Spot 執行個體也會增加中斷風險。

`launch`

當現有競價型執行個體發出重新平衡通知時，Amazon 會EC2啟動替換 Spot 執行個體。Amazon EC2 不會終止收到重新平衡通知的執行個體。您可以終止舊的執行個體，也可以讓它們保持執行中狀態。所有執行個體在執行時，您需要支付所有執行個體的費用。

考量事項

如果您設定EC2叢集或 Spot 叢集以進行容量重新平衡，請考慮下列事項：

盡可能在請求中提供更多 Spot 容量集區

將您的叢集設定為使用多個執行個體類型和可用區域。這提供了在各種 Spot 容量集區中啟動 Spot 執行個體的彈性。如需詳細資訊，請參閱 對執行個體類型和可用區域具有彈性。

避免中斷替代 Spot 執行個體的風險提高

為了避免中斷的風險增加，我們建議使用`capacity-optimized`或`capacity-optimized-prioritized`配置策略。這些策略可確保在最佳 Spot 容量集區中啟動替代 Spot 執行個體，因此不太可能在不久的將來中斷。如需詳細資訊，請參閱使用價格和容量最佳化分配策略。

如果您使用`lowest-price`配置策略，替換的 Spot 執行個體可能會增加中斷的風險。這是因為 Amazon 始終EC2會在當時具有可用容量的最低價集區中啟動執行個體，即使您的替換 Spot 執行個體在啟動後很快就會中斷。

Amazon 只EC2會在可用性相同或更好的情況下啟動新執行個體

容量重新平衡的其中一個目標是改善 Spot 執行個體的可用性。如果現有競價型執行個體收到重新平衡建議，Amazon 只EC2會在新執行個體提供與現有執行個體相同或更好的可用性時啟動新執行

個體。如果新執行個體的中斷風險比現有執行個體更糟，Amazon 將不EC2會啟動新執行個體。但是，Amazon EC2 將繼續評估 Spot 容量集區，並在可用性改善時啟動新執行個體。

如果 Amazon EC2 主動啟動新執行個體，您的現有執行個體有可能會中斷。發生這種情況時，Amazon EC2 將嘗試啟動新執行個體，無論新執行個體是否有高中斷風險。

容量重新平衡不會增加 Spot 執行個體中斷率

啟用容量重新平衡時，不會增加 Spot 執行個體中斷率 (Amazon EC2 需要返回容量時回收的 Spot 執行個體數量)。但是，如果容量重新平衡偵測到執行個體有中斷的風險，Amazon EC2 將立即嘗試啟動新的執行個體。結果是，與在風險執行個體中斷後等待 Amazon 啟動新執行個體EC2的情況下，可能會取代更多的執行個體。

雖然您可以在啟用容量重新平衡的情況下取代更多執行個體，但您可以從主動式而非被動式中受益，因為在執行個體中斷之前您有更多時間採取動作。透過 Spot 執行個體中斷通知，您通常最多只有兩分鐘時間來正常關閉您的執行個體。透過容量重新平衡預先啟動新的執行個體，可讓現有程序更有可能在有風險的執行個體上完成，您可以啟動執行個體關閉程序，並防止在有風險的執行個體上排程新的工作。您也可以開始準備新啟動的執行個體以接管應用程式。透過容量重新平衡的主動式取代，您可以從正常連續性中獲益。

作為示範使用容量重新平衡的風險和益處的理論範例，請考慮下列案例：

- 下午 2:00 — 收到執行個體 A 的重新平衡建議，Amazon 會EC2立即開始嘗試啟動替換執行個體 B，讓您有時間開始關機程序。 *
- 下午 2:30 – 收到執行個體 B 的重新平衡建議，會取代為執行個體 C，讓您有時間開始關機程序。*
- 下午 2:32 – 如果未啟用容量重新平衡，並且如果在下午 2:32 收到執行個體 A 的 Spot 執行個體中斷通知，則您最多只有兩分鐘時間採取動作，但是執行個體 A 會一直執行到目前。

* 如果指`launch-before-terminate`定，Amazon EC2 將在替換執行個體上線後終止有風險的執行個體。

Amazon EC2 可以啟動新的替代 Spot 執行個體，直到完成的容量達到目標容量

將叢集設定為容量重新平衡時，叢集會嘗試為每個接收重新平衡建議的 Spot 執行個體啟動新的替代 Spot 執行個體。Spot 執行個體收到重新平衡建議後，該執行個體不再計入以滿足容量的一部分。根據替換策略，Amazon EC2 會在預先設定的終止延遲後終止執行個體，或讓執行個體保持執行。這讓您有機會在執行個體上執行重新平衡動作。

如果您的機群達到目標容量的兩倍，即使替代執行個體本身收到重新平衡建議，它仍會停止啟動新的替代執行個體。

例如，您建立的叢集目標容量為 100 個 Spot 執行個體。所有 Spot 執行個體都會收到重新平衡建議，這會導致 Amazon 啟動 100 EC2 個替代 Spot 執行個體。這會將已滿足 Spot 執行個體的數目提高至 200 個，這是目標容量的兩倍。部分取代執行個體會收到重新平衡建議，但不會再啟動取代執行個體，因為叢集不能超過其目標容量的兩倍。

請注意，所有執行個體在執行時都需支付這些執行個體的費用。

建議您將叢集設定為終止接收重新平衡建議的 Spot 執行個體

如果您設定叢集進行容量重新平衡，建議您只`launch-before-terminate`有在能夠預測執行個體關閉程序需要多長時間才能完成時，才選擇適當的終止延遲。這將確保只在關機程序完成之後，才會終止舊執行個體。

如果您選擇自行終止建議用於重新平衡的執行個體，建議您監控機群中 Spot 執行個體收到的重新平衡建議訊號。透過監控訊號，您可以在 Amazon EC2 中斷受影響的執行個體之前，快速對受影響的執行個體執行重新平衡動作，然後手動終止它們。如果您未終止執行個體，則會在執行時繼續為這些執行個體付費。Amazon EC2 不會自動終止接收重新平衡建議的執行個體。

您可以使用 Amazon EventBridge 或執行個體中繼資料設定通知。如需詳細資訊，請參閱監控重新平衡建議訊號。

在擴展或擴展期間計算已滿足容量時，叢集不會計算接收重新平衡建議的執行個體

如果您的叢集設定為「容量重新平衡」，且您將目標容量變更為擴充或向外擴充，則叢集不會將標示為重新平衡作為已履行容量一部分的執行個體計算，如下所示：

- 擴充 — 如果您減少所需的目標容量，Amazon EC2 會終止未標示為重新平衡的執行個體，直到達到所需的容量為止。標示要重新平衡的執行個體不會計入已滿足的容量。

  例如，您建立一個目標容量為 100 個 Spot 執行個體的叢集。10 個執行個體會收到重新平衡建議，因此 Amazon EC2 啟動 10 個新的替換執行個體，產生 110 個執行個體的可滿足容量。然後您將目標容量減少到 50 (擴充)，但已滿足的容量實際上是 60 個執行個體，因為 Amazon EC2 不會終止標記為重新平衡的 10 個執行個體。您需要手動終止這些執行個體，或者您也可以讓它們保持運行。
- 向外擴充 — 如果您增加所需的目標容量，Amazon 會EC2啟動新執行個體，直到達到所需的容量為止。標示要重新平衡的執行個體不會計入已滿足的容量。

  例如，您建立的叢集目標容量為 100 個 Spot 執行個體。10 個執行個體會收到重新平衡建議，因此叢集會啟動 10 個新的替代執行個體，產生 110 個執行個體的滿足容量。然後，您將目標容量增加至 200 (擴展)，但已滿足容量實際上是 210 個執行個體，因為標示要重新平衡的 10 個執行個體不會被機群計為目標容量的一部分。您需要手動終止這些執行個體，或者您也可以讓它們保持運行。

使用容量保留在EC2叢集中保留隨需容量

隨需容量預留可讓您在特定的可用區域中，為隨需執行個體預訂任何期間的運算容量。您可以將EC2叢集設定為在啟動隨需執行個體時先使用容量保留。

隨需容量保留僅適用於請求類型設定為的EC2叢集`instant`。

容量預留設定為 `open` 或 `targeted`。EC2叢集可以將隨需執行個體啟動到`open`或`targeted`容量保留中，如下所示：

- 如果容量預留為 `open`，則具有相符屬性的隨需執行個體會以預留容量自動執行。
- 如果容量預留為 `targeted`，則隨需執行個體必須特別針對它使用預留容量執行。這對於用完特定容量預留或控制何時該使用特定容量預留的情況下十分有用。

如果您在EC2叢集中使用`targeted`容量保留，則必須有足夠的容量保留來滿足目標隨需容量，否則啟動會失敗。為了避免啟動失敗，最好將 `targeted` 容量預留新增至資源群組，然後針對資源群組為目標來執行。資源群組不需要有足夠的容量預留；如果在滿足目標隨需容量之前用完容量預留，則機群可以將剩餘的目標容量啟動為一般隨需容量。

若要搭配EC2叢集使用容量保留

1. 將機群配置為 `instant` 類型。您無法針對其他類型的機群使用容量預留。
2. 將容量預留的使用策略配置為 `use-capacity-reservations-first`。
3. 在啟動範本的 Capacity reservation (容量保留) 中，選擇 Open (開放) 或 Target by group (依群組分類的目標)。如果您選擇 Target by group (依群組分類的目標)，請指定容量預留資源群組 ID。

當機群嘗試滿足隨需容量時，如果發現多個執行個體集區具有相符的未使用容量預留，則會根據隨需分配策略 (`lowest-price` 或 `prioritized`) 決定啟動隨需執行個體的集區。

相關資源

- 如需CLI如何設定叢集以使用容量保留來滿足隨需容量的範例EC2叢集的範例CLI組態，請參閱範例 5 到 7。
- 如需教學課程，可引導您完成建立容量保留、在叢集中使用它們，以及檢視剩餘容量保留數目的步驟，請參閱 教學課程：設定EC2叢集以使用目標容量保留啟動隨需執行個
- 如需設定容量保留的相關資訊，請參閱透過隨需容量保留來保留運算容量和隨選容量保留FAQs。

與EC2艦隊合作

若要開始使用EC2叢集，請建立要求，其中包括總目標容量、隨需容量、Spot 容量，以及指定叢集中執行個體組態設定的啟動範本。您可以選擇性地指定其他參數，或讓叢集使用預設值。您也可以在建立叢集時標記叢集要求及其執行個體和磁碟區。

叢集會在有可用容量時啟動隨需執行個體，並在您的最高價超過 Spot 價格和可用容量時啟動 Spot 執行個體。

叢集啟動後，您可以描述叢集要求、叢集中的執行個體以及任何叢集事件。您也可以視需要指派其他標籤。

如果您需要變更任何叢集參數 (例如目標容量總計)，您可以修改叢集 (前提是已設定為維持容量)。一次性請求提交後，您無法修改該要求的容量。

叢集請求會保持作用中狀態，直到到期或您將其刪除為止。刪除叢集要求時，您可以終止執行個體或讓執行個體保持執行中。如果您選擇讓它們繼續執行，隨需執行個體會一直執行，直到您終止它們為止，Spot 執行個體會一直執行，直到它們中斷或您終止它們為止。

主題

- EC2艦隊請求狀態
- 建立EC2艦隊
- 標記新的或現有的EC2叢集請求，以及其啟動的執行個體和磁碟區
- 說明EC2叢集的組態、執行個體和事件歷史記錄
- 修改EC2叢集
- 刪除EC2叢集請求和叢集中的執行個體

EC2艦隊請求狀態

EC2叢集要求可以是各種狀態之一，每個狀態都指示要求生命週期的不同階段及其對執行個體的管理。

EC2叢集請求可以處於下列其中一種狀態：

`submitted`

系統正在評估EC2叢集請求，Amazon EC2 正準備啟動目標執行個體數量。超出機群限制的請求會立即刪除。

`active`

EC2叢集請求已通過驗證，Amazon 正在嘗試維護EC2正在執行的執行個體的目標數量。此請求會一直維持在該狀態，直到修改或刪除。

`modifying`

EC2叢集要求正在修改中。此請求會一直維持在該狀態，直到已完成修改或已刪除請求。只有 `maintain` 機群類型可以修改。此狀態不適用於其他請求類型。

`deleted_running`

EC2叢集請求已刪除，且不會啟動額外的 Spot 執行個體。其現有執行個體將繼續執行直到中斷或手動終止。此請求會一直維持在該狀態，直到所有執行個體中斷或終止。刪除EC2叢集請求後，只有叢集類型`maintain`或`request`可以擁有執行中EC2的執行個體。不支援具有執行中執行個體的已刪除 `instant` 機群。此狀態不適用於 `instant` 機群。

`deleted_terminating`

EC2叢集要求會遭到刪除，且其執行個體正在終止。此請求會一直維持在該狀態，直到所有執行個體終止。

`deleted`

EC2叢集已刪除，且沒有執行中的執行個體。請求會在其執行個體終止之後兩天刪除。

建立EC2艦隊

若要建立EC2叢集，請在JSON檔案中定義叢集配置，並使用 create- AWS CLI fleet 指令參考檔案。在JSON檔案中，您必須指定叢集的目標容量總計、Spot 執行個體和隨需執行個體的個別目標容量，以及定義叢集中執行個體組態的啟動範本，例如執行個體類型AMI、子網路或可用區域，以及一或多個安全群組。您可以選擇性地指定其他組態，例如用於覆寫啟動範本設定的參數、從EC2容量集區選取 Spot 執行個體和隨需執行個體的配置策略，以及願意為叢集支付的最高金額。如需詳細資訊，請參閱您的EC2車隊或 Spot 叢集的組態選項。

EC2叢集會在容量可用時啟動隨需執行個體，並在您的最高價超過 Spot 價格和可用容量時啟動 Spot 執行個體。

如果您的叢集包含 Spot 執行個體且屬於類型`maintain`，Amazon EC2 會在 Spot 執行個體中斷時嘗試維持叢集目標容量。

EC2機隊限制

下列限制適用於「EC2艦隊」：

- 只有透過 Amazon EC2 API、、AWS CLI和才能建立EC2叢集AWS CloudFormation。AWS SDKs
- EC2叢集請求無法跨越 AWS 區域。您需要為每個區域建立單獨的EC2艦隊。
- EC2叢集要求無法跨越來自相同可用區域的不同子網路。

EC2機隊先決條

若要建立EC2叢集，必須符合下列先決條件：

- 啟動範本
- 叢集的EC2服務連結角色
- 授予客戶受管金鑰的存取權，以便搭配加密AMIs和EBS快照使用
- EC2叢集使用者的權限

啟動範本

啟動範本會指定要啟動之執行個體的組態資訊，例如執行個體類型和可用區域。如需啟動範本的詳細資訊，請參閱在 Amazon 啟動範本中存放執行個體EC2啟動參數。

叢集的EC2服務連結角色

此`AWSServiceRoleForEC2Fleet`角色授與EC2叢集權限，以代表您要求、啟動、終止和標記執行個體。Amazon EC2 使用此服務連結角色來完成下列動作：

- `ec2:RunInstances` – 啟動執行個體。
- `ec2:RequestSpotInstances` – 請求 競價型執行個體。
- `ec2:TerminateInstances` – 終止執行個體。
- `ec2:DescribeImages`— 描述 Amazon 機器映像（AMIs）的實例。
- `ec2:DescribeInstanceStatus`— 描述執行個體的狀態。
- `ec2:DescribeSubnets`— 描述執行個體的子網路。
- `ec2:CreateTags`— 將標籤新增至EC2叢集、執行個體和磁碟區。

在使用 AWS CLI 或建立EC2叢集之前，請確定此角色已存在。API

> **Note**
>
> `instant`EC2艦隊不需要此角色。

若要建立角色，請依照下列方式使用IAM主控台。

若要建立EC2艦隊 AWSServiceRoleForEC2Fleet 角色

1. 在開啟IAM主控台https://console.aws.amazon.com/iam/。
2. 在導覽窗格中，選擇 Roles (角色)。
3. 選擇 Create Role (建立角色)。
4. 在 Select trusted entity (選取信任的實體) 頁面上，執行以下作業：
   a. 針對信任的實體類型，請選擇 AWS 服務。
   b. 在 [使用案例] 下，針對 [服務] 或 [使用案例]，選擇 EC2 [叢集]

      > **Tip**
      >
      > 一定要選擇 EC2-艦隊。如果您選擇 EC2，[使用案例] 清單中不會顯示 [EC2-叢集] 使用案例。EC2-叢集使用案例會自動建立具有所需IAM權限的原則，並將建議AWSServiceRoleForEC2Fleet做為角色名稱。

   c. 選擇 Next (下一步)。
5. 在 Add permissions (新增許可) 頁面上，選擇 Next (下一步)。
6. 在命名、檢閱和建立頁面上，選擇建立角色。

如果您不再需要使用EC2艦隊，建議您刪除該AWSServiceRoleForEC2Fleet角色。從您的帳戶中刪除此角色後，如果您建立其他機群，則可以再次建立此角色。

如需詳細資訊，請參閱IAM使用指南中的服務連結角色。

授予客戶受管金鑰的存取權，以便搭配加密AMIs和EBS快照使用

如果您在EC2叢集中指定加密AMI或加密的 Amazon EBS 快照，並使用 AWS KMS 金鑰進行加密，則必須授與AWSServiceRoleForEC2Fleet角色使用客戶受管金鑰的權限，以便 Amazon EC2 可以代表您啟動執行個體。若要執行此動作，您必須將授予新增至客戶受管金鑰，如下列程序所示。

提供許可時，授與為金鑰政策的替代方案。如需詳細資訊，請參閱《AWS Key Management Service 開發人員指南》中的使用授與和在 AWS KMS中使用金鑰政策。

授與 AWSServiceRoleForEC2Fleet 角色使用客戶管理金鑰的權限

- 使用 create-grant 命令可將授權新增至客戶管理的金鑰，並指定授與執行授權之作業的主體 (AWSServiceRoleForEC2Fleet服務連結角色)。客戶管理金鑰由`key-id`參數和客戶管理金鑰指定。ARN主體是由`grantee-principal`參數和AWSServiceRoleForEC2Fleet服務連結角色所指定。ARN

```
aws kms create-grant \
    --region us-east-1 \
    --key-id arn:aws:kms:us-
east-1:444455556666:key/1234abcd-12ab-34cd-56ef-1234567890ab \
    --grantee-principal arn:aws:iam::111122223333:role/AWSServiceRoleForEC2Fleet \
    --operations "Decrypt" "Encrypt" "GenerateDataKey"
 "GenerateDataKeyWithoutPlaintext" "CreateGrant" "DescribeKey" "ReEncryptFrom"
 "ReEncryptTo"
```

EC2叢集使用者的權限

如果您的使用者將建立或管理EC2叢集，請務必授予他們必要的權限。

若要為EC2叢集建立原則

1. 在開啟IAM主控台https://console.aws.amazon.com/iam/。
2. 在導覽窗格中，選擇政策。
3. 選擇 Create policy (建立政策)。
4. 在 [建立原則] 頁面上，選擇索JSON引標籤，以下列項目取代文字，然後選擇 [檢閱原則]。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:*"
            ],
            "Resource": "*"
        },
```

```
        {
            "Effect": "Allow",
            "Action": [
              "iam:ListRoles",
              "iam:PassRole",
              "iam:ListInstanceProfiles"
            ],
            "Resource":"arn:aws:iam::123456789012:role/DevTeam*"
        }
    ]
}
```

`ec2:*`授予用戶調用所有 Amazon EC2 API 操作的權限。若要限制使用者使用特定 Amazon EC2 API 動作，請改為指定這些動作。

使用者必須具有呼叫`iam:ListRoles`動作以列舉現有IAM角色的權限、指定EC2叢集角色的`iam:PassRole`動作，以及列舉現有執行個體設定檔的`iam:ListInstanceProfiles`動作。

(選擇性) 若要讓使用者能夠使用IAM主控台建立角色或執行個體設定檔，您還必須將下列動作新增至原則：

- `iam:AddRoleToInstanceProfile`
- `iam:AttachRolePolicy`
- `iam:CreateInstanceProfile`
- `iam:CreateRole`
- `iam:GetRole`
- `iam:ListPolicies`

5. 在 Review policy (檢閱政策) 頁面上，輸入政策名稱和說明，然後選擇 Create policy (建立政策)。
6. 若要提供存取權，請新增權限至您的使用者、群組或角色：

   - 使用者和群組位於 AWS IAM Identity Center：

     建立權限合集。請按照 AWS IAM Identity Center 使用者指南 中的 建立權限合集 說明進行操作。

   - IAM透過身分識別提供者管理的使用者：

     建立聯合身分的角色。請遵循《使用指南》中的〈為第三方身分識別提供IAM者 (同盟) 建立角色〉中的指示進行。

- IAM使用者：
  - 建立您的使用者可擔任的角色。請按照《用戶指南》中的「為IAM用戶創建角色」中的IAM說明進行操作。
  - (不建議) 將政策直接附加至使用者，或將使用者新增至使用者群組。遵循《使用者指南》中的「向使用者 (主控台) 新增權限」IAM 中的指示進行。

建立EC2艦隊

若要使用叢集啟動執行個體EC2叢集，您只需在叢集請求中指定下列參數，叢集將使用其他參數的預設值：

- `LaunchTemplateId`或 `LaunchTemplateName` — 指定要使用的啟動範本 (其中包含要啟動之執行個體的參數，例如執行個體類型和可用區域)
- `TotalTargetCapacity` – 指定機群的總目標容量
- `DefaultTargetCapacityType` – 指定預設購買選項是隨需還是 Spot

若要覆寫啟動範本中指定的參數，您可以指定一或多個覆寫。每個覆寫可能會因執行個體類型、可用區域、子網路和最高價格而有所不同，並且可能包含不同的加權容量。作為指定執行個體類型的替代方法，您可以指定執行個體必須具有的屬性，Amazon EC2 將使用這些屬性識別所有執行個體類型。如需詳細資訊，請參閱指定EC2叢集或 Spot 叢集的執行個體類型選取屬性。

對於類型的EC2叢集`instant`，您可以指定「Systems Manager」參數而不是 AMI ID。您可以在取代或啟動樣板中指定「Systems Manager」參數。如需詳細資訊，請參閱使用「Systems Manager」參數而不是 AMI ID。

您可以在JSON檔案中指定叢集參數。如需有關可指定之所有可能參數的資訊，請參閱檢視所有EC2車隊配置選項。

如需叢集組態範例，請參閱EC2叢集的範例CLI組態。

目前不支援建立EC2叢集的主控台。

若要建立EC2叢集

- 使用 create-fleet (AWS CLI) 指令來建立叢集，並指定包含叢集組態參數的JSON檔案。

```
aws ec2 create-fleet --cli-input-json file://file_name.json
```

以下是類型 `request` 或 `maintain` 的叢集輸出範例。

```
{
    "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE"
}
```

以下是啟動目標容量之類型 `instant` 的叢集輸出範例。

```
{
  "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE",
  "Errors": [],
  "Instances": [
    {
      "LaunchTemplateAndOverrides": {
        "LaunchTemplateSpecification": {
          "LaunchTemplateId": "lt-01234a567b8910abcEXAMPLE",
          "Version": "1"
        },
        "Overrides": {
          "InstanceType": "c5.large",
          "AvailabilityZone": "us-east-1a"
        }
      },
      "Lifecycle": "on-demand",
      "InstanceIds": [
        "i-1234567890abcdef0",
        "i-9876543210abcdef9"
      ],
      "InstanceType": "c5.large",
      "Platform": null
    },
    {
      "LaunchTemplateAndOverrides": {
        "LaunchTemplateSpecification": {
          "LaunchTemplateId": "lt-01234a567b8910abcEXAMPLE",
          "Version": "1"
        },
        "Overrides": {
          "InstanceType": "c4.large",
          "AvailabilityZone": "us-east-1a"
        }
      },
      "Lifecycle": "on-demand",
```

```
      "InstanceIds": [
        "i-5678901234abcdef0",
        "i-5432109876abcdef9"
      ]
  ]
}
```

以下是啟動部分目標容量並附有未啟動之執行個體錯誤之類型 `instant` 的叢集輸出範例。

```
{
  "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE",
  "Errors": [
    {
      "LaunchTemplateAndOverrides": {
        "LaunchTemplateSpecification": {
          "LaunchTemplateId": "lt-01234a567b8910abcEXAMPLE",
          "Version": "1"
        },
        "Overrides": {
          "InstanceType": "c4.xlarge",
          "AvailabilityZone": "us-east-1a",
        }
      },
      "Lifecycle": "on-demand",
      "ErrorCode": "InsufficientInstanceCapacity",
      "ErrorMessage": ""
    },
  ],
  "Instances": [
    {
      "LaunchTemplateAndOverrides": {
        "LaunchTemplateSpecification": {
          "LaunchTemplateId": "lt-01234a567b8910abcEXAMPLE",
          "Version": "1"
        },
        "Overrides": {
          "InstanceType": "c5.large",
          "AvailabilityZone": "us-east-1a"
        }
      },
      "Lifecycle": "on-demand",
      "InstanceIds": [
        "i-1234567890abcdef0",
```

```
        "i-9876543210abcdef9"
      ]
  ]
}
```

以下是未啟動任何執行個體之類型 `instant` 的叢集輸出範例。

```
{
  "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE",
  "Errors": [
    {
      "LaunchTemplateAndOverrides": {
        "LaunchTemplateSpecification": {
          "LaunchTemplateId": "lt-01234a567b8910abcEXAMPLE",
          "Version": "1"
        },
        "Overrides": {
          "InstanceType": "c4.xlarge",
          "AvailabilityZone": "us-east-1a",
        }
      },
      "Lifecycle": "on-demand",
      "ErrorCode": "InsufficientCapacity",
      "ErrorMessage": ""
    },
    {
      "LaunchTemplateAndOverrides": {
        "LaunchTemplateSpecification": {
          "LaunchTemplateId": "lt-01234a567b8910abcEXAMPLE",
          "Version": "1"
        },
        "Overrides": {
          "InstanceType": "c5.large",
          "AvailabilityZone": "us-east-1a",
        }
      },
      "Lifecycle": "on-demand",
      "ErrorCode": "InsufficientCapacity",
      "ErrorMessage": ""
    },
  ],
  "Instances": []
}
```

建立可取代狀態不良的 Spot 執行個體的EC2叢集

EC2叢集每兩分鐘檢查叢集中執行個體的健全狀態。執行個體的運作狀態會是 `healthy` 或 `unhealthy`。

EC2叢集使用 Amazon 提供的狀態檢查來判斷執行個體的運作狀態EC2。如果執行個體狀態檢查或系統狀態檢查的運作狀態檢查，出現連續三次為 `impaired`，則該執行個體即為 `unhealthy`。如需詳細資訊，請參閱Amazon EC2 執行個體的狀態檢查。

您可以設定機群以取代運作狀態不佳的 競價型執行個體。`ReplaceUnhealthyInstances` 設為 `true` 以後，在報告為 `unhealthy` 時，則 Spot 執行個體會被取代。當取代一個狀態不佳的 Spot 執行個體時，機群可能會低於其目標容量長達幾分鐘。

要求

- 只有維持目標容量 (類型的機EC2隊`maintain`) 的機隊才支援 Health 狀態檢查更換，而不支援或類型的機隊。`request` `instant`
- 運作狀態檢查取代僅支援 競價型執行個體。此功能不支援 隨需執行個體。
- 您可以將EC2叢集設定為只有在建立時才更換運作狀態不良的執行個體。
- 使用者只有在擁有呼叫 `ec2:DescribeInstanceStatus` 動作的許可時，才能使用運作狀態檢查取代功能。

設定EC2叢集以取代運作狀態不良的 Spot 執行個體

1. 使用資訊在中建立EC2叢集建立EC2艦隊。
2. 若要設定叢集以取代運作狀態不良的 Spot 執行個體，請在JSON檔案中指定`true`。`ReplaceUnhealthyInstances`

檢視所有EC2車隊配置選項

若要檢視EC2叢集組態參數的完整清單，您可以產生JSON檔案。如需每個參數的說明，請參閱 AWS CLI 命令參考中的 create-fleet。

使用所有可能的EC2叢集參數產生JSON檔案

使用 create-fleet (AWS CLI) 命令和`--generate-cli-skeleton`參數來生成一個 EC2 Fleet JSON 文件，並將輸出導向到一個文件進行保存。

```
aws ec2 create-fleet \
    --generate-cli-skeleton input > ec2createfleet.json
```

範例輸出

```
{
    "DryRun": true,
    "ClientToken": "",
    "SpotOptions": {
        "AllocationStrategy": "price-capacity-optimized",
        "MaintenanceStrategies": {
            "CapacityRebalance": {
                "ReplacementStrategy": "launch"
            }
        },
        "InstanceInterruptionBehavior": "hibernate",
        "InstancePoolsToUseCount": 0,
        "SingleInstanceType": true,
        "SingleAvailabilityZone": true,
        "MinTargetCapacity": 0,
        "MaxTotalPrice": ""
    },
    "OnDemandOptions": {
        "AllocationStrategy": "prioritized",
        "CapacityReservationOptions": {
            "UsageStrategy": "use-capacity-reservations-first"
        },
        "SingleInstanceType": true,
        "SingleAvailabilityZone": true,
        "MinTargetCapacity": 0,
        "MaxTotalPrice": ""
    },
    "ExcessCapacityTerminationPolicy": "termination",
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateId": "",
                "LaunchTemplateName": "",
                "Version": ""
            },
            "Overrides": [
                {
                    "InstanceType": "r5.metal",
```

```
                    "MaxPrice": "",
                    "SubnetId": "",
                    "AvailabilityZone": "",
                    "WeightedCapacity": 0.0,
                    "Priority": 0.0,
                    "Placement": {
                        "AvailabilityZone": "",
                        "Affinity": "",
                        "GroupName": "",
                        "PartitionNumber": 0,
                        "HostId": "",
                        "Tenancy": "dedicated",
                        "SpreadDomain": "",
                        "HostResourceGroupArn": ""
                    },
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 0
                        },
                        "MemoryMiB": {
                            "Min": 0,
                            "Max": 0
                        },
                        "CpuManufacturers": [
                            "amd"
                        ],
                        "MemoryGiBPerVCpu": {
                            "Min": 0.0,
                            "Max": 0.0
                        },
                        "ExcludedInstanceTypes": [
                            ""
                        ],
                        "InstanceGenerations": [
                            "previous"
                        ],
                        "SpotMaxPricePercentageOverLowestPrice": 0,
                        "OnDemandMaxPricePercentageOverLowestPrice": 0,
                        "BareMetal": "included",
                        "BurstablePerformance": "required",
                        "RequireHibernateSupport": true,
                        "NetworkInterfaceCount": {
                            "Min": 0,
```

```
                            "Max": 0
                        },
                        "LocalStorage": "excluded",
                        "LocalStorageTypes": [
                            "ssd"
                        ],
                        "TotalLocalStorageGB": {
                            "Min": 0.0,
                            "Max": 0.0
                        },
                        "BaselineEbsBandwidthMbps": {
                            "Min": 0,
                            "Max": 0
                        },
                        "AcceleratorTypes": [
                            "inference"
                        ],
                        "AcceleratorCount": {
                            "Min": 0,
                            "Max": 0
                        },
                        "AcceleratorManufacturers": [
                            "amd"
                        ],
                        "AcceleratorNames": [
                            "a100"
                        ],
                        "AcceleratorTotalMemoryMiB": {
                            "Min": 0,
                            "Max": 0
                        }
                    }
                }
            ]
        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 0,
        "OnDemandTargetCapacity": 0,
        "SpotTargetCapacity": 0,
        "DefaultTargetCapacityType": "on-demand",
        "TargetCapacityUnitType": "memory-mib"
    },
    "TerminateInstancesWithExpiration": true,
```

```
    "Type": "instant",
    "ValidFrom": "1970-01-01T00:00:00",
    "ValidUntil": "1970-01-01T00:00:00",
    "ReplaceUnhealthyInstances": true,
    "TagSpecifications": [
        {
            "ResourceType": "fleet",
            "Tags": [
                {
                    "Key": "",
                    "Value": ""
                }
            ]
        }
    ],
    "Context": ""
}
```

標記新的或現有的EC2叢集請求，以及其啟動的執行個體和磁碟區

為了協助分類和管理EC2叢集請求及其啟動的執行個體和磁碟區，您可以使用自訂中繼資料來標記它們。您可以在建立EC2叢集請求時或之後將標籤指派給叢集請求。同樣地，您可以在叢集啟動或之後為執行個體和磁碟區指派標籤。

當您標記機群請求時，由機群啟動的執行個體和磁碟區不會自動加上標記。您需要明確標記由機群啟動的執行個體和磁碟區。您可以選擇只將標籤指派給叢集請求，或僅指派給叢集啟動的執行個體，或僅指派給叢集啟動的執行個體所連接的磁碟區，或指派給所有執行個體的磁碟區。

> **Note**
>
> 針對 `instant` 機群類型，您可以為連接至 隨需執行個體 和 競價型執行個體 的磁碟區加標籤。針對 `request` 或 `maintain` 機群類型，您只可以為連接至 隨需執行個體 的磁碟區加標籤。

如需有關標籤運作方式的詳細資訊，請參閱標記您的 Amazon EC2 資源。

必要條件

授予使用者標記資源的許可。如需詳細資訊，請參閱範例：標籤資源。

授予使用者標記資源的許可

建立包含下列項目的IAM策略：

- `ec2:CreateTags` 動作。這會授予使用者建立標籤的許可。
- `ec2:CreateFleet` 動作。這會授與使用者建立EC2叢集要求的權限。
- 針對 `Resource`，建議您指定 `"*"`。這可讓使用者標記所有資源類型。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "TagEC2FleetRequest",
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags",
                "ec2:CreateFleet"
            ],
            "Resource": "*"
}
```

> **Important**
>
> 我們目前不支援 `create-fleet` 資源的資源層級許可。如果您指定 `create-fleet` 作為資源，當您嘗試標記機群時，會收到未經授權的例外狀況。下列範例說明如何不要設定政策。
>
> ```
> {
>  "Effect": "Allow",
>  "Action": [
>  "ec2:CreateTags",
>  "ec2:CreateFleet"
>  ],
>  "Resource": "arn:aws:ec2:us-east-1:111122223333:create-fleet/*"
> }
> ```

若要提供存取權，請新增權限至您的使用者、群組或角色：

- 使用者和群組位於 AWS IAM Identity Center：

  建立權限合集。請按照 AWS IAM Identity Center 使用者指南 中的 建立權限合集 說明進行操作。

- IAM透過身分識別提供者管理的使用者：

  建立聯合身分的角色。請遵循《使用指南》中的〈為第三方身分識別提供IAM者 (同盟) 建立角色〉中的指示進行。
- IAM使用者：
  - 建立您的使用者可擔任的角色。請按照《用戶指南》中的「為IAM用戶創建角色」中的IAM說明進行操作。
  - (不建議) 將政策直接附加至使用者，或將使用者新增至使用者群組。遵循《使用者指南》中的「向使用者 (主控台) 新增權限」IAM 中的指示進行。

標記新的EC2叢集請求

若要在建立EC2叢集要求時加上標記，請在用來建立叢集的JSON檔案中指定索引鍵值組。`ResourceType` 的值必須為 `fleet`。若您指定其他值，機群請求將會失敗。

標記EC2叢集啟動的執行個體和磁碟區

若要在叢集啟動執行個體和磁碟區時加上標籤，請在叢集要求中參考的啟動範本中EC2指定標籤。

> **Note**
>
> 您無法標記連接至 競價型執行個體、由 `request` 或 `maintain` 機群類型啟動的磁碟區。

標記現有的EC2叢集請求、執行個體和磁碟區

使用建立標籤命令來標記現有資源。

```
aws ec2 create-tags \
    --resources fleet-12a34b55-67cd-8ef9-
ba9b-9208dEXAMPLE i-1234567890abcdef0 vol-1234567890EXAMPLE \
    --tags Key=purpose,Value=test
```

說明EC2叢集的組態、執行個體和事件歷史記錄

您可以說明EC2叢集設定、叢集中的執行個體，以及EC2叢集的EC2事件歷史記錄。

主題

- 描述您所有的EC2艦隊

- 說明指定EC2叢集中的所有執行個體
- 描述EC2艦隊的事件歷史

描述您所有的EC2艦隊

使用描述叢集指令來描述您所有的艦隊。EC2

```
aws ec2 describe-fleets
```

> **Important**
>
> 若機群類型為 `instant`，則必須指定機群 ID，否則其不會顯示於回應中。包括 `--fleet-ids`，如下所示：
>
> ```
> aws ec2 describe-fleets --fleet-ids fleet-8a22eee4-f489-ab02-06b8-832a7EXAMPLE
> ```

範例輸出

```
{
    "Fleets": [
        {
            "ActivityStatus": "fulfilled",
            "CreateTime": "2022-02-09T03:35:52+00:00",
            "FleetId": "fleet-364457cd-3a7a-4ed9-83d0-7b63e51bb1b7",
            "FleetState": "active",
            "ExcessCapacityTerminationPolicy": "termination",
            "FulfilledCapacity": 2.0,
            "FulfilledOnDemandCapacity": 0.0,
            "LaunchTemplateConfigs": [
                {
                    "LaunchTemplateSpecification": {
                        "LaunchTemplateName": "my-launch-template",
                        "Version": "$Latest"
                    }
                }
            ],
            "TargetCapacitySpecification": {
                "TotalTargetCapacity": 2,
                "OnDemandTargetCapacity": 0,
```

```
                "SpotTargetCapacity": 2,
                "DefaultTargetCapacityType": "spot"
            },
            "TerminateInstancesWithExpiration": false,
            "Type": "maintain",
            "ReplaceUnhealthyInstances": false,
            "SpotOptions": {
                "AllocationStrategy": "capacity-optimized",
                "InstanceInterruptionBehavior": "terminate"
            },
            "OnDemandOptions": {
                "AllocationStrategy": "lowestPrice"
            }
        }
    ]
}
```

說明指定EC2叢集中的所有執行個體

使用命describe-fleet-instances令描述指定EC2叢集的執行個體。執行中執行個體的傳回清單會定期重新整理且可能過期。

```
aws ec2 describe-fleet-instances  --fleet-id fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE
```

範例輸出

```
{
    "ActiveInstances": [
        {
            "InstanceId": "i-09cd595998cb3765e",
            "InstanceHealth": "healthy",
            "InstanceType": "m4.large",
            "SpotInstanceRequestId": "sir-86k84j6p"
        },
        {
            "InstanceId": "i-09cf95167ca219f17",
            "InstanceHealth": "healthy",
            "InstanceType": "m4.large",
            "SpotInstanceRequestId": "sir-dvxi7fsm"
        }
    ],
    "FleetId": "fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE"
```

```
}
```

描述EC2艦隊的事件歷史

使用命describe-fleet-history令來描述指定EC2叢集在指定時間內的事件。如需輸出中傳回之事件的詳細資訊，請參閱EC2艦隊事件類型。

```
aws ec2 describe-fleet-history \
    --fleet-id fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
    --start-time 2018-04-10T00:00:00Z
```

範例輸出

```
{
    "HistoryRecords": [
        {
            "EventInformation": {
                "EventSubType": "submitted"
            },
            "EventType": "fleetRequestChange",
            "Timestamp": "2020-09-01T18:26:05.000Z"
        },
        {
            "EventInformation": {
                "EventSubType": "active"
            },
            "EventType": "fleetRequestChange",
            "Timestamp": "2020-09-01T18:26:15.000Z"
        },
        {
            "EventInformation": {
                "EventDescription": "t2.small, ami-07c8bc5c1ce9598c3, ...",
                "EventSubType": "progress"
            },
            "EventType": "fleetRequestChange",
            "Timestamp": "2020-09-01T18:26:17.000Z"
        },
        {
            "EventInformation": {
                "EventDescription": "{\"instanceType\":\"t2.small\", ...}",
                "EventSubType": "launched",
                "InstanceId": "i-083a1c446e66085d2"
```

```
            },
            "EventType": "instanceChange",
            "Timestamp": "2020-09-01T18:26:17.000Z"
        },
        {
            "EventInformation": {
                "EventDescription": "{\"instanceType\":\"t2.small\", ...}",
                "EventSubType": "launched",
                "InstanceId": "i-090db02406cc3c2d6"
            },
            "EventType": "instanceChange",
            "Timestamp": "2020-09-01T18:26:17.000Z"
        }
    ],
    "FleetId": "fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",
    "LastEvaluatedTime": "1970-01-01T00:00:00.000Z",
    "StartTime": "2018-04-09T23:53:20.000Z"
}
```

修改EC2叢集

您可以修改EC2叢集的總目標容量、Spot 容量和隨需容量。如果新的目標總容量減少到目前叢集的大小以下，您也可以修改執行中的執行個體是否應終止。

考量事項

修改EC2叢集時，請考慮下列事項：

- [叢集類型] — 您只能修改類型的 [EC2叢集] `maintain`。您無法修改`request`或類型的EC2叢集`instant`。
- 叢集參數 — 您可以修改「EC2叢集」的下列參數：
  - `target-capacity-specification`— 增加或減少以下目標容量：
    - `TotalTargetCapacity`
    - `OnDemandTargetCapacity`
    - `SpotTargetCapacity`
  - `excess-capacity-termination-policy`— 如果叢集的總目標容量減少到目前的EC2叢集大小，是否應終止執行中的執行個體。有效的 值如下：
    - `no-termination`
    - `termination`

- 增加總目標容量時的叢集行為 — 當您增加總目標容量時，EC2叢集會根據針對`DefaultTargetCapacityType`指定的執行個體購買選項 (即隨需執行個體或 Spot 執行個體) 以及指定的配置策略啟動額外的執行個體。
- 降低 Spot 目標容量時的叢集行為 — 當您減少 Spot 目標容量時，EC2叢集會刪除任何超出新目標容量的開啟請求。您可以要求叢集終止 Spot 執行個體，直到叢集的大小達到新的目標容量為止。如果分配策略為 `lowest-price`，則機群會以最高每單位價格終止執行個體。如果分配策略為 `diversified`，則機群會在集區中終止執行個體。或者，您可以要求EC2叢集保持其目前大小，但不能取代任何已中斷的 Spot 執行個體或手動終止的任何執行個體。

  當EC2叢集因目標容量減少而終止 Spot 執行個體時，執行個體會收到 Spot 執行個體中斷通知。
- [叢集] 狀態 — 您可以修改處於`submitted`或`active`狀態的 [EC2叢集]。當您修改機群時，機群會進入 `modifying` 狀態。

用於修改EC2叢集的指令

您可以使用「修改叢集」 AWS CLI 指令來修改「叢集」。EC2

若要修改EC2叢集的總目標容量

使用修改叢集指令來更新指定叢集的目標容量。EC2

```
aws ec2 modify-fleet \
    --fleet-id fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
    --target-capacity-specification TotalTargetCapacity=20
```

若要指定在減少叢集的總目標容量時，不得刪除多餘的執行中執行個EC2體

如果您要降低目標容量，但希望保持目前的大小，則可以按照下列方式修改前一個命令。

```
aws ec2 modify-fleet \
    --fleet-id fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
    --target-capacity-specification TotalTargetCapacity=10 \
    --excess-capacity-termination-policy no-termination
```

刪除EC2叢集請求和叢集中的執行個體

如果您不再需要EC2叢集請求，您可以將其刪除。刪除叢集請求後，會取消與叢集相關聯的所有 Spot 請求，因此不會啟動新的 Spot 執行個體。

刪除EC2叢集請求時，您還必須指定是否要終止其所有執行個體。這包含隨需執行個體和 Spot 執行個體。對於instant叢集，EC2叢集必須在刪除叢集時終止執行個體。不支援具有執行中執行個體的已刪除 instant 機群。

如果您指定在刪除叢集要求時必須終止執行個體，叢集要求就會進入deleted_terminating狀態。否則會進入 deleted_running 狀態，且執行個體會一直執行直到中斷或您手動終止。

限制

- 您可以在單一作業中刪除多達 25 個類型instant的叢集。
- 您最多可以刪除 100 個類型的叢集，也可以request在maintain單一作業中刪除。
- 如上所述，您可以在單一作業中刪除多達 125 個叢集，前提是您不超過每個叢集類型的配額。
- 如果超過要刪除的叢集指定數目，則不會刪除任何叢集。
- 單一作業最多可終止 1000 個執行個體以刪除instant叢集。

若要刪除EC2叢集並終止其執行個體

使用 delete fleets 命令和--terminate-instances參數可刪除指定的EC2叢集並終止其關聯的執行個體。

```
aws ec2 delete-fleets \
    --fleet-ids fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
    --terminate-instances
```

範例輸出

```
{
    "UnsuccessfulFleetDeletions": [],
    "SuccessfulFleetDeletions": [
        {
            "CurrentFleetState": "deleted_terminating",
            "PreviousFleetState": "active",
            "FleetId": "fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE"
        }
    ]
}
```

刪除EC2叢集而不終止其執行處理

您可以使用`--no-terminate-instances`參數修改上一個命令，以刪除指定的EC2叢集，而不會終止其關聯的執行個體。

> **Note**
>
> `--no-terminate-instances` 不支援 `instant` 機群。

```
aws ec2 delete-fleets \
    --fleet-ids fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
    --no-terminate-instances
```

範例輸出

```
{
    "UnsuccessfulFleetDeletions": [],
    "SuccessfulFleetDeletions": [
        {
            "CurrentFleetState": "deleted_running",
            "PreviousFleetState": "active",
            "FleetId": "fleet-4b8aaae8-dfb5-436d-a4c6-3dafa4c6b7dcEXAMPLE"
        }
    ]
}
```

當機群無法刪除時進行故障診斷

如果EC2叢集無法刪除，則`UnsuccessfulFleetDeletions`在輸出中傳回EC2叢集的 ID、錯誤代碼和錯誤訊息。

錯誤代碼為：

- `ExceededInstantFleetNumForDeletion`
- `fleetIdDoesNotExist`
- `fleetIdMalformed`
- `fleetNotInDeletableState`
- `NoTerminateInstancesNotSupported`
- `UnauthorizedOperation`

- unexpectedError

ExceededInstantFleetNumForDeletion 疑難排解

如果您在單一請求中嘗試刪除超過 25 個 instant 機群，則會傳回 ExceededInstantFleetNumForDeletion 錯誤。下列為此錯誤的範例輸出。

```
{
    "UnsuccessfulFleetDeletions": [
     {
          "FleetId": " fleet-5d130460-0c26-bfd9-2c32-0100a098f625",
          "Error": {
                  "Message": "Can’t delete more than 25 instant fleets in a single
 request.",
                  "Code": "ExceededInstantFleetNumForDeletion"
           }
     },
     {
           "FleetId": "fleet-9a941b23-0286-5bf4-2430-03a029a07e31",
           "Error": {
                  "Message": "Can’t delete more than 25 instant fleets in a single
 request.",
                  "Code": "ExceededInstantFleetNumForDeletion"
            }
     }
     .
     .
     .
     ],
     "SuccessfulFleetDeletions": []
}
```

NoTerminateInstancesNotSupported 疑難排解

如果您指定在刪除 instant 機群時不得終止該機群中的執行個體，則會傳回 NoTerminateInstancesNotSupported 錯誤。--no-terminate-instances 不支援 instant 機群。下列為此錯誤的範例輸出。

```
{
      "UnsuccessfulFleetDeletions": [
            {
                  "FleetId": "fleet-5d130460-0c26-bfd9-2c32-0100a098f625",
```

```
                  "Error": {
                          "Message": "NoTerminateInstances option is not supported for
 instant fleet",
                          "Code": "NoTerminateInstancesNotSupported"
                   }
            }
      ],
      "SuccessfulFleetDeletions": []
```

UnauthorizedOperation 疑難排解

如果沒有終止執行個體的許可，您在刪除必須終止其執行個體的機群時，會收到 UnauthorizedOperation 錯誤。以下是錯誤回應。

```
<Response><Errors><Error><Code>UnauthorizedOperation</Code><Message>You are not
 authorized to perform this
operation. Encoded authorization failure message: VvuncIxj7Z_CPGNYXWqnuFV-
YjByeAU66Q9752NtQ-I3-qnDLWs6JLFd
KnSMMiq5s6cGqjjPtEDpsnGHzzyHasFHOaRYJpaDVravoW25azn6KNkUQQlFwhJyujt2dtNCdduJfrqcFYAjlEiRMkfDHt7
BHturzDK6A560Y2nDSUiMmAB1y9UNtqaZJ9SNe5sNxKMqZaqKtjRbk02RZu5V2vn9VMk6fm2aMVHbY9JhLvGypLcMUjtJ76
VPiU5v2s-
UgZ7h0p2yth6ysUdhlONg6dBYu8_y_HtEI54invCj4CoK0qawqzMNe6rcmCQHvtCxtXsbkgyaEbcwmrm2m01-
EMhekLFZeJLr
DtYOpYcEl4_nWFX1wtQDCnNNCmxnJZAoJvb3VMDYpDTsxjQv1PxODZuqWHs23YXWVywzgnLtHeRf2o4lUhGBw17mXsS07k7
PT9vrHtQiILor5VVTsjSPWg7edj__1rsnXhwPSu8gI48ZLRGrPQqFq0RmKO_QIE8N8s6NWzCK4yoX-9gDcheurOGpkprPIC
</Message></Error></Errors><RequestID>89b1215c-7814-40ae-a8db-41761f43f2b0</
RequestID></Response>
```

若要解決錯誤，您必須將ec2:TerminateInstances動作新增至IAM原則，如下列範例所示。

```
{
      "Version": "2012-10-17",
      "Statement": [
           {
               "Sid": "DeleteFleetsAndTerminateInstances",
               "Effect": "Allow",
               "Action": [
                      "ec2:DeleteFleets"
                      "ec2:TerminateInstances"
               ],
               "Resource": "*"
           }
      ]
```

```
    }
```

使用現貨艦隊

若要開始使用 Spot 叢集，請建立一個請求，其中包括 Spot 執行個體的總目標容量、選用的隨需部分，以AMI及手動指定和 key pair 組，或指定包含叢集中執行個體組態的啟動範本。您可以選擇性地指定其他參數，或讓叢集使用預設值。您也可以在建立叢集時標記叢集要求及其執行個體和磁碟區。

叢集會在有可用容量時啟動隨需執行個體，並在您的最高價超過 Spot 價格和可用容量時啟動 Spot 執行個體。

叢集啟動後，您可以描述叢集要求、叢集中的執行個體以及任何叢集事件。您也可以視需要指派其他標籤。

如果您需要變更任何叢集參數 (例如目標容量總計)，您可以修改叢集 (前提是已設定為維持容量)。一次性請求提交後，您無法修改該要求的容量。

叢集請求會保持作用中狀態，直到到期或您取消 (刪除) 它為止。當您取消叢集要求時，您可以終止執行個體或讓執行個體保持執行。如果您選擇讓它們繼續執行，隨需執行個體會一直執行，直到您終止它們為止，Spot 執行個體會一直執行，直到它們中斷或您終止它們為止。

主題

- Spot Fleet 請求狀態
- 建立 Spot 機群
- 標記新的或現有的 Spot 叢集請求及其啟動的執行個體和磁碟區
- 描述 Spot 叢集設定、其執行個體和事件歷史記錄
- 修改 Spot Fleet 請求
- 取消 (刪除) Spot 叢集請求
- 瞭解 Spot 叢集的自動擴充

Spot Fleet 請求狀態

Spot Fleet 請求可以是各種狀態之一，每個狀態都指示請求生命週期的不同階段及其對執行個體的管理。

Spot Fleet 請求可以是下列任一狀態：

`submitted`

正在評估 Spot 叢集請求，Amazon EC2 正準備啟動目標執行個體數量。如果請求超出您的 Spot 叢集配額，則會立即取消該請求。

`active`

Spot 叢集已通過驗證，Amazon 正在嘗試維護EC2正在執行的 Spot 執行個體的目標數量。此請求會一直維持在該狀態，直到修改或取消。

`modifying`

正在修改 Spot Fleet 請求。請求會保持此狀態，直到完全處理修改或取消請求為止。只有 `maintain` 機群類型可以修改。此狀態不適用於一次性`request`叢集類型。

`cancelled_running`

Spot 叢集已取消 (刪除)，且不會啟動額外的 Spot 執行個體。其現有執行個體將繼續執行直到中斷或手動終止。此請求會一直維持在該狀態，直到所有執行個體中斷或終止。

`cancelled_terminating`

Spot 叢集已取消 (刪除)，且其執行個體正在終止。此請求會一直維持在該狀態，直到所有執行個體終止。

`cancelled`

Spot 叢集已取消 (刪除)，且沒有執行中的執行個體。請求會在其執行個體終止之後兩天刪除。

建立 Spot 機群

使用 AWS Management Console，透過僅選擇AMI和您想要的目標總容量來快速建立 Spot 叢集請求。Amazon EC2 將設定最符合您需求的叢集，並遵循 Spot 最佳實務。如需詳細資訊，請參閱快速建立 Spot Fleet 請求 (主控台)。否則，您可以修改任何預設設定。如需詳細資訊，請參閱 使用已定義的參數建立 Spot Fleet 請求 (主控台) 和 使用建立競價型叢集 AWS CLI。

如果您想要在叢集中包含隨需執行個體，則需要在請求中指定啟動範本，並指定所需的隨需容量。

叢集會在容量可用時啟動隨需執行個體，並在您的最高價超過 Spot 價格和可用容量時啟動 Spot 執行個體。

如果您的叢集包含 Spot 執行個體且屬於類型`maintain`，Amazon EC2 會在 Spot 執行個體中斷時嘗試維持叢集目標容量。

主題

- Spot Fleet 許可
- 快速建立 Spot Fleet 請求 (主控台)
- 使用已定義的參數建立 Spot Fleet 請求 (主控台)
- 使用建立競價型叢集 AWS CLI
- 建立可取代狀態不良的 Spot 執行個體的 Spot 叢集

Spot Fleet 許可

如果使用者將會建立或管理 Spot 機群，您必須授予這些使用者所需的許可。

如果您使用 Amazon EC2 主控台建立 Spot 叢集，它會建立兩個名為`AWSServiceRoleForEC2SpotFleet`和的服務連結角色`AWSServiceRoleForEC2Spot`，以及一個名為`aws-ec2-spot-fleet-tagging-role`的角色，授與 Spot 叢集代表您請求、啟動、終止和標記資源的許可。如果您使用 AWS CLI 或API，則必須確定這些角色存在。

請使用下列指示，授與必要的許可並建立角色。

許可和角色

- 將 Spot 機群的許可授予使用者
- Spot Fleet 的服務連結角色
- Spot 執行個體的服務連結角色
- IAM標記 Spot 叢集的角色

將 Spot 機群的許可授予使用者

如果使用者將會建立或管理 Spot 機群，請務必授予這些使用者所需的許可。

為 Spot 機群建立政策

1. 在開啟IAM主控台https://console.aws.amazon.com/iam/。
2. 在導覽窗格中，選擇政策、建立政策。
3. 在 [建立原則] 頁面上 JSON，選擇並以下列項目取代文字。

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

```
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances",
                "ec2:CreateTags",
                "ec2:RequestSpotFleet",
                "ec2:ModifySpotFleetRequest",
                "ec2:CancelSpotFleetRequests",
                "ec2:DescribeSpotFleetRequests",
                "ec2:DescribeSpotFleetInstances",
                "ec2:DescribeSpotFleetRequestHistory"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::*:role/aws-ec2-spot-fleet-tagging-role"
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:CreateServiceLinkedRole",
                "iam:ListRoles",
                "iam:ListInstanceProfiles"
            ],
            "Resource": "*"
        }
    ]
}
```

上述範例政策會向使用者授予大多數 Spot Fleet 使用案例所需的許可。若要限制使用者執行特定API動作，請僅指定這些API動作。

必要EC2和 IAM APIs

政策中APIs必須包含以下內容：

- `ec2:RunInstances` – 在 Spot Fleet 中啟動執行個體所需
- `ec2:CreateTags` – 標記 Spot Fleet 請求、執行個體或磁碟區所需
- `iam:PassRole` – 指定 Spot Fleet 角色所需
- `iam:CreateServiceLinkedRole` – 建立服務連結角色所需

- `iam:ListRoles`— 列舉現有IAM角色所需
- `iam:ListInstanceProfiles` – 列舉現有執行個體描述檔所需

> **Important**
>
> 如果您在啟動規格或啟動範本中指定IAM執行個體設定檔的角色，則必須授與使用者將角色傳遞給服務的權限。要做到這一點，在IAM策略包括作"`arn:aws:iam::*:role/`*`IamInstanceProfile-role`*"為`iam:PassRole`動作的資源。如需詳細資訊，請參閱《使用指南》中的授與使用IAM者將角色傳遞給 AWS 服務的權限。

現貨艦隊 APIs

視需要將下列 Spot 叢集API動作新增至您的原則：

- `ec2:RequestSpotFleet`
- `ec2:ModifySpotFleetRequest`
- `ec2:CancelSpotFleetRequests`
- `ec2:DescribeSpotFleetRequests`
- `ec2:DescribeSpotFleetInstances`
- `ec2:DescribeSpotFleetRequestHistory`

可選 IAM APIs

(選擇性) 若要讓使用者能夠使用IAM主控台建立角色或執行個體設定檔，您必須將下列動作新增至原則：

- `iam:AddRoleToInstanceProfile`
- `iam:AttachRolePolicy`
- `iam:CreateInstanceProfile`
- `iam:CreateRole`
- `iam:GetRole`
- `iam:ListPolicies`

4. 選擇檢閱政策。
5. 在 Review policy (檢閱政策) 頁面上，輸入政策名稱和說明，然後選擇 Create policy (建立政策)。
6. 若要提供存取權，請新增權限至您的使用者、群組或角色：
   - 使用者和群組位於 AWS IAM Identity Center：

     建立權限合集。請按照 AWS IAM Identity Center 使用者指南 中的 建立權限合集 說明進行操作。
   - IAM透過身分識別提供者管理的使用者：

     建立聯合身分的角色。請遵循《使用指南》中的〈為第三方身分識別提供IAM者 (同盟) 建立角色〉中的指示進行。
   - IAM使用者：
     - 建立您的使用者可擔任的角色。請按照《用戶指南》中的「為IAM用戶創建角色」中的IAM說明進行操作。
     - (不建議) 將政策直接附加至使用者，或將使用者新增至使用者群組。遵循《使用者指南》中的「向使用者 (主控台) 新增權限」IAM 中的指示進行。

Spot Fleet 的服務連結角色

Amazon EC2 使用服務連結角色來取得代表您呼叫其他 AWS 服務所需的許可。服務連結角色是直接連結至 AWS 服務的唯一IAM角色類型。服務連結角色提供將權限委派給 AWS 服務的安全方式，因為只有連結的服務可以擔任服務連結角色。如需詳細資訊，請參閱IAM使用指南中的服務連結角色。

Amazon EC2 使用名為AWSServiceRoleForEC2SpotFleet的服務連結角色代表您啟動和管理執行個體。

> **Important**
>
> 如果您在 Spot 叢集中指定加密AMI或加密的 Amazon EBS 快照，則必須授與該AWSServiceRoleForEC2SpotFleet角色使用的權限，以CMK便 Amazon EC2 可以代表您啟動執行個體。如需詳細資訊，請參閱授予存取權，以CMKs便與加密AMIs和EBS快照搭配使用。

授予的權限 AWSServiceRoleForEC2SpotFleet

此AWSServiceRoleForEC2SpotFleet角色授與 Spot 叢集權限，以代表您要求、啟動、終止和標記執行個體。Amazon EC2 使用此服務連結角色來完成下列動作：

- `ec2:RequestSpotInstances` - 請求 競價型執行個體
- `ec2:RunInstances` - 啟動執行個體
- `ec2:TerminateInstances` - 終止執行個體
- `ec2:DescribeImages`-描述執行個體的 Amazon 機器映像檔 (AMIs)
- `ec2:DescribeInstanceStatus` - 監控執行個體的狀態
- `ec2:DescribeSubnets` - 描述執行個體的子網絡
- `ec2:CreateTags` - 將標籤新增至 Spot Fleet 請求、執行個體和磁碟區
- `elasticloadbalancing:RegisterInstancesWithLoadBalancer` - 將指定執行個體新增到指定的負載平衡器
- `elasticloadbalancing:RegisterTargets` - 向指定的目標群組登錄指定的目標

建立服務連結角色

在大部分情況下，您不需要手動建立服務連結角色。Amazon EC2 會在您第一次使用主控台建立 Spot 叢集時建立AWSServiceRoleForEC2SpotFleet服務連結角色。

如果您在 2017 年 10 月之前有一個有效的 Spot 叢集請求，那麼當 Amazon EC2 開始支援此服務連結角色時，Amazon 就會在您的 AWS 帳戶中EC2建立該AWSServiceRoleForEC2SpotFleet角色。如需詳細資訊，請參閱IAM使用者指南中的新角色出現在我的 AWS 帳戶中。

如果您使用 AWS CLI 或建立競價型叢集，則必須先確定此角色存在。API

使用主控台建立 Spot 叢集的 AWSServiceRoleForEC2SpotFleet 角色

1. 在開啟IAM主控台https://console.aws.amazon.com/iam/。
2. 在導覽窗格中，選擇 Roles (角色)。
3. 選擇 Create Role (建立角色)。
4. 在 Select trusted entity (選取信任的實體) 頁面上，執行以下作業：
   a. 針對信任的實體類型，請選擇 AWS 服務。
   b. 在 [使用案例] 下，針對服務或使用案例，選擇EC2。
   c. 對於使用案例，請選擇 EC2-現貨艦隊。

> **Note**
>
> EC2-Spot Fleet 使用案例將自動建立具有所需IAM權限的策略，並將建議AWSEC2SpotFleetServiceRolePolicy作為角色名稱。

d. 選擇 Next (下一步)。

5. 在 Add permissions (新增許可) 頁面上，選擇 Next (下一步)。
6. 在命名、檢閱和建立頁面上，選擇建立角色。

若要使用 AWSServiceRoleForEC2SpotFleet AWS CLI

使用 create-service-linked-role 命令，如下所示。

```
aws iam create-service-linked-role --aws-service-name spotfleet.amazonaws.com
```

如果您不再需要使用 Spot 叢集，建議您刪除該AWSServiceRoleForEC2SpotFleet角色。從您的帳戶中刪除此角色後，如果您使用主控台請求 Spot 叢集，Amazon EC2 將再次建立該角色。如需詳細資訊，請參閱IAM使用指南中的刪除服務連結角色。

授予存取權，以CMKs便與加密AMIs和EBS快照搭配使用

如果您在 Spot 叢集請求中指定加密AMI或加密的 Amazon EBS 快照，並使用客戶受管金鑰進行加密，則必須授與AWSServiceRoleForEC2SpotFleet角色使用權限，以CMK便 Amazon EC2 可以代表您啟動執行個體。若要執行此操作，您必須將授權新增至CMK，如下列程序所示。

提供許可時，授與為金鑰政策的替代方案。如需詳細資訊，請參閱 開發人員指南中的使用授予 AWS KMS和在 AWS Key Management Service 中使用金鑰政策。

若要授與 AWSServiceRoleForEC2SpotFleet 角色使用的權限 CMK

- 使用 create-grant 命令將授權新增至CMK和，以指定被授與執行授權之作業的主體 (AWSServiceRoleForEC2SpotFleet服務連結角色)。由`key-id`參數和ARN的指定CMK。CMK主體是由`grantee-principal`參數和AWSServiceRoleForEC2SpotFleet服務連結角色所指定。ARN

```
aws kms create-grant \
    --region us-east-1 \
    --key-id arn:aws:kms:us-
east-1:444455556666:key/1234abcd-12ab-34cd-56ef-1234567890ab \
```

```
    --grantee-principal arn:aws:iam::111122223333:role/
AWSServiceRoleForEC2SpotFleet \
    --operations "Decrypt" "Encrypt" "GenerateDataKey"
 "GenerateDataKeyWithoutPlaintext" "CreateGrant" "DescribeKey" "ReEncryptFrom"
 "ReEncryptTo"
```

Spot 執行個體的服務連結角色

Amazon EC2 使用名為AWSServiceRoleForEC2Spot的服務連結角色代表您啟動和管理 Spot 執行個體。如需詳細資訊，請參閱Spot 執行個體請求的服務連結角色。

IAM標記 Spot 叢集的角色

此`aws-ec2-spot-fleet-tagging-role`IAM角色授與 Spot 叢集權限，以標記 Spot 叢集請求、執行個體和磁碟區。如需詳細資訊，請參閱標記新的或現有的 Spot 叢集請求及其啟動的執行個體和磁碟區。

> **Important**
>
> 如果您選擇標記機群中的執行個體，並也選擇維護目標容量 (Spot 機群請求為類型 `maintain`)，為使用者與 `IamFleetRole` 所設定的許可差異可能會導致機群中執行個體標記行為的不一致。如果 `IamFleetRole` 不包含 `CreateTags` 許可，則可能無法標記由機群啟動的某些執行個體。雖然我們正在努力修正此不一致，但為了確保標記由機群啟動的所有執行個體，建議您針對 `IamFleetRole` 使用 `aws-ec2-spot-fleet-tagging-role` 角色。或者，若要使用現有角色，請將受`AmazonEC2SpotFleetTaggingRole` AWS 管理策略附加至現有角色。否則，您需要手動將 `CreateTags` 許可新增至現有政策。

建立標記競價型叢集的IAM角色

1. 在開啟IAM主控台https://console.aws.amazon.com/iam/。
2. 在導覽窗格中，選擇 Roles (角色)。
3. 選擇 Create Role (建立角色)。
4. 在 Select trusted entity (選取受信任實體) 頁面，於 Trusted entity type (受信任實體的類型) 下，選擇 AWS service (服務)。
5. 在使用案例下，從其他 AWS 服務的使用案例中選擇 EC2，然後選擇 EC2-Spot 叢集標記。
6. 選擇 Next (下一步)。

7. 在 Add permissions (新增許可) 頁面上，選擇 Next (下一步)。
8. 在 Name, review, and create (名稱，檢閱和建立) 頁面上，針對 Role name (角色名稱) 輸入角色的名稱 (例如 **`aws-ec2-spot-fleet-tagging-role`**)。
9. 檢視頁面上的資訊，然後選擇 Create role (建立角色)。

預防跨服務混淆代理人

混淆代理人問題屬於安全性議題，其中沒有執行動作許可的實體可以強制具有更多權限的實體執行該動作。若要限制 Spot 機群為資源提供另一項服務的許可，我們建議在 `aws-ec2-spot-fleet-tagging-role` 信任政策中使用 `aws:SourceArn` 和 `aws:SourceAccount` 全域條件內容金鑰。

將 aws: SourceArn 和 aws: SourceAccount 條件金鑰新增至**`aws-ec2-spot-fleet-tagging-role`**信任政策

1. 在開啟IAM主控台https://console.aws.amazon.com/iam/。
2. 在導覽窗格中，選擇角色。
3. 尋找您之前建立的 `aws-ec2-spot-fleet-tagging-role` 並選擇連結 (而非核取方塊)。
4. 在 Summary (摘要) 下，選擇 Trust relationships (信任關係) 索引標籤，然後選擇 Edit trust policy (編輯信任政策)。
5. 在JSON語句中，添加一個包含您`aws:SourceAccount`和`aws:SourceArn`全局條件上下文鍵的`Condition`元素，以防止混淆的副問題，如下所示：

```
"Condition": {
      "ArnLike": {
        "aws:SourceArn": "arn:aws:ec2:us-east-1:account_id:spot-fleet-request/sfr-
*"
      },
      "StringEquals": {
        "aws:SourceAccount": "account_id"
      }
```

> **Note**
>
> 如果 `aws:SourceArn` 值包含帳戶 ID，且同時使用這兩個全域條件內容金鑰，則在相同政策陳述式中使用 `aws:SourceAccount` 值和 `aws:SourceArn` 值中的帳戶時，必須使用相同的帳戶 ID。

最終信任政策如下所示：

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Sid": "ConfusedDeputyPreventionExamplePolicy",
    "Effect": "Allow",
    "Principal": {
      "Service": "spotfleet.amazonaws.com"
    },
    "Action": "sts:AssumeRole",
    "Condition": {
      "ArnLike": {
        "aws:SourceArn": "arn:aws:ec2:us-east-1:account_id:spot-fleet-request/sfr-
*"
      },
      "StringEquals": {
        "aws:SourceAccount": "account_id"
      }
    }
  }
}
```

6. 選擇更新政策。

下表提供了 `aws:SourceArn` 的潛在值，以不同程度地限制 `aws-ec2-spot-fleet-tagging-role` 的範圍。

API操作	呼叫的服務	範圍	**aws:SourceArn**
RequestSpotFleet	AWS STS (AssumeRole)	將AssumeRole 功能限制aws-ec2-spot-fleet-tagging-role spot-fleet-requests 在指定帳戶中。	arn:aws:ec2:*: *123456789012* :spot-fleet-request/sfr-*

API操作	呼叫的服務	範圍	aws:SourceArn
RequestSpotFleet	AWS STS (AssumeRole)	aws-ec2-spot-fleet-tagging-role 將AssumeRole 功能限制spot-fleet-requests 在指定帳戶和指定區域中。請注意，此角色在其他區域中無法使用。	arn:aws:ec2: *us-east-1* :*123456789012* :spot-fleet-request/sfr-*
RequestSpotFleet	AWS STS (AssumeRole)	將 aws-ec2-spot-fleet-tagging-role 上的 AssumeRole 功能限制為僅限影響機群 sfr-11111111-1111-1111-1111-111111111111 的動作。請注意，此角色可能無法用於其他 Spot 機群。此外，此角色無法用於透過request-spot-fleet啟動任何新的 Spot 艦隊。	arn:aws:ec2: *us-east-1* :*123456789012* :spot-fleet-request/sfr-*11111111-1111-1111-1111-111111111111*

快速建立 Spot Fleet 請求 (主控台)

依照這些步驟快速建立 Spot Fleet 請求。

若要使用建議的設定建立 Spot Fleet 請求 (主控台)

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 如果您是第一次使用 Spot，將會顯示歡迎頁面；請選擇開始使用。否則，請選擇請求競價型執行個體。

4. 在 Launch parameters (啟動參數) 下，選擇 Manually configure launch parameters (手動設定啟動參數)。
5. 對於 AMI，選擇一個AMI.
6. 在 Target capacity (目標容量) 下， 針對 Total target capacity (總目標容量)，指定要請求的單位數目。對於單位類型，您可以選擇執行個體vCPUs、或記憶體 (GiB)。
7. 對於 Your fleet request at a glance (您的機群請求一覽)，檢閱您的機群組態，然後選擇 Launch (啟動)。

使用已定義的參數建立 Spot Fleet 請求 (主控台)

您可以使用您定義的參數建立 Spot Fleet。

若要使用已定義的參數建立 Spot Fleet 請求 (主控台)

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 如果您是第一次使用 Spot，將會顯示歡迎頁面；請選擇開始使用。否則，請選擇請求競價型執行個體。
4. 對於 Launch 參數，您可以手動設定啟動參數，也可以使用啟動範本，如下所示：

   a. [手動設定] 若要在 Amazon EC2 主控台中定義啟動參數，請選擇「手動設定啟動參數」，然後執行下列操作：

      i. 對於 AMI，請選擇AMIs提供的其中一個基本內容 AWS，或選擇 [搜尋] 以使用AMIAMI來自我們的使用者社群 AWS Marketplace、或您自己的使用者社群。

      > **Note**
      > 如果啟動參數中AMI指定的已取消註冊或停用，則無法從中啟動新實例。AMI對於設定為維持目標容量的車隊，將不會維持目標容量。

      ii. (選用) 針對金鑰對名稱，選擇現有的金鑰對或建立新的金鑰對。

      [現有的金鑰對] 選擇金鑰對。

      [新 key pair] 選擇 [建立新 key pair] 以前往 [金鑰配對] 頁面。當您完成時，請返回 Spot Requests (Spot 請求) 頁面並重新整理清單。

      iii. (選用) 展開 Additional launch parameters (其他啟動參數)，並執行下列動作：

A. (選用) 若要啟用 Amazon EBS最佳EBS化，對於最佳化，請選取啟動優化執EBS行個體。

B. (選用) 若要新增執行個體的暫時區塊層級儲存，請針對執行個體存放區，選擇連接啟動。

C. (選擇性) 若要新增儲存，請選擇 [新增磁碟區]，然後根據執行個體類型指定其他執行個體存放EBS磁碟區或 Amazon 磁碟區。

D. (選用) 執行個體預設為啟用基本監控。若要啟用詳細監視，對於監視，請選取啟用 CloudWatch 詳細監視。

E. (選用) 若要執行專用 Spot 執行個體，對於租用，請選擇 Dedicated - run a dedicated instance (專用 - 執行專用執行個體)。

F. (選用) 針對安全群組，選擇一或多個安全群組或建立新的安全群組。

   [現有的安全群組] 選擇一或多個安全群組。

   [新的安全群組] 選擇 Create new security group (建立新安全群組) 以前往 Security Groups (安全群組) 頁面。當您完成時，請返回 Spot Requests (Spot 請求) 並重新整理清單。

G. (選擇性) 若要讓您的執行個體可從網際網路存取，請針對「自動指派IPv4公用 IP」選擇「啟用」。

H. (選擇性) 若要使用IAM角色啟動 Spot 執行個體，IAM例如設定檔，請選擇角色。

I. (選用) 若要執行啟動指令碼，請將指令碼複製到使用者資料。

J. (選用) 若要新增標籤，請選擇 Create tag (建立標籤) 並輸入標籤的鍵和值，然後選擇 Create (建立)。針對每個標籤，重複執行上述動作。

   對於每個標籤，若要使用相同標籤標記執行個體和 Spot Fleet 請求，請確定同時選取 Instances (執行個體) 和 Fleet (機群)。若只要標記由機群啟動的執行個體，請清除 Fleet (機群)。若只要標記 Spot Fleet 請求，請清除 Instances (執行個體)。

b. [啟動範本] 若要使用您在啟動範本中建立的組態，請選擇 [使用啟動範本]，然後針對 Launch 範本選擇啟動範本。

> **Note**
>
> 如果您想要 Spot 叢集中的隨需容量，則必須指定啟動範本。

5. 針對其他請求詳細資訊，執行下列操作：

a. 檢閱其他的請求詳細資訊。若要進行變更，請清除套用預設值。

b. (選擇性) 對於IAM叢集角色，您可以使用預設角色或選擇其他角色。若要在變更角色後使用預設角色，請選擇使用預設角色。

c. (選用) 若要建立僅在特定期間內有效的請求，請編輯請求有效性起始日期和請求有效性結束日期。

d. (選用) 根據預設，Amazon 會在 Spot 叢集請求到期時EC2終止您的競價型執行個體。若要在請求過期後繼續執行這些執行個體，請清除在請求過期時終止執行個體。

e. (選用) 若要登錄 競價型執行個體與負載平衡器，請選擇從一個或多個負載平衡器接收流量，然後選擇一個或多個 Classic Load Balancer 或目標群組。

6. 針對 Target capacity (目標容量)，執行下列動作：

a. 針對 Total target capacity (總目標容量)，指定目標容量要請求的單位數目。對於單位類型，您可以選擇執行個體vCPUs、或記憶體 (MiB)。若要將目標容量指定為 0，以便稍後可以新增容量，您必須先選取維持目標容量。

b. (選用) 針對 Include On-Demand base capacity (包含隨需基礎容量)，指定要請求的隨需單位數量。數量必須小於總目標容量。Amazon EC2 計算差異，並將差異分配給要請求的現貨單位。

> **Important**
>
> 若要指定選用的隨需容量，您必須先選擇啟動範本。

c. (選擇性) Amazon 預設EC2會在 Spot 執行個體中斷時終止。若要維持目標容量，請選取維持目標容量。然後，您可以指定 Amazon 在 Spot 執行個體中斷時EC2終止、停止或休眠。若要這麼做，請在中斷行為中選擇對應的選項。

> **Note**
>
> 如果啟動參數中AMI指定的已取消註冊或停用，則無法從中啟動新實例。AMI在這種情況下，對於設定為維持目標容量的車隊，將不會維持目標容量。

d. (選用) 若要允許 Spot Fleet 在機群中為現有的 Spot 執行個體發出執行個體重新平衡通知時啟動替代 Spot 執行個體，請選取 Capacity rebalance (容量重新平衡)，然後選擇執行個體取代策略。如果您選擇在終止前啟動，請指定 Amazon EC2 終止舊執行個體之前的延遲時間 (以秒為單位)。如需詳細資訊，請參閱使用叢集和 S EC2 pot 叢集中的容量重新平衡來取代有風險的 Spot 執行個體。

e. (選用) 若要控制您為了在機群中所有的 Spot 執行個體所支付的每小時金額，請選取 Set maximum cost for Spot Instances (設定 Spot 執行個體的最大成本)，然後輸入您願意支付的每小時金額上限。達到總額上限時，Spot Fleet 會停止啟動 Spot 執行個體，即使它未符合目標容量亦然。如需詳細資訊，請參閱為您的EC2車隊或 Spot 艦隊設定消費上限。

7. 針對 Network (網路)，請執行下列動作：

   a. 在「網路」中，選擇現有的VPC或建立新的。

      [現有VPC] 選擇VPC.

      [新VPC] 選擇創建新的VPC去 Amazon VPC 控制台。完成後，請返回此畫面並重新整理清單。

   b. (選擇性) 對於可用區域，讓 Amazon 為您的競價型執行個體EC2選擇可用區域，或指定一或多個可用區域。

      如果您在可用區域中擁有一或多個子網，請從子網選擇適當的子網。若要新增子網路，請選擇「建立新子網路」以前往 Amazon 主VPC控台。完成後，返回此畫面並重新整理清單。

8. 對於執行個體類型要求，您可以指定執行個體屬性，並讓 Amazon 使用這些屬性EC2識別最佳執行個體類型，或者指定執行個體清單。如需詳細資訊，請參閱指定EC2叢集或 Spot 叢集的執行個體類型選取屬性。

   a. 如果選擇 Specify instance attributes that match your compute requirements (指定符合運算需求的執行個體屬性)，請指定您的執行個體屬性，如下所示：

      i. 對於 vCPUs，輸入所需的最小和最大數目vCPUs。若要指定無限制，請選取「無最小值」或「無最大值」，或兩者皆選。

      ii. 對於 Memory (GiB) (記憶體 (GiB))，輸入所需記憶體數量的下限和上限。若要指定無限制，請選取「無最小值」或「無最大值」，或兩者皆選。

      iii. (選擇性) 對於其他執行個體屬性，您可以選擇性地指定一或多個屬性，以更詳細地表示您的運算需求。每個額外屬性都會將進一步的限制新增至您的請求。您可以省略額外屬性；省略時，就會使用預設值。如需每個屬性及其預設值的說明，請參閱 Amazon 命EC2令列參考get-spot-placement-scores中的。

      iv. (選用) 若要檢視具有所指定屬性的執行個體類型，請展開 Preview matching instance types (預覽相符的執行個體類型)。若要排除執行個體類型，使其不在請求中使用，請選取執行個體，然後選擇 Exclude selected instance types (排除選取的執行個體類型)。

   b. 如果選擇 Manually select instance types (手動選取執行個體類型)，Spot Fleet 會提供預設的執行個體類型清單。若要選取更多執行個體類型，請選擇 Add instance types (新增執行個體

類型)、選取要在請求中使用的執行個體類型，然後選擇 Select (選取)。若要刪除執行個體類型，請選取執行個體類型並選擇 Delete (刪除)。

9. 對於配置策略，請選擇符合您需求的 Spot 配置策略和隨需配置策略。如需詳細資訊，請參閱使用配置策略來確定EC2叢集或 Spot 叢集如何滿足 Spot 和隨需容量。
10. 針對 Your fleet request at a glance (您的機群請求一覽)，檢閱您的機群配置，並視需要進行任何調整。
11. (選擇性) 若要下載啟動組態的複本以搭配使用 AWS CLI，請選擇 JSONconfig。
12. 當您準備好啟動 Spot 艦隊時，請選擇啟動。

    Spot 機群請求類型為 `fleet`。請求處理完成時，會新增類型為 `instance` 的請求，其狀態為 `active` 和 `fulfilled`。

使用建立競價型叢集 AWS CLI

若要使用 AWS CLI

使用命request-spot-fleet令建立 Spot 叢集請求。

```
aws ec2 request-spot-fleet --spot-fleet-request-config file://config.json
```

如需組態檔案範例，請參閱 範例CLI組態競價型叢集。

下列為範例輸出：

```
{
    "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE"
}
```

建立可取代狀態不良的 Spot 執行個體的 Spot 叢集

Spot Fleet 會每兩分鐘定期檢查機群中 Spot 執行個體的運作狀態。執行個體的運作狀態會是 `healthy` 或 `unhealthy`。

Spot 叢集使用 Amazon 提供的狀態檢查來判斷執行個體的運作狀態EC2。如果執行個體狀態檢查或系統狀態檢查等運作狀態檢查，出現連續三次為 `unhealthy`，則該執行個體即為 `impaired`。如需詳細資訊，請參閱 Amazon EC2 執行個體的狀態檢查。

您可以設定機群以取代運作狀態不佳的 競價型執行個體。啟用運作狀態檢查取代後，Spot 執行個體會在被報告為 `unhealthy` 時被取代。當取代一個狀態不佳的 Spot 執行個體時，機群可能會低於其目標容量長達幾分鐘。

要求

- 運作狀態檢查替代僅支援維持目標容量的 Spot Fleets (類型 `maintain` 的機群)，不支援一次性的 Spot Fleets (類型 `request` 的機群)。
- 運作狀態檢查取代僅支援 競價型執行個體。此功能不支援 隨需執行個體。
- 您可以設定 Spot Fleet，只在建立執行個體時取代狀況不良的執行個體。
- 使用者只有在擁有呼叫 `ec2:DescribeInstanceStatus` 動作的許可時，才能使用運作狀態檢查取代功能。

Console

若要使用主控台來設定 Spot Fleet 以取代運作狀態不佳的 Spot 執行個體

1. 請遵循在中建立競價型叢集的步驟使用已定義的參數建立 Spot Fleet 請求 (主控台)。
2. 若要設定叢集以取代運作 Health 態不良的 Spot 執行個體，請展開其他啟動參數，然後在健全狀態檢查下選取取代運作狀態不良的執 若要啟用此選項，首先必須選擇維護目標容量。

AWS CLI

若要使用 AWS CLI來設定 Spot Fleet 以取代運作狀態不佳的 Spot 執行個體

1. 請遵循在中建立競價型叢集的步驟使用建立競價型叢集 AWS CLI。
2. 若要設定機群以取代運作狀態不佳的 競價型執行個體，請在 `ReplaceUnhealthyInstances` 中輸入 `true`。

標記新的或現有的 Spot 叢集請求及其啟動的執行個體和磁碟區

為了協助對 Spot 叢集請求及其啟動的執行個體和磁碟區進行分類和管理，您可以使用自訂中繼資料對其進行標記。您可以在建立 Spot Fleet 請求時或之後將標記指派給請求。同樣地，您可以在叢集啟動或之後為執行個體和磁碟區指派標籤。

當您標記機群請求時，由機群啟動的執行個體和磁碟區不會自動加上標記。您需要明確標記由機群啟動的執行個體和磁碟區。您可以選擇只將標籤指派給叢集請求，或僅指派給叢集啟動的執行個體，或僅指派給叢集啟動的執行個體所連接的磁碟區，或指派給所有執行個體的磁碟區。

> **Note**
>
> 您只能標記連接到隨需執行個體的磁碟區。您無法為連接至 競價型執行個體 的磁碟區加標籤。

您可以使用 Amazon EC2 主控台或命令列工具指派標籤。

如需標籤運作方式的詳細資訊，請參閱標記您的 Amazon EC2 資源。

目錄

- 必要條件
- 標記新的 Spot Fleet 以及執行個體和其啟動的磁碟區
- 標記現有 Spot Fleet
- 檢視 Spot Fleet 請求標籤

必要條件

授予使用者標記資源的許可。如需詳細資訊，請參閱範例：標籤資源。

授予使用者標記資源的許可

建立包含下列項目的IAM策略：

- `ec2:CreateTags` 動作。這會授予使用者建立標籤的許可。
- `ec2:RequestSpotFleet` 動作。這會授予使用者建立 Spot 機群請求的許可。
- 對於 `Resource`，您必須指定 `"*"`。這可讓使用者標記所有資源類型。

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

```
            "Action": [
                "ec2:CreateTags",
                "ec2:RequestSpotFleet"
            ],
            "Resource": "*"
        }
    ]
}
```

> **Important**
>
> 我們目前不支援 `spot-fleet-request` 資源的資源層級許可。如果您指定 `spot-fleet-request` 作為資源，當您嘗試標記機群時，會收到未經授權的例外狀況。下列範例說明如何不要設定政策。
>
> ```
> {
>  "Effect": "Allow",
>  "Action": [
>  "ec2:CreateTags",
>  "ec2:RequestSpotFleet"
>  ],
>  "Resource": "arn:aws:ec2:us-east-1:111122223333:spot-fleet-request/*"
> }
> ```

若要提供存取權，請新增權限至您的使用者、群組或角色：

- 使用者和群組位於 AWS IAM Identity Center：

  建立權限合集。請按照 AWS IAM Identity Center 使用者指南 中的 建立權限合集 說明進行操作。

- IAM透過身分識別提供者管理的使用者：

  建立聯合身分的角色。請遵循《使用指南》中的〈為第三方身分識別提供IAM者 (同盟) 建立角色〉中的指示進行。

- IAM使用者：
  - 建立您的使用者可擔任的角色。請按照《用戶指南》中的「為IAM用戶創建角色」中的IAM說明進行操作。
  - (不建議) 將政策直接附加至使用者，或將使用者新增至使用者群組。遵循《使用者指南》中的「向使用者 (主控台) 新增權限」IAM 中的指示進行。

標記新的 Spot Fleet 以及執行個體和其啟動的磁碟區

標記新的 Spot 叢集請求及其使用主控台啟動的執行個體和磁碟區

1. 遵循使用已定義的參數建立 Spot Fleet 請求 (主控台)程序。
2. 新增標籤的方式取決於您是手動設定叢集還是使用啟動範本。

   - 如果您手動設定叢集，請執行下列動作：

     若要新增標籤，請展開其他啟動參數，選擇建立標籤，然後輸入標籤的機碼和值。針對每個標籤，重複執行上述動作。

     對於每個標籤，您可以標記 Spot Fleet 請求和具有相同標籤的執行個體。若要標記兩者，請確定同時選取執行個體和叢集。若只要標記 Spot Fleet 請求，請清除 Instances (執行個體)。若只要標記由機群啟動的執行個體，請清除 Fleet (機群)。

     > **Note**
     > 手動設定叢集時，沒有標記磁碟區的選項。磁碟區標籤僅支援連接至 隨需執行個體 的磁碟區。手動設定叢集時，無法指定隨需執行個體。

   - 如果您使用啟動範本，請執行下列動作：

     若要將標記新增至叢集要求，請在 [標籤] 底下選擇 [建立標籤]，然後輸入標籤的金鑰和值。針對每個標籤，重複執行上述動作。

     若要標記叢集中的資源，您必須在啟動範本中指定標籤。

標記新的 Spot 叢集請求及其啟動的執行個體和磁碟區 AWS CLI

若要在建立 Spot Fleet 請求時加上標記，並在機群啟動執行個體和磁碟區時給它們加上標記，請依下列方式設定 Spot Fleet 請求組態：

Spot Fleet 請求標籤：

- 在 `SpotFleetRequestConfig` 中指定 Spot Fleet 請求的標籤。
- 對於 `ResourceType`，請指定 `spot-fleet-request`。若指定其他值，機群請求將會失敗。
- 對於 `Tags`，指定金鑰值對。您可以指定多個金鑰值對。

執行個體標籤：

- 在 `LaunchSpecifications` 中指定執行個體的標籤。
- 對於 `ResourceType`，請指定 `instance`。若指定其他值，機群請求將會失敗。
- 對於 `Tags`，指定金鑰值對。您可以指定多個金鑰值對。

  或者，您可以在 Spot Fleet 請求中參照的啟動範本中，指定執行個體的標籤。

磁碟區標籤：

- 在 Spot Fleet 請求中參照的啟動範本中，指定磁碟區的標籤。不支援在 `LaunchSpecifications` 中標記磁碟區。

在下面的例子中，Spot Fleet 請求標記了兩個標籤：Key=Environment and Value=Production 和 Key=Cost-Center and Value=123。由機群啟動的執行個體標記了一個標籤 (這與 Spot Fleet 請求的標籤之一相同)：Key=Cost-Center and Value=123。

```
{
    "SpotFleetRequestConfig": {
        "AllocationStrategy": "priceCapacityOptimized",
        "ExcessCapacityTerminationPolicy": "default",
        "IamFleetRole": "arn:aws:iam::111122223333:role/aws-ec2-spot-fleet-tagging-
role",
        "LaunchSpecifications": [
            {
                "ImageId": "ami-0123456789EXAMPLE",
                "InstanceType": "c4.large",
                "TagSpecifications": [
                    {
                        "ResourceType": "instance",
                        "Tags": [
                            {
                                "Key": "Cost-Center",
                                "Value": "123"
                            }
                        ]
                    }
                ]
            }
        ],
```

```
        "SpotPrice": "5",
        "TargetCapacity": 2,
        "TerminateInstancesWithExpiration": true,
        "Type": "maintain",
        "ReplaceUnhealthyInstances": true,
        "InstanceInterruptionBehavior": "terminate",
        "InstancePoolsToUseCount": 1,
        "TagSpecifications": [
            {
                "ResourceType": "spot-fleet-request",
                "Tags": [
                    {
                        "Key": "Environment",
                        "Value":"Production"
                    },
                    {
                        "Key": "Cost-Center",
                        "Value":"123"
                    }
                ]
            }
        ]
    }
}
```

若要使用標記 Spot 叢集啟動的執行個體 AWS CLI

若要在機群啟動執行個體時加上標記，您可以在 Spot Fleet 請求中參照的啟動範本中指定標籤，或者您可以指定 Spot Fleet 請求組態中的標籤，如下所示：

- 在 `LaunchSpecifications` 中指定執行個體的標籤。
- 對於 `ResourceType`，請指定 `instance`。若指定其他值，機群請求將會失敗。
- 對於 `Tags`，指定金鑰值對。您可以指定多個金鑰值對。

在下列範例中，機群啟動的執行個體均標記了一個標籤：Key=Cost-Center and Value=123。

```
{
    "SpotFleetRequestConfig": {
        "AllocationStrategy": "priceCapacityOptimized",
        "ExcessCapacityTerminationPolicy": "default",
        "IamFleetRole": "arn:aws:iam::111122223333:role/aws-ec2-spot-fleet-tagging-
role",
```

```
        "LaunchSpecifications": [
            {
                "ImageId": "ami-0123456789EXAMPLE",
                "InstanceType": "c4.large",
                "TagSpecifications": [
                    {
                        "ResourceType": "instance",
                        "Tags": [
                            {
                                "Key": "Cost-Center",
                                "Value": "123"
                            }
                        ]
                    }
                ]
            }
        ],
        "SpotPrice": "5",
        "TargetCapacity": 2,
        "TerminateInstancesWithExpiration": true,
        "Type": "maintain",
        "ReplaceUnhealthyInstances": true,
        "InstanceInterruptionBehavior": "terminate",
        "InstancePoolsToUseCount": 1
    }
}
```

若要使用標記 Spot 叢集啟動的隨需執行個體連接的磁碟區 AWS CLI

若要在機群建立磁碟區時標記磁碟區，請在 Spot Fleet 請求中參考的啟動範本中指定標籤。

> **Note**
>
> 磁碟區標籤僅支援連接至 隨需執行個體 的磁碟區。您無法為連接至 競價型執行個體 的磁碟區加標籤。
> 不支援在 `LaunchSpecifications` 中標記磁碟區。

標記現有 Spot Fleet

若要使用主控台標記現有的 Spot Fleet 請求

建立 Spot 叢集請求後，您可以使用主控台將標籤新增至叢集請求。

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選取您的 Spot Fleet 請求。
4. 選擇標籤索引標籤，然後選擇建立標籤。

若要使用標記現有的 Spot 叢集請求 AWS CLI

使用 create-tags 命令來標記現有資源。在下面的例子中，現有的 Spot Fleet 請求會使用 Key=purpose 和 Value=test 標籤。

```
aws ec2 create-tags \
    --resources sfr-11112222-3333-4444-5555-66666EXAMPLE \
    --tags Key=purpose,Value=test
```

檢視 Spot Fleet 請求標籤

若要使用主控台檢視 Spot Fleet 請求標籤

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選取您的 Spot Fleet 請求並選擇 Tags (標籤)。

若要描述 Spot Fleet 請求標籤

使用 describe-tag 命令檢視指定資源的標籤。在下面的範例中，會描述指定 Spot Fleet 請求的標籤。

```
aws ec2 describe-tags \
    --filters "Name=resource-id,Values=sfr-11112222-3333-4444-5555-66666EXAMPLE"
```

```
{
    "Tags": [
        {
            "Key": "Environment",
            "ResourceId": "sfr-11112222-3333-4444-5555-66666EXAMPLE",
            "ResourceType": "spot-fleet-request",
            "Value": "Production"
        },
        {
            "Key": "Another key",
```

```
            "ResourceId": "sfr-11112222-3333-4444-5555-66666EXAMPLE",
            "ResourceType": "spot-fleet-request",
            "Value": "Another value"
        }
    ]
}
```

您也可以透過描述 Spot Fleet 請求來檢視 Spot Fleet 請求的標籤。

使用此describe-spot-fleet-requests命令檢視指定的 Spot 叢集要求的組態，其中包括針對叢集要求指定的任何標籤。

```
aws ec2 describe-spot-fleet-requests \
    --spot-fleet-request-ids sfr-11112222-3333-4444-5555-66666EXAMPLE
```

```
{
    "SpotFleetRequestConfigs": [
        {
            "ActivityStatus": "fulfilled",
            "CreateTime": "2020-02-13T02:49:19.709Z",
            "SpotFleetRequestConfig": {
                "AllocationStrategy": "capacityOptimized",
                "OnDemandAllocationStrategy": "lowestPrice",
                "ExcessCapacityTerminationPolicy": "Default",
                "FulfilledCapacity": 2.0,
                "OnDemandFulfilledCapacity": 0.0,
                "IamFleetRole": "arn:aws:iam::111122223333:role/aws-ec2-spot-fleet-
tagging-role",
                "LaunchSpecifications": [
                    {
                        "ImageId": "ami-0123456789EXAMPLE",
                        "InstanceType": "c4.large"
                    }
                ],
                "TargetCapacity": 2,
                "OnDemandTargetCapacity": 0,
                "Type": "maintain",
                "ReplaceUnhealthyInstances": false,
                "InstanceInterruptionBehavior": "terminate"
            },
            "SpotFleetRequestId": "sfr-11112222-3333-4444-5555-66666EXAMPLE",
            "SpotFleetRequestState": "active",
            "Tags": [
```

```
                {
                    "Key": "Environment",
                    "Value": "Production"
                },
                {
                    "Key": "Another key",
                    "Value": "Another value"
                }
            ]
        }
    ]
}
```

描述 Spot 叢集設定、其執行個體和事件歷史記錄

您可以說明 Spot 叢集組態、Spot 叢集中的執行個體，以及 Spot 叢集的事件歷史記錄。

如要說明您的 Spot 機群 (主控台)

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選取您的 Spot Fleet 請求。識別碼以 sfr- 開頭。若要查看組態的詳細資訊，請選擇描述。
4. 若要列出 Spot Fleet 的 Spot 執行個體，請選擇 Instances (執行個體)。
5. 若要檢視 Spot Fleet 的歷史記錄，請選擇 History (歷史記錄)。

如要說明您的 Spot 機群 (AWS CLI)

使用命describe-spot-fleet-requests令描述您的 Spot 叢集請求。

```
aws ec2 describe-spot-fleet-requests
```

使用命describe-spot-fleet-instances令描述指定 Spot 叢集的 Spot 執行個體。

```
aws ec2 describe-spot-fleet-instances \
    --spot-fleet-request-id sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE
```

使用 describe-spot-fleet-request-history 命令來描述指定競價型叢集請求的事件歷史記錄。

```
aws ec2 describe-spot-fleet-request-history \
    --spot-fleet-request-id sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
```

```
    --start-time 2015-05-18T00:00:00Z
```

修改 Spot Fleet 請求

您可以修改作用中的 Spot Fleet 請求，來完成下列任務：

- 增加總目標容量和隨需部分
- 減少總目標容量和隨需部分

> **Note**
> 您無法修改一次性的 Spot Fleet 請求。當您建立 Spot Fleet 請求時，如果已選取 Maintain target capacity (維護目標容量)，則只能修改 Spot Fleet 請求。

當您增加總目標容量時，Spot 叢集會啟動額外的 Spot 執行個體。當您增加隨需部分時，Spot Fleet 會啟動額外的隨需執行個體。

當您增加目標容量總計時，Spot 叢集會根據其 Spot 叢集請求的配置策略啟動額外的 Spot 執行個體。

當您減少總目標容量時，Spot 叢集會取消任何超出新目標容量的開啟請求。您可以請求 Spot Fleet 終止 Spot 執行個體，直到機群大小達到新的目標容量。如果分配策略為 `diversified`，則 Spot Fleet 會在集區中終止執行個體。或者，您可以請求該 Spot Fleet 將機群保持為目前的大小，但不能取代任何中斷的或您手動終止的 Spot 執行個體。

當 Spot Fleet 因為目標容量減少而終止執行個體時，執行個體會收到 Spot 執行個體中斷通知。

若要修改 Spot Fleet 請求 (主控台)

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選取您的 Spot Fleet 請求。
4. 選擇動作和修改目標容量。
5. 在修改目標容量中，執行下列動作：
   a. 輸入新的目標容量和隨需部分。
   b. (選用) 如果您正在減少目標容量，但希望叢集保持目前的大小，請清除終止執行個體。
   c. 選擇提交。

若要使用修改 Spot 叢集請求 AWS CLI

使用命modify-spot-fleet-request令更新指定 Spot 叢集請求的目標容量。

```
aws ec2 modify-spot-fleet-request \
    --spot-fleet-request-id sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
    --target-capacity 20
```

如果要減少所指定 Spot Fleet 的目標容量，而不會因此終止任何 Spot 執行個體，您可以如下修改先前的命令：

```
aws ec2 modify-spot-fleet-request \
    --spot-fleet-request-id sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
    --target-capacity 10 \
    --excess-capacity-termination-policy NoTermination
```

取消 (刪除) Spot 叢集請求

如果您不再需要 Spot 叢集，您可以取消 Spot 叢集請求，這會刪除該請求。取消機群請求後，也會取消與機群相關的所有 Spot 請求，因此不會啟動新的 Spot 執行個體。

刪除時 Spot 機群請求時，必須指定是否也要終止其全部執行個體。這包含隨需執行個體和 Spot 執行個體。

如果指定在刪除機群請求時必須終止執行個體，則機群請求會進入 `cancelled_terminating` 狀態。否則會進入 `cancelled_running` 狀態，且執行個體會一直執行直到中斷或您手動終止。

限制

- 您可以在單一要求中取消最多 100 個叢集。如果超過指定的數目，則不會取消任何叢集。

若要取消 (刪除) Spot 叢集請求 (主控台)

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選取您的 Spot Fleet 請求。
4. 選擇動作、取消請求。
5. 在取消 Spot 請求對話方塊中，執行下列操作：

a. 若要在取消 Spot 機群請求的同時終止關聯的執行個體，請將終止執行個體核取方塊保持為選取狀態。若要取消 Spot 機群請求，而不終止關聯的執行個體，請清除終止執行個體核取方塊。

b. 選擇確認。

若要取消 (刪除) Spot 叢集請求並使用 AWS CLI

使用cancel-spot-fleet-requests命令取消指定的 Spot 叢集請求，並終止其隨需執行個體和 Spot 執行個體。

```
aws ec2 cancel-spot-fleet-requests \
    --spot-fleet-request-ids sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
    --terminate-instances
```

範例輸出

```
{
    "SuccessfulFleetRequests": [
        {
            "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",
            "CurrentSpotFleetRequestState": "cancelled_terminating",
            "PreviousSpotFleetRequestState": "active"
        }
    ],
    "UnsuccessfulFleetRequests": []
}
```

若要取消 (刪除) Spot 叢集請求，而不使用 AWS CLI

您可以使用 `--no-terminate-instances` 參數來修改前一個命令，以取消指定的 Spot 機群請求，而無需終止其隨需執行個體和 Spot 執行個體。

```
aws ec2 cancel-spot-fleet-requests \
    --spot-fleet-request-ids sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
    --no-terminate-instances
```

範例輸出

```
{
```

```
    "SuccessfulFleetRequests": [
        {
            "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",
            "CurrentSpotFleetRequestState": "cancelled_running",
            "PreviousSpotFleetRequestState": "active"
        }
    ],
    "UnsuccessfulFleetRequests": []
}
```

瞭解 Spot 叢集的自動擴充

自動擴展可讓您的 Spot 叢集根據需求增加或減少其目標容量。透過自動擴展，Spot Fleet 可以在指定範圍內啟動執行個體 (向外擴充) 或終止執行個體 (擴充)，以回應一或多個擴展政策。

結合 Amazon EC2、Amazon CloudWatch 和應用程式自動擴展功能，可實現 Spot Fleet 的自動擴展 APIs。Spot 叢集請求是使用 Amazon 建立的EC2，使用建立警示 CloudWatch，並使用應用程式自動擴展建立擴展政策。

自動調整比例的類型

Spot Fleet 支援以下類型的自動擴展：

- 目標追蹤縮放 — 增加或減少 以特定量度的值為目標，以目標為叢集的目前容量。這與恆溫器維持家中溫度的方式類似 — 您可以選擇所需的溫度，恆溫器會執行其餘工作。
- 步進擴展 – 根據一組擴展調整 (也稱為步進調整)，來增加或減少機群目前的容量，其會根據警示違規的程度而有不同。
- 排程的擴展 – 根據日期和時間，來增加或減少機群目前的容量。

考量事項

使用 Spot 叢集的自動調整規模時，請考慮下列事項：

- 執行個體加權 — 如果您使用執行個體加權，請記住，Spot 叢集可以視需要超過目標容量。已實現的容量可以是浮點數，但目標容量必須是整數，因此 Spot Fleet 會四捨五入為下一個整數。如果檢視警示觸發時擴展政策的結果，您必須將這些行為列入考量。例如，假設目標容量為 30、已實現的容量為 30.1，則規模調整政策會減掉 1。當警示觸發時，自動擴展程序會從 30.1 減掉 1 而得出 29.1，然後再四捨五入為 30，因此不需採取規模調整動作。在另一個範例中，假設選擇了 2、4 和 8 的執行個體權重，而目標容量為 10，但沒有可用的權重 2 執行個體，因此 Spot Fleet 針對已實現的容量

12 佈建權重為 4 和 8 的執行個體。如果規模調整政策將目標容量減少了 20%，而且警示已觸發，則自動擴展程序會從 12 減掉 12*0.2 而得出 9.6，然後再四捨五入為 10，因此不需採取規模調整動作。

- 冷卻時間-您為 Spot 艦隊建立的擴展政策支援一段冷卻時間。這是在擴展活動完成之後，先前的觸發相關擴展活動還能影響未來擴展活動的秒數。如果使用向外擴展政策，則在冷卻時間仍有效時，由先前起始冷卻時間的向外擴展事件所新增的容量，會在計算時列入下次向外擴展所需容量的一部分。這種做法的目的是連續的向外擴展 (但並非過度)。如果使用向內擴展政策，在冷卻時間尚未到期之前，會利用此期間來封鎖請求中後續的擴展。這種做法的目的是保守地進行擴展，以保障應用程式的可用性。不過，如果在向內擴展後的冷卻時間內，有另一個警示觸發了向外擴展政策，則自動擴展功能會立即擴展可擴展的目標。
- 使用詳細監控 — 我們建議您以 1 分鐘的頻率根據執行個體指標進行擴展，因為這樣可以更快地回應使用率變更。5 分鐘頻率的擴展指標可能會導致較慢的回應時間和用到過時的指標資料進行擴展。若要 CloudWatch在 1 分鐘內將執行個體的指標資料傳送至，您必須特別啟用詳細監控。如需詳細資訊，請參閱 管理EC2執行個體的詳細監控 和 使用已定義的參數建立 Spot Fleet 請求 (主控台)。
- AWS CLI— 如果您使用用 AWS CLI 於設定 Spot 叢集的擴展，您將使用應用程式自動CLI調整規模。如需詳細資訊，請參閱下列資源：
  - AWS CLI 命令參考中的應用程式自動擴展一節
  - Application Auto Scaling 放API參
  - Application Auto Scaling 使用者指南

IAMSpot 叢集自動擴充所需的權限

結合 Amazon EC2、Amazon CloudWatch 和應用程式自動擴展功能，可實現 Spot Fleet 的自動擴展 APIs。Spot 叢集請求是使用 Amazon 建立的EC2，使用建立警示 CloudWatch，並使用應用程式自動擴展建立擴展政策。除了使用 Spot Fleet 和 Amazon 所需的IAM許可之外EC2，存取叢集擴展設定的使用者還必須擁有支援自動擴展的服務的適當許可。

若要針對 Spot Fleet 使用自動調整規模，使用者必須擁有使用下列範例原則中顯示之動作的權限。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "application-autoscaling:*",
                "ec2:DescribeSpotFleetRequests",
```

```
                "ec2:ModifySpotFleetRequest",
                "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": "*"
        }
    ]
}
```

您也可以建立自己的IAM原則，允許對 Application Auto Scaling API 的呼叫提供更精細的權限。如需詳細資訊，請參閱應用程式 Auto Scaling 使用者指南中的應用 Application Auto Scaling 的身分識別與存取

應用程式 Auto Scaling 服務還需要權限來描述您的競價型叢集和 CloudWatch 警報，以及代表您修改 Spot 叢集目標容量的權限。如果您為 Spot Fleet 啟用自動擴展，則會建立名為 `AWSServiceRoleForApplicationAutoScaling_EC2SpotFleetRequest` 的服務連結角色。此服務連結角色會授與 Application Auto Scaling 許可以描述政策的警示、監控機群目前的容量，以及修改機群的容量。Application Auto Scaling 的原始受管 Spot Fleet 角色為 `aws-ec2-spot-fleet-autoscale-role`，但已不再需要。服務連結的角色預設為 Application Auto Scaling。如需詳細資訊，請參閱《Application Auto Scaling 使用者指南》中的適用於 Application Auto Scaling 的服務連結角色。

目標追蹤擴展：透過定位特定量度的值來擴展 Spot 叢集

使用目標追蹤擴展，您可以透過選取指標並設定目標值來建立目標追蹤擴展政策。然後，Spot Fleet 會建立和管理觸發擴展政策的CloudWatch 警示，並根據所選指標和目標值計算擴展調整。資源調整政策會視需要新增或移除執行個體來調整容量，使指標保持在或接近指定的目標值。目標追蹤原則不僅可以讓測量結果接近目標值，還可以調整因負載模式波動而導致的測量結果波動，並將快速容量波動降到最低。

如果每個政策使用不同的指標，您可以為 Spot 叢集建立多個目標追蹤擴展政策。叢集會根據指定最大叢集容量的原則進行擴展。這可讓您涵蓋多種案例，以確保應用程式工作負載有足夠的容量。

為了確保應用程式的可用性，機群可以配合指標按比例快速地擴展規模，但是以更漸進的方式縮減規模。

當 Spot 叢集因目標容量減少而終止 Spot 執行個體時，執行個體會收到 Spot 執行個體中斷通知。

> **Note**
>
> 請勿編輯或刪除 Spot 叢集針對目標追蹤擴展政策管理的 CloudWatch 警示。當您刪除目標追蹤擴展政策時，Spot Fleet 會自動刪除警示。

必要條件

- Spot 機群請求必須具有 `maintain` 的請求類型。類型 `request` 的請求不支援自動擴展功能。
- 設定 IAMSpot 叢集自動擴充所需的權限。
- 請參閱考量事項。

設定目標追蹤政策 (主控台)

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選取您的 Spot Fleet 請求。
4. 選擇螢幕底部附近的「Auto Scaling」標籤。如果您選取 Spot 叢集的連結，則沒有索引標籤；而是向下捲動至「Auto Scaling」區段。
5. 如果未設定自動擴展，請選擇設定。
6. 利用容量擴展的範圍來設定機群的最低和最高容量。自動擴展功能不會將機群的規模擴展到小於最低容量或大於最高容量。
7. 針對 Policy Name (政策名稱)，請輸入政策的名稱。
8. 選擇目標指標。
9. 輸入指標的目標值。
10. 針對冷卻時間，請指定一個新值 (以秒為單位) 或保留預設值。
11. (選擇性) 若要省略根據目前的組態建立縮放原則，請選取停用擴充。您可以使用不同的組態來建立向內擴展政策。

12. 選擇 Save (儲存)。

使用設定目標追蹤原則 AWS CLI

1. 使用register-scalable-target命令將 Spot 叢集請求註冊為可擴展的目標。
2. 使用put-scaling-policy命令建立擴展政策。

步驟擴展：使用步驟擴展政策擴展 Spot 叢集

使用步驟擴展政策，您可以指定 CloudWatch 警示以觸發擴展程序。例如，如果您想要在使用CPU率達到特定等級時向外擴充，請使用 Amazon 提供的`CPUUtilization`指標建立警示EC2。

建立步進規模調整政策時，您必須指定下列規模調整類型的其中一種：

- 新增 – 根據指定的容量單位數量或指定的目前容量百分比，來增加機群的目標容量。
- 移除 – 根據指定的容量單位數量或指定的目前容量百分比，來減少機群的目標容量。
- 設定為 – 將機群的目標容量設定為指定的容量單位數量。

警示觸發時，自動擴展程序會利用已實現的容量和規模調整政策，來計算出新的目標容量，然後據以更新目標容量。例如，假設目標容量為和已實現的容量為 10，而且規模調整政策會增加 1。當警示觸發時，自動擴展程序會將 10 加上 1 而得出 11，因此 Spot Fleet 會啟動 1 個執行個體。

當 Spot 叢集因目標容量減少而終止 Spot 執行個體時，執行個體會收到 Spot 執行個體中斷通知。

必要條件

- Spot 機群請求必須具有 `maintain` 的請求類型。類型 `request` 的請求不支援自動擴展功能。
- 設定 IAMSpot 叢集自動擴充所需的權限。
- 請考慮哪些 CloudWatch 指標對您的應用程式很重要。您可以根據提供的指標 AWS 或您自己的自訂指標建立 CloudWatch 警示。
- 對於您將在資源調 AWS 度政策中使用的指標，如果提供CloudWatch 指標的服務預設未啟用指標，請啟用指標收集。
- 請參閱考量事項。

建立 CloudWatch 鬧鐘

1. 在開啟 CloudWatch 主控台https://console.aws.amazon.com/cloudwatch/。

2. 在功能窗格中，展開 [警報]，然後選擇 [所有鬧鐘]。
3. 選擇建立警示。
4. 在指定指標和條件頁面上，選擇選取指標。
5. 選擇EC2競價型，然後選擇叢集要求量度，然後選取量度 (例如，TargetCapacity)，然後選擇選取量度。

   指定指標和條件頁面隨即出現，顯示您選取指標的圖形及其他資訊。
6. 對於「週期」，選擇警示的評估期間，例如 1 分鐘。評估警示時，每個期間都會彙整為一個資料點。

   > **Note**
   >
   > 期間越短會建立更敏感的警示。

7. 針對條件，透過定義閾值條件來定義警示。例如，您可以定義一個閾值，設定每當指標值大於或等於 80% 時，即會觸發警示。
8. 在其他組態下，對於要發出警示的資料點，指定ALARM狀態必須處於觸發警示的資料點 (評估週期) 數目，例如 1 個評估週期或 3 個評估期中的 2 個。如果連續多個週期違反，這將創建一個警報，該警報將進入ALARM狀態。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的評估警示。
9. 對於遺失資料處理，選擇其中之一個選項 (或保留預設的 將遺失資料視為遺失)。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的設定 CloudWatch 警示如何處理遺失的資料。
10. 選擇 Next (下一步)。
11. （可選）要接收擴展事件的通知，對於通知，您可以選擇或建立要用於接收通知的 Amazon SNS 主題。否則，您可以立即刪除通知，並根據需要在之後新增通知。
12. 選擇下一步。
13. 在 [新增名稱和說明] 底下，輸入警示的名稱和說明，然後選擇 [下一步]。
14. 選擇建立警示。

為您的 Spot Fleet 設定步進擴展政策 (主控台)

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選取您的 Spot Fleet 請求。

4. 選擇螢幕底部附近的「Auto Scaling」標籤。如果您選取 Spot 叢集的連結，則沒有索引標籤；而是向下捲動至「Auto Scaling」區段。
5. 如果未設定自動擴展，請選擇設定。
6. 利用容量擴展的範圍來設定機群的最低和最高容量。擴展政策不會將機群的規模擴展到小於最低容量或大於最高容量。
7. 在資源調整政策下，對於原則類型，選擇步驟資源調整政策。
8. 一開始，擴展政策包含名為 ScaleUp 和 ScaleDown 的步驟擴展政策。您可以完成這些政策，或是選擇移除政策來刪除政策。您也可以選擇新增政策。
9. 若要定義政策，請執行下列的動作：

   a. 針對 Policy Name (政策名稱)，請輸入政策的名稱。
   b. 對於政策觸發器，請選取現有警示，或選擇建立警示以開啟 Amazon CloudWatch 主控台並建立警示。
   c. 針對修改容量，請定義要擴展的量以及步驟調整的下限與上限。您可以新增或移除特定數量的執行個體，或現有機群大小的百分比，或將機群設為確切大小。

      例如，若要建立將叢集容量增加 30% 的步驟擴展政策，請選擇 [新增]，在下一個欄位中輸入 30，然後選擇百分比。根據預設，新增政策的下限為警示閾值，而上限為無限大正數 (+) 值。根據預設，移除政策的上限為警示閾值，而下限為無限小負數 (-) 值。
   d. (選用) 若要新增另一個步驟，請選擇新增步驟。
   e. 針對冷卻時間，請指定一個新值 (以秒為單位) 或保留預設值。
10. 選擇 Save (儲存)。

若要使用以下方式設定 Spot 叢集的步驟擴展原則 AWS CLI

1. 使用register-scalable-target命令將 Spot 叢集請求註冊為可擴展的目標。
2. 使用put-scaling-policy命令建立擴展政策。
3. 使用命令建立觸發擴展政策的警put-metric-alarm示。

排定的擴展：按排程擴展 Spot 叢集

按排程調整叢集可讓您調整應用程式，以因應可預測的需求變更。透過建立排程動作，您可以指示 Spot 叢集在特定時間執行擴展活動。若要建立排程動作，您必須指定現有的 Spot 叢集、必須執行擴展活動的時間，以及所需的最小和最大容量。排程動作可設定為一次或按週期性排程進行擴充。如果您需要變更，可以編輯或刪除已排程的動作。

必要條件

- 剪切動作只能針對現有的 Spot 叢集建立。您無法在建立 Spot 叢集時建立排程動作。
- Spot 機群請求必須具有 `maintain` 的請求類型。類型 `request` 的請求不支援自動擴展功能。
- 設定 IAMSpot 叢集自動擴充所需的權限。
- 請參閱考量事項。

建立一次性排程動作

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選取您的 Spot Fleet 請求。
4. 選擇畫面底部附近的「預約縮放比例」標籤。如果您選取競價型叢集的連結，則沒有索引標籤；而是向下捲動至「排程擴展」區段。
5. 選擇「建立排程動作」。
6. 在名稱 中，為排程動作指定名稱。
7. 為容量下限、容量上限或兩者輸入一值。
8. 針對重複，選擇一次。
9. (選用) 針對開始時間、結束時間或兩者選擇日期和時間。
10. 選擇 Create (建立)。

若要建立週期性排程動作

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選取您的 Spot Fleet 請求。
4. 選擇畫面底部附近的「預約縮放比例」標籤。如果您選取競價型叢集的連結，則沒有索引標籤；而是向下捲動至「排程擴展」區段。
5. 在名稱 中，為排程動作指定名稱。
6. 為容量下限、容量上限或兩者輸入一值。
7. 針對重複，選擇其中一個預先定義的排程 (例如，每天)，或選擇 自訂，然後輸入 Cron 表達式。如需排程擴展支援的 Cron 運算式的詳細資訊，請參閱 Amazon EventBridge 使用者指南中的 Cron 運算式。

8. (選用) 針對開始時間、結束時間或兩者選擇日期和時間。
9. 選擇提交。

編輯排程動作

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選取您的 Spot Fleet 請求。
4. 選擇畫面底部附近的「預約縮放比例」標籤。如果您選取競價型叢集的連結，則沒有索引標籤；而是向下捲動至「排程擴展」區段。
5. 選取排程動作，然後選擇動作、編輯。
6. 執行需要的變更，然後選擇提交。

刪除排程動作

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在瀏覽窗格中，選擇 Spot 請求。
3. 選取您的 Spot Fleet 請求。
4. 選擇畫面底部附近的「預約縮放比例」標籤。如果您選取競價型叢集的連結，則沒有索引標籤；而是向下捲動至「排程擴展」區段。
5. 選取排程動作，然後選擇動作、刪除。
6. 出現確認提示時，請選擇 Delete (刪除)。

使用管理排程的縮放 AWS CLI

使用下列命令：

- put-scheduled-action
- describe-scheduled-actions
- delete-scheduled-action

監控您的EC2車隊或 Spot 車隊

有效監控您的EC2車隊或競價型艦隊對於維持最佳效能並確保可靠性至關重要。有各種工具可以幫助您實現這一目標 EventBridge，包括本主題介紹的 Amazon CloudWatch 和 Amazon。

使用 CloudWatch，您可以收集和追蹤指標、設定警示，並自動回應車隊狀態的變化。

使用 EventBridge，您可以透過程式設計方式監控並回應叢集所發出的事件。透過在中定義規則 EventBridge，您可以自動回應特定叢集事件 (例如執行個體終止或叢集狀態變更)，從而提高營運效率。

主題

- 使用以下方式監控您的車EC2隊或 Spot CloudWatch
- 使用 Amazon 監控並以程式設計方式回應EC2叢集或 Spot 叢集所發出的事件 EventBridge

使用以下方式監控您的車EC2隊或 Spot CloudWatch

您EC2可以使用本節所述的 Amazon CloudWatch 指標來監控叢集或 Spot 叢集。

> **Important**
>
> 為確保準確性，建議您在使用這些指標時，啟用詳細監控功能。如需詳細資訊，請參閱管理EC2執行個體的詳細監控。

如需使用的更多資訊 CloudWatch，請參閱使用監控執行個體 CloudWatch。

EC2車隊和競價型艦隊指標

`AWS/EC2Spot`命名空間包含叢集的下列指標，以及叢集中 Spot 執行個體的 CloudWatch 指標。如需詳細資訊，請參閱執行個體指標。

指標	描述
`AvailableInstancePoolsCount`	叢集請求中指定的競價型容量集區。 單位：計數

指標	描述
BidsSubmittedForCapacity	Amazon EC2 已提交叢集請求的容量。 單位：計數
EligibleInstancePoolCount	叢集請求中指定的競價型容量集區，Amazon 可EC2以在其中履行請求。如果您願意為競價型執行個體支付的最高價格低於競價型價格或競價型價格高於隨需執行個體的價格，Amazon 不EC2會滿足集區中的請求。 單位：計數
FulfilledCapacity	Amazon EC2 已經實現的容量。 單位：計數
MaxPercentCapacityAllocation	叢集要求中指定的PercentCapacityAllocation 所有叢集集區的最大值。 單位：百分比
PendingCapacity	TargetCapacity 與 FulfilledCapacity 之間的差異。 單位：計數
PercentCapacityAllocation	已為指定維度的 Spot 容量集區配置的容量。若要取得所有 Spot 容量集區所記錄的最大值，請使用 MaxPercentCapacityAllocation 。 單位：百分比
TargetCapacity	叢集要求的目標容量。 單位：計數
TerminatingCapacity	由於佈建容量而終止的容量大於目標容量。 單位：計數

如果指標的測量單位是 `Count`，則最有用的統計資訊是 `Average`。

EC2車隊和現貨車隊維度

若要篩選叢集的資料，請使用下列維度。

維度	描述
`AvailabilityZone`	依可用區域篩選資料。
`FleetRequestId`	按車隊請求過濾數據。
`InstanceType`	依執行個體類型篩選資料。

檢視您的EC2車隊或 Spot 叢集的 CloudWatch 指標

您可以使用 Amazon CloudWatch 主控台檢視叢集的 CloudWatch 指標。這些指標會以監控圖表的形式顯示。如果叢集處於作用中狀態，這些圖表會顯示資料點。

指標會先依據命名空間進行分組，然後再根據各命名空間內不同的維度組合來分組。例如，您可以依叢集要求識別碼、執行個體類型或可用區域來檢視所有叢集指標或叢集指標群組。

若要檢視叢集度量

1. 在開啟 CloudWatch 主控台https://console.aws.amazon.com/cloudwatch/。
2. 在導覽窗格中，展開「量度」，然後選擇「所有量度」。
3. 選擇 S EC2pot 命名空間。

> **Note**
>
> 如果未顯示 EC2Spot 命名空間，則有兩個原因。您可能從未在區域中使用EC2過叢集或 Spot Fleet，只有您使用的 AWS 服務將指標傳送到 Amazon。 CloudWatch或者，如果您在區域中使用EC2叢集或 Spot 叢集，但過去兩週沒有使用，則命名空間不會出現。

4. 若要依維度篩選量度，請選擇下列其中一項：
   - 叢集請求指標 — 按叢集請求分組
   - 依可用區域 — 依叢集請求和可用區域分組
   - 依執行個體類型 — 依叢集請求和執行個體類型分組

- 按可用區域/執行個體類型 — 按叢集請求、可用區域和執行個體類型分組

5. 若要檢視指標的資料，請勾選指標旁的核取方塊。

使用 Amazon 監控並以程式設計方式回應EC2叢集或 Spot 叢集所發出的事件 EventBridge

當EC2叢集或 Spot 叢集的狀態變更時，會發出通知。該通知作為發送到 Amazon 的事件提供 EventBridge（以前稱為 Amazon CloudWatch 活動）。盡可能發出事件。

您可以使用 Amazon 建 EventBridge 立規則，以便觸發程式設計動作以回應事件。例如，您可以建立兩個 EventBridge 規則：一個規則在叢集狀態變更時觸發，另一個則在叢集中的執行個體終止時觸發。在此範例中，您可以設定第一個規則，以便在叢集狀態變更時，規則會叫用SNS主題，並傳送電子郵件通知給您。您可以設定第二個規則，以便在叢集中的執行個體終止時，規則會叫用 Lambda 函數來啟動新執行個體。

> **Note**
>
> 只有類型 `maintain` 和 `request` 的機群會發出事件。類型 `instant` 的機群不會發出事件，因為它們提交一次性的同步請求，而且在回應中會立即知道機群的狀態。若要使用 Amazon EventBridge 監控叢集事件，請求類型必須是`maintain`或`request`。

如需如何描述叢集事件歷程記錄的指示，請參閱描述EC2艦隊的事件歷史。

主題

- 建立 Amazon EventBridge 規則以監控EC2叢集或競價型叢集事件
- EC2艦隊事件類型
- Spot Fleet 事件類型

建立 Amazon EventBridge 規則以監控EC2叢集或競價型叢集事件

當針對EC2叢集或 Spot 叢集發出狀態變更通知時，會以事件形式傳送至 Amazon EventBridge 作為 JSON檔案。如果 EventBridge 偵測到符合規則中定義的模式的事件模式，則會 EventBridge 叫用規則中指定的目標 (或多個目標)。

您可以撰寫 EventBridge 規則，根據相符的事件模式自動執行動作。

事件中的下列欄位會形成規則中定義的事件模式：

`"source": "aws.ec2fleet"`

識別事件來自「EC2叢集」。

`"detail-type": "EC2 Fleet State Change"`

識別事件類型。

`"detail": { "sub-type": "submitted" }`

識別事件子類型。

如需EC2叢集和 Spot 叢集事件清單以及事件資料範例，請參閱EC2艦隊事件類型和Spot Fleet 事件類型。

範例

- 建立 EventBridge 規則以傳送通知
- 建立 EventBridge 規則以觸發 Lambda 函數

建立 EventBridge 規則以傳送通知

下列範例會建立 EventBridge 規則，以便在 Amazon EC2 每次發出EC2叢集狀態變更通知時傳送電子郵件、文字訊息或行動推播通知。將發出此範例中的訊號作為 `EC2 Fleet State Change` 事件，這會觸發規則定義的動作。

先決條件

在建立 EventBridge 規則之前，您必須為電子郵件、文字訊息或行動推播通知建立 Amazon SNS 主題。

若要建立 EventBridge 規則，以便在EC2叢集狀態變更時傳送通知

1. 在打開 Amazon EventBridge 控制台https://console.aws.amazon.com/events/。
2. 選擇建立規則。
3. 針對 Define rule detail (定義規則詳細資訊) 執行下列動作：
   a. 輸入規則的Name (名稱)，或者輸入描述。

      在同一個區域和同一個事件匯流排上，規則不能與另一個規則同名。

b. 針對事件匯流排選擇預設值。當您帳戶中的某個 AWS 服務產生事件時，它一律會移至您帳戶的預設事件匯流排。

c. 針對規則類型，選擇具有事件模式的規則。

d. 選擇 Next (下一步)。

4. 針對 Build event pattern (建置事件模式) 執行下列動作：

a. 對於事件來源，請選擇AWS 事件或 EventBridge 合作夥伴事件。

b. 針對此範例的 Event pattern (事件模式)，您需指定下列事件模式，以便與 `EC2 Fleet Instance Change` 事件相符。

```
{
 "source": ["aws.ec2fleet"],
 "detail-type": ["EC2 Fleet Instance Change"]
}
```

若要新增事件模式，您可以選擇事件模式表單來使用範本，或選擇自訂模式 (JSON編輯器) 來指定您自己的模式，如下所示：

i. 若要使用範本建立事件模式，請執行下列動作：

A. 選擇 Event pattern form (事件模式表單)。

B. 在 Event source (事件來源) 欄位中，選擇 AWS services (服務)。

C. 對於 [AWS 服務]，選擇 [EC2機隊]

D. 針對事件類型，選擇EC2叢集執行個體變更。

E. 若要自訂範本，請選擇 Edit pattern (編輯模式) 並進行變更以與範例事件模式相符。

ii. (替代) 若要指定自訂事件模式，請執行下列動作：

A. 選擇自定義模式（JSON編輯器）。

B. 在 Event pattern (事件模式) 方塊中，為此範例新增事件模式。

c. 選擇 Next (下一步)。

5. 針對 Select target(s) (選取目標) 執行下列動作：

a. 在 Target types (目標類型) 欄位中，選擇 AWS service (服務)。

b. 針對 [選取目標]，選擇要在事件發生時傳送電子郵件、簡訊或行動推播通知的SNS主題。

c. 針對 Topic (主題)，請選擇現有的主題。您首先需要使用 Amazon 控制台創建一個 Amazon SNS SNS 主題。如需詳細資訊，請參閱 Amazon 簡單通知服務開發人員指南中的使用 Amazon SNS 進行 application-to-person (A2P) 簡訊。

d. (選用) 在 Additional settings (其他設定) 下，您可以選擇性地設定其他設定。如需詳細資訊，請參閱 Amazon EventBridge 使用者指南中的建立對事件做出反應的 Amazon EventBridge 規則 (步驟 16)。

e. 選擇 Next (下一步)。

6. (選用) 針對 Tags (標籤)，您可以選擇性地將一或多個標籤指派給您的規則，然後選擇 Next (下一步)。

7. 針對 Review and create (檢閱和建立) 執行下列動作：

   a. 檢閱規則的詳細資訊，然後視需求進行修改。

   b. 選擇建立規則。

有關更多信息，請參閱 Amazon EventBridge 用戶指南中的 Amazon EventBridge 規則和 Amazon EventBridge 事件模式

建立 EventBridge 規則以觸發 Lambda 函數

下列範例會建立 EventBridge 規則，以便在每次 Amazon 在執行個體啟動時EC2發出EC2叢集執行個體變更通知時觸發 Lambda 函數。將發出此範例中的訊號作為 `EC2 Fleet Instance Change` 事件 (子類型 `launched`)，這會觸發規則定義的動作。

在建立 EventBridge 規則之前，您必須先建立 Lambda 函數。

若要建立要在 EventBridge 規則中使用的 Lambda 函數

1. 在開啟 AWS Lambda 主控台https://console.aws.amazon.com/lambda/。

2. 選擇 Create function (建立函式)。

3. 輸入函數的名稱、設定程式碼，然後選擇 Create function (建立函數)。

   如需有關使用 Lambda 的詳細資訊，請參閱 AWS Lambda 開發人員指南中的使用主控台建立 Lambda 函數。

若要建立 EventBridge 規則以在EC2叢集中的執行個體變更狀態時觸發 Lambda 函數

1. 在打開 Amazon EventBridge 控制台https://console.aws.amazon.com/events/。

2. 選擇建立規則。
3. 針對 Define rule detail (定義規則詳細資訊) 執行下列動作：
   a. 輸入規則的Name (名稱)，或者輸入描述。

      在同一個區域和同一個事件匯流排上，規則不能與另一個規則同名。
   b. 針對事件匯流排選擇預設值。當您帳戶中的某個 AWS 服務產生事件時，它一律會移至您帳戶的預設事件匯流排。
   c. 針對規則類型，選擇具有事件模式的規則。
   d. 選擇 Next (下一步)。
4. 針對 Build event pattern (建置事件模式) 執行下列動作：
   a. 對於事件來源，請選擇AWS 事件或 EventBridge 合作夥伴事件。
   b. 針對此範例的 Event pattern (事件模式)，您需指定下列事件模式，以便與 `EC2 Fleet Instance Change` 事件和 `launched` 子類型相符。

```
{
 "source": ["aws.ec2fleet"],
 "detail-type": ["EC2 Fleet Instance Change"],
 "detail": {
   "sub-type": ["launched"]
}
```

      若要新增事件模式，您可以選擇事件模式表單來使用範本，或選擇自訂模式 (JSON編輯器) 來指定您自己的模式，如下所示：

      i. 若要使用範本建立事件模式，請執行下列動作：
         A. 選擇 Event pattern form (事件模式表單)。
         B. 在 Event source (事件來源) 欄位中，選擇 AWS services (服務)。
         C. 對於 [AWS 服務]，選擇 [EC2機隊]
         D. 針對事件類型，選擇EC2叢集執行個體變更。
         E. 選擇 Edit pattern (編輯模式)，並新增 `"detail": {"sub-type": ["launched"]` 以符合範例事件模式。若要取得適當的JSON格式，請在前面的方括號 (`,`) 之後插入逗號 (`]`)。
      ii. (替代) 若要指定自訂事件模式，請執行下列動作：

A. 選擇自定義模式（JSON編輯器）。

B. 在 Event pattern (事件模式) 方塊中，為此範例新增事件模式。

c. 選擇 Next (下一步)。

5. 針對 Select target(s) (選取目標) 執行下列動作：

a. 在 Target types (目標類型) 欄位中，選擇 AWS service (服務)。

b. 針對 [選取目標]，選擇要在事件發生時傳送電子郵件、簡訊或行動推播通知的SNS主題。

c. 針對 Topic (主題)，請選擇 Lambda function (Lambda 函數)，然後在 Function (函數) 中選擇您建立的函數，以便在事件發生時回應。

d. (選用) 在 Additional settings (其他設定) 下，您可以選擇性地設定其他設定。如需詳細資訊，請參閱 Amazon EventBridge 使用者指南中的建立對事件做出反應的 Amazon EventBridge 規則 (步驟 16)。

e. 選擇 Next (下一步)。

6. (選用) 針對 Tags (標籤)，您可以選擇性地將一或多個標籤指派給您的規則，然後選擇 Next (下一步)。

7. 針對 Review and create (檢閱和建立) 執行下列動作：

a. 檢閱規則的詳細資訊，然後視需求進行修改。

b. 選擇建立規則。

如需如何建立 Lambda 函數和執行 Lambda 函數的 EventBridge 規則的教學課程，請參閱AWS Lambda 開發人員指南 EventBridge中的教學課程：使用記錄 Amazon EC2 執行個體的狀態。

EC2艦隊事件類型

有五種EC2艦隊事件類型。每個事件類型都有數個子類型。

事件類型

- EC2車隊狀態變更
- EC2叢集競價型執行個體請求
- EC2機隊實例變更
- EC2車隊資訊
- EC2艦隊錯誤

EC2車隊狀態變更

EC2叢集變更狀態 EventBridge 時，EC2叢集會傳送`EC2 Fleet State Change`事件至 Amazon。

以下是此事件的範例資料。

```
{
    "version": "0",
    "id": "715ed6b3-b8fc-27fe-fad6-528c7b8bf8a2",
    "detail-type": "EC2 Fleet State Change",
    "source": "aws.ec2fleet",
    "account": "123456789012",
    "time": "2020-11-09T09:00:20Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ec2:us-east-1:123456789012:fleet/fleet-598fb973-87b7-422d-
be4d-6b0809bfff0a"
    ],
    "detail": {
        "sub-type": "active"
    }
}
```

`sub-type` 的可能值為：

`active`

EC2叢集請求已通過驗證，Amazon EC2 正在嘗試維護執行中執行個體的目標數量。

`deleted`

EC2叢集要求已刪除，且沒有執行中的執行個體。EC2叢集將在執行個體終止兩天後刪除。

`deleted_running`

EC2叢集要求已刪除，且不會啟動其他執行個體。其現有執行個體將繼續執行直到中斷或終止。此請求會一直維持在該狀態，直到所有執行個體中斷或終止。

`deleted_terminating`

EC2叢集要求會遭到刪除，且其執行個體正在終止。此請求會一直維持在該狀態，直到所有執行個體終止。

`expired`

EC2叢集要求已過期。如果請求是使用 `TerminateInstancesWithExpiration` 集建立的，後續 `terminated` 事件會指出已終止執行個體。

`modify_in_progress`

EC2叢集要求正在修改中。請求會一直維持在此狀態，直到已完成修改。

`modify_succeeded`

已修改「EC2叢集」要求。

`submitted`

系統正在評估EC2叢集請求，Amazon EC2 正準備啟動目標執行個體數量。

`progress`

EC2艦隊請求正在完成。

EC2叢集競價型執行個體請求

EC2叢集中的 Spot 執`EC2 Fleet Spot Instance Request Change`行個體請求變更狀態 EventBridge 時，叢集會傳送事件至 Amazon。

以下是此事件的範例資料。

```
{
    "version": "0",
    "id": "19331f74-bf4b-a3dd-0f1b-ddb1422032b9",
    "detail-type": "EC2 Fleet Spot Instance Request Change",
    "source": "aws.ec2fleet",
    "account": "123456789012",
    "time": "2020-11-09T09:00:05Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ec2:us-east-1:123456789012:fleet/
fleet-83fd4e48-552a-40ef-9532-82a3acca5f10"
    ],
    "detail": {
        "spot-instance-request-id": "sir-rmqske6h",
        "description": "SpotInstanceRequestId sir-rmqske6h, PreviousState:
 cancelled_running",
        "sub-type": "cancelled"
```

```
    }
}
```

`sub-type` 的可能值為：

`active`

Spot 執行個體請求已履行，並具有關聯的 Spot 執行個體。

`cancelled`

您已取消 Spot 執行個體請求，或 Spot 執行個體請求已過期。

`disabled`

您已停用 Spot 執行個體。

`submitted`

已提交 Spot 執行個體請求。

EC2機隊實例變更

EC2叢集中的執行`EC2 Fleet Instance Change`個體狀態變更 EventBridge時，叢集會傳送事件至 Amazon。

以下是此事件的範例資料。

```
{
    "version": "0",
    "id": "542ce428-c8f1-0608-c015-e8ed6522c5bc",
    "detail-type": "EC2 Fleet Instance Change",
    "source": "aws.ec2fleet",
    "account": "123456789012",
    "time": "2020-11-09T09:00:23Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ec2:us-east-1:123456789012:fleet/fleet-598fb973-87b7-422d-
be4d-6b0809bfff0a"
    ],
    "detail": {
        "instance-id": "i-0c594155dd5ff1829",
        "description": "{\"instanceType\":\"c5.large\",\"image\":\"ami-6057e21a\",
\"productDescription\":\"Linux/UNIX\",\"availabilityZone\":\"us-east-1d\"}",
```

```
        "sub-type": "launched"
    }
}
```

`sub-type` 的可能值為：

`launched`

已啟動新執行個體。

`terminated`

已終止執行個體。

`termination_notified`

Amazon EC2 在縮減期間終止 Spot 執行個體時，會傳送執行個體終止通知，例如從 4 的目標容量降低到目標容量 3 時。

EC2車隊資訊

EC2Fleet 會在履行期間發生錯誤 EventBridge 時，將`EC2 Fleet Information`事件傳送至 Amazon。此資訊事件不會封鎖機群嘗試履行其目標容量。

以下是此事件的範例資料。

```
{
    "version": "0",
    "id": "76529817-d605-4571-7224-d36cc1b2c0c4",
    "detail-type": "EC2 Fleet Information",
    "source": "aws.ec2fleet",
    "account": "123456789012",
    "time": "2020-11-09T08:17:07Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ec2:us-east-1:123456789012:fleet/fleet-8becf5fe-
bb9e-415d-8f54-3fa5a8628b91"
    ],
    "detail": {
        "description": "c4.xlarge, ami-0947d2ba12ee1ff75, Linux/UNIX, us-east-1a,
 Spot price in either SpotFleetRequestConfigData or SpotFleetLaunchSpecification or
 LaunchTemplate or LaunchTemplateOverrides is less than Spot market price $0.0619",
        "sub-type": "launchSpecUnusable"
    }
```

```
}
```

`sub-type` 的可能值為：

`fleetProgressHalted`

每個啟動規格中的價格無效，因為價格低於 Spot 價格 (所有啟動規格都已產生 `launchSpecUnusable` 事件)。如果 Spot 價格變動，啟動規格則可能會變成有效。

`launchSpecTemporarilyBlacklisted`

組態無效，且多次嘗試啟動執行個體失敗。如需詳細資訊，請參閱事件的描述。

`launchSpecUnusable`

啟動規格中的價格無效，因為價格低於 Spot 價格。

`registerWithLoadBalancersFailed`

嘗試在負載平衡器失敗時註冊執行個體。如需詳細資訊，請參閱事件的描述。

EC2艦隊錯誤

EC2Fleet 會在履行期間發生錯誤 EventBridge 時，將`EC2 Fleet Error`事件傳送至 Amazon。錯誤事件會阻止機群嘗試履行其目標容量。

以下是此事件的範例資料。

```
{
    "version": "0",
    "id": "69849a22-6d0f-d4ce-602b-b47c1c98240e",
    "detail-type": "EC2 Fleet Error",
    "source": "aws.ec2fleet",
    "account": "123456789012",
    "time": "2020-10-07T01:44:24Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ec2:us-east-1:123456789012:fleet/fleet-9bb19bc6-60d3-4fd2-ae47-
d33e68eafa08"
    ],
    "detail": {
        "description": "m3.large, ami-00068cd7555f543d5, Linux/UNIX: IPv6 is not
 supported for the instance type 'm3.large'. ",
        "sub-type": "spotFleetRequestConfigurationInvalid"
```

```
    }
}
```

`sub-type` 的可能值為：

`iamFleetRoleInvalid`

EC2叢集沒有啟動或終止執行個體的必要權限。

`allLaunchSpecsTemporarilyBlacklisted`

沒有任何組態是有效的，且多次嘗試啟動執行個體失敗。如需詳細資訊，請參閱事件的描述。

`spotInstanceCountLimitExceeded`

您已達到可啟動 Spot 執行個體的數目上限。

`spotFleetRequestConfigurationInvalid`

組態無效。如需詳細資訊，請參閱事件的描述。

Spot Fleet 事件類型

有五種 Spot Fleet 事件類型。每個事件類型都有數個子類型。

事件類型

- EC2發現艦隊狀態變更
- EC2競價型叢集 Spot 執行個體請求
- EC2競價型叢集實例變更
- EC2現貨車隊資訊
- EC2發現艦隊錯誤

EC2發現艦隊狀態變更

Spot 叢集會在 Spot 叢集變更狀態 EventBridge時向 Amazon 傳送`EC2 Spot Fleet State Change`事件。

以下是此事件的範例資料。

```
{
    "version": "0",
```

```
    "id": "d1af1091-6cc3-2e24-203a-3b870e455d5b",
    "detail-type": "EC2 Spot Fleet State Change",
    "source": "aws.ec2spotfleet",
    "account": "123456789012",
    "time": "2020-11-09T08:57:06Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ec2:us-east-1:123456789012:spot-fleet-request/sfr-4b6d274d-0cea-4b2c-
b3be-9dc627ad1f55"
    ],
    "detail": {
        "sub-type": "submitted"
    }
}
```

`sub-type` 的可能值為：

`active`

Spot 叢集請求已經過驗證，Amazon EC2 正在嘗試維護執行中執行個體的目標數量。

`cancelled`

Spot Fleet 請求已取消，且無執行中的執行個體。Spot Fleet 將會在其執行個體終止後兩天刪除。

`cancelled_running`

Spot Fleet 請求已取消，不會啟動其他執行個體。其現有執行個體將繼續執行直到中斷或終止。此請求會一直維持在該狀態，直到所有執行個體中斷或終止。

`cancelled_terminating`

Spot Fleet 請求已取消，其執行個體正在終止。此請求會一直維持在該狀態，直到所有執行個體終止。

`expired`

Spot Fleet 請求已過期。如果請求是使用 `TerminateInstancesWithExpiration` 集建立的，後續 `terminated` 事件會指出已終止執行個體。

`modify_in_progress`

正在修改 Spot Fleet 請求。請求會一直維持在此狀態，直到已完成修改。

`modify_succeeded`

已修改 Spot Fleet 請求。

`submitted`

正在評估 Spot 叢集請求，Amazon EC2 正準備啟動目標執行個體數量。

`progress`

正在履行 Spot Fleet 請求。

EC2競價型叢集 Spot 執行個體請求

當叢集中的 Spot 執`EC2 Spot Fleet Spot Instance Request Change`行個體請求變更狀態 EventBridge 時，Spot 叢集會傳送事件至 Amazon。

以下是此事件的範例資料。

```
{
    "version": "0",
    "id": "cd141ef0-14af-d670-a71d-fe46e9971bd2",
    "detail-type": "EC2 Spot Fleet Spot Instance Request Change",
    "source": "aws.ec2spotfleet",
    "account": "123456789012",
    "time": "2020-11-09T08:53:21Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ec2:us-east-1:123456789012:spot-fleet-request/sfr-
a98d2133-941a-47dc-8b03-0f94c6852ad1"
    ],
    "detail": {
        "spot-instance-request-id": "sir-a2w9gc5h",
        "description": "SpotInstanceRequestId sir-a2w9gc5h, PreviousState:
 cancelled_running",
        "sub-type": "cancelled"
    }
}
```

`sub-type` 的可能值為：

`active`

Spot 執行個體請求已履行，並具有關聯的 Spot 執行個體。

`cancelled`

您已取消 Spot 執行個體請求，或 Spot 執行個體請求已過期。

`disabled`

您已停用 Spot 執行個體。

`submitted`

已提交 Spot 執行個體請求。

EC2競價型叢集實例變更

當叢集中的執行`EC2 Spot Fleet Instance Change`個體狀態變更 EventBridge 時，Spot 叢集會傳送事件至 Amazon。

以下是此事件的範例資料。

```
{
    "version": "0",
    "id": "11591686-5bd7-bbaa-eb40-d46529c2710f",
    "detail-type": "EC2 Spot Fleet Instance Change",
    "source": "aws.ec2spotfleet",
    "account": "123456789012",
    "time": "2020-11-09T07:25:02Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ec2:us-east-1:123456789012:spot-fleet-request/sfr-c8a764a4-bedc-4b62-
af9c-0095e6e3ba61"
    ],
    "detail": {
        "instance-id": "i-08b90df1e09c30c9b",
        "description": "{\"instanceType\":\"r4.2xlarge\",\"image\":
\"ami-032930428bf1abbff\",\"productDescription\":\"Linux/UNIX\",\"availabilityZone\":
\"us-east-1a\"}",
        "sub-type": "launched"
    }
}
```

`sub-type` 的可能值為：

`launched`

已啟動新執行個體。

`terminated`

已終止執行個體。

`termination_notified`

Amazon EC2 在縮減期間終止 Spot 執行個體時，會傳送執行個體終止通知，例如從 4 的目標容量降低到目標容量 3 時。

EC2現貨車隊資訊

如 EventBridge果在履行期間發生錯誤，Spot 叢集就會傳送`EC2 Spot Fleet Information`事件給 Amazon。此資訊事件不會封鎖機群嘗試履行其目標容量。

以下是此事件的範例資料。

```
{
    "version": "0",
    "id": "73a60f70-3409-a66c-635c-7f66c5f5b669",
    "detail-type": "EC2 Spot Fleet Information",
    "source": "aws.ec2spotfleet",
    "account": "123456789012",
    "time": "2020-11-08T20:56:12Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ec2:us-east-1:123456789012:spot-fleet-request/sfr-2531ea06-
af18-4647-8757-7d69c94971b1"
    ],
    "detail": {
        "description": "r3.8xlarge, ami-032930428bf1abbff, Linux/UNIX, us-east-1a, Spot
 bid price is less than Spot market price $0.5291",
        "sub-type": "launchSpecUnusable"
    }
}
```

`sub-type` 的可能值為：

`fleetProgressHalted`

每個啟動規格中的價格無效，因為價格低於 Spot 價格 (所有啟動規格都已產生 `launchSpecUnusable` 事件)。如果 Spot 價格變動，啟動規格則可能會變成有效。

`launchSpecTemporarilyBlacklisted`

組態無效，且多次嘗試啟動執行個體失敗。如需詳細資訊，請參閱事件的描述。

`launchSpecUnusable`

啟動規格中的價格無效，因為價格低於 Spot 價格。

`registerWithLoadBalancersFailed`

嘗試在負載平衡器失敗時註冊執行個體。如需詳細資訊，請參閱事件的描述。

EC2發現艦隊錯誤

如 EventBridge 果在履行期間發生錯誤，Spot 叢集就會傳送`EC2 Spot Fleet Error`事件給 Amazon。錯誤事件會阻止機群嘗試履行其目標容量。

以下是此事件的範例資料。

```
{
    "version": "0",
    "id": "10adc4e7-675c-643e-125c-5bfa1b1ba5d2",
    "detail-type": "EC2 Spot Fleet Error",
    "source": "aws.ec2spotfleet",
    "account": "123456789012",
    "time": "2020-11-09T06:56:07Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ec2:us-east-1:123456789012:spot-fleet-request/
sfr-38725d30-25f1-4f30-83ce-2907c56dba17"
    ],
    "detail": {
        "description": "r4.2xlarge, ami-032930428bf1abbff, Linux/UNIX: The
 associatePublicIPAddress parameter can only be specified for the network interface
 with DeviceIndex 0. ",
        "sub-type": "spotFleetRequestConfigurationInvalid"
    }
}
```

`sub-type` 的可能值為：

`iamFleetRoleInvalid`

Spot Fleet 沒有啟動或終止執行個體所需的許可。

`allLaunchSpecsTemporarilyBlacklisted`

沒有任何組態是有效的，且多次嘗試啟動執行個體失敗。如需詳細資訊，請參閱事件的描述。

`spotInstanceCountLimitExceeded`

您已達到可啟動 Spot 執行個體的數目上限。

`spotFleetRequestConfigurationInvalid`

組態無效。如需詳細資訊，請參閱事件的描述。

EC2艦隊教程

有不同的方法可以配置EC2艦隊。您選擇的組態視您的特定使用案例而定。

下列自學課程涵蓋一些可能的使用案例，並提供實作這些案例所需的工作。

使用案例	連結至教學課程
使用執行個體加權來管理EC2叢集的可用性和效能。 透過執行個體權重，您可以為EC2叢集中的每個執行個體類型指派加權，以代表其相對於彼此的運算容量和效能。根據權重，叢集可以使用指定執行個體類型的任意組合，只要它能夠滿足所需的目標容量即可。	教學課程：將EC2叢集設定為使用執行個體加權
使用隨需容量確保尖峰期間的可用性，但以較低的成本從額外的 Spot 容量中受益。 將EC2叢集設定為使用隨需執行個體做為主要容量，以確保尖峰期間的可用容量。此外，請將一些容量分配給 Spot 執行個體以享受折扣定價，同時請記住，如果 Amazon EC2 需要恢復容量，Spot 執行個體可能會中斷。	教學課程：將EC2叢集設定為使用隨需執行個體做為主要容量
使用容量保留為隨需執行個體保留運算容量。	教學課程：設定EC2叢集以使用目標容量保留啟動隨需執行個

使用案例	連結至教學課程
將EC2叢集設定為在啟動隨需執行個體時先使用targeted容量保留。如果您有嚴格的容量需求，並且正在執行需要特定等級的長期或短期容量保證的關鍵業務工作負載，我們建議您建立EC2容量保留，以確保在需要時隨時都能存取 Amazon 容量。	
使用容量區塊為您的 ML 工作負載保留備受歡迎的GPU執行個體。 設定EC2叢集以將執行個體啟動至容量區塊。	教學課程：設定EC2叢集以將執行個體啟動至容量區塊

教學課程：將EC2叢集設定為使用執行個體加權

本教程使用一個名為 Example Corp 的虛構公司來說明使用實例加權請求EC2艦隊的過程。

目標

Example Corp 是一家製藥公司，希望利用 Amazon 的計算能力來篩選可能用於EC2對抗癌症的化合物。

規劃

Example Corp 首先檢視了 Spot 最佳實務。接下來，實施例公司確定其EC2艦隊的要求。

執行個體類型

Example Corp 具有運算密集型和記憶體密集型應用程式，具有至少 60 GB 記憶體和八個虛擬 CPUs () vCPUs 的效能最佳。他們希望以最低的價格為應用程式最大限度利用這些資源。範例公司決定下列任何一種EC2執行個體類型都能滿足其需求：

執行個體類型	記憶體 (GiB)	vCPUs
r3.2xlarge	61	8
r3.4xlarge	122	16

執行個體類型	記憶體 (GiB)	vCPUs
r3.8xlarge	244	32

目標容量 (單位)

使用執行個體加權時，目標容量可以等於執行個體數目 (預設值) 或核心 (vCPUs)、記憶體 () 和儲存 (GiBsGBs) 等因素的組合。通過將其應用程序的基礎（60 GB RAM 和八個vCPUs）視為一個單元，Example Corp 決定 20 倍這個數量將滿足他們的需求。因此，該公司將其EC2艦隊請求的目標容量設置為 20 個單位。

執行個體權重

Example Corp 決定目標容量後，即計算執行個體權重。為了計算每個執行個體類型的執行個體權重，將決定達成目標容量所需的每個執行個體類型單位，如下所示：

- 大容量 (61.0 GB, 8vCPUs) = 一個單位的二十
- 大容量 (122.0 GB，16vCPUs) = 兩個單位的 20 個單位
- 大容量 (24 GB、32vCPUs) = 四個單位的 20 個單位

因此，Example Corp 會將 1、2 和 4 的執行個體權重指派給其EC2叢集請求中的相應啟動設定。

每個單位小時的價格

Example Corp 會使用每執行個體小時的隨需價格，來做為其價格的起點。他們也可以使用最近的 Spot 價格，或兩者的組合。為計算每單位小時價格，他們將每個執行個體小時的起始價格除以權重。例如：

執行個體類型	隨需價格	執行個體權重	每個單位小時的價格
R3.2 xLarge	0.7 USD	1	0.7 USD
r3.4 xLarge	1.4 USD	2	0.7 USD
r3.8 xLarge	$2.8	4	0.7 USD

Example Corp 可以使用每單位小時 0.7 USD 的全球價格，並且對於所有三種執行個體類型都具有競爭力。在 `r3.8xlarge` 啟動規格中，他們也可以使用每單位小時 0.7 USD 全局價格和 0.9 USD 每單位小時特定價格。

驗證許可

在建立EC2叢集之前，Example Corp 會驗證其具有必要權限的IAM角色。如需詳細資訊，請參閱EC2機隊先決條。

建立啟動範本

接下來，範例公司會建立一個啟動範本。啟動範本 ID 會在下列步驟中使用。如需詳細資訊，請參閱創建一個 Amazon EC2 啟動模板。

建立EC2艦隊

實施例公司創建一個文件`config.json`，其EC2艦隊具有以下配置。在下列範例中，以您自己的資源識別符取代資源識別符。

```
{
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateId": "lt-07b3bc7625cdab851",
                "Version": "1"
            },
            "Overrides": [
                {
                    "InstanceType": "r3.2xlarge",
                    "SubnetId": "subnet-482e4972",
                    "WeightedCapacity": 1
                },
                {
                    "InstanceType": "r3.4xlarge",
                    "SubnetId": "subnet-482e4972",
                    "WeightedCapacity": 2
                },
                {
                    "InstanceType": "r3.8xlarge",
                    "MaxPrice": "0.90",
                    "SubnetId": "subnet-482e4972",
                    "WeightedCapacity": 4
                }
```

```
            ]
        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 20,
        "DefaultTargetCapacityType": "spot"
    }
}
```

實施例公司使用以下創建EC2艦隊命令創建艦隊。

```
aws ec2 create-fleet --cli-input-json file://config.json
```

如需詳細資訊，請參閱建立EC2艦隊。

實現

分配策略決定您的 競價型執行個體 來自哪個 Spot 容量集區。

採用 `lowest-price` 策略 (預設的策略)，競價型執行個體 來自實現時每單位價格最低的集區。為了提供 20 個容量單位，EC2叢集會啟動 20 `r3.2xlarge` 個執行個體 (20 除以 1)、10 `r3.4xlarge` 個執行個體 (20 除以 2) 或 5 `r3.8xlarge` 個執行個體 (20 除以 4)。

如果 Example Corp 使用 `diversified` 策略，競價型執行個體 將來自所有三個集區。EC2艦隊將啟動 6 個`r3.2xlarge`實例（提供 6 個單元），3 個`r3.4xlarge`實例（提供 6 個單元）和 2 個`r3.8xlarge`實例（提供 8 個單位），總共 20 個單元。

教學課程：將EC2叢集設定為使用隨需執行個體做為主要容量

本教學課程使用名為 ABC Online 的虛擬公司來說明請求以隨需作為主要容量的EC2叢集和 Spot 容量 (如果有的話) 的程序。

目標

ABC餐廳外送公司 Online 旨在跨EC2執行個體類型和購買選項佈建 Amazon EC2 容量，以達到所需的規模、效能和成本。

計畫

ABC線上需要固定容量來處理尖峰期，但想要以較低的成本從額外容量中獲益。該公司決定其EC2艦隊的以下要求：

- 隨需執行個體容量 — ABC 線上需要 15 個隨需執行個體，以確保它們能夠在尖峰時段容納流量。
- 競價型執行個體容量 — ABC 線上計劃佈建 5 個 Spot 執行個體，以較低的價格提升效能。

驗證許可

在建立EC2叢集之前，ABCOnline 會驗證其IAM角色具有必要權限。如需詳細資訊，請參閱EC2機隊先決條。

建立啟動範本

接下來，「ABC線上」會建立啟動範本。啟動範本 ID 會在下列步驟中使用。如需詳細資訊，請參閱創建一個 Amazon EC2 啟動模板。

建立EC2艦隊

ABC在線創建一個文件`config.json`，其EC2艦隊具有以下配置。在下列範例中，以您自己的資源識別符取代資源識別符。

```
{
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateId": "lt-07b3bc7625cdab851",
                "Version": "2"
            }

        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 20,
        "OnDemandTargetCapacity":15,
        "DefaultTargetCapacityType": "spot"
    }
}
```

ABC線上使用下列「建立EC2叢集」指令建立「艦隊」。

```
aws ec2 create-fleet --cli-input-json file://config.json
```

如需詳細資訊，請參閱建立EC2艦隊。

實現

配置策略決定始終滿足隨需容量，而如果有可用容量，則目標容量的餘額會以 Spot 的形式履行。

教學課程：設定EC2叢集以使用目標容量保留啟動隨需執行個

本教學課程將引導您完成必須執行的所有步驟，以便EC2叢集將隨需執行個體啟動到targeted容量保留中。

您將會了解到如何設定機群，以在啟動隨需執行個體時先使用 targeted 隨需容量預留。您也會了解到如何設定機群，以便機群在隨需目標容量總計超過可用的未使用容量預留數目時，使用指定的分配策略來選取要在其中啟動剩餘目標容量的執行個體集區。

EC2車隊配置

在本教學課程中，叢集的設定方式如下：

- 目標容量：10 個隨需執行個體
- 總計未使用的 targeted 容量預留：6 (小於機群的隨需目標容量：10 個隨需執行個體)
- 容量預留集區數目：2 (us-east-1a 和 us-east-1b)
- 每個集區的容量預留數目：3
- 隨需分配策略：lowest-price (當未使用的容量預留數目小於隨需目標容量時，機群會根據隨需分配策略決定要用來啟動剩餘隨需容量的集區。)

  請注意，您也可以使用 prioritized 分配策略，而不使用 lowest-price 分配策略。

若要按照 targeted 容量預留來啟動隨需執行個體，必須執行以下幾個步驟：

- 步驟 1：建立容量預留
- 步驟 2：建立容量預留資源群組
- 步驟 3：將容量預留新增至容量預留資源群組
- (選用) 步驟 4：檢視資源群組中的容量預留
- 步驟 5：建立啟動範本，此範本指定容量預留以特定資源群組為目標
- (選用) 步驟 6：描述啟動範本
- 步驟 7：建立EC2艦隊
- (選用) 步驟 8：檢視剩餘未使用的容量預留數目

步驟 1：建立容量預留

使用指create-capacity-reservation令建立容量保留，三個用於us-east-1a，另外三個用於us-east-1b。除了可用區域之外，容量預留的其他屬性皆相同。

在 **us-east-1a** 中的 3 個容量預留

```
aws ec2 create-capacity-reservation \
    --availability-zone us-east-1a \
    --instance-type c5.xlarge \
    --instance-platform Linux/UNIX \
    --instance-count 3 \
    --instance-match-criteria targeted
```

最後產生的容量預留 ID 範例

```
cr-1234567890abcdef1
```

在 **us-east-1b** 中的 3 個容量預留

```
aws ec2 create-capacity-reservation \
    --availability-zone us-east-1b \
    --instance-type c5.xlarge \
    --instance-platform Linux/UNIX \
    --instance-count 3 \
    --instance-match-criteria targeted
```

最後產生的容量預留 ID 範例

```
cr-54321abcdef567890
```

步驟 2：建立容量預留資源群組

使用 `resource-groups` 服務和 create-group 命令來建立容量預留資源群組。在此範例中，資源群組會命名為 `my-cr-group`。如需為何必須建立資源群組的相關資訊，請參閱 使用容量保留在EC2叢集中保留隨需容量。

```
aws resource-groups create-group \
    --name my-cr-group \
```

```
    --configuration '{"Type":"AWS::EC2::CapacityReservationPool"}'
 '{"Type":"AWS::ResourceGroups::Generic", "Parameters": [{"Name": "allowed-resource-
types", "Values": ["AWS::EC2::CapacityReservation"]}]}'
```

步驟 3：將容量預留新增至容量預留資源群組

使用 `resource-groups` 服務和 group-resources 命令，將您在步驟 1 中建立的容量預留新增至容量預留資源群組。請注意，您必須依照隨需容量保留參考隨需容量保留ARNs。

```
aws resource-groups group-resources \
    --group my-cr-group \
    --resource-arns \
      arn:aws:ec2:us-east-1:123456789012:capacity-reservation/cr-1234567890abcdef1 \
      arn:aws:ec2:us-east-1:123456789012:capacity-reservation/cr-54321abcdef567890
```

範例輸出

```
{
   "Failed": [],
   "Succeeded": [
   "arn:aws:ec2:us-east-1:123456789012:capacity-reservation/cr-1234567890abcdef1",
   "arn:aws:ec2:us-east-1:123456789012:capacity-reservation/cr-54321abcdef567890"
   ]
}
```

(選用) 步驟 4：檢視資源群組中的容量預留

您可以使用`resource-groups`服務與命list-group-resources令，選擇性地描述資源群組，以檢視其「產能保留量」。

```
aws resource-groups list-group-resources --group my-cr-group
```

範例輸出

```
{
    "ResourceIdentifiers": [
        {
            "ResourceType": "AWS::EC2::CapacityReservation",
            "ResourceArn": "arn:aws:ec2:us-east-1:123456789012:capacity-reservation/
cr-1234567890abcdef1"
        },
```

```
        {
            "ResourceType": "AWS::EC2::CapacityReservation",
            "ResourceArn": "arn:aws:ec2:us-east-1:123456789012:capacity-reservation/
cr-54321abcdef567890"
        }
    ]
}
```

步驟 5：建立啟動範本，此範本指定容量預留以特定資源群組為目標

使用指create-launch-template令建立要在其中指定要使用的容量保留的啟動範本。在此範例中，機群會使用新增至資源群組的 targeted 容量預留。因此，啟動範本資料會指定容量預留以特定資源群組為目標。在此範例中，啟動範本會命名為 my-launch-template。

```
aws ec2 create-launch-template \
    --launch-template-name my-launch-template \
    --launch-template-data \
        '{"ImageId": "ami-0123456789example",
          "CapacityReservationSpecification":
            {"CapacityReservationTarget":
                { "CapacityReservationResourceGroupArn": "arn:aws:resource-groups:us-
east-1:123456789012:group/my-cr-group" }
            }
        }'
```

(選用) 步驟 6：描述啟動範本

使用指describe-launch-template-versions令可選擇性地描述啟動範本，以檢視其組態。

```
aws ec2 describe-launch-template-versions --launch-template-name my-launch-template
```

範例輸出

```
{
    "LaunchTemplateVersions": [
        {
            "LaunchTemplateId": "lt-01234567890example",
            "LaunchTemplateName": "my-launch-template",
            "VersionNumber": 1,
            "CreateTime": "2021-01-19T20:50:19.000Z",
            "CreatedBy": "arn:aws:iam::123456789012:user/Admin",
            "DefaultVersion": true,
```

```
            "LaunchTemplateData": {
                "ImageId": "ami-0947d2ba12ee1ff75",
                "CapacityReservationSpecification": {
                    "CapacityReservationTarget": {
                        "CapacityReservationResourceGroupArn": "arn:aws:resource-
groups:us-east-1:123456789012:group/my-cr-group"
                    }
                }
            }
        }
    ]
}
```

步驟 7：建立EC2艦隊

建立EC2叢集，指定要啟動之執行個體的組態資訊。下列「EC2叢集」組態僅顯示此範例的相關組態。啟動範本 `my-launch-template` 是您在步驟 5 中建立的啟動範本。有兩個執行個體集區，每個集區內的執行個體類型相同 (`c5.xlarge`)，但使用不同的可用區域 (`us-east-1a` 和 `us-east-1b`)。執行個體集區的價格相同，因為定價是根據區域界定，而不是根據可用區域界定的。總目標容量為 10，預設目標容量類型為 `on-demand`。隨需分配策略為 `lowest-price`。容量預留的使用策略為 `use-capacity-reservations-first`。

> **Note**
>
> 機群類型必須為 `instant`。其他機群類型不支援 `use-capacity-reservations-first`。

```
{
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateName": "my-launch-template",
                "Version": "1"
            },
            "Overrides": [
               {
                   "InstanceType": "c5.xlarge",
                   "AvailabilityZone": "us-east-1a"
               },
               {
                    "InstanceType": "c5.xlarge",
```

```
                    "AvailabilityZone": "us-east-1b"
                }
            ]
        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 10,
        "DefaultTargetCapacityType": "on-demand"
    },
    "OnDemandOptions": {
        "AllocationStrategy": "lowest-price",
        "CapacityReservationOptions": {
             "UsageStrategy": "use-capacity-reservations-first"
         }
    },
    "Type": "instant"
}
```

使用上述組態建立 `instant` 機群之後，會啟動以下 10 個執行個體以符合目標容量：

- 會先使用容量預留來啟動 6 個隨需執行個體，如下所示：
  - 3 個隨需執行個體已在 `us-east-1a` 中啟動至 3 個 `c5.xlarge` `targeted` 容量預留
  - 3 個隨需執行個體已在 `c5.xlarge` 中啟動至 3 個 `targeted` `us-east-1b` 容量預留
- 為了滿足目標容量，會根據隨需分配策略 (在本例中為 `lowest-price`) 使用一般隨需容量來啟動 4 個額外的隨需執行個體。不過，由於集區的價格相同 (因為價格是按區域而非可用區域計價)，因此機群會使用任一個集區來啟動剩餘的 4 個隨需執行個體。

(選用) 步驟 8：檢視剩餘未使用的容量預留數目

啟動叢集之後，您可以選擇性地執行describe-capacity-reservations以查看剩餘的未使用容量保留數量。在此範例中，您應該會看到下列回應，其中顯示所有集區中的所有 容量預留 都已使用。

```
{ "CapacityReservationId":  "cr-111",
     "InstanceType":  "c5.xlarge",
     "AvailableInstanceCount":  0
}

 { "CapacityReservationId":  "cr-222",
     "InstanceType":  "c5.xlarge",
     "AvailableInstanceCount":  0
```

```
}
```

教學課程：設定EC2叢集以將執行個體啟動至容量區塊

本教學課程將逐步引導您完成必須執行的步驟，讓EC2叢集將執行個體啟動至容量區塊。

在大多數情況下，EC2叢集請求的目標容量應小於或等於您目標的容量區塊保留區的可用容量。超出容量區塊預留上限的目標容量請求將無法滿足。如果目標容量請求超出容量區塊保留的限制，您將收到超出容量區塊保留限制的容量。`Insufficient Capacity Exception`

> **Note**
>
> 對於容量區塊，EC2叢集不會在所需目標容量的剩餘部分後退至啟動隨需執行個體。

如果EC2叢集無法在可用容量區塊保留區中滿足要求的目標容量，EC2叢集將盡可能多地履行容量，並傳回能夠啟動的執行個體。您可以重複呼叫EC2叢集，直到佈建完所有執行個體為止。

設定EC2叢集要求之後，您必須等到容量區塊保留的開始日期為止。如果您向EC2叢集提出要求以啟動至尚未啟動的容量區塊，您將會收到`Insufficient Capacity Error`.

容量區塊保留變為作用中後，您可以根據選取的參數API撥打EC2叢集呼叫，並將執行個體佈建到容量區塊中。在容量區塊中執行的執行個體會繼續執行，直到您手動停止或終止它們，或直到容量區塊保留結束時 Amazon EC2 終止執行個體為止。

如需容量區塊的更多資訊，請參閱ML 的容量區塊。

考量事項

- 將執行個體啟動到容量區塊時，只支援EC2叢集類型`instant`的叢集要求。如需詳細資訊，請參閱設定類型的EC2叢集 instant。
- 不支援相同EC2叢集要求中的多個容量區塊。
- 不支援在使用 `OnDemandTargetCapacity` 或 `SpotTargetCapacity` 時，同時將 `capacity-block` 設為 `DefaultTargetCapacity`。
- 如果將 `DefaultTargetCapacityType` 設為 `capacity-block`，則無法提供 `OnDemandOptions::CapacityReservationOptions`。一旦提供，便會發生例外狀況。

設定EC2叢集以將執行個體啟動至容量區塊

1. 建立啟動範本。

   在啟動範本中，執行下列動作：

   - 對於InstanceMarketOptionsRequest，設定MarketType為capacity-block。
   - 若要以容量區塊保留為目標，請針對CapacityReservationID指定容量區塊保留區識別碼。

   記下啟動範本名稱和版本。您將在下一個步驟中使用此資訊。

   如需建立啟動範本的詳細資訊，請參閱創建一個 Amazon EC2 啟動模板。

2. 設定「EC2艦隊」。

   為您的EC2叢集建立具有下列組態的檔案。config.json在下列範例中，以您自己的資源識別符取代資源識別符。

   如需配置EC2叢集的詳細資訊，請參閱建立EC2艦隊。

```
{
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateName": "CBR-launch-template",
                "Version": "1"
            },
            "Overrides": [
                {
                    "InstanceType": "p5.48xlarge",
                    "AvailabilityZone": "us-east-1a"
                },
            ]
        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 10,
        "DefaultTargetCapacityType": "capacity-block"
    },
    "Type": "instant"
}
```

3. 啟動艦隊。

使用以下 create-fleet 命令。

```
aws ec2 create-fleet --cli-input-json file://config.json
```

如需詳細資訊，請參閱建立EC2艦隊。

EC2叢集的範例CLI組態

您可以在JSON檔案中定義EC2叢集配置，然後使用 create-fleet AWS CLI 指令參考該檔案來建立叢集，如下所示：

```
aws ec2 create-fleet --cli-input-json file://file_name.json
```

下列範例說明各種EC2叢集使用案例的啟動設定。如需有關組態參數的詳細資訊，請參閱命令參考中的建立叢集。AWS CLI

範例

- 範例 1：啟動 競價型執行個體 做為預設購買選項
- 範例 2：啟動 隨需執行個體 做為預設購買選項
- 範例 3：啟動 隨需執行個體 做為主要容量
- 範例 4：使用多個容量保留啟動隨需執行個體
- 範例 5：當目標容量總計超過未使用的容量保留數時，使用容量保留啟動隨需執行個體
- 範例 6：使用目標容量保留啟動隨需執行個體
- 範例 7：設定容量重新平衡以啟動替代 Spot 執行個體
- 範例 8：在容量最佳化叢集中啟動 Spot 執行個體
- 範例 9：在具有優先順序的容量最佳化叢集中啟動 Spot 執行個體
- 範例 10：在 price-capacity-optimized 叢集中啟動 Spot 執行個體
- 範例 11：設定以屬性為基礎的執行個體類型選取

如需叢集類型的更多CLI範例`instant`，請參閱設定類型的EC2叢集 instant。

範例 1：啟動 競價型執行個體 做為預設購買選項

下列範例會指定EC2叢集中所需的最小參數：啟動範本、目標產能及預設採購選項。啟動範本由其啟動範本 ID 和版本編號識別。機群的目標容量為 2 個執行個體，預設購買選項為 spot，會讓機群啟動 2 個 競價型執行個體。

```
{
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateId": "lt-0e8c754449b27161c",
                "Version": "1"
            }

        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 2,
        "DefaultTargetCapacityType": "spot"
    }
}
```

範例 2：啟動 隨需執行個體 做為預設購買選項

下列範例會指定EC2叢集中所需的最小參數：啟動範本、目標產能及預設採購選項。啟動範本由其啟動範本 ID 和版本編號識別。機群的目標容量為 2 個執行個體，預設購買選項為 on-demand，會讓機群啟動 2 個 隨需執行個體。

```
{
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateId": "lt-0e8c754449b27161c",
                "Version": "1"
            }

        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 2,
        "DefaultTargetCapacityType": "on-demand"
    }
```

```
}
```

範例 3：啟動 隨需執行個體 做為主要容量

下列範例指定機群 2 個執行個體的總目標容量，以及 1 個隨需執行個體的目標容量。預設購買選項為 spot。機群依照指定啟動 1 個隨需執行個體，但需要再啟動一個執行個體才能實現總目標容量。其差值的購買選項計算式為 TotalTargetCapacity – OnDemandTargetCapacity = DefaultTargetCapacityType，這會讓機群啟動 1 個 Spot 執行個體。

```
{
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateId": "lt-0e8c754449b27161c",
                "Version": "1"
            }

        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 2,
        "OnDemandTargetCapacity": 1,
        "DefaultTargetCapacityType": "spot"
    }
}
```

範例 4：使用多個容量保留啟動隨需執行個體

您可以將容量預留的使用策略設為 use-capacity-reservations-first，以將機群設定為在啟動隨需執行個體 時首先使用 隨需容量預留。本例示範當容量預留比所需目標容量更多時，機群如何選取要使用的容量預留。

本例使用的機群組態如下：

- 目標容量：12 個隨需執行個體
- 總計未使用的容量預留：15 (大於機群的目標容量：12 個隨需執行個體)
- 容量預留集區數目：3 (m5.large、m4.xlarge 和 m4.2xlarge)
- 每個集區的容量預留數目：5
- 隨需分配策略：lowest-price (當多個執行個體集區中有多個未使用的容量預留時，機群會根據隨需分配策略決定要用來啟動隨需執行個體的集區。)

請注意，您也可以使用 prioritized 分配策略，而不使用 lowest-price 分配策略。

Capacity Reservations

帳戶在 3 個不同的集區中有以下 15 個未使用的 容量預留。每個集區中的 容量預留 數目由 AvailableInstanceCount 表示。

```
{
    "CapacityReservationId": "cr-111",
    "InstanceType": "m5.large",
    "InstancePlatform": "Linux/UNIX",
    "AvailabilityZone": "us-east-1a",
    "AvailableInstanceCount": 5,
    "InstanceMatchCriteria": "open",
    "State": "active"
}

{
    "CapacityReservationId": "cr-222",
    "InstanceType": "m4.xlarge",
    "InstancePlatform": "Linux/UNIX",
    "AvailabilityZone": "us-east-1a",
    "AvailableInstanceCount": 5,
    "InstanceMatchCriteria": "open",
    "State": "active"
}

{
    "CapacityReservationId": "cr-333",
    "InstanceType": "m4.2xlarge",
    "InstancePlatform": "Linux/UNIX",
    "AvailabilityZone": "us-east-1a",
    "AvailableInstanceCount":5,
    "InstanceMatchCriteria": "open",
    "State": "active"
}
```

機群組態

下列機群組態僅顯示此範例的相關組態。總目標容量為 12，預設目標容量類型為 on-demand。隨需分配策略為 lowest-price。容量預留的使用策略為 use-capacity-reservations-first。

在此範例中， 隨需執行個體 價格如下：

- `m5.large` – 每小時 0.096 美元
- `m4.xlarge` – 每小時 0.20 美元
- `m4.2xlarge` – 每小時 0.40 美元

> **Note**
>
> 機群類型必須為 `instant`。其他機群類型不支援 `use-capacity-reservations-first`。

```
{
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateId": "lt-abc1234567example",
                "Version": "1"
            }
            "Overrides": [
                {
                  "InstanceType": "m5.large",
                  "AvailabilityZone": "us-east-1a",
                  "WeightedCapacity": 1
                },
                {
                  "InstanceType": "m4.xlarge",
                  "AvailabilityZone": "us-east-1a",
                  "WeightedCapacity": 1
                },
                {
                  "InstanceType": "m4.2xlarge",
                  "AvailabilityZone": "us-east-1a",
                  "WeightedCapacity": 1
                }
              ]

        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 12,
        "DefaultTargetCapacityType": "on-demand"
```

```
    },
    "OnDemandOptions": {
        "AllocationStrategy": "lowest-price"
        "CapacityReservationOptions": {
            "UsageStrategy": "use-capacity-reservations-first"
        }
    },
    "Type": "instant",
}
```

使用上述組態建立 `instant` 叢集之後，會啟動以下 12 個執行個體以符合目標容量：

- 5 個 `m5.large` 隨需執行個體在 `us-east-1a` 中 (`us-east-1a` 中的 `m5.large` 價格最低)，並且有 5 個可用的未使用 `m5.large` 容量預留。
- 5 個 `m4.xlarge` 隨需執行個體在 us-east-1a 中 – `us-east-1a` 中的 `m4.xlarge` 價格次低，並且有 5 個可用的未使用 `m4.xlarge` 容量預留。
- 2 個 `m4.2xlarge` 隨需執行個體在 us-east-1a 中 – `us-east-1a` 中的 `m4.2xlarge` 價格第三低，並且有 5 個可用的未使用 `m4.2xlarge` 容量預留，只需要其中 2 個就可滿足目標容量

啟動叢集之後，您可以執行describe-capacity-reservations以查看剩餘的未使用容量保留數量。在本例中，您會看到下面回應，這顯示所有的 `m5.large` 和 `m4.xlarge` 容量預留都已使用，只剩 3 個 `m4.2xlarge` 容量預留未使用。

```
{
    "CapacityReservationId": "cr-111",
    "InstanceType": "m5.large",
    "AvailableInstanceCount": 0
}

{
    "CapacityReservationId": "cr-222",
    "InstanceType": "m4.xlarge",
    "AvailableInstanceCount": 0
}

{
    "CapacityReservationId": "cr-333",
    "InstanceType": "m4.2xlarge",
    "AvailableInstanceCount": 3
}
```

範例 5：當目標容量總計超過未使用的容量保留數時，使用容量保留啟動隨需執行個體

您可以將容量預留的使用策略設為 `use-capacity-reservations-first`，以將機群設定為在啟動 隨需執行個體 時首先使用 隨需容量預留。本例示範當總計目標容量超過可用的未使用容量預留數目時，機群會如何選取用來啟動隨需執行個體的執行個體集區。

本例使用的機群組態如下：

- 目標容量：16 個隨需執行個體
- 總計未使用的容量預留：15 (小於機群的目標容量：16 個隨需執行個體)
- 容量預留集區數目：3 (`m5.large`、`m4.xlarge` 和 `m4.2xlarge`)
- 每個集區的容量預留數目：5
- 隨需分配策略：`lowest-price` (當未使用的容量預留數目小於隨需目標容量時，機群會根據隨需分配策略決定要用來啟動剩餘隨需容量的集區。)

  請注意，您也可以使用 `prioritized` 分配策略，而不使用 `lowest-price` 分配策略。

Capacity Reservations

帳戶在 3 個不同的集區中有以下 15 個未使用的 容量預留。每個集區中的 容量預留 數目由 `AvailableInstanceCount` 表示。

```
{
    "CapacityReservationId": "cr-111",
    "InstanceType": "m5.large",
    "InstancePlatform": "Linux/UNIX",
    "AvailabilityZone": "us-east-1a",
    "AvailableInstanceCount": 5,
    "InstanceMatchCriteria": "open",
    "State": "active"
}

{
    "CapacityReservationId": "cr-222",
    "InstanceType": "m4.xlarge",
    "InstancePlatform": "Linux/UNIX",
    "AvailabilityZone": "us-east-1a",
    "AvailableInstanceCount": 5,
```

```
    "InstanceMatchCriteria": "open",
    "State": "active"
}

{
    "CapacityReservationId": "cr-333",
    "InstanceType": "m4.2xlarge",
    "InstancePlatform": "Linux/UNIX",
    "AvailabilityZone": "us-east-1a",
    "AvailableInstanceCount":5,
    "InstanceMatchCriteria": "open",
    "State": "active"
}
```

機群組態

下列機群組態僅顯示此範例的相關組態。總目標容量為 16，預設目標容量類型為 on-demand。隨需分配策略為 lowest-price。容量預留的使用策略為 use-capacity-reservations-first。

在此範例中， 隨需執行個體 價格如下：

- m5.large – 每小時 0.096 USD
- m4.xlarge – 每小時 0.20 USD
- m4.2xlarge – 每小時 0.40 USD

> **Note**
>
> 機群類型必須為 instant。其他機群類型不支援 use-capacity-reservations-first。

```
{
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateId": "lt-0e8c754449b27161c",
                "Version": "1"
            }
            "Overrides": [
                {
```

```
                    "InstanceType": "m5.large",
                    "AvailabilityZone": "us-east-1a",
                    "WeightedCapacity": 1
                  },
                  {
                    "InstanceType": "m4.xlarge",
                    "AvailabilityZone": "us-east-1a",
                    "WeightedCapacity": 1
                  },
                  {
                    "InstanceType": "m4.2xlarge",
                    "AvailabilityZone": "us-east-1a",
                    "WeightedCapacity": 1
                  }
                ]

        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 16,
        "DefaultTargetCapacityType": "on-demand"
    },
    "OnDemandOptions": {
        "AllocationStrategy": "lowest-price"
        "CapacityReservationOptions": {
            "UsageStrategy": "use-capacity-reservations-first"
        }
    },
    "Type": "instant",
}
```

使用上述組態建立 instant 叢集之後，會啟動以下 16 個執行個體以符合目標容量：

- 6 個 m5.large 隨需執行個體在 us-east-1a 中 – us-east-1a 中的 m5.large 價格最低，並且有 5 個可用的未使用 m5.large 容量預留。會先使用容量預留來啟動 5 個隨需執行個體。在使用了其餘的 m4.xlarge 和 m4.2xlarge 容量預留之後，為了滿足目標容量，會根據隨需分配策略啟動額外的隨需執行個體，在本例中為 lowest-price。
- 5 個 m4.xlarge 隨需執行個體在 us-east-1a 中 – us-east-1a 中的 m4.xlarge 價格次低，並且有 5 個可用的未使用 m4.xlarge 容量預留。
- 5 個 m4.2xlarge 隨需執行個體在 us-east-1a 中 – us-east-1a 中的 m4.2xlarge 價格第三低，並且有 5 個可用的未使用 m4.2xlarge 容量預留。

啟動叢集之後，您可以執行describe-capacity-reservations以查看剩餘的未使用容量保留數量。在此範例中，您應該會看到下列回應，其中顯示所有集區中的所有 容量預留 都已使用。

```
{
    "CapacityReservationId": "cr-111",
    "InstanceType": "m5.large",
    "AvailableInstanceCount": 0
}

{
    "CapacityReservationId": "cr-222",
    "InstanceType": "m4.xlarge",
    "AvailableInstanceCount": 0
}

{
    "CapacityReservationId": "cr-333",
    "InstanceType": "m4.2xlarge",
    "AvailableInstanceCount": 0
}
```

範例 6：使用目標容量保留啟動隨需執行個體

您可設定機群，使其在啟動隨需執行個體時首先使用 `targeted` 隨需容量預留，設定方法是將容量預留的使用策略設為 `use-capacity-reservations-first`。本例示範如何使用 `targeted` 容量預留來啟動隨需執行個體，其中容量預留的屬性皆相同，但其可用區域 (`us-east-1a` 和 `us-east-1b`) 不同。本例亦示範當總計目標容量超過可用的未使用容量預留數目時，機群會如何選取用來啟動隨需執行個體的執行個體集區。

本例使用的機群組態如下：

- 目標容量：10 個隨需執行個體
- 總計未使用的 `targeted` 容量預留：6 (小於機群的隨需目標容量：10 個隨需執行個體)
- 容量預留集區數目：2 (`us-east-1a` 和 `us-east-1b`)
- 每個集區的容量預留數目：3
- 隨需分配策略：`lowest-price` (當未使用的容量預留數目小於隨需目標容量時，機群會根據隨需分配策略決定要用來啟動剩餘隨需容量的集區。)

  請注意，您也可以使用 `prioritized` 分配策略，而不使用 `lowest-price` 分配策略。

如需完成此範例所須執行程序的逐步演練，請參閱 教學課程：設定EC2叢集以使用目標容量保留啟動隨需執行個。

Capacity Reservations

帳戶在 2 個不同的集區中有以下 6 個未使用的容量預留。在本例中，集區的可用區域有所不同。每個集區中的 容量預留 數目由 `AvailableInstanceCount` 表示。

```
{
    "CapacityReservationId": "cr-111",
    "InstanceType": "c5.xlarge",
    "InstancePlatform": "Linux/UNIX",
    "AvailabilityZone": "us-east-1a",
    "AvailableInstanceCount": 3,
    "InstanceMatchCriteria": "open",
    "State": "active"
}

{
    "CapacityReservationId": "cr-222",
    "InstanceType": "c5.xlarge",
    "InstancePlatform": "Linux/UNIX",
    "AvailabilityZone": "us-east-1b",
    "AvailableInstanceCount": 3,
    "InstanceMatchCriteria": "open",
    "State": "active"
}
```

機群組態

下列機群組態僅顯示此範例的相關組態。總目標容量為 10，預設目標容量類型為 `on-demand`。隨需分配策略為 `lowest-price`。容量預留的使用策略為 `use-capacity-reservations-first`。

在本例中，`us-east-1` 中的 `c5.xlarge` 隨需執行個體價格為每小時 0.17 美元。

> **Note**
>
> 機群類型必須為 `instant`。其他機群類型不支援 `use-capacity-reservations-first`。

```
{
    "LaunchTemplateConfigs": [
```

```
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateName": "my-launch-template",
                "Version": "1"
            },
            "Overrides": [
               {
                   "InstanceType": "c5.xlarge",
                   "AvailabilityZone": "us-east-1a"
               },
               {
                    "InstanceType": "c5.xlarge",
                    "AvailabilityZone": "us-east-1b"
               }
            ]
        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 10,
        "DefaultTargetCapacityType": "on-demand"
    },
    "OnDemandOptions": {
        "AllocationStrategy": "lowest-price",
        "CapacityReservationOptions": {
             "UsageStrategy": "use-capacity-reservations-first"
         }
    },
    "Type": "instant"
}
```

使用上述組態建立 `instant` 機群之後，會啟動以下 10 個執行個體以符合目標容量：

- 會先使用容量預留來啟動 6 個隨需執行個體，如下所示：
  - 3 個隨需執行個體已在 `us-east-1a` 中啟動至 3 個 `c5.xlarge` `targeted` 容量預留
  - 3 個隨需執行個體已在 `c5.xlarge` 中啟動至 3 個 `targeted` `us-east-1b` 容量預留
- 為了滿足目標容量，會根據隨需分配策略 (在本例中為 `lowest-price`) 使用一般隨需容量來啟動 4 個額外的隨需執行個體。不過，由於集區的價格相同 (因為價格是按區域而非可用區域計價)，因此機群會使用任一個集區來啟動剩餘的 4 個隨需執行個體。

啟動叢集之後，您可以執行describe-capacity-reservations以查看剩餘的未使用容量保留數量。在此範例中，您應該會看到下列回應，其中顯示所有集區中的所有 容量預留 都已使用。

```
{
    "CapacityReservationId": "cr-111",
    "InstanceType": "c5.xlarge",
    "AvailableInstanceCount": 0
}

{
    "CapacityReservationId": "cr-222",
    "InstanceType": "c5.xlarge",
    "AvailableInstanceCount": 0
}
```

範例 7：設定容量重新平衡以啟動替代 Spot 執行個體

下列範例會將EC2叢集設定為在 Amazon 針對叢集中的 Spot 執行個EC2體發出重新平衡建議時啟動替代 Spot 執行個體。若要為 `ReplacementStrategy` 設定 競價型執行個體 的自動替代，請指定 `launch-before-terminate`。若要設定從啟動新替代 Spot 執行個體到自動刪除舊 Spot 執行個體的時間延遲，請針對 `termination-delay`，指定一值 (以秒為單位)。如需詳細資訊，請參閱 組態選項。

> **Note**
>
> 建議您僅在可以預測執行個體關閉程序需要多長時間才能完成時，才使用 `launch-before-terminate`，以便只在這些程序完成後才會終止舊執行個體。所有執行個體在執行時，您需要支付所有執行個體的費用。

容量重新平衡策略的有效性取決於EC2叢集請求中指定的 Spot 容量集區數量。我們建議您使用多樣化的執行個體類型和可用區域的集合來設定機群，若要 `AllocationStrategy`，請指定 `capacity-optimized`。如需設定EC2叢集進行容量重新平衡時應考量之事項的詳細資訊，請參閱使用叢集和 S EC2 pot 叢集中的容量重新平衡來取代有風險的 Spot 執行個體。

```
{
    "ExcessCapacityTerminationPolicy": "termination",
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateName": "LaunchTemplate",
                "Version": "1"
            },
```

```
                  "Overrides": [
                        {
                            "InstanceType": "c3.large",
                            "WeightedCapacity": 1,
                             "Placement": {
                                "AvailabilityZone": "us-east-1a"
                            }
                        },
                        {
                            "InstanceType": "c4.large",
                            "WeightedCapacity": 1,
                             "Placement": {
                                "AvailabilityZone": "us-east-1a"
                            }
                        },
                        {
                            "InstanceType": "c5.large",
                            "WeightedCapacity": 1,
                             "Placement": {
                                "AvailabilityZone": "us-east-1a"
                            }
                        }
                ]
            }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 5,
        "DefaultTargetCapacityType": "spot"
    },
    "SpotOptions": {
        "AllocationStrategy": "capacity-optimized",
        "MaintenanceStrategies": {
            "CapacityRebalance": {
                "ReplacementStrategy": "launch-before-terminate",
                "TerminationDelay": "720"
            }
        }
    }
}
```

範例 8：在容量最佳化叢集中啟動 Spot 執行個體

下列範例示範如何使用可最佳化容量的 Spot 配置策略來設定EC2叢集。若要最佳化容量，您必須將 `AllocationStrategy` 設為 `capacity-optimized`。

在下列範例中，三個啟動規格指定三個 Spot 容量集區。目標容量為 50 個 Spot 執行個體。EC2叢集會嘗試將 50 個 Spot 執行個體啟動到 Spot 容量集區，並針對正在啟動的執行個體數量提供最佳容量。

```
{
    "SpotOptions": {
        "AllocationStrategy": "capacity-optimized",
        },
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateName": "my-launch-template",
                "Version": "1"
            },
                 "Overrides": [
                       {
                           "InstanceType": "r4.2xlarge",
                           "Placement": {
                               "AvailabilityZone": "us-west-2a"
                           },
                      },
                       {
                           "InstanceType": "m4.2xlarge",
                           "Placement": {
                               "AvailabilityZone": "us-west-2b"
                           },
                       },
                       {
                           "InstanceType": "c5.2xlarge",
                           "Placement": {
                               "AvailabilityZone": "us-west-2b"
                           }
                       }
                 ]
            }
    ],
    "TargetCapacitySpecification": {
            "TotalTargetCapacity": 50,
            "DefaultTargetCapacityType": "spot"
```

```
    }
}
```

範例 9：在具有優先順序的容量最佳化叢集中啟動 Spot 執行個體

下列範例示範如何使用 Spot 配置策略來設定EC2叢集，以最佳化容量，同時盡力使用優先順序。

當您使用 `capacity-optimized-prioritized` 分配策略時，您可以使用 `Priority` 參數來指定 Spot 容量集區的優先順序，其中數字越小，優先順序越高。如果您對它們一視同仁，也可以為數個 Spot 容量集區設定相同的優先順序。如果您沒有設定集區的優先順序，集區將被視為最後一個優先順序。

若要排定 Spot 容量集區的優先順序，您必須將 `AllocationStrategy` 設為 `capacity-optimized-prioritized`。EC2艦隊將首先針對容量進行最佳化，但會盡最大努力遵守優先事項（例如，如果遵守優先順序不會對EC2艦隊提供最佳容量的能力造成重大影響）。對於必須將中斷可能性降至最低的工作負載來說，這是一個很好的選擇，而且某些執行個體類型的偏好也很重要。

在下列範例中，三個啟動規格指定三個 Spot 容量集區。每個集區都有優先順序，其中數字越小，優先順序越高。目標容量為 50 個 Spot 執行個體。EC2叢集嘗試以最高優先順序將 50 個 Spot 執行個體啟動至 Spot 容量集區，但會先針對容量進行最佳化。

```
{
    "SpotOptions": {
        "AllocationStrategy": "capacity-optimized-prioritized"
        },
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateName": "my-launch-template",
                "Version": "1"
            },
                 "Overrides": [
                        {
                           "InstanceType": "r4.2xlarge",
                           "Priority": 1,
                           "Placement": {
                               "AvailabilityZone": "us-west-2a"
                           },
                       },
                        {
```

```
                            "InstanceType": "m4.2xlarge",
                            "Priority": 2,
                            "Placement": {
                                "AvailabilityZone": "us-west-2b"
                            },
                        },
                        {
                            "InstanceType": "c5.2xlarge",
                            "Priority": 3,
                            "Placement": {
                                "AvailabilityZone": "us-west-2b"
                            }
                        }
                    ]
                }
    ],
    "TargetCapacitySpecification": {
            "TotalTargetCapacity": 50,
            "DefaultTargetCapacityType": "spot"
}
```

範例 10：在 price-capacity-optimized 叢集中啟動 Spot 執行個體

以下範例示範如何使用 Spot 配置策略來設定EC2叢集，以同時針對容量和最低價格進行最佳化。若要在將價格納入考慮的同時最佳化容量，您必須將 Spot `AllocationStrategy` 設定為 `price-capacity-optimized`。

在下列範例中，三個啟動規格指定三個 Spot 容量集區。目標容量為 50 個 Spot 執行個體。EC2叢集會嘗試針對正在啟動的執行個體數量，將 50 個 Spot 執行個體啟動至 Spot 容量集區，並針對正在啟動的執行個體數量，同時選擇價格最低的集區。

```
{
    "SpotOptions": {
        "AllocationStrategy": "price-capacity-optimized",
        "MinTargetCapacity": 2,
        "SingleInstanceType": true
    },
    "OnDemandOptions": {
        "AllocationStrategy": "lowest-price"
    },
    "LaunchTemplateConfigs": [
        {
```

```
            "LaunchTemplateSpecification": {
                "LaunchTemplateName": "my-launch-template",
                "Version": "1"
            },
                 "Overrides": [
                       {
                           "InstanceType": "r4.2xlarge",
                           "Placement": {
                               "AvailabilityZone": "us-west-2a"
                           },
                      },
                       {
                           "InstanceType": "m4.2xlarge",
                           "Placement": {
                               "AvailabilityZone": "us-west-2b"
                           },
                       },
                       {
                           "InstanceType": "c5.2xlarge",
                           "Placement": {
                               "AvailabilityZone": "us-west-2b"
                           }
                       }
                 ]
            }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 50,
        "OnDemandTargetCapacity":0,
        "SpotTargetCapacity":50,
        "DefaultTargetCapacityType": "spot"
    },
    "Type": "instant"
}
```

範例 11：設定以屬性為基礎的執行個體類型選取

下列範例示範如何將EC2叢集設定為使用屬性型執行個體類型選取項來識別執行個體類型。若要指定必要的執行個體屬性，您可以在 `InstanceRequirements` 結構中指定屬性。

在下列範例中，指定了兩個執行個體屬性：

- `VCpuCount`— 至少指 vCPUs 定 2。由於未指定最大值，因此沒有最大限制。

- `MemoryMiB` – 至少指定 4 MiB 的記憶體。由於未指定最大值，因此沒有最大限制。

任何具有 2 或更多記憶體 vCPUs 和 4 MiB 或更多記憶體的執行個體類型都會被識別。不過，當EC2叢集佈建叢集時，價格保護和配置策略可能會排除某些執行個體類型。

如需您可以指定的所有可能屬性的清單和說明，請參閱 Amazon EC2 API 參考InstanceRequirements中的。

```
{
 "SpotOptions": {
  "AllocationStrategy": "price-capacity-optimized"
 },
 "LaunchTemplateConfigs": [{
  "LaunchTemplateSpecification": {
   "LaunchTemplateName": "my-launch-template",
   "Version": "1"
  },
  "Overrides": [{
   "InstanceRequirements": {
    "VCpuCount": {
     "Min": 2
    },
    "MemoryMiB": {
     "Min": 4
    }
   }
  }]
 }],
 "TargetCapacitySpecification": {
  "TotalTargetCapacity": 20,
  "DefaultTargetCapacityType": "spot"
 },
 "Type": "instant"
}
```

範例CLI組態競價型叢集

您可以在JSON檔案中定義 Spot Fleet 組態，然後使用request-spot-fleet AWS CLI 指令參考該檔案來建立叢集，如下所示：

```
aws ec2 request-spot-fleet --spot-fleet-request-config file://file_name.json
```

下列範例說明各種 Spot 叢集使用案例的啟動設定。如需有關組態參數的更多資訊，請參閱《AWS CLI 指令參考》request-spot-fleet中的〈〉。如需建立 Spot 叢集的詳細資訊，請參閱建立 Spot 機群。

> **Note**
>
> 對於 Spot 機群，您無法在啟動範本或啟動規格中指定網路介面 ID。確保省略啟動範本或啟動規格中的 `NetworkInterfaceID` 參數。

範例

- 範例 1：使用區域中最低價格的可用區域或子網來啟動 競價型執行個體
- 範例 2：使用指定清單中最低價格的可用區域或子網來啟動 競價型執行個體
- 範例 3：使用指定清單中最低價格的執行個體類型來啟動 競價型執行個體
- 範例 4。覆寫請求的價格
- 範例 5：使用多樣化分配策略來啟動 Spot Fleet
- 範例 6：使用執行個體權重來啟動 Spot Fleet
- 範例 7：使用隨需容量來啟動 Spot Fleet
- 範例 8：設定容量重新平衡以啟動替代 競價型執行個體
- 範例 9：在容量最佳化機群中啟動 Spot 執行個體
- 範例 10：使用優先順序在容量最佳化機群中啟動 Spot 執行個體
- 範例 11：在 priceCapacityOptimized 叢集中啟動 Spot 執行個體
- 範例 12：設定屬性型執行個體類型選取

範例 1：使用區域中最低價格的可用區域或子網來啟動 競價型執行個體

下列的範例指定了單一啟動規格，其中未包含可用區域或子網。Spot Fleet 會在具有預設子網的最低價格可用區域中，啟動執行個體。您所支付的價格不會超過隨需價格。

```
{
  "TargetCapacity": 20,
  "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
  "LaunchSpecifications": [
      {
          "ImageId": "ami-1a2b3c4d",
```

```
            "KeyName": "my-key-pair",
            "SecurityGroups": [
                {
                    "GroupId": "sg-1a2b3c4d"
                }
            ],
            "InstanceType": "m3.medium",
            "IamInstanceProfile": {
                "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
            }
        }
    ]
}
```

範例 2：使用指定清單中最低價格的可用區域或子網來啟動 競價型執行個體

下列範例會指定兩個具有不同可用區域或子網路的啟動規格，但執行個體類型和AMI相同。

可用區域

Spot Fleet 會在您所指定最低價格可用區域的預設子網中，啟動執行個體。

```
{
  "TargetCapacity": 20,
  "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
  "LaunchSpecifications": [
      {
          "ImageId": "ami-1a2b3c4d",
          "KeyName": "my-key-pair",
          "SecurityGroups": [
              {
                  "GroupId": "sg-1a2b3c4d"
              }
          ],
          "InstanceType": "m3.medium",
          "Placement": {
              "AvailabilityZone": "us-west-2a, us-west-2b"
          },
          "IamInstanceProfile": {
              "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
          }
      }
  ]
```

```
}
```

子網

您可以指定預設子網路或非預設子網路，而非預設子網路可以來自預設或非預設子網路。VPC VPCSpot 服務會在最低價格可用區域的任何可用子網中，啟動執行個體。

您不能在 Spot Fleet 請求中指定來自同一個可用區域的不同子網。

```
{
  "TargetCapacity": 20,
  "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
  "LaunchSpecifications": [
      {
          "ImageId": "ami-1a2b3c4d",
          "KeyName": "my-key-pair",
          "SecurityGroups": [
              {
                  "GroupId": "sg-1a2b3c4d"
              }
          ],
          "InstanceType": "m3.medium",
          "SubnetId": "subnet-a61dafcf, subnet-65ea5f08",
          "IamInstanceProfile": {
              "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
          }
      }
  ]
}
```

如果執行個體是預設啟動的VPC，則預設會收到公開IPv4位址。如果執行個體是以非預設值啟動VPC，則預設不會收到公開IPv4位址。使用啟動規格中的網路介面，將公用IPv4位址指派給在非預設值VPC中啟動的執行個體。指定網路介面時，必須加入使用網路介面的子網 ID 和安全群組 ID。

```
  ...
      {
          "ImageId": "ami-1a2b3c4d",
          "KeyName": "my-key-pair",
          "InstanceType": "m3.medium",
          "NetworkInterfaces": [
              {
                  "DeviceIndex": 0,
                  "SubnetId": "subnet-1a2b3c4d",
```

```
                "Groups": [ "sg-1a2b3c4d" ],
                "AssociatePublicIpAddress": true
            }
        ],
        "IamInstanceProfile": {
            "Arn": "arn:aws:iam::880185128111:instance-profile/my-iam-role"
        }
    }
...
```

範例 3：使用指定清單中最低價格的執行個體類型來啟動 競價型執行個體

下列範例會指定兩個具有不同執行個體類型但相同的啟動設定，以AMI及可用區域或子網路。Spot Fleet 會使用具有最低價格的指定執行個體類型，來啟動執行個體。

可用區域

```
{
  "TargetCapacity": 20,
  "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
  "LaunchSpecifications": [
      {
          "ImageId": "ami-1a2b3c4d",
          "SecurityGroups": [
              {
                  "GroupId": "sg-1a2b3c4d"
              }
          ],
          "InstanceType": "c5.4xlarge",
          "Placement": {
            "AvailabilityZone": "us-west-2b"
          }
      },
      {
          "ImageId": "ami-1a2b3c4d",
          "SecurityGroups": [
              {
                  "GroupId": "sg-1a2b3c4d"
              }
          ],
          "InstanceType": "r3.8xlarge",
          "Placement": {
              "AvailabilityZone": "us-west-2b"
```

```
            }
        }
    ]
}
```

子網

```
{
  "TargetCapacity": 20,
  "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
  "LaunchSpecifications": [
      {
          "ImageId": "ami-1a2b3c4d",
          "SecurityGroups": [
              {
                  "GroupId": "sg-1a2b3c4d"
              }
          ],
          "InstanceType": "c5.4xlarge",
          "SubnetId": "subnet-1a2b3c4d"
      },
      {
          "ImageId": "ami-1a2b3c4d",
          "SecurityGroups": [
              {
                  "GroupId": "sg-1a2b3c4d"
              }
          ],
          "InstanceType": "r3.8xlarge",
          "SubnetId": "subnet-1a2b3c4d"
      }
  ]
}
```

範例 4。覆寫請求的價格

我們建議您使用預設的最高價格，也就是隨需價格。如果想要的話，您可以指定機群請求的最高價格，和個別啟動規格的最高價格。

下列的範例指定了機群請求的最高價格，和三種啟動規格中其中兩種的最高價格。對於未指定最高價格的任何啟動規格，則會使用機群請求的最高價格。Spot Fleet 會使用具有最低價格的執行個體類型，來啟動執行個體。

可用區域

```
{
  "SpotPrice": "1.00",
  "TargetCapacity": 30,
  "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
  "LaunchSpecifications": [
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "c3.2xlarge",
          "Placement": {
              "AvailabilityZone": "us-west-2b"
          },
          "SpotPrice": "0.10"
      },
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "c3.4xlarge",
          "Placement": {
              "AvailabilityZone": "us-west-2b"
          },
          "SpotPrice": "0.20"
      },
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "c3.8xlarge",
          "Placement": {
              "AvailabilityZone": "us-west-2b"
          }
      }
    ]
}
```

子網

```
{
  "SpotPrice": "1.00",
  "TargetCapacity": 30,
  "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
  "LaunchSpecifications": [
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "c3.2xlarge",
```

```
            "SubnetId": "subnet-1a2b3c4d",
            "SpotPrice": "0.10"
        },
        {
            "ImageId": "ami-1a2b3c4d",
            "InstanceType": "c3.4xlarge",
            "SubnetId": "subnet-1a2b3c4d",
            "SpotPrice": "0.20"
        },
        {
            "ImageId": "ami-1a2b3c4d",
            "InstanceType": "c3.8xlarge",
            "SubnetId": "subnet-1a2b3c4d"
        }
    ]
}
```

範例 5：使用多樣化分配策略來啟動 Spot Fleet

下列範例使用 `diversified` 分配策略。啟動規格具有不同的執行個體類型，但相同AMI，可用區域或子網路。Spot Fleet 會將 30 個執行個體分配給 3 種啟動規格，如此每種規格就有 10 個執行個體。如需詳細資訊，請參閱使用配置策略來確定EC2叢集或 Spot 叢集如何滿足 Spot 和隨需容量。

可用區域

```
{
  "SpotPrice": "0.70",
  "TargetCapacity": 30,
  "AllocationStrategy": "diversified",
  "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
  "LaunchSpecifications": [
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "c4.2xlarge",
          "Placement": {
              "AvailabilityZone": "us-west-2b"
          }
      },
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "m3.2xlarge",
          "Placement": {
              "AvailabilityZone": "us-west-2b"
```

```
            }
        },
        {
            "ImageId": "ami-1a2b3c4d",
            "InstanceType": "r3.2xlarge",
            "Placement": {
                "AvailabilityZone": "us-west-2b"
            }
        }
    ]
}
```

子網

```
{
    "SpotPrice": "0.70",
    "TargetCapacity": 30,
    "AllocationStrategy": "diversified",
    "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
    "LaunchSpecifications": [
        {
            "ImageId": "ami-1a2b3c4d",
            "InstanceType": "c4.2xlarge",
            "SubnetId": "subnet-1a2b3c4d"
        },
        {
            "ImageId": "ami-1a2b3c4d",
            "InstanceType": "m3.2xlarge",
            "SubnetId": "subnet-1a2b3c4d"
        },
        {
            "ImageId": "ami-1a2b3c4d",
            "InstanceType": "r3.2xlarge",
            "SubnetId": "subnet-1a2b3c4d"
        }
    ]
}
```

在其中一個可用區域發生中斷時，增加 Spot 請求可以按EC2容量滿足的機會的最佳做法是跨區域進行多樣化。針對此案例，請將可供您使用的各可用區域納入啟動規格之中。不要每次都使用相同的子網，而是使用三個唯一的子網 (各個都映射至不同的區域)。

可用區域

```
{
  "SpotPrice": "0.70",
  "TargetCapacity": 30,
  "AllocationStrategy": "diversified",
  "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
  "LaunchSpecifications": [
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "c4.2xlarge",
          "Placement": {
              "AvailabilityZone": "us-west-2a"
          }
      },
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "m3.2xlarge",
          "Placement": {
              "AvailabilityZone": "us-west-2b"
          }
      },
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "r3.2xlarge",
          "Placement": {
              "AvailabilityZone": "us-west-2c"
          }
      }
  ]
}
```

子網

```
{
    "SpotPrice": "0.70",
    "TargetCapacity": 30,
    "AllocationStrategy": "diversified",
    "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
    "LaunchSpecifications": [
        {
            "ImageId": "ami-1a2b3c4d",
            "InstanceType": "c4.2xlarge",
            "SubnetId": "subnet-1a2b3c4d"
        },
```

```
        {
            "ImageId": "ami-1a2b3c4d",
            "InstanceType": "m3.2xlarge",
            "SubnetId": "subnet-2a2b3c4d"
        },
        {
            "ImageId": "ami-1a2b3c4d",
            "InstanceType": "r3.2xlarge",
            "SubnetId": "subnet-3a2b3c4d"
        }
    ]
}
```

範例 6：使用執行個體權重來啟動 Spot Fleet

下列範例使用執行個體權重，這表示價格為每單位小時而非每執行個體小時的價格。每個啟動組態都列出了不同的執行個體類型和不同的權重。Spot Fleet 會使用具有最低每單位小時價格的執行個體類型。Spot Fleet 會將目標容量除以執行個體的權重，來計算出要啟動的 Spot Fleet 數量。如果結果不是整數，則 Spot Fleet 會將其捨入到下一個整數，以便您的機群大小不低於其目標容量。

如果 `r3.2xlarge` 的請求成功，Spot 會佈建 4 個這類執行個體。將 20 除以 6，會得出總共 3.33 個執行個體，接著再四捨五入為 4 個執行個體。

如果 `c3.xlarge` 的請求成功，Spot 會佈建 7 個這類執行個體。將 20 除以 3，會得出總共 6.66 個執行個體，接著再四捨五入為 7 個執行個體。

如需詳細資訊，請參閱使用執行個體加權來管理EC2叢集或 Spot 叢集的成本和效能。

可用區域

```
{
  "SpotPrice": "0.70",
  "TargetCapacity": 20,
  "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
  "LaunchSpecifications": [
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "r3.2xlarge",
          "Placement": {
              "AvailabilityZone": "us-west-2b"
          },
          "WeightedCapacity": 6
```

```
      },
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "c3.xlarge",
          "Placement": {
              "AvailabilityZone": "us-west-2b"
          },
          "WeightedCapacity": 3
      }
    ]
}
```

子網

```
{
  "SpotPrice": "0.70",
  "TargetCapacity": 20,
  "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
  "LaunchSpecifications": [
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "r3.2xlarge",
          "SubnetId": "subnet-1a2b3c4d",
          "WeightedCapacity": 6
      },
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "c3.xlarge",
          "SubnetId": "subnet-1a2b3c4d",
          "WeightedCapacity": 3
      }
  ]
}
```

範例 7：使用隨需容量來啟動 Spot Fleet

為確保永遠有執行個體容量，您可以在 Spot Fleet 請求中加入對隨需容量的請求。如果有容量，則隨需請求一律已履行。如有容量和可用性，則目標容量的餘額已履行做為 Spot 使用。

下列範例將所要的目標容量指定為 10，其中 5 個必須為隨需容量。未指定 Spot 容量；它暗含在目標容量減去隨需容量的餘額中。如果有可用的 Amazon 容量和可用性，Amazon 會以隨需的形式EC2啟動 5 個容量單位，以及 5 個EC2容量單位 (10-5=5) 做為 Spot。

```
{
  "IamFleetRole": "arn:aws:iam::781603563322:role/aws-ec2-spot-fleet-tagging-role",
  "AllocationStrategy": "lowestPrice",
  "TargetCapacity": 10,
  "SpotPrice": null,
  "ValidFrom": "2018-04-04T15:58:13Z",
  "ValidUntil": "2019-04-04T15:58:13Z",
  "TerminateInstancesWithExpiration": true,
  "LaunchSpecifications": [],
  "Type": "maintain",
  "OnDemandTargetCapacity": 5,
  "LaunchTemplateConfigs": [
    {
      "LaunchTemplateSpecification": {
        "LaunchTemplateId": "lt-0dbb04d4a6cca5ad1",
        "Version": "2"
      },
      "Overrides": [
        {
          "InstanceType": "t2.medium",
          "WeightedCapacity": 1,
          "SubnetId": "subnet-d0dc51fb"
        }
      ]
    }
  ]
}
```

範例 8：設定容量重新平衡以啟動替代 競價型執行個體

下列範例會將 Spot 叢集設定為在 Amazon 針對叢集中的競價型執行個EC2體發出重新平衡建議時啟動替代 Spot 執行個體。若要為 `ReplacementStrategy` 設定 競價型執行個體 的自動替代，請指定 `launch-before-terminate`。若要設定從啟動新替代 Spot 執行個體到自動刪除舊 Spot 執行個體的時間延遲，請針對 `termination-delay`，指定一值 (以秒為單位)。如需詳細資訊，請參閱組態選項。

> **Note**
>
> 建議僅在您可以預測執行個體關閉程序將需要多長時間才能完成時，才使用 `launch-before-terminate`。這可確保只在關機程序完成之後，才會終止舊執行個體。所有執行個體在執行時，您需要支付所有執行個體的費用。

容量重新平衡策略的有效性取決於 Spot Fleet 請求中指定的 Spot 容量集區數量。我們建議您使用多樣化的執行個體類型和可用區域的集合來設定機群，若要 `AllocationStrategy`，請指定 `capacityOptimized`。如需當設定 Spot Fleet 進行容量重新平衡時應考量哪些項目的詳細資訊，請參閱 使用叢集和 S EC2 pot 叢集中的容量重新平衡來取代有風險的 Spot 執行個體。

```
{
    "SpotFleetRequestConfig": {
        "AllocationStrategy": "capacityOptimized",
        "IamFleetRole": "arn:aws:iam::000000000000:role/aws-ec2-spot-fleet-tagging-
role",
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "LaunchTemplate",
                    "Version": "1"
                },
                 "Overrides": [
                        {
                            "InstanceType": "c3.large",
                            "WeightedCapacity": 1,
                            "Placement": {
                                "AvailabilityZone": "us-east-1a"
                            }
                        },
                        {
                            "InstanceType": "c4.large",
                            "WeightedCapacity": 1,
                            "Placement": {
                                "AvailabilityZone": "us-east-1a"
                            }
                        },
                        {
                            "InstanceType": "c5.large",
                            "WeightedCapacity": 1,
                            "Placement": {
```

```
                                "AvailabilityZone": "us-east-1a"
                            }
                        }
                ]
            }
    ],
        "TargetCapacity": 5,
        "SpotMaintenanceStrategies": {
            "CapacityRebalance": {
                "ReplacementStrategy": "launch-before-terminate",
                "TerminationDelay": "720"
            }
        }
    }
}
```

範例 9：在容量最佳化機群中啟動 Spot 執行個體

下列範例示範如何使用可最佳化容量的 Spot 分配策略來設定 Spot Fleet 。若要最佳化容量，您必須將 `AllocationStrategy` 設為 `capacityOptimized`。

在下列範例中，三個啟動規格指定三個 Spot 容量集區。目標容量為 50 個 Spot 執行個體。Spot Fleet 嘗試將 50 個 Spot 執行個體啟動到 Spot 容量集區中，且擁有啟動中執行個體數目之最佳容量。

```
{
    "TargetCapacity": "50",
    "SpotFleetRequestConfig": {
        "AllocationStrategy": "capacityOptimized",
    },
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateName": "my-launch-template",
                "Version": "1"
            },
                 "Overrides": [
                       {
                           "InstanceType": "r4.2xlarge",
                           "AvailabilityZone": "us-west-2a"
                        },
                        {
                           "InstanceType": "m4.2xlarge",
                           "AvailabilityZone": "us-west-2b"
```

```
                        },
                        {
                            "InstanceType": "c5.2xlarge",
                            "AvailabilityZone": "us-west-2b"
                        }
                    ]
            }
        ]
}
```

範例 10：使用優先順序在容量最佳化機群中啟動 Spot 執行個體

下列範例示範在盡最大努力使用優先順序時，如何使用可最佳化容量的 Spot 分配策略來設定 Spot Fleet 。

當您使用 `capacityOptimizedPrioritized` 分配策略時，您可以使用 `Priority` 參數來指定 Spot 容量集區的優先順序，其中數字越小，優先順序越高。如果您對它們一視同仁，也可以為數個 Spot 容量集區設定相同的優先順序。如果您沒有設定集區的優先順序，集區將被視為最後一個優先順序。

若要排定 Spot 容量集區的優先順序，您必須將 `AllocationStrategy` 設為 `capacityOptimizedPrioritized`。Spot Fleet 會先針對容量進行最佳化，但會盡力遵守優先順序 (例如，如果遵守優先順序不會顯著影響 Spot Fleet 佈建最佳容量的能力)。對於必須將中斷可能性降至最低的工作負載來說，這是一個很好的選擇，而且某些執行個體類型的偏好也很重要。

在下列範例中，三個啟動規格指定三個 Spot 容量集區。每個集區都有優先順序，其中數字越小，優先順序越高。目標容量為 50 個 Spot 執行個體。Spot Fleet 嘗試以最高優先順序在 Spot 容量集區中啟動 50 個 Spot 執行個體，但首先針對容量進行最佳化。

```
{
    "TargetCapacity": "50",
    "SpotFleetRequestConfig": {
        "AllocationStrategy": "capacityOptimizedPrioritized"
    },
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateName": "my-launch-template",
                "Version": "1"
            },
             "Overrides": [
```

```
                        {
                            "InstanceType": "r4.2xlarge",
                            "Priority": 1,
                            "AvailabilityZone": "us-west-2a"
                        },
                        {
                            "InstanceType": "m4.2xlarge",
                            "Priority": 2,
                            "AvailabilityZone": "us-west-2b"
                        },
                        {
                            "InstanceType": "c5.2xlarge",
                            "Priority": 3,
                            "AvailabilityZone": "us-west-2b"
                        }
                    ]
                }
        ]
}
```

範例 11：在 priceCapacityOptimized 叢集中啟動 Spot 執行個體

下列範例示範如何使用可最佳化容量且價格最低的 Spot 分配策略來設定 Spot 機群。若要在將價格納入考慮的同時最佳化容量，您必須將 Spot `AllocationStrategy` 設定為 `priceCapacityOptimized`。

在下列範例中，三個啟動規格指定三個 Spot 容量集區。目標容量為 50 個 Spot 執行個體。Spot 機群嘗試將 50 個 Spot 執行個體啟動到 Spot 容量集區中，且擁有啟動中執行個體數目之最佳容量，同時還會選擇價格最低的集區。

```
{
    "SpotFleetRequestConfig": {
        "AllocationStrategy": "priceCapacityOptimized",
        "OnDemandAllocationStrategy": "lowestPrice",
        "ExcessCapacityTerminationPolicy": "default",
        "IamFleetRole": "arn:aws:iam::111111111111:role/aws-ec2-spot-fleet-tagging-
role",
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateId": "lt-0123456789example",
                    "Version": "1"
```

```
                },
                "Overrides": [
                     {
                           "InstanceType": "r4.2xlarge",
                           "AvailabilityZone": "us-west-2a"
                      },
                      {
                           "InstanceType": "m4.2xlarge",
                           "AvailabilityZone": "us-west-2b"
                      },
                      {
                           "InstanceType": "c5.2xlarge",
                           "AvailabilityZone": "us-west-2b"
                      }
                ]
            }
        ],
        "TargetCapacity": 50,
        "Type": "request"
    }
}
```

範例 12：設定屬性型執行個體類型選取

下列範例示範如何將 Spot 機群設定為使用屬性型執行個體類型選取來識別執行個體類型。若要指定必要的執行個體屬性，您可以在 `InstanceRequirements` 結構中指定屬性。

在下列範例中，指定了兩個執行個體屬性：

- `VCpuCount`— 至少指 vCPUs 定 2。由於未指定最大值，因此沒有最大限制。
- `MemoryMiB` – 至少指定 4 MiB 的記憶體。由於未指定最大值，因此沒有最大限制。

任何具有 2 或更多記憶體 vCPUs 和 4 MiB 或更多記憶體的執行個體類型都會被識別。但是，當 Spot 機群佈建機群時，價格保護和分配策略可能會排除某些執行個體類型。

如需您可以指定的所有可能屬性的清單和說明，請參閱 Amazon EC2 API 參考InstanceRequirements中的。

```
{
 "AllocationStrategy": "priceCapacityOptimized",
 "TargetCapacity": 20,
```

```
 "Type": "request",
 "LaunchTemplateConfigs": [{
  "LaunchTemplateSpecification": {
   "LaunchTemplateName": "my-launch-template",
   "Version": "1"
  },
  "Overrides": [{
   "InstanceRequirements": {
    "VCpuCount": {
     "Min": 2
    },
    "MemoryMiB": {
     "Min": 4
    }
   }
  }]
 }]
}
```

EC2車隊和競價型艦隊的配額

您的每項 AWS 服務都 AWS 帳戶 有預設配額 (先前稱為限制)。除非另有說明，否則每個配額都是區域特定規定。

一般 Amazon EC2 配額適用於EC2叢集或 Spot 叢集啟動的執行個體，例如競價型執行個體限制和磁碟區限制。

此外，您還 AWS 帳戶 具有下列與「艦隊」和「競價型EC2艦隊」相關的配額：

配額說明	配額
每個區域類型以及active、deleted_running 和州request中的艦隊maintain和 Spot 叢集數 EC2 cancelled_running	1,000 [1] [2] [3]
類型的EC2艦隊數 instant	無限制
叢集和 Spot 叢集類型和 Spot EC2 叢集的 Spot 容量集區數量 (執行個體類型和子網路的唯一組合) maintain request	300 [1]

配額說明	配額
EC2叢集類型的 Spot 容量集區數量 (執行個體類型和子網路的唯一組合) `instant`	無限制
啟動規格中使用者資料的大小	16 KB [2]
每個EC2叢集或 Spot 叢集的目標容量	10,000
一個區域中所有EC2車隊和 Spot 艦隊的目標容量	100,000 [1]
EC2叢集請求或 Spot 叢集請求無法跨越區域。	
EC2叢集請求或 Spot 叢集請求無法跨越來自同一可用區域的不同子網路。	

[1] 這些配額同時適用於您的EC2車隊和 Spot 車隊。

[2] 這些是硬性配額。您無法要求提高這些配額。

[3] 刪除EC2叢集或取消 Spot Fleet 請求後，如果您指定叢集不應在刪除或取消請求時終止其 Spot 執行個體，則叢集請求會進入 (叢集) 或 `deleted_running` `cancelled_running` (Spot EC2 Fleet) 狀態，並且執行個體會繼續執行，直到執行個體被中斷或您手動終止。如果您終止執行個體，叢集請求會進入 `deleted_terminating` (EC2叢集) 或 `cancelled_terminating` (Spot Fleet) 狀態，且不會計入此配額中。如需詳細資訊，請參閱 刪除EC2叢集請求和叢集中的執行個體 和 取消 (刪除) Spot 叢集請求。

請求增加目標容量的配額

如果您需要的容量超過目標容量的預設配額，則可以請求增加配額。

請求增加目標容量的配額

1. 開啟「中 AWS Support 心建立案例」表單。
2. 選擇提高服務限制。
3. 在「限制類型」中選擇「EC2艦隊」。
4. 在「地區」中，選擇要求提高配額的 AWS 地區。

5. 對於 Limit (限制)，請選擇 Target Fleet Capacity per Fleet (in units) (每個機群的目標機群容量 (單位))，或者 Target Fleet Capacity per Region (in units) (每個區域的目標機群容量 (單位))，具體取決於您要增加的配額。
6. 針對 New limit value (新的限值)，輸入新的配額值。
7. 若要請求增加另一個配額，請選擇 Add another request (新增另一個請求)，然後重複步驟 4–6。
8. 對於 Use case description (使用案例描述)，輸入您請求增加配額的原因。
9. 在 Contact options (聯絡選項)，請指定您偏好的聯絡語言和聯絡方式。
10. 選擇提交。

Amazon 網絡 EC2

Amazon VPC 可讓您將 AWS 資源 (例如 Amazon EC2 執行個體) 啟動到專用於您 AWS 帳戶的虛擬網路中，稱為虛擬私有雲 (VPC)。啟動執行個體時，您可以從中選取子網路VPC。執行個體設定為主要網路介面，這是邏輯虛擬網路卡。執行個體會從子網路的位址接收主要私有 IP 位IPv4址，並指派給主要網路介面。

您可以控制執行個體是否從 Amazon 的公有 IP 地址集區接收公有 IP 地址。執行個體的公有 IP 地址只會與您的執行個體關聯，直到執行個體停止或終止為止。如果您需要永久的公用 IP 位址，您可以為您的 AWS 帳戶配置彈性 IP 位址，並將其與執行個體或網路介面建立關聯。彈性 IP 地址會與您的 AWS 帳戶保持關聯，直到您釋放它為止，您可以根據需要將其從一個實例移動到另一個實例。您可以將自己的 IP 地址範圍帶到您的 AWS 帳戶中，它會顯示為地址集區，然後從您的地址集區配置彈性 IP 地址。

若要提高網路效能並減少延遲，您可以在配置群組中啟動執行個體。使用增強型網路，您可以大幅提高每秒封包 (PPS) 效能。您可以使用 Elastic Fabric 配接卡 () 來加速高效能運算和機器學習應用程式，而 Elastic Fabric Adapter (EFA) 是可連接至支援執行個體類型的網路裝置。

功能

- 區域 (Region) 和區域 (Zone)
- Amazon EC2 實例 IP 定址
- Amazon EC2 執行個體主機名稱
- 將您自己的 IP 地址（BYOIP）帶到 Amazon EC2
- 彈性 IP 地址
- 彈性網絡介面
- Amazon EC2 實例網絡帶寬
- Amazon EC2 執行個體上的增強聯網
- 適用於 Amazon 上的機器學習HPC工作負載的彈性光纖 EC2
- Amazon EC2 實例拓撲
- Amazon EC2 執行個體的放置群組
- EC2執行個體的網路最大傳輸單元 (MTU)
- 適用於EC2執行個體的虛擬私有雲

區域 (Region) 和區域 (Zone)

Amazon 託管EC2在世界各地的多個地點。這些位置由可用區域 AWS 區域、Local Zones 和 Wavelength 區組成。 AWS Outposts

- 各個區域為獨立的地理區域。
- 可用區域是每個區域內的多個隔離位置。
- Local Zones 可讓您將資源 (例如運算和儲存) 放置在靠近最終使用者的多個位置。
- AWS Outposts 將原生 AWS 服務、基礎架構和作業模型帶到幾乎任何資料中心、主機代管空間或內部部署設施。
- Wavelength 區域可讓開發人員為 5G 裝置與最終使用者建立提供極低延遲的應用程式。Wavelength 將標準 AWS 運算和儲存服務部署到電信業者的 5G 網路邊緣。

AWS 操作 state-of-the-art、高可用性的資料中心。儘管故障極為少見，但仍可能影響相同位置內執行個體的可用性。若您將所有執行個體都託管於單一位置，一旦該位置受故障影響，所有執行個體都將無法使用。

內容

- 區域
  - 可用的區域
  - 區域端點
- 可用區域
  - AZ IDs
  - 可用的可用區域
  - 可用區域的執行個體
- 本機區域
  - 可用 Local Zones
  - Local Zones 的執行個體
- Wavelength 區域
  - 可用 Wavelength 區
  - Wavelength 區域中的實例
- AWS Outposts
  - 前哨上的執行個體

- 前哨機架上的磁碟區

區域

每個區域皆設計為與其他區域隔離。如此可達到最高的容錯能力與穩定性。

當檢視資源時，您只會看到與您指定的區域繫結的資源。這是因為區域彼此隔離，而且我們不會自動跨區域複寫資源。

啟動執行個體時，您必須選取AMI位於相同區域中的執行個體。如果位AMI於另一個區域中，您可以將其複製AMI到您正在使用的區域。如需詳細資訊，請參閱複製 Amazon EC2 AMI。

請注意，跨區域傳輸資料會產生費用。如需詳細資訊，請參閱 Amazon EC2 定價-資料傳輸。

可用的區域

您的帳戶決定您可用的區域。

- AWS 帳戶 提供多個區域，因此您可以在符合需求的位置啟動 Amazon EC2 執行個體。例如，您可能想在歐洲啟動執行個體，以拉近與歐洲客戶的距離或符合法規要求。
- AWS GovCloud (美國西部) 帳戶提供 (美國西部) 區域和 AWS GovCloud (美國東部) 區域的 AWS GovCloud 存取權。如需詳細資訊，請參閱AWS GovCloud (US)。
- Amazon AWS (中國) 帳戶僅提供北京和寧夏地區的存取權。如需詳細資訊，請參閱 Amazon Web Services in China (Amazon Web Services (中國))。

下表列出由提供的區域 AWS 帳戶。您無法描述或存取其他地區 AWS 帳戶，例如 AWS GovCloud (US) Regions 或中國區域。若要使用 2019 年 3 月 20 日之後引進的區域，您必須啟用該區域。如需詳細資訊，請參閱AWS Account Management 參考指南中的「指定您的帳戶可以使用的 AWS 區域」。

Code	名稱	選擇參加狀態
us-east-1	美國東部 (維吉尼亞北部)	非必要
us-east-2	美國東部 (俄亥俄)	非必要
us-west-1	美國西部 (加州北部)	非必要
us-west-2	美國西部 (奧勒岡)	非必要

Code	名稱	選擇參加狀態
af-south-1	非洲 (開普敦)	必要
ap-east-1	亞太區域 (香港)	必要
ap-south-2	亞太區域 (海德拉巴)	必要
ap-southeast-3	亞太區域 (雅加達)	必要
ap-southeast-5	亞太區域 (馬來西亞)	必要
ap-southeast-4	亞太區域 (墨爾本)	必要
ap-south-1	亞太區域 (孟買)	非必要
ap-northeast-3	亞太區域 (大阪)	非必要
ap-northeast-2	亞太區域 (首爾)	非必要
ap-southeast-1	亞太區域 (新加坡)	非必要
ap-southeast-2	亞太區域 (悉尼)	非必要
ap-northeast-1	亞太區域 (東京)	非必要
ca-central-1	加拿大 (中部)	非必要
ca-west-1	加拿大西部 (卡加利)	必要
cn-north-1	中國 (北京)	非必要
cn-northwest-1	中國 (寧夏)	非必要
eu-central-1	歐洲 (法蘭克福)	非必要
eu-west-1	歐洲 (愛爾蘭)	非必要
eu-west-2	歐洲 (倫敦)	非必要
eu-south-1	歐洲 (米蘭)	必要

Code	名稱	選擇參加狀態
eu-west-3	Europe (Paris)	非必要
eu-south-2	歐洲 (西班牙)	必要
eu-north-1	歐洲 (斯德哥爾摩)	非必要
eu-central-2	歐洲 (蘇黎世)	必要
il-central-1	以色列 (特拉維夫)	必要
me-south-1	Middle East (Bahrain)	必要
me-central-1	中東 (UAE)	必要
sa-east-1	南美洲 (聖保羅)	非必要

如需詳細資訊，請參閱 AWS 全球基礎設施。

區域端點

使用命令列介面或API動作處理執行個體時，必須指定其區域端點。如需有關 Amazon 區域和端點的詳細資訊EC2，請參閱 Amazon EC2開發人員指南中的 Amazon EC2 服務端點。

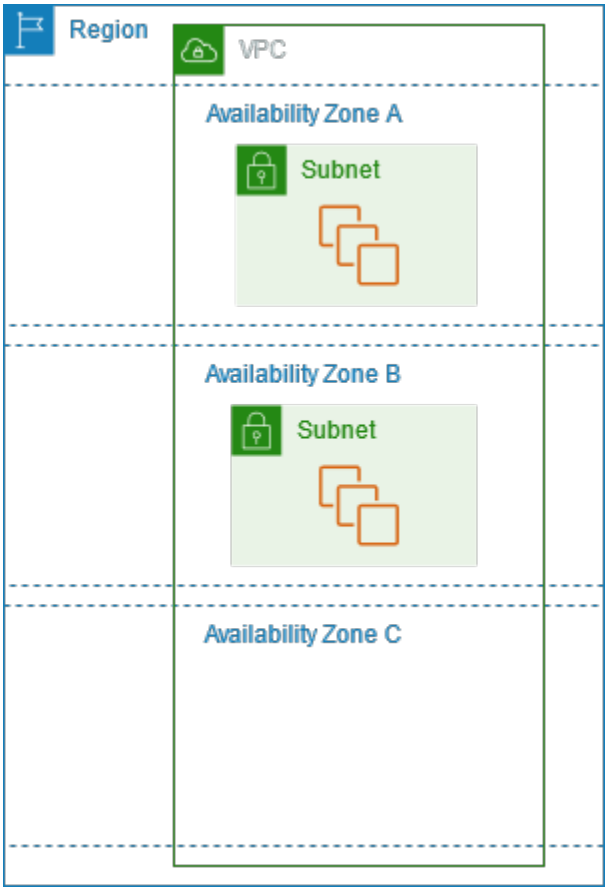
如需有關 AWS GovCloud (美國西部) 中端點和通訊協定的詳細資訊，請參閱AWS GovCloud (US) 使用手冊中的「服務端點」。

可用區域

每個區域擁有多個隔離位置，稱為可用區域。可用區域的代碼為其區域代碼，後續跟著一個字母識別符。例如：`us-east-1a`。

啟動執行個體時，請選取區域和虛擬私有雲 (VPC)，然後您可以從其中一個可用區域選取子網路，或讓我們為您選擇一個子網路。如果您要將執行個體分散至多個可用區域，但其中一個執行個體故障，則您可設計應用程式，讓其他可用區域內的執行個體來處理要求。您也可用彈性 IP 地址快速將地址重新映射到另一個可用區域中的執行個體，以遮罩某一可用區域內執行個體的故障。

下圖說明一 AWS 個區域中的多個可用區域。可用區域 A 和可用區域 B 各有一個子網，每個子網都有執行個體。可用區域 C 沒有子網，因此您無法在此可用區域中啟動執行個體。



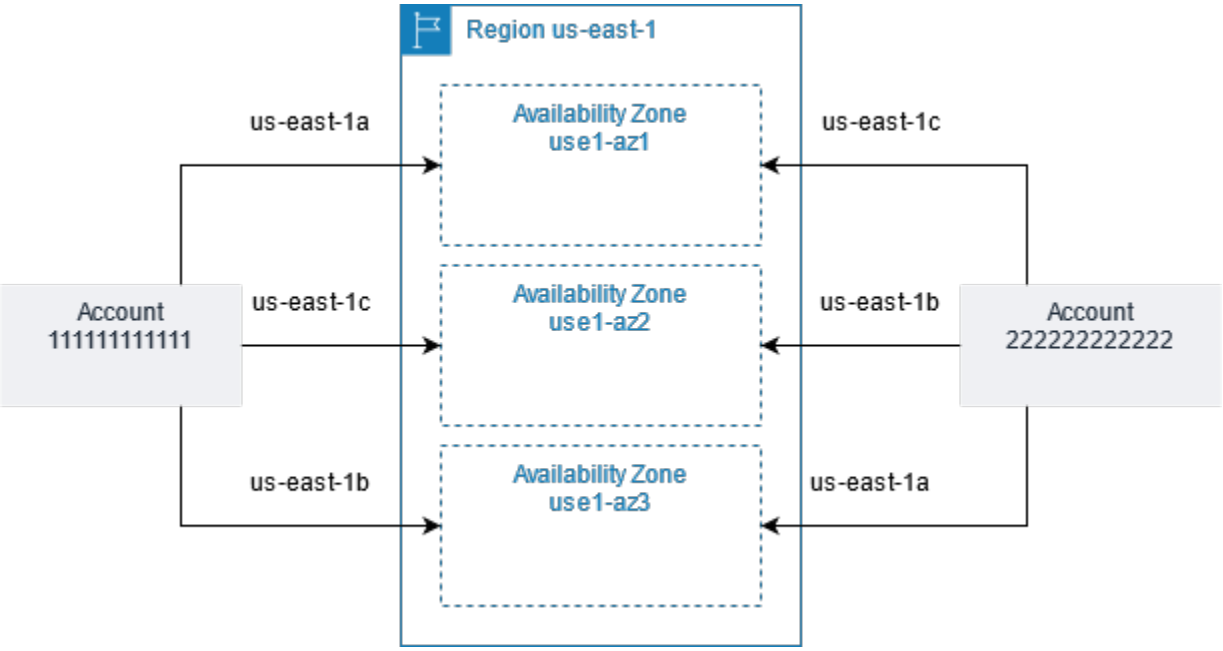
可用區域隨時間擴大時，我們擴展可用區域的能力將受到限制。假如出現此狀況，我們可能禁止您從受限的可用區域啟動執行個體，除非您在該可用區域已有執行個體。我們甚至可能需要為新帳戶將受限的可用區域從可用區域清單中移除。因此您帳戶在某區域能取得的可用區域數量可能與其他帳戶的數量不同。

AZ IDs

為了確保資源分散到某個區域的可用區域，我們會將可用區域獨立對應至最舊區域 AWS 帳戶 中每個區域的代碼。例如，您的`us-east-1a`的實際位置 AWS 帳戶 可能與其他位置不同 AWS 帳戶。`us-east-1a`

若要協調所有區域中帳戶 (即使對應可用區域的帳戶) 的可用區域，請使用 AZ IDs，這是可用區域的唯一且一致的識別碼。例如，`use1-az1`是區域的 AZ ID，並且在每個`us-east-1`區域中都有相同的實體位置 AWS 帳戶。您可以檢視帳戶IDs的 AZ，以判斷資源相對於其他帳戶中資源的實際位置。例如，如果您與另一個帳戶共享 AZ ID 為 `use1-az2` 的可用區域子網，則 AZ ID 也是 `use1-az2` 之可用區域中的該帳戶就可以使用此子網。

下圖說明兩個具有不同可用區域代碼至 AZ ID 映射的帳戶。



可用的可用區域

每個區域都有多個可用區域，如下列清單所示。

- 美國東部 (維吉尼亞北部) — use1-az1 use1-az2 use1-az3 | use1-az4 | use1-az5 | use1-az6
- 美國東部 (俄亥俄) — use2-az1 use2-az2 | use2-az3
- 美國西部 (加利佛尼亞北部) — usw1-az1 usw1-az2 | | usw1-az3 †
- 美國西部 (奧勒岡) — usw2-az1 usw2-az2 | usw2-az3 | usw2-az4
- 非洲 (開普敦) — afs1-az1 | afs1-az2 | afs1-az3
- 亞太區域 (香港) — ape1-az1 | ape1-az2 | ape1-az3
- 亞太區域 (海德拉巴) — aps2-az1 | aps2-az2 | aps2-az3
- 亞太區域 (雅加達) — apse3-az1 | apse3-az2 | apse3-az3
- 亞太區域 (馬來西亞) — apse5-az1 | apse5-az2 | apse5-az3
- 亞太區域 (墨爾本) — apse4-az1 | apse4-az2 | apse4-az3
- 亞太區域 (孟買) — aps1-az1 | aps1-az2 | aps1-az3
- 亞太區域 (大阪) — apne3-az1 apne3-az2 | apne3-az3
- 亞太區域 (首爾) — apne2-az1 apne2-az2 | apne2-az3 | apne2-az4
- 亞太區域 (新加坡) — apse1-az1 | apse1-az2 | apse1-az3
- 亞太區域 (雪梨) — apse2-az1 | apse2-az2 | apse2-az3

- 亞太區域 (東京) — apne1-az1 apne1-az2 | apne1-az3 | apne1-az4
- 加拿大（中部）— cac1-az1 | cac1-az2 | cac1-az4
- 加拿大西部 (卡加利) — caw1-az1 | caw1-az2 | caw1-az3
- 歐洲 (法蘭克福) — euc1-az1 | euc1-az2 | euc1-az3
- 歐洲 (愛爾蘭) — euw1-az1 | euw1-az2 | euw1-az3
- 歐洲 (倫敦) — euw2-az1 euw2-az2 | euw2-az3
- 歐洲 (米蘭) — eus1-az1 | eus1-az2 | eus1-az3
- 歐洲 (巴黎) — euw3-az1 euw3-az2 | euw3-az3
- 歐洲 (西班牙) — eus2-az1 | eus2-az2 | eus2-az3
- 歐洲 (斯德哥爾摩) — eun1-az1 eun1-az2 | eun1-az3
- 歐洲 (蘇黎世) — euc2-az1 euc2-az2 | euc2-az3
- 以色列（特拉維夫）— ilc1-az1 | ilc1-az2 | ilc1-az3
- 中東（巴林）— mes1-az1 | mes1-az2 | mes1-az3
- 中東 (UAE) — mec1-az1 mec1-az2 | mec1-az3
- 南美洲（聖保羅）— sae1-az1 sae1-az2 | sae1-az3
- AWS GovCloud (美國東部) — usge1-az1 | | usge1-az2 usge1-az3
- AWS GovCloud (美國西部) — usgw1-az1 | | usgw1-az2 usgw1-az3

† 較新的帳戶可以存取美國西部 (加利佛尼亞北部) 的兩個可用區域。

可用區域的執行個體

啟動執行個體時，選取可讓執行個體更接近特定客戶，或符合法律或您其他要求的區域。透過在不同的可用區域中啟動執行個體，您可以保護應用程式免於區域中單一位置的故障。

您可在啟動執行個體時在您使用的區域中選擇性指定可用區域。如果您沒有指定可用區域，我們會為您選取可用區域。啟動初始執行個體時，建議您接受預設的可用區域，因為這可讓我們依照系統運作狀態和可用容量為您選取最適合的可用區域。如果您啟動其他的執行個體，則只有當您的新執行個體必須接近或隔離於執行中的執行個體時，您才能指定可用區域。

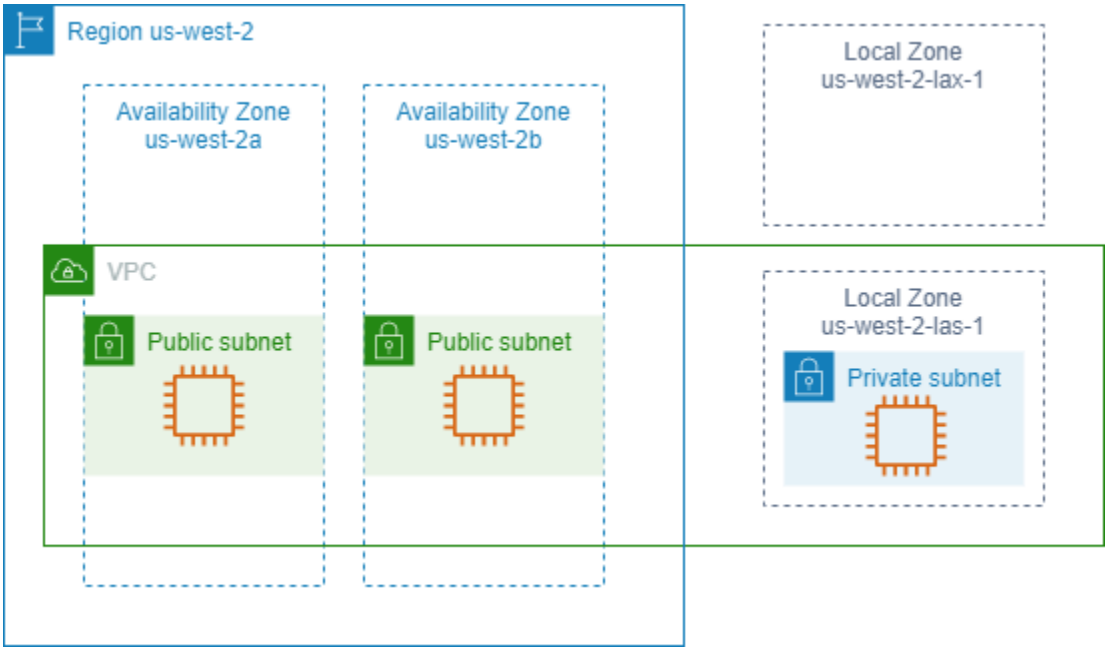
本機區域

「本地區域」是您使用者所在地理位置附近的「 AWS 區域」的延伸。Local Zones 具有自己的網際網路連線和支援 AWS Direct Connect，因此在本機區域中建立的資源可以透過低延遲通訊為本機使用者

提供服務。如需詳細資訊，請參閱什麼是 L AWS ocal Zones？ 在「L AWS ocal Zones 使用者指南」中。

本機區域的代碼為其區域代碼，後續跟著一個字母識別符，代表其實體位置。例如：`us-west-2-lax-1` 位於洛杉磯。

下圖說明「區 AWS 域」`us-west-2`、其兩個可用區域及其兩個「Local Zones」。VPC跨越可用區域和其中一個 Local Zones。中的每個區域都VPC有一個子網路，而每個子網路都有一個執行個體。



可用 Local Zones

如需可用 Local Zones 的清單，請參閱 L ocal Zones 使用者指南中的可用 L AWS ocal Zones。如需已宣佈的本機區域清單，請參閱 L AWS ocal Zones 位置。

Local Zones 的執行個體

若要使用 Local Zone，首先您必須啟用它。然後，在本機區域中建立子網路。您可以在啟動執行個體時指定本機區域子網路，以便將其置於本機區域的本機區域子網路中。

當您在本機區域中啟動執行個體時，也會從網路邊界群組配置 IP 位址。網路邊界群組是一組唯一的可用區域、Local Zones 或 Wavelength 區域，可從中 AWS 通告 IP 位址，例如。`us-west-2-lax-1a`您可以從網路邊界群組配置下列 IP 地址：

- 亞馬遜提供的彈性地址 IPv4
- 亞馬遜提供的IPv6VPC地址（僅適用於洛杉磯地區）

如需如何在本機區域中啟動執行個體的詳細資訊，請參閱 L AWS ocal Zones 使用者指南中的使用 L AWS ocal Zones 入門。

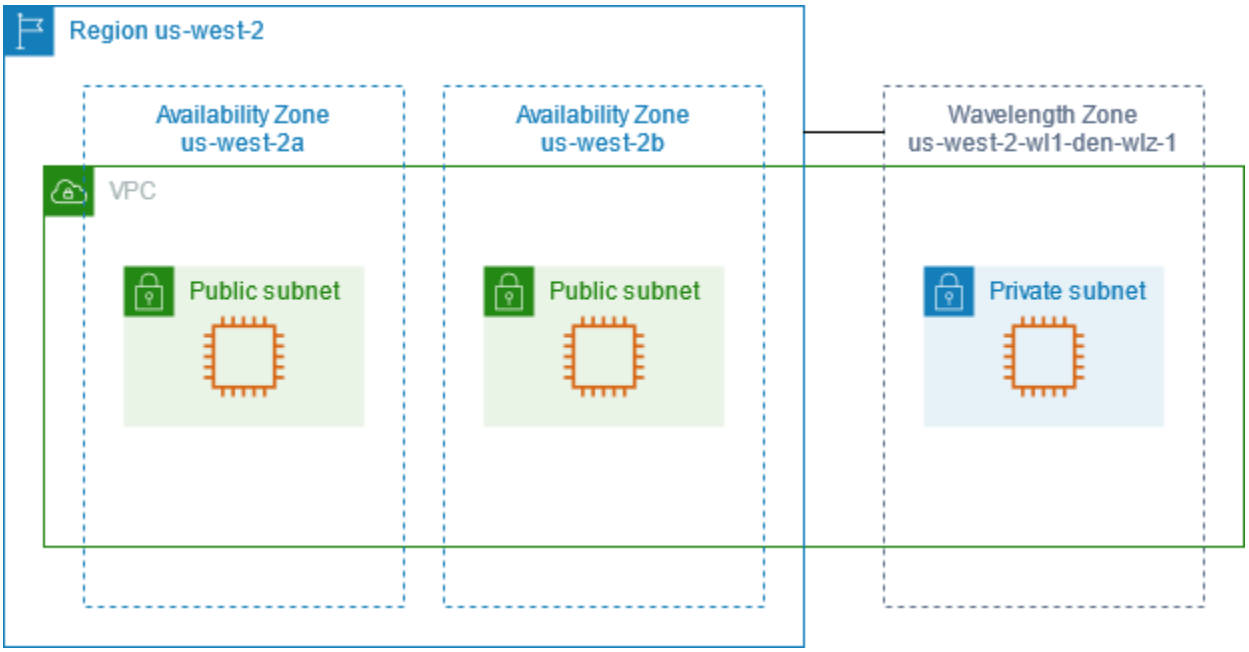
Wavelength 區域

AWS Wavelength 讓開發人員能夠建置可為行動裝置和使用者提供超低延遲的應用程式。Wavelength 將標準 AWS 運算和儲存服務部署到電信業者的 5G 網路邊緣。開發人員可以將虛擬私有雲 (VPC) 擴展到一個或多個 Wavelength 區域，然後使用 Amazon EC2 執行個體等 AWS 資源執行需要超低延遲和區域中 AWS 服務連線的應用程式。

Wavelength 區域是部署 Wavelength 基礎設施之電信業者位置的隔離區域。Wavelength 區域繫結至區域。Wavelength 區域是區域的邏輯延伸，並且由區域中的控制平面管理。

Wavelength 區域的代碼為其區域代碼，後續跟著一個字母識別符，代表其實體位置。例如：`us-east-1-wl1-bos-wlz-1` 位於波士頓。

下圖說明了區 AWS 域`us-west-2`，其兩個可用區域和一個 Wavelength 區。VPC跨越可用區域和 Wavelength 區域。中的每個區域都VPC有一個子網路，而每個子網路都有一個執行個體。



並非每個區域都有 Wavelength 區域。如需支援 Wavelength 區域的區域相關資訊，請參閱 AWS Wavelength 開發人員指南中的可用 Wavelength 區域。

可用 Wavelength 區

有關可用 Wavelength 區域的清單，請參閱指AWS Wavelength 南中的可用 Wavelength 區域。

Wavelength 區域中的實例

若要使用 Wavelength 區域，您必須先選擇加入「Zone (區域)」。然後，在 Wavelength 區域中創建一個子網。您可以在啟動執行個體時指定 Wavelength 子網路。您也可以從網路邊界群組配置電信業者 IP 地址，該網路邊界群組是一組唯一的可用區域、Local Zones 或 Wavelength 區域， AWS 會從中公告 IP 地址，例如，`us-east-1-wl1-bos-wlz-1`。

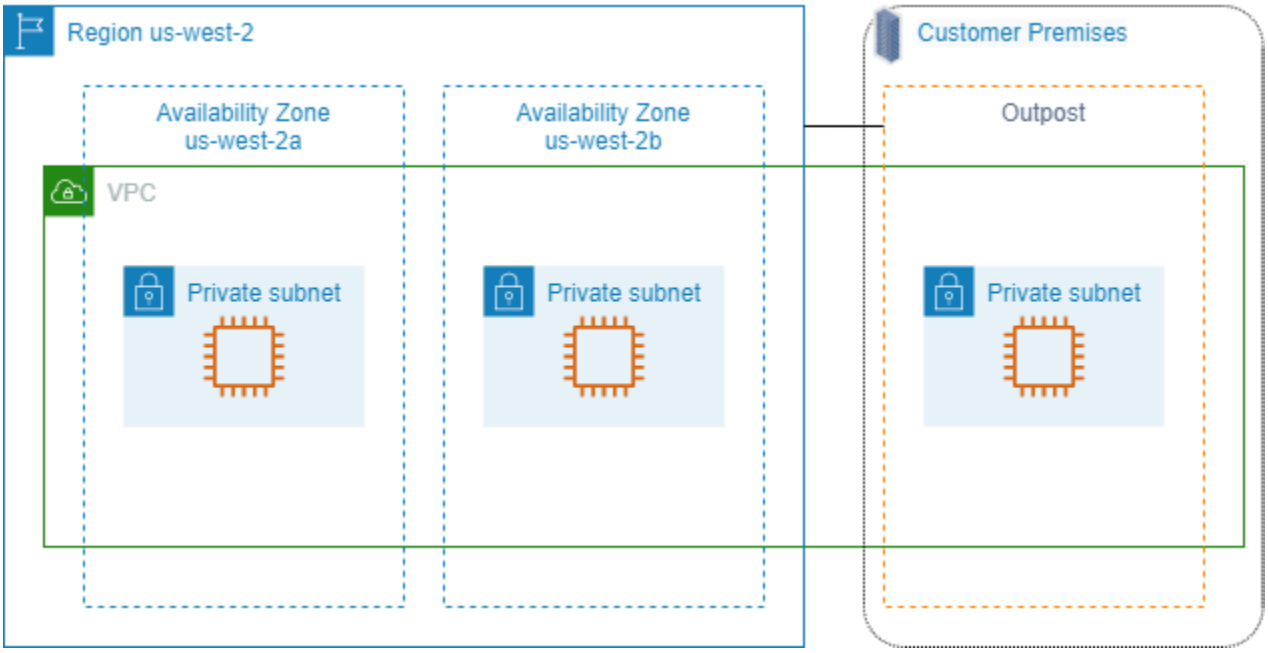
如需在「Wavelength 區域」中啟動執行個體的指 step-by-step 示，請參閱開AWS Wavelength 發人員指南 AWS Wavelength中的入門使用。

AWS Outposts

AWS Outposts 是一項全受管服務，可將 AWS 基礎架構APIs、服務和工具延伸至客戶場所。透過提供 AWS 受管理基礎架構的本機存取權， AWS Outposts 讓客戶能夠使用與 Region 相同的程式設計介面在 AWS 內部部署建置和執行應用程式，同時使用本機運算和儲存資源來降低延遲和本機資料處理需求。

Outpost 是部署在客戶站點的 AWS 計算和儲存容量集區。 AWS 作為 AWS 區域的一部分來操作、監控和管理此容量。您可以在 Outpost 上建立子網路，並在建立 AWS 資源時指定子網路。Outpost 子網路中的執行個體會使用私有 IP 位址與 AWS 區域中的其他執行個體進行通訊，全部都在相同的範圍內。VPC

下圖說明「區 AWS 域」`us-west-2`、其兩個可用區域及一個「前哨」。VPC跨越可用區域和前哨站。Outpost 位於內部部署客戶資料中心。中的每個區域都VPC有一個子網路，而每個子網路都有一個執行個體。



前哨上的執行個體

要開始使用 AWS Outposts，您必須創建一個前哨並訂購前哨容量。如需 Outposts 組態的詳細資訊，請參閱我們的目錄。安裝 Outpost 設備後，當您在 Outpost 上啟動 Amazon EC2 執行個體時，即可使用運算和儲存容量。

若要啟動EC2執行個體，您必須建立 Outpost 子網路。安全群組控制 Outpost 子網路中執行個體的入站和輸出流量，就像對可用區域子網路中的執行個體一樣。若要連線至 Outpost 子網路中的EC2執行個體，您可以在啟動執行個體時指定 key pair，就像在可用區域子網路中執行個體一樣。

建議您將 Outpost 機架上執行個體的根磁碟區限制為 30 GiB 或更小。您可以在AMI或執行個體的區塊裝置對應中指定資料磁碟區，以提供額外的儲存體。若要從開機磁碟區裁剪未使用的區塊，請參閱 AWS 合作夥伴網路部落格中的如何建立稀疏EBS磁碟區。

我們建議您增加根磁碟區的NVMe逾時時間。如需詳細資訊，請參閱 I/O 作業逾時。

如需有關如何建立 Outpost 的資訊，請參閱《AWS Outposts 使用者指南》中的 AWS Outposts入門。

前哨機架上的磁碟區

AWS Outposts 提供機架和伺服器外型規格。如果您的容量位於 Outpost 機架上，您可以在您建立的 Outpost 子網路中建立EBS磁碟區。建立磁碟區時，請指定前哨的 Amazon 資源名稱 (ARN)。

以下 create-volume 命令會在指定的 Outpost 上建立空的 50 GB 磁碟區。

```
aws ec2 create-volume --availability-zone us-east-2a --outpost-arn arn:aws:outposts:us-east-2:123456789012:outpost/op-03e6fecad652a6138 --size 50
```

您可以動態修改 Amazon EBS gp2 磁碟區的大小，而無需分離它們。如需有關在不分離磁碟區的情況下修改磁碟區的詳細資訊，請參閱要求修改EBS磁碟區。

Amazon EC2 實例 IP 定址

Amazon EC2 和 Amazon VPC 支持IPv4和IPv6尋址協議。根據預設，Amazon VPC 會使用定IPv4址協定；您無法停用此行為。當您建立時VPC，您必須指定一個IPv4CIDR區塊 (私人IPv4位址的範圍)。您可以選擇性地將IPv6CIDR圖塊指派給您的區塊，VPC並將該區塊中的IPv6位址指派給子網路中的執行個體。

目錄

- 私人IPv4地址

- 公共IPv4地址
- 公共IPv4位址最佳化
- IPv6地址
- EC2實例主機名稱
- Link-local 地址
- 管理執EC2行個體的IPv4位址
- 管理執EC2行個體的IPv6位址
- EC2執行個體的多個 IP 位址
- 設定 Windows 執行個體的次要私人IPv4位址

私人IPv4地址

私人位IPv4址是無法透過網際網路存取的 IP 位址。您可以使用私有IPv4地址在同一個實例之間進行通信VPC。如需私人IPv4位址標準與規格的詳細資訊，請參閱 RFC1918。我IPv4們使用DHCP.

> **Note**
>
> 您可以創建一個包VPC含位于 RFC 1918 年指定的私有IPv4地址范圍之外的可公開路由CIDR塊。不過，就本文件而言，我們將私人位IPv4址 (或稱為「私人 IP 位址」) 稱為您的 IPv4 CIDR IP 位址。VPC

VPC子網路可以是下列其中一種類型：

- IPv4只有子網路 — 您只能在這些子網路中建立具有指派IPv4位址的資源。
- IPv6只有子網路 — 您只能在這些子網路中建立具有指派IPv6位址的資源。
- IPv4和IPv6子網路 — 您可以在這些子網路中建立資源，並指定給這些子網路的IPv4或IPv6位址。

當您將EC2執行個體啟動至IPv4唯一或雙重堆疊 (IPv4和IPv6) 子網路時，執行個體會從子網路的位IPv4址範圍接收主要私有 IP 位址。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的 IP 定址。如果您在啟動執行個體時未指定主要私有 IP 位址，我們會在子網路IPv4範圍內為您選取可用的 IP 位址。每個執行個體都有指派主要私人IPv4位址的預設網路介面 (eth0)。您也可以指定其他私人IPv4位址，稱為次要私人IPv4位址。與主要私有 IP 地址不同，輔助私有 IP 地址可從原執行個體重新指派至別的執行個體。如需詳細資訊，請參閱EC2執行個體的多個 IP 位址。

私有IPv4位址 (無論是主要或次要位址) 在執行個體停止和啟動或休眠和啟動時都會與網路介面保持關聯，並在執行個體終止時釋放。

公共IPv4地址

公用 IP 位IPv4址是可從網際網路存取的位址。您可以使用公有地址，在執行個體與網際網路之間通訊。

當您以預設值啟動執行個體時VPC，我們預設會為其指派公用 IP 位址。當您將執行個體啟動至非預設值時VPC，子網路具有一個屬性，可決定啟動至該子網路的執行個體是否會從公用位址集區接收公用 IP 位IPv4址。根據預設，我們不會將公有 IP 地址指派給非預設子網中啟動的執行個體。

您可以控制您的執行個體是否接收公有 IP 地址，如下所示：

- 修改子網的公有 IP 定址屬性。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的修改子網路的公用IPv4定址屬性。
- 在啟動期間啟用或停用公有 IP 定址，其可覆寫子網的公有 IP 定址屬性。如需詳細資訊，請參閱在執行個體啟動期間指派公用IPv4地址。
- 啟動後，您可以管理與網路介面關聯的 IP 位址，從執行個體取消指派公用 IP 位址。

公有 IP 地址會從 Amazon 的公共地IPv4址集區指派給您的執行個體，而且與您的 AWS 帳戶不相關聯。如果公用 IP 位址與您的執行個體取消關聯，就會將其釋放回公用位IPv4址集區，而且您無法重複使用該位址。

在某些情況下，我們會從您的執行個體釋出公有 IP 位址，或指派新的 IP 位址：

- 當您的執行個體停止、休眠或終止時，我們會釋出其公有 IP 地址。啟動您已停止或休眠的執行個體時，其會收到新的公有 IP 地址。
- 當您將彈性 IP 地址與執行個體建立關聯時，我們會釋出該執行個體的公有 IP 地址。當您取消彈性 IP 地址與執行個體的關聯時，執行個體收到新的公有 IP 地址。
- 如果執行個體中的公有 IP 位址VPC已經釋出，如果您的執行個體連接了一個以上的網路介面，就不會收到新的 IP 位址。
- 當您的執行個體具有與彈性 IP 地址相關聯的次要私有 IP 地址時，如果釋放該執行個體的公有 IP 地址，則執行個體不會接收新的公有 IP 地址。

如果您需要可建立和取消與執行個體之關聯的持久性公有 IP 地址，請改用彈性 IP 地址。

如果您使用 dynamic DNS 將現有DNS名稱對應至新執行個體的公有 IP 位址，IP 位址最多可能需要 24 小時才能透過網際網路傳播。因此，已終止的執行個體還會持續收到請求，而新的執行個體可能尚未開始接收流量。若要解決此問題，請使用彈性 IP 地址。您可以配置自己的彈性 IP 地址，並建立地址與您執行個體的關聯。如需詳細資訊，請參閱彈性 IP 地址。

> **Note**
>
> - AWS 所有公共IPv4地址的費用，包括與執行中執行個體和彈性 IP IPv4 地址相關聯的公共地址。如需詳細資訊，請參閱 Amazon VPC 定價頁面上的公開IPv4位址索引標籤。
> - 透過其公有 NAT IP 位址存取其他執行個體的執行個體需支付地區或網際網路資料傳輸的費用，視執行個體是否位於同一區域而定。

公共IPv4位址最佳化

AWS 所有公共IPv4地址的費用，包括與執行中執行個體和彈性 IP IPv4 地址相關聯的公共地址。如需詳細資訊，請參閱 Amazon VPC 定價頁面上的公開IPv4位址索引標籤。

下列清單包含您可以採取的動作來最佳化您使用的公用IPv4位址數目：

- 使用彈性負載平衡器負載平衡執行個體的流量，並在指派給EC2執行個體的主要伺服器上停用自動ENI指派公用 IP。負載平衡器使用單一公用IPv4位址，因此可減少您的公用IPv4位址計數。您可能還想要合併現有的負載平衡器，以進一步減少公用IPv4位址計數。
- 如果使用NAT閘道的唯一原因是要SSH進入私有子網路中的EC2執行個體進行維護或緊急情況，請考慮改用EC2執行個體 Connect 端點。透過EC2執行個體 Connect 端點，您可以從網際網路連線至執行個體，而不需要執行個體擁有公開IPv4位址。
- 如果您的EC2執行個體位於已分配公有 IP 位址的公有子網路中，請考慮將執行個體移至私有子網路、移除公有 IP 位址，並使用公用NAT閘道允許執行個體的存取和存取EC2執行個體。使用NAT閘道需要考量成本。使用此計算方法來決定NAT閘道是否符合成本效益。您可以建立「Number of public IPv4 addresses帳 AWS 單成本與用量報表」來取得此計算所需的資料。

  ```
  NAT gateway per hour + NAT gateway public IPs + NAT gateway transfer / Existing
   public IP cost
  ```

  其中：

  - `NAT gateway per hour = $0.045 * 730 hours in a month * Number of Availability Zones the NAT gateways are in`

- `NAT gateway public IPs = $0.005 * 730 hours in a month * Number of IPs associated with your NAT gateways`
- `NAT gateway transfer = $0.045 * Number of GBs that will go through the NAT gateway in a month`
- `Existing public IP cost = $0.005 * 730 hours in a month * Number of public IPv4 addresses`

如果總數小於 1，則NAT閘道會比公用IPv4位址便宜。

- 用於私下連線AWS PrivateLink至 AWS 由其他 AWS 帳戶託管的服務或服務，而不是使用公用IPv4位址和網際網路閘道。
- 將您自己的 IP 地址範圍（BYOIP）帶到公共地IPv4址 AWS並使用該範圍，而不是使用亞馬遜擁有的公共IPv4地址。
- 針對啟動至子網路的執行個體關閉自動指派公用IPv4位址。當您建立子網路VPCs時，通常預設會停用此選項，但是您應該檢查現有的子網路以確保已停用。
- 如果您的EC2執行個體不需要公開位IPv4址，請檢查連接至執行個體的網路介面是否已停用自動指派公用 IP。
- 針對私有子網路中 AWS Global Accelerator的EC2執行個體設定加速器端點，讓網際網路流量直接流向您中的端點，VPCs而不需要公用 IP 位址。您也可以將自己的位址帶到加速器的靜態 IP 位IPv4址， AWS Global Accelerator並使用您自己的位址。

IPv6地址

IPv6位址是全域唯一的，可設定為保持私密或透過網際網路存取。公共和私人IPv6定址都可用在 AWS:

- 私人 IPv6： AWS 將私人IPv6地址視為那些未廣告且無法從互聯網上廣告的私人地址。 AWS
- 公共 IPv6： AWS 考慮從 AWS互聯網上廣告的公共IPv6地址。

如需有關公用和私有IPv6地址的詳細資訊，請參閱 Amazon VPC 使用者指南中的IPv6地址。

如果IPv6CIDR區塊與您的VPC和子網路相關聯，且下列其中一項為真，則您的EC2執行個體會收到IPv6位址：

- 您的子網路設定為在啟動期間自動指派IPv6位址給執行個體。如需詳細資訊，請參閱修改子網路的 IP 位址屬性。
- 您可以在啟動期間為執行個體指派IPv6位址。

- 啟動後，您可以為執行個體的主要網路介面指派IPv6位址。
- 您可以將IPv6位址指派給相同子網路中的網路介面，並在啟動後將網路介面附加至執行個體。

當您的執行個體在啟動期間收到位IPv6址時，該位址就會與執行個體的主要網路介面 (eth0) 相關聯。您可以透過下列方式管理執行個體主要網路介面 (eth0) 的IPv6位址：

- 從網路介面指派和取消指派IPv6位址。您可以指派給網路介面的IPv6位址數目，以及可連接至執行個體的網路介面數目，視執行個體類型而有所不同。如需詳細資訊，請參閱每個網路介面的 IP 位址上限。
- 啟用主要IPv6位址。主要IPv6位址可讓您避免中斷執行個體或ENIs的流量。如需詳細資訊，請參閱建立網路介面 或 管理網路介面的 IP 位址。

當您停止並啟動或休眠並啟動執行個體時，IPv6地址仍會持續存在，並在您終止執行個體時釋放。將IPv6位址指派給其他網路介面時，您無法重新指派位址，您必須先取消指派該位址。

您可以控制子網路的路由或使用安全群組和網路ACL規則，來控制執行個體是否可透過其IPv6位址存取。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的網路間流量隱私權。

如需有關保留IPv6位址範圍的詳細資訊，請參閱IANAIPv6特殊用途地址登錄和. RFC4291

EC2實例主機名稱

建立EC2執行個體時， AWS 會建立該執行個體的主機名稱。如需主機名稱類型及其佈建方式的詳細資訊 AWS，請參閱Amazon EC2 執行個體主機名稱。Amazon 提供了一個DNS服務器，可以將亞馬遜提供的主機名解析為IPv4和地址。IPv6Amazon DNS 服務器位於您的VPC網絡範圍的基礎加上兩個。如需詳細資訊，請參閱 Amazon VPC 使用者指南VPC中的DNS屬性。

Link-local 地址

Link-local 地址是眾所周知不可路由的 IP 地址。Amazon EC2 使用來自連結本機位址空間的地址來提供只能從執行個EC2體存取的服務。這些服務不在執行個體上執行，而是在基礎主機上執行。當您存取這些服務的 Link-local 地址時，您正在與 Xen 虛擬化管理程序或 Nitro 控制器進行通訊。

Link-local 地址範圍

- IPv4— 169.254.0.0/16 (169.254.0.0 至 169.254.255.255)
- IPv6— 二月八十:: /10

使用 link-local 地址存取的服務

- 執行個體中繼資料服務
- Amazon Route 53 Resolver（也稱為 Amazon 服DNS務器）
- Amazon Time Sync Service
- AWS KMS伺服器

管理執EC2行個體的IPv4位址

您可以在啟動執行個體時為執行個體指派公用IPv4位址。您可以透過「執行個體」頁面或「網路介面」頁面，在主控台中檢視執行個體的IPv4位址。

目錄

- 檢視地IPv4址
- 在執行個體啟動期間指派公用IPv4地址

檢視地IPv4址

您可以使用 Amazon EC2 主控台檢視執行個體的公有和私有IPv4地址。您也可以使用執行個體中繼資料，從執行個體中判斷執行個體的公開IPv4和私有IPv4地址。如需詳細資訊，請參閱使用執行個體中繼資料管理EC2執行個。

公用IPv4位址會在主控台中顯示為網路介面的屬性，但會透過對應至主要私人IPv4位址NAT。因此，如果您在執行個體上檢查網路介面的屬性，例如透過 `ifconfig` (Linux) 或 `ipconfig` (Windows)，就不會顯示公用IPv4位址。若要從執行個體判斷執行個體的公開IPv4位址，請使用執行個體中繼資料。

若要使用指令列檢視執行個體的IPv4位址

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 訪問 Amazon EC2。

- describe-instances (AWS CLI)
- Get-EC2Instance (AWS Tools for Windows PowerShell).

使用執行個體中繼資料判斷執行個體的IPv4位址

1. 連線到您的執行個體。如需詳細資訊，請參閱Connect 至您的EC2執行個體。
2. 使用下列命令來存取私人 IP 位址。

IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H
 "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/
meta-data/local-ipv4
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/local-ipv4
```

Tools for Windows PowerShell

```
PS C:\> Invoke-RestMethod http://169.254.169.254/latest/meta-data/local-ipv4
```

3. 使用下列命令存取公用 IP 位址。如果建立彈性 IP 地址與執行個體的關聯，傳回的值即為彈性 IP 地址的值。

IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H
 "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/
meta-data/public-ipv4
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/public-ipv4
```

Tools for Windows PowerShell

```
PS C:\> Invoke-RestMethod http://169.254.169.254/latest/meta-data/public-ipv4
```

在執行個體啟動期間指派公用IPv4地址

每一個子網都會有一個屬性，以判定是否將公有 IP 地址指派至啟動到該子網的執行個體。非預設子網預設會將此屬性設定為 false，而預設子網會將此屬性設定為 true。當您啟動執行個體時，您也可以使

用公開位IPv4址功能來控制執行個體是否已指派公用位IPv4址；您可以覆寫子網路 IP 位址屬性的預設行為。公共IPv4地址是從 Amazon 的公共地IPv4址集區分配的，並指派給具有 eth0 裝置索引的網路界面。此功能取決於啟動執行個體時的特定條件。

考量事項

- 啟動後，您可以管理與網路介面關聯的 IP 位址，從執行個體取消指派公用 IP 位址。如需有關公用IPv4位址的詳細資訊，請參閱公共IPv4地址。
- 如果您指定多個網路介面，則無法自動指派公有 IP 地址。此外，如果您為 eth0 指定現有網路介面，則無法使用自動指派公有 IP 功能覆寫子網設定。
- 無論您是否在啟動期間為執行個體指派公用 IP 位址，都可以在啟動後將彈性 IP 位址與執行個體建立關聯。如需詳細資訊，請參閱彈性 IP 地址。您也可以修改子網路的公開IPv4定址行為。如需詳細資訊，請參閱修改子網路的公用位IPv4址屬性。

使用主控台在執行個體啟動期間指派公用IPv4位址

遵循程序啟動執行個體，當您設定 Network Settings (網路設定) 時，選擇選項以 Auto-assign Public IP (自動指派公有 IP)。

使用命令列啟用或停用公有 IP 定址功能

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 訪問 Amazon EC2。

- 使用 `--associate-public-ip-address` 或 `--no-associate-public-ip-address` 選項並搭配 run-instances 命令 (AWS CLI)
- 透過New-EC2Instance指令使用`-AssociatePublicIp`參數 (AWS Tools for Windows PowerShell)

管理執EC2行個體的IPv6位址

您可以檢視指派給執行個體的IPv6位址、指派公用位IPv6址給執行個體，或是取消指派執行個體的IPv6地址。您可以在主控台中透過 Instances (執行個體) 頁面或 Network Interfaces (網路介面) 頁面檢視這些地址。

目錄

- 指派位IPv6址給執行個體
- 檢視地IPv6址
- 從執行個體取消指派IPv6位址

指派位IPv6址給執行個體

如果您的子網路VPC和子網路有相關聯的IPv6CIDR區塊，您可以在啟動期間或之後為執行個體指派IPv6位址。位IPv6址是從子網路的IPv6位址範圍指派的，並且會以 eth0 的裝置索引指派給網路介面。

若要在執行個體啟動期間指派IPv6位址

依照程序啟動執行個體，並在設定網路設定時選擇自動指派 IPv6 IP 選項。

若要在啟動後指派IPv6位址

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後選取 Actions (動作)、Networking (聯網)、Manage IP addresses (管理 IP 地址)。
4. 展開網路介面。在 [位IPv6址] 下方，選擇 [指派新 IP 位址]。輸入子網路範圍內的位IPv6址，或將欄位留空，讓 Amazon 為您選擇一個IPv6地址。
5. 選擇 Save (儲存)。

使用指令行指定IPv6位址的步驟

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 訪問 Amazon EC2。

- 使用 `--ipv6-addresses` 選項並搭配 run-instances 命令 (AWS CLI)
- 在New-EC2Instance指令 (AWS Tools for Windows PowerShell) `-NetworkInterface` 中使用的`Ipv6Addresses`屬性
- assign-ipv6-addresses (AWS CLI)
- Register-EC2Ipv6 AddressList (AWS Tools for Windows PowerShell)

檢視地IPv6址

您可以使用 Amazon EC2 主控台和執行個體中繼資料來檢視執行個體的IPv6地址。 AWS CLI

使用主控台檢視執行個體的IPv6位址

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。

3. 選取執行個體。
4. 在 [網路] 索引標籤上，找出IPv6位址。

若要使用指令列檢視執行個體的IPv6位址

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 訪問 Amazon EC2。

- describe-instances (AWS CLI)
- Get-EC2Instance (AWS Tools for Windows PowerShell).

使用執行個體中繼資料檢視執行個體的IPv6位址

1. 連線到您的執行個體。如需詳細資訊，請參閱Connect 至您的EC2執行個體。
2. 從中取得執行個體的MAC位址`http://169.254.169.254/latest/meta-data/network/interfaces/macs/`。
3. 使用下面的命令來查看IPv6地址。

   IMDSv2

   ```
   [ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H
    "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
   && curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/
   meta-data/network/interfaces/macs/mac-address/ipv6s
   ```

   IMDSv1

   ```
   [ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/network/interfaces/
   macs/mac-address/ipv6s
   ```

   Tools for Windows PowerShell

   ```
   PS C:\> Invoke-RestMethod http://169.254.169.254/latest/meta-data/network/
   interfaces/macs/mac-address/ipv6s
   ```

從執行個體取消指派IPv6位址

您可以隨時從執行個體取消指派IPv6位址。

使用主控台取消指派執行個體的IPv6位址

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後選取 Actions (動作)、Networking (聯網)、Manage IP addresses (管理 IP 地址)。
4. 展開網路介面。在「地IPv6址」下方，選擇IPv6地址旁邊的「取消指派」。
5. 選擇 Save (儲存)。

若要使用指令列取消指派執行個體的IPv6位址

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 訪問 Amazon EC2。

- unassign-ipv6-addresses (AWS CLI)
- Unregister-EC2Ipv6 AddressList (AWS Tools for Windows PowerShell)。

EC2執行個體的多個 IP 位址

您可以為執行個體指定多個私有IPv6位址IPv4和位址。您可以為執行個體指定的網路介面IPv4、私有和IPv6位址數目，視執行個體類型而定。如需詳細資訊，請參閱每個網路介面的 IP 位址上限。

將多個 IP 位址指派給執行個體中的執行個體可能很有用，VPC以執行下列動作：

- 在單一伺服器上使用多個SSL憑證，並將每個憑證與特定 IP 位址建立關聯，在單一伺服器上託管多個網站。
- 操作網路應用裝置 (例如防火牆或負載平衡器)，其中的每個網路介面有多個 IP 地址。
- 將輔助 IP 地址重新指派給待命執行個體，以在執行個體失敗時將內部流量重新導向至備用執行個體。

內容

- 多個 IP 地址運作方式
- 使用多個IPv4地址
- 使用多個IPv6地址

多個 IP 地址運作方式

下列清單說明多個 IP 地址如何與網路介面運作：

- 您可以將次要私人IPv4位址指派給任何網路介面。
- 您可以將多個IPv6位址指派給位於具有關聯IPv6CIDR區塊之子網路中的網路介面。
- 您必須從網路介面的子網路IPv4CIDR區塊範圍中選擇次要IPv4位址。
- 您必須從網路介面的子網路IPv6CIDR區塊範圍中選擇IPv6位址。
- 您可以將安全群組與網路介面 (而非個別的 IP 地址) 建立關聯。因此，您在網路介面中指定的每個 IP 地址會受其網路介面安全群組的規範。
- 多個 IP 地址可以指派和取消指派至連接至執行中或已停止執行個體的網路介面。
- 如果您明確允許，指派給網路介面的次要私人IPv4位址可以重新指派給另一個專用位址。
- 無法將IPv6位址重新指定給其他網路介面；您必須先從現有的網路介面取消指派IPv6位址。
- 使用命令列工具將多個 IP 位址指派給網路介面時API，或者如果無法指派其中一個 IP 位址，則整個作業會失敗。
- 當次要網路介面從執行個體中斷連結或連接至執行個體時，主要私人IPv4位IPv6址、次要私人位址、彈性 IP 位址和位址仍會保留在次要網路介面中。IPv4
- 雖然您無法將主要網路介面與執行個體分離，但您可以將主要網路介面的次要私人IPv4位址重新指派給另一個網路介面卡。

下列清單說明多個 IP 位址如何與彈性 IP 位址搭配使用 (IPv4僅限)：

- 每個私人IPv4地址都可以與單個彈性 IP 地址相關聯，反之亦然。
- 當次要私人位IPv4址重新指派給另一個介面時，次要私人位IPv4址會保留其與彈性 IP 位址的關聯。
- 從介面取消指派次要私人位IPv4址時，相關聯的彈性 IP 位址會自動取消與次要私人IPv4位址的關聯。

使用多個IPv4地址

您可以將次要私人IPv4位址指派給執行個體、將彈性IPv4位址與次要私人位IPv4址建立關聯，以及取消指派次要私人IPv4位址。

任務

- 指派次要私人IPv4位址

- 設定作業系統以辨識次要私人IPv4位址
- 將彈性 IP 位址與次要私人位IPv4址建立關聯
- 檢視您的次要私人IPv4地址
- 取消指派次要私人IPv4位址

指派次要私人IPv4位址

您可以在啟動執行個體時或執行個體執行時，將次要私人IPv4位址指派給執行個體的網路介面。

若要在啟動執行個體時指派次要私人IPv4位址

1. 請遵循該程序來啟動執行個體。針對網路設定，選擇編輯。
2. 選取子網路VPC和子網路。
3. 展開 進階網路組態。
4. 對於次要 IP，請選擇「自動指派」並輸入 IP 位址數目 (Amazon 會自動指派次要IPv4地址)，或選擇「手動指派」並輸入IPv4地址。
5. 完成其餘步驟以啟動執行個體。

使用指令列在啟動期間指派次要IPv4位址

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 訪問 Amazon EC2。

- `--secondary-private-ip-addresses` 選項搭配 run-instances 命令 (AWS CLI)
- 使用指New-EC2Instance令定義`-NetworkInterface`並指定`PrivateIpAddresses`參數 (AWS Tools for Windows PowerShell)。

將次要私人IPv4位址指派給網路介面

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在瀏覽窗格中，選擇 [網路介面]，然後選取執行個體的網路介面。
3. 選擇 Actions (動作)、Manage IP Addresses (管理 IP 地址)。
4. 展開網路介面。在 [位IPv4址] 下方，選擇 [指派新 IP 位址]。
5. 輸入位於執行個體子網路範圍內的特定位IPv4址，或將欄位留空，讓 Amazon 為您選取IPv4地址。
6. (選擇性) 選取允許在次要私人 IP 位址已指派給其他網路介面時重新指定該 IP 位址。

7. 選擇 Save (儲存)。

或者，您也可以將次要私人IPv4位址指派給執行個體。在導覽窗格中選取執行個體，並選取執行個體，然後選取動作、聯網、管理 IP 地址。您可以設定如同上述步驟中所設定的資訊。IP 地址會指派給執行個體的主要網路介面 (eth0)。

使用命令列將次要私人IPv4位址指派給現有執行個體

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱訪問 Amazon EC2。

- assign-private-ip-addresses (AWS CLI)
- Register-EC2PrivateIpAddress (AWS Tools for Windows PowerShell)

設定作業系統以辨識次要私人IPv4位址

將次要私人位IPv4址指派給執行個體之後，您必須在執行個體上設定作業系統，以辨識次要私有 IP 位址。

Linux 執行個體

- 如果您使用的是 Amazon Linux，ec2-net-utils 套件可以為您處理此步驟。它會設定執行個體執行時連接的其他網路介面、在DHCP租用續約期間重新整理次要IPv4位址，以及更新相關的路由規則。您可以使用指令立即重新整理介面清單，`sudo service network restart`然後使用檢視up-to-date 清單`ip addr li`。如果您需要手動控制網路組態，則可以移除 ec2-net-utils 套件。如需詳細資訊，請參閱使用 ec2-net-utils 設定您的網路介面。
- 如果您使用的是其他 Linux 發行版本，請參閱 Linux 發行版本的文件。搜尋有關設定其他網路介面和次要IPv4位址的資訊。如果執行個體在相同的子網上有兩個或以上的介面，請搜尋有關使用路由規則暫時處理非對稱路由的資訊。

Windows 執行個體

如需詳細資訊，請參閱設定 Windows 執行個體的次要私人IPv4位址。

將彈性 IP 位址與次要私人位IPv4址建立關聯

將彈性 IP 位址與次要私人位IPv4址建立關聯

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 [彈性] IPs。

3. 選取彈性 IP 位址的核取方塊
4. 選擇 [動作]、[關聯彈性 IP 位址]。
5. 針對 [資源類型]，選擇 [網路介面]。選取網路介面，然後從 [私人 IP 位址] 清單中選取次要 IP 位址。
6. 在 [網路介面] 中，選取網路介面。從 [私人 IP 位址] 清單中選取次要 IP 位址。
7. 對於私人 IP 位址，請選取次要 IP 位址。
8. 選擇關聯。

使用命令列將彈性 IP 位址與次要私人位IPv4址建立關聯

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 訪問 Amazon EC2。

- associate-address (AWS CLI)
- Register-EC2Address (AWS Tools for Windows PowerShell)

檢視您的次要私人IPv4地址

檢視指派給網路介面的私人IPv4位址

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面的核取方塊。
4. 在 [詳細資料] 索引標籤的 [IP 位址] 下，找出 [私人位IPv4址] 和 [次要私人位IPv4址

若要檢視指派給執行個體的私人IPv4位址

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取該執行個體的核取方塊。
4. 在 [網路功能] 索引標籤的 [網路詳細資料] 下，找出 [私人IPv4位IPv4址] 和 [次要私

取消指派次要私人IPv4位址

如果您不再需要次要私人IPv4位址，您可以從執行個體或網路介面取消指派它。從網路介面取消指派次要私人位IPv4址時，彈性 IP 位址 (如果存在) 也會取消關聯。

若要從執行個體取消指派次要私人IPv4位址

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後選取動作、聯網、管理 IP 地址。
4. 展開網路介面。對於地IPv4址，請針對要取消指定的IPv4地址選擇「取消指定」。
5. 選擇 Save (儲存)。

從網路介面取消指派次要私人IPv4位址

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面，選擇 [動作]、[管理 IP 位址]。
4. 展開網路介面。對於地IPv4址，請針對要取消指定的IPv4地址選擇「取消指定」。
5. 選擇 Save (儲存)。

若要使用命令列取消指派次要私人IPv4位址

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱訪問 Amazon EC2。

- unassign-private-ip-addresses (AWS CLI)
- Unregister-EC2PrivateIpAddress (AWS Tools for Windows PowerShell)

使用多個IPv6地址

您可以為執行個體指派多個IPv6地址、檢視指派給執行個體的IPv6位址，以及從執行個體取消指派IPv6地址。

目錄

- 指派多個IPv6地址
- 檢視您的IPv6地址
- 取消指派地址 IPv6

指派多個IPv6地址

您可以在啟動期間或啟動後為執行個體指派一或多個IPv6位址。若要指派IPv6位址給執行個體，啟動執行個體所在的VPC和子網路必須具有關聯的IPv6CIDR區塊。

若要在啟動期間指派多個IPv6位址

1. 請遵循該程序來啟動執行個體。針對網路設定，選擇編輯。
2. 選取子網路VPC和子網路。
3. 展開 進階網路組態。
4. 對於 IPv6IPs，選擇自動指派和 IP 位址的數量 (Amazon 會自動指派位IPv6址)，或選擇手動指派並輸入位IPv6址。
5. 完成其餘步驟以啟動執行個體。

您可以使用「執行個體」畫面 Amazon EC2 主控台將多個IPv6地址指派給現有執行個體。這會將位IPv6址指派給執行個體的主要網路介面 (eth0)。若要指派特定IPv6位址給執行個體，請確定該IPv6位址尚未指派給其他執行個體或網路介面。

若要將多個IPv6位址指派給現有執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，選擇 [動作]、[網路]、[管理 IP 位址]。
4. 展開網路介面。對於位IPv6址，請選擇為每個要新增的位IPv6址指派新 IP 位址。您可以從子網路範圍中指定位IPv6址，或將欄位保留空白，讓 Amazon 為您選擇一個IPv6地址。
5. 選擇 Save (儲存)。

或者，您可以將多個IPv6位址指派給現有的網路介面。網路介面必須是在具有關聯IPv6CIDR區塊的子網路中建立的。若要為網路介面指派特定IPv6位址，請確定該IPv6位址尚未指派給其他網路介面卡。

若要將多個IPv6位址指派給網路介面

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取您的網路介面，選擇 [動作]、[管理 IP 位址]。

4. 展開網路介面。對於位IPv6址，請選擇為每個要新增的位IPv6址指派新 IP 位址。您可以從子網路範圍中指定位IPv6址，或將欄位保留空白，讓 Amazon 為您選擇一個IPv6地址。
5. 選擇 Save (儲存)。

CLI概述

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱訪問 Amazon EC2。

- 在啟動期間指派IPv6位址：
  - 使用 `--ipv6-addresses` 或 `--ipv6-address-count` 選項並搭配 run-instances 命令 (AWS CLI)
  - 使用指New-EC2Instance令定義`-NetworkInterface`並指定`Ipv6Addresses`或`Ipv6AddressCount`參數 (AWS Tools for Windows PowerShell)。
- 指定網路介面的IPv6位址：
  - assign-ipv6-addresses (AWS CLI)
  - Register-EC2Ipv6 AddressList (AWS Tools for Windows PowerShell)

檢視您的IPv6地址

您可以檢視執行個體或網路介面的IPv6位址。

若要檢視指派給執行個體的IPv6位址

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體的核取方塊。
4. 在 [網路] 索引標籤上，找出 [位IPv6址] 欄位。

檢視指派給網路介面的IPv6位址

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面的核取方塊。
4. 在 [詳細資料] 索引標籤的 [IP 位址] 下，找出位IPv6址欄位。

CLI概述

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱訪問 Amazon EC2。

- 檢視執行個體的IPv6位址：
  - describe-instances (AWS CLI)
  - Get-EC2Instance (AWS Tools for Windows PowerShell).
- 檢視網路介面的IPv6位址：
  - describe-network-interfaces (AWS CLI)
  - Get-EC2NetworkInterface (AWS Tools for Windows PowerShell)

取消指派地址 IPv6

您可以從執行個體的主要網路介面取消指派IPv6位址，也可以從網路介面取消指派IPv6位址。

若要從執行個體取消指派IPv6位址

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體的核取方塊，然後選擇 [動作]、[網路]、[管理 IP 位址]。
4. 展開網路介面。在「地IPv6址」下方，選擇IPv6地址旁邊的「取消指派」。
5. 選擇 Save (儲存)。

從網路介面取消指派IPv6位址

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面的核取方塊，然後選擇 [動作] > [管理 IP 位址]。
4. 展開網路介面。在「地IPv6址」下方，選擇IPv6地址旁邊的「取消指派」。
5. 選擇 Save (儲存)。

CLI概述

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 訪問 Amazon EC2。

- unassign-ipv6-addresses (AWS CLI)

- Unregister-EC2Ipv6 AddressList (AWS Tools for Windows PowerShell)

設定 Windows 執行個體的次要私人IPv4位址

您可以為執行個體指定多個私有IPv4地址。將次要私人IPv4位址指派給執行個體之後，您必須在執行個體上設定作業系統以辨識次要私人IPv4位址。

> **Note**
>
> 這些說明是基於視窗服務器 2022。這些步驟的實作可能會因 Windows 執行個體的作業系統而有所不同。

任務

- 必要條件
- 步驟 1：在執行個體中設定靜態 IP 位址
- 步驟 2：設定您的執行個體的次要私有 IP 地址
- 步驟 3：設定應用程式使用次要私有 IP 地址

必要條件

1. 將次要私人IPv4位址指派給執行個體的網路介面。您可以在啟動執行個體時或執行個體執行時指派次要私人IPv4位址。如需詳細資訊，請參閱指派次要私人IPv4位址。
2. 分配彈性 IP 地址並將其與次要私人IPv4地址相關聯。如需詳細資訊，請參閱 配置彈性 IP 地址 和 將彈性 IP 位址與次要私人位IPv4址建立關聯。

步驟 1：在執行個體中設定靜態 IP 位址

若要讓 Windows 執行個體使用多個 IP 位址，您必須將執行個體設定為使用靜態 IP 位址而非DHCP伺服器。

> **Important**
>
> 在執行個體中設定靜態 IP 位址時，IP 位址必須與主控台、CLI或中顯示的 IP 位址完全相符 API。如果您不正確地輸入這些 IP 地址，執行個體可能會變成無法存取。

在 Windows 執行個體上設定靜態 IP 定址

1. 連線到您的執行個體。
2. 執行下列步驟，尋找執行個體的 IP 地址、子網遮罩和預設閘道地址：
   - 在中執行下列命令 PowerShell：

     ```
     ipconfig /all
     ```

     檢閱輸出並記下網路介面的IPv4位址、子網路遮罩、預設閘道和DNS伺服器值。您的輸出應該類似於下列範例：

     ```
     ...

     Ethernet adapter Ethernet 4:

        Connection-specific DNS Suffix  . : us-west-2.compute.internal
        Description . . . . . . . . . . . : Amazon Elastic Network Adapter #2
        Physical Address. . . . . . . . . : 02-9C-3B-FC-8E-67
        DHCP Enabled. . . . . . . . . . . : Yes
        Autoconfiguration Enabled . . . . : Yes
        Link-local IPv6 Address . . . . . : fe80::f4d1:a773:5afa:cd1%7(Preferred)
        IPv4 Address. . . . . . . . . . . : 10.200.0.128(Preferred)
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        Lease Obtained. . . . . . . . . . : Monday, April 8, 2024 12:19:29 PM
        Lease Expires . . . . . . . . . . : Monday, April 8, 2024 4:49:30 PM
        Default Gateway . . . . . . . . . : 10.200.0.1
        DHCP Server . . . . . . . . . . . : 10.200.0.1
        DHCPv6 IAID . . . . . . . . . . . : 151166011
        DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-2D-67-AC-FC-12-34-9A-BE-A5-
     E7
        DNS Servers . . . . . . . . . . . : 10.200.0.2
        NetBIOS over Tcpip. . . . . . . . : Enabled
     ```

3. 在中執行下列命令，以開啟「網路和共用中心」 PowerShell：

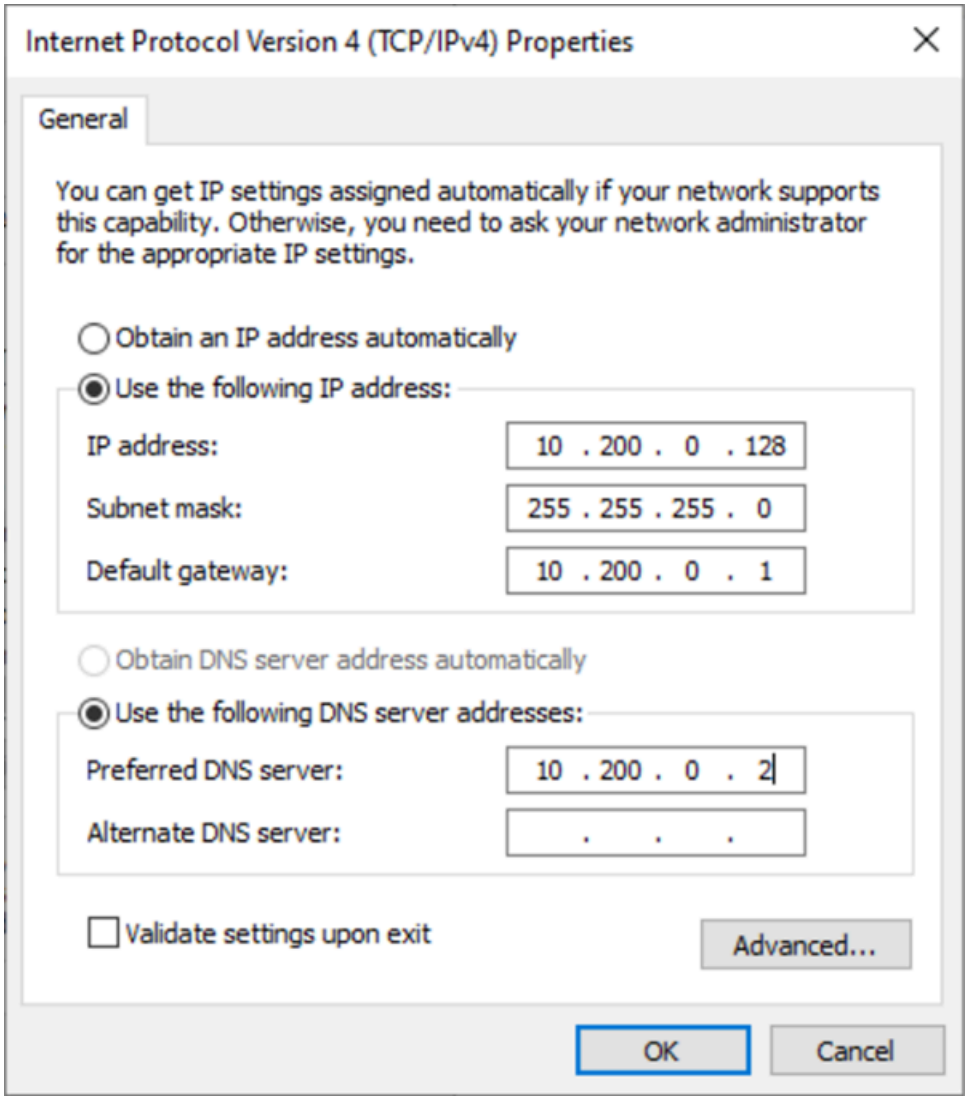
   ```
   & $env:SystemRoot\system32\control.exe ncpa.cpl
   ```

4. 開啟網路介面 (「區域連線」或「乙太網路」) 的內容 (按一下滑鼠右鍵) 選單，然後選擇「內容」。
5. 選擇網際網路通訊協定版本 4 (TCP/IPv4)、[內容]。

6. 在 [網際網路通訊協定版本 4 (TCP/IPv4) 內容] 對話方塊中，選擇 [使用下列 IP 位址]，輸入下列值，然後選擇 [確定]。

欄位	Value
IP 地址	在上述步驟 2 中獲得的IPv4地址。
子網遮罩	前述步驟 2 中取得的子網遮罩。
預設閘道	前述步驟 2 中取得的預設閘道地址。
偏好DNS伺服器	在上述步驟 2 中獲得的DNS伺服器。
替代DNS伺服器	在上述步驟 2 中取得的替代DNS伺服器。如果未列出替代DNS伺服器，請將此欄位保留空白。

> **Important**
>
> 如果您將 IP 地址設定為目前 IP 地址以外的任何值，則會中斷與執行個體的連線。



當執行個體從使用轉換為靜態位址時，幾秒鐘內就會失去與 Windows 執行DHCP個體的RDP連線。執行個體會保留與之前相同的 IP 位址資訊，但現在這項資訊是靜態的，而不是由管理DHCP。

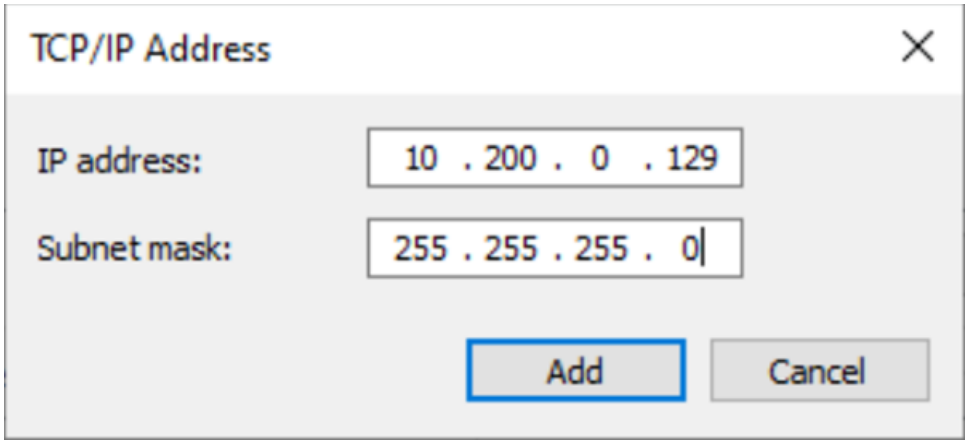
步驟 2：設定您的執行個體的次要私有 IP 地址

在您設定 Windows 執行個體的靜態 IP 定址之後，即可準備第二個私有 IP 地址。

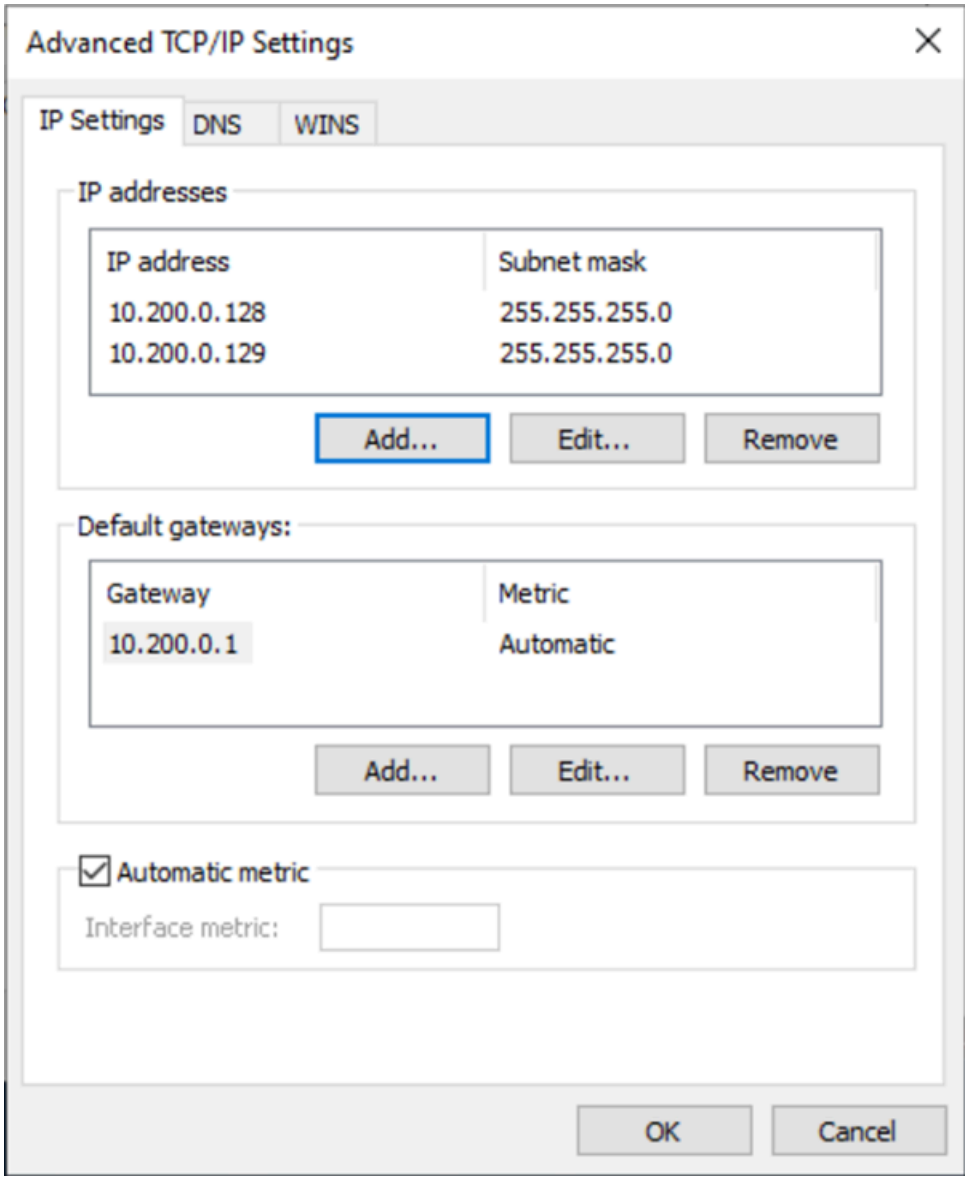
設定次要 IP 地址

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選取 Instances (執行個體)，然後選取您的執行個體。
3. 在 Networking (聯網) 標籤上，記下輔助 IP 地址。

4. 連線到您的執行個體。
5. 在 Windows 執行個體上，選擇開始、控制台。
6. 選擇網路和網際網路、網路和共用中心。
7. 選取網路介面 (「區域連線」或「乙太網路」)，然後選擇「內容」。
8. 在 [區域連線內容] 頁面上，選擇 [網際網路通訊協定版本 4 (TCP/IPv4)]、[內容]、[進階]。
9. 選擇新增。
10. 在 [TCP/IP 位址] 對話方塊中，輸入 IP 位址的次要私人 IP 位址。針對子網遮罩，輸入您在步驟 1：在執行個體中設定靜態 IP 位址中於主要私有 IP 地址輸入的相同子網遮罩，然後選擇新增。



11. 確認 IP 地址設定，然後選擇確定。



12. 選擇確定、關閉。
13. 若要確認次要 IP 位址已新增至作業系統，請在中執行`ipconfig /all`指令 PowerShell。輸出應類似以下內容：

```
Ethernet adapter Ethernet 4:

   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : Amazon Elastic Network Adapter #2
   Physical Address. . . . . . . . . : 02-9C-3B-FC-8E-67
   DHCP Enabled. . . . . . . . . . . : No
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::f4d1:a773:5afa:cd1%7(Preferred)
   IPv4 Address. . . . . . . . . . . : 10.200.0.128(Preferred)
```

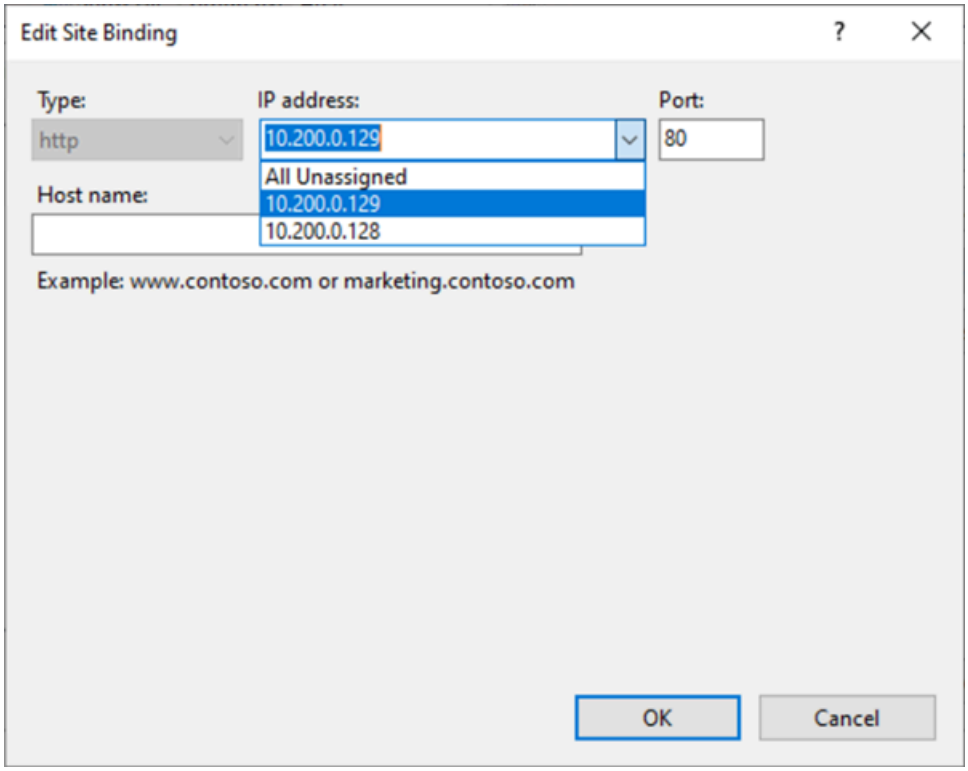
```
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   IPv4 Address. . . . . . . . . . . : 10.200.0.129(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 10.200.0.1
   DHCPv6 IAID . . . . . . . . . . . : 151166011
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-2D-67-AC-FC-12-34-9A-BE-A5-E7
   DNS Servers . . . . . . . . . . . : 10.200.0.2
   NetBIOS over Tcpip. . . . . . . . : Enabled
```

步驟 3：設定應用程式使用次要私有 IP 地址

您可以設定任何應用程式使用輔助私有 IP 地址。例如，如果您的執行個體正在執行網站IIS，您可以設定IIS為使用次要私有 IP 位址。

若要設IIS定使用次要私人 IP 位址

1. 連線到您的執行個體。
2. 開啟網際網路資訊服務 (IIS) 管理員。
3. 在連線窗格中，展開網站。
4. 開啟您網站的內容 (右鍵) 選單，然後選擇編輯繫結。
5. 在網站繫結對話方塊中，針對類型，選擇 http、編輯。
6. 在編輯網站繫結對話方塊中，針對 IP 地址，選取輔助私有 IP 地址。（默認情況下，每個網站接受來自所有 IP 地址的HTTP請求。）



7. 選擇確定、關閉。

Amazon EC2 執行個體主機名稱

本節說明您在VPC子網路中啟動執行個體時可用的 Amazon EC2 執行個體客體作業系統主機名稱類型。

主機名稱會區分您網路上的EC2執行個體。例如，如果您希望執行指令碼，與網路上的部分或全部執行個體通訊，則可以使用執行個體的主機名稱。

目錄

- EC2主機名稱的類型
- 在哪裡可以找到資源名稱和 IP 名稱
- 在資源名稱和 IP 名稱之間進行選擇
- 更改 Amazon 的基於資源的命名選項 EC2

EC2主機名稱的類型

以下列方式啟動執行個EC2體時，客體作業系統主機名稱有兩種主機名稱類型VPC：

- IP 名稱：舊版命名配置，當您啟動執行個體時，執行個體的私有位IPv4址會包含在執行個體的主機名稱中。IP 名稱在EC2執行個體的生命週期內存在。當用作私人DNS主機名時，它只會返回私人IPv4地址（A 記錄）。
- 資源名稱：啟動執行個體時，執行個EC2體 ID 會包含在執行個體的主機名稱中。資源名稱在EC2實例的生命週期內存在。當用作私人DNS主機名稱時，它可以同時傳回私人IPv4位址 (A 記錄) 和/或IPv6全域單點傳播位址 (AAAA記錄)。

EC2執行個體客體作業系統主機名稱類型取決於子網路設定：

- 如果執行個體啟動至IPv4唯一子網路，您可以選取 IP 名稱或資源名稱。
- 如果執行個體啟動至雙堆疊 (IPv4+IPv6) 子網路，您可以選取 IP 名稱或資源名稱。
- 如果執行個體啟動至IPv6唯一子網路，則會自動使用資源名稱。

目錄

- IP 名稱
- 資源名稱
- IP 名稱與資源名稱之間的區別

IP 名稱

當您使用 IP 名稱的主機名稱類型啟動EC2執行個體時，客體作業系統主機名稱會設定為使用私人位IPv4址。

- us-east-1 中執行個體的格式：*private-ipv4-address*.ec2.internal
- 範例：*ip-10-24-34-0*.ec2.internal
- 任何其他 AWS 區域中執行個體的格式：*private-ipv4-address.region*.compute.internal
- 範例：*ip-10-24-34-0.us-west-2*.compute.internal

資源名稱

當您在IPv6僅有子網路中啟動EC2執行個體時，預設會選取 [資源名稱] 的 [主機名稱] 類型。當您在IPv4唯一或雙堆疊 (IPv4+IPv6) 子網路中啟動執行個體時，您可以選取 [資源名稱] 選項。在啟動執行個體之後，您可以管理主機名稱組態。如需詳細資訊，請參閱更改 Amazon 的基於資源的命名選項EC2。

當您啟動主機名稱類型為資源名稱的EC2執行個體時，客體作業系統主機名稱會設定為使用EC2執行個體 ID。

- us-east-1 中執行個體的格式：*ec2-instance-id*.ec2.internal
- 範例：*i-0123456789abcdef*.ec2.internal
- 任何其他 AWS 區域中執行個體的格式：*ec2-instance-id.region*.compute.internal
- 範例：*i-0123456789abcdef.us-west-2*.compute.internal

IP 名稱與資源名稱之間的區別

DNSIP 名稱和資源名稱的查詢會共存，以確保回溯相容性，並可讓您從主機名稱的以 IP 為基礎的命名移轉至以資源為基礎的命名。對於以 IP DNS 名稱為基礎的私人主機名稱，您無法設定是否回應執行個體的 DNS A 記錄查詢。DNS無論客體作業系統主機名稱設定為何，都會回應記錄查詢。相反地，對於以資源DNS名稱為基礎的私人主機名稱，您可以設DNS定是否回應執行個體的 A 和/或DNSAAAA查詢。您可以在啟動執行個體或修改子網時設定回應行為。如需詳細資訊，請參閱更改 Amazon 的基於資源的命名選項 EC2。

在哪裡可以找到資源名稱和 IP 名稱

您可以在 Amazon 主EC2控台中查看主機名稱類型、資源名稱和 IP 名稱。

目錄

- 建立EC2執行個體時
- 檢視現有EC2執行個體的詳細資訊時

建立EC2執行個體時

建立EC2執行個體時，視您選取的子網路類型而定，資源名稱的主機名稱類型可能可用，或者可能已選取且無法修改。本節介紹您查看主機名稱類型資源名稱和 IP 名稱的案例。

案例 1

您可以在精靈中建立EC2執行個體 (請參閱使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體)，並在設定詳細資料時選擇您設定為IPv6僅限的子網路。

在此情況下，Hostname type (主機名稱類型) 的 Resource name (資源名稱) 會自動選取且無法修改。DNS[啟用 IP 名稱 IPv4 (A 記錄) DNS 要求] 和 [啟用資源型 IPv4 (A 記錄) 要求] 的主機名稱選項會自動取消選取，且無法修改。DNS預設會選取 「啟用以資源為基礎 IPv6 (AAAA記錄)」的DNS請求，

但是可以修改。如果選取此選項，對資源名稱的DNS要求將解析為此EC2執行處理的IPv6位址 (AAAA記錄)。

案例 2

您可以在精靈中建立EC2執行個體 (請參閱使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體)，當您設定詳細資料時，您可以選擇設定IPv4CIDR區塊或與區IPv6CIDR塊 (「雙重堆疊」) 的子網路。IPv4

在此情況下，會自動選取 「啟用 IP 名稱」IPv4 (A 記錄) DNS 要求，且無法變更。這表示對 IP 名稱的要求會解析為此EC2執行個體的位IPv4址 (A 記錄)。

選項預設為子網的組態，但您可以修改此執行個體的選項，取決於子網設定：

- 主機名稱類型：決定您要將執行個體的客EC2體作業系統主機名稱作為資源名稱還是 IP 名稱。預設值為 IP name (IP 名稱)。
- 啟用以資源為基礎的 IPv4 (A 記錄) DNS 請求：決定對資源名稱的請求是否解析為此EC2執行處理的專用IPv4地址 (A 記錄)。預設不會選取此選項。
- 啟用以資源為基礎的 IPv6 (AAAA記錄) DNS 請求：決定對資源名稱的請求是否解析為此EC2執行環境的IPv6GUA地址 (AAAA記錄)。預設不會選取此選項。

檢視現有EC2執行個體的詳細資訊時

您可以在執行個體的 [詳細資料] 索引標籤中查看現有EC2執行個EC2體的主機名稱值：

- Hostname type (主機名稱類型)：IP 名稱或資源名稱格式的主機名稱。
- 私人 IP DNS 名稱 (IPv4僅限)：一律解析為執行個體私人位IPv4址的 IP 名稱。
- 專用資源DNS名稱：解析為此執行處理選取之DNS記錄的資源名稱。
- 回答私人資源DNS名稱：資源名稱解析為IPv4（A），IPv6（AAAA）或IPv4和IPv6（A 和 AAAA）DNS記錄。

此外，如果您直接透過連線至EC2執行個體SSH並輸入`hostname`指令，就會看到 IP 名稱或資源名稱格式的主機名稱。

在資源名稱和 IP 名稱之間進行選擇

啟動EC2執行個體時 (請參閱使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體)，如果您選擇 [資源名稱] 名稱的 [主機名稱] 類型，則EC2執行個體會以資源名稱格式的主機名稱

啟動。在這種情況下，此EC2實例的DNS記錄也可以指向資源名稱。這可讓您彈性選擇該主機名稱是否解析為執行個體的IPv6位址、位址，或同時解析為IPv4與IPv6位址。IPv4如果您打算IPv6在 future 使用，或者如果您現在使用雙堆疊子網路，最好使用資源名稱的主機名稱類型，這樣您就可以變更執行個體主機名稱的DNS解析度，而不會對記錄本身進行任何變更。DNS資源名稱允許您在EC2實例上添加IPv4和刪除和IPv6DNS解析。

如果您改為選擇 IP 名稱的主機名稱類型，並將其用作DNS主機名稱，則它只能解析為執行個體的位IPv4址。即使執行個體同時具有相關聯的IPv6位址和IPv4位址，也不會解析為執行個體的IPv6位址。

更改 Amazon 的基於資源的命名選項 EC2

您可以變更子網路的主機DNS名稱類型和主機名稱組態，這會影響該主旨中所有後續的執行個體啟動，也可以在啟動EC2執行個體之後變更它們。

子網

在 Amazon VPC 主控台中選取子網路，然後選擇動作、編輯子網路設定，以修改子網路的組態。

> **Note**
>
> 變更子網路設定並不會變更已在子網路中啟動之EC2執行個體的組態。

- 主機名稱類型：決定是否要將子網路中啟動之EC2執行個體的客體作業系統主機名稱預設設定為資源名稱或 IP 名稱。
- 啟用DNS主機名稱IPv4（A 記錄）DNS請求：確定對您的資源名稱的請求/查詢是否解析為此EC2實例的私有IPv4地址（A 記錄）。
- 啟用DNS主機名稱IPv6（AAAA記錄）DNS請求：確定對您的資源名稱的請求/查詢是否解析為此EC2實例的IPv6地址（AAAA記錄）。

EC2 執行個體

請遵循本節中的步驟來修改執行處理的「主機名稱」類型和「DNS主機名稱」組EC2態。

考量事項

- 若要變更 Use resource based naming as guest OS hostname (使用資源型命名作為客體作業系統主機名稱) 設定，必須首先停止執行個體。若要變更回應DNS主機名稱 IPv4 (A 記錄) 要求或回應DNS主機名稱 IPv6 (AAAA記錄) 要求設定，您不必停止執行個體。

- 若要修改非EBS支援EC2執行個體類型的任何設定，您無法停止執行個體。您必須終止執行個體，並啟動具有所需主機名稱類型和主機名DNS稱組態的新執行個體。

若要修改執行個體的主機DNS名稱類型和主機名稱組態

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 如果您要變更 [使用以資源為基礎的命名做為客體作業系統主機名稱] 設定，請先停止EC2執行個體。否則，請跳過這個步驟。

   若要停止執行個體，請選取執行個體，然後選取 Instance state (執行個體狀態)、Stop instance (停止執行個體)。
3. 選取執行個體，然後選取 Actions (動作)、Instance settings (執行個體設定)、Change resource based naming options (變更資源型命名選項)。
   - 使用以資源為基礎的命名作為客體作業系統主機名稱：決定您要將EC2執行個體的客體作業系統主機名稱作為資源名稱還是 IP 名稱。
   - 回答DNS主機名稱IPv4（A 記錄）請求：確定對您的資源名稱的DNS請求/查詢是否解析為此EC2實例的私有IPv4地址。
   - 回答DNS主機名稱IPv6（AAAA記錄）DNS請求：確定對您的資源名稱的請求/查詢是否解析為此EC2實例的IPv6地址（AAAA記錄）。
4. 選擇 Save (儲存)。
5. 如果已停止執行個體，請重新啟動它。

將您自己的 IP 地址（BYOIP）帶到 Amazon EC2

您可以將部分或全部的公開路由IPv4或IPv6位址範圍從內部部署網路帶到您的 AWS 帳戶。您可以繼續控制地址範圍，並且可以通過在 Internet 上通告地址範圍 AWS。將地址範圍帶到 Amazon 後EC2，它將顯示在您 AWS 帳戶 的地址池中。

> **Note**
>
> 本文件說明如何將您自己的 IP 地址範圍EC2僅用於 Amazon。若要使用自己的 IP 位址範圍 AWS Global Accelerator，請參閱AWS Global Accelerator 開發人員指南中的「攜帶您自己的 IP 位址」(BYOIP)。若要攜帶您自己的 IP 位址範圍以供搭配使用 Amazon VPC IP Address Manager，請參閱 Amazon 使用VPCIPAM者指南IPAM中的教學課程：將 IP 位址帶入。

當您將 IP 位址範圍帶到時 AWS，會 AWS 驗證您是否控制 IP 位址範圍。您可以使用兩種方法來表明您控制範圍：

- 如果您的 IP 位址範圍是在支援 RDAP (例如ARINRIPE和APNIC) 的網際網路登錄中註冊，您可以使用此頁面上的程序，以 X.509 憑證驗證網域的控制權。
- 無論您的網際網路註冊表是否支援RDAP，您都可以使用 Amazon VPC IPAM 透過DNSTXT記錄驗證網域的控制權。該程序已在 Amazon VPC IPAM 使用者指南中的教學：將您的 IP 位址帶 IPAM入。

如需詳細資訊，請參閱 AWS 線上技術講座深入探討自攜 IP。

目錄

- BYOIP定義
- 需求與配額
- 區域可用性
- Local Zone 可用性
- Amazon 的BYOIP先決條件 EC2
- 在 Amazon 上使用您的地址範圍 EC2
- 在 Amazon 使用您的BYOIP地址範圍 EC2

BYOIP定義

- X.509 自我簽署的憑證 - 最常用來加密和驗證網路中資料的憑證標準。它是用來驗證從RDAP記錄 IP 空間的控制權的證書。 AWS 如需有關 X.509 憑證的詳細資訊，請參閱 RFC3280。
- Su@@ blic System Number (ASN) — 全域唯一識別碼，可定義由一或多個網路營運商執行的 IP 首碼群組，這些 IP 字首會維護單一、明確定義的路由原則。
- 區域網際網路登記處 (RIR) — 管理 IP 位址配置與註冊的組織，並ASNs在全球某個地區內。
- 登錄資料存取通訊協定 (RDAP) — 一種唯讀通訊協定，用於查詢RIR. 查詢的RIR資料庫中的項目稱為「RDAP記錄」。客戶需要透過RIR提供的機制更新某些記錄類型。系統會查詢這些記錄， AWS 以驗證中的位址空間的控制權。RIR
- 路由來源授權 (ROA) — 由客戶建立的物件，RIRs用於在特定的自主系統中驗證 IP 廣告。有關概述，請參閱ARIN網站上的路由來源授權（ROAs）。
- 本機網際網路登錄 (LIR) — 例如網際網路服務供應商，可為其客戶配置 IP 位址區塊的 Organizations。RIR

需求與配額

- 地址範圍必須在您的區域互聯網註冊處註冊 (RIR)。有關地理區域RIR的任何政策，請參閱您的。BYOIP目前支援在美國網際網路號碼註冊處 (ARIN)、IP 歐洲網路協調中心 (RIPE) 或亞太網路資訊中心 () 註冊。APNIC它必須以商業或機構實體註冊，而無法以個人身分註冊。
- 您可以帶來的最具體的IPv4地址範圍是 /24。
- 您可以提供CIDRs的最具體的IPv6地址範圍是 /48 公開廣告，而 /56 則不可公開廣CIDRs告。
- ROAs對於不可公開廣告的CIDR範圍不需要，但記RDAP錄仍需要更新。
- 您一次可以將每個地址範圍帶到一個地 AWS 區。
- 您可以將每個地 AWS 區總共五個BYOIPIPv4和IPv6地址範圍帶到您的 AWS 帳戶中。您無法調整BYOIPCIDRs使用 Service Quotas 主控台的配額，但可以透過連絡 Sup AWS port 中心來要求增加配額，如中的AWS 服務配額中所述AWS 一般參考。
- 除非您使用 Amazon IP 地址管理員 (IPAM) 並與 Organ AWS izations 整合， AWS RAM 否則無法IPAM與其他帳戶共用 VPC IP 位址範圍。如需詳細資訊，請參閱 Amazon VPC IPAM 使用者指南中的「IPAM與 Organ AWS izations 整合」。
- IP 地址範圍中的地址都必須有良好的歷史記錄。我們會調查 IP 地址範圍的評價，如果 IP 地址範圍包含的 IP 地址評價不佳或與惡意行為有關，我們保留拒絕該範圍的權利。
- 舊版位址空間是由網際網路指派號碼管理局 (IANA) 中央登錄機構 () 在形成區域網際網路登錄 (RIR) 系統之前所散佈的IPv4位址空間，仍需要對應的ROA物件。
- 對於LIRs，他們通常會使用手動程序來更新記錄。這可能需要數天的時間來部署，具體取決於LIR.
- 大型CIDR圖塊需要單一ROA物件和RDAP記錄。您可以使用單一物件和記錄 AWS，將多個較小的 AWS 區CIDR塊從該範圍引入，甚至跨多個區域。
- BYOIP不支援 Wavelength 區域或開啟 AWS Outposts。
- 請勿針對BYOIP中或其他任何其他項目進行RADb任何手動變更IRR。BYOIP會自動更新RADb。任何包含的手動變更都BYOIPASN會導致BYOIP佈建作業失敗。
- 引入位IPv4址範圍後 AWS，您就可以使用範圍內的所有 IP 位址，包括第一個位址 (網路位址) 和最後一個位址 (廣播位址)。

區域可用性

此BYOIP功能目前適用於所有商業AWS 地區，中國地區除外。

Local Zone 可用性

「本地區域」是您使用者所在地理位置附近的「 AWS 區域」的延伸。Local Zones 會分組為「網路邊界群組」。在中 AWS，網路邊界群組是可用區域 (AZs)、Local Zones 或 Wavelength 區域的集合，用於 AWS 通告公用 IP 位址。Local Zones 的網路邊界群組可能與 AWS 區域AZs中的網路邊界群組不同，以確保 AWS 網路與存取這些區域中資源的客戶之間的最小延遲或實際距離。

您可以使用`--network-border-group`選項佈建BYOIPv4位址範圍，並在下列區域網路邊界群組中進行通告：

- us-east-1-dfw-2
- us-west-2-lax-1
- us-west-2-phx-2

如果您已啟用 Local Zones (請參閱啟用本機區域)，您可以在佈建和通告時為 Local Zones 選擇網路邊界群組BYOIPv4CIDR。請謹慎選擇網路邊界群組，因為與其關聯的 AWS 資源必須位於相同的網路邊界群組中。EIP

> **Note**
> 您目前無法在 Local Zones 中佈建或通告BYOIPv6位址範圍。

Amazon 的BYOIP先決條件 EC2

的上線程序BYOIP有兩個階段，您必須執行三個步驟。這些步驟與下圖中說明的步驟相對應。我們在本文檔中包含了手動步驟，但您RIR可能會提供託管服務來幫助您完成這些步驟。

> **Tip**
> 本節中的任務需要 Linux 終端，而且可以使用 Linux、AWS CloudShell 或 Windows Subsystem for Linux 執行。

目錄

- 概要
- 建立私密金鑰並產生 X.509 憑證

- 將 X.509 憑證上傳至您的RDAP記錄 RIR
- 在您的中創建一個ROA對象 RIR

概要

準備階段

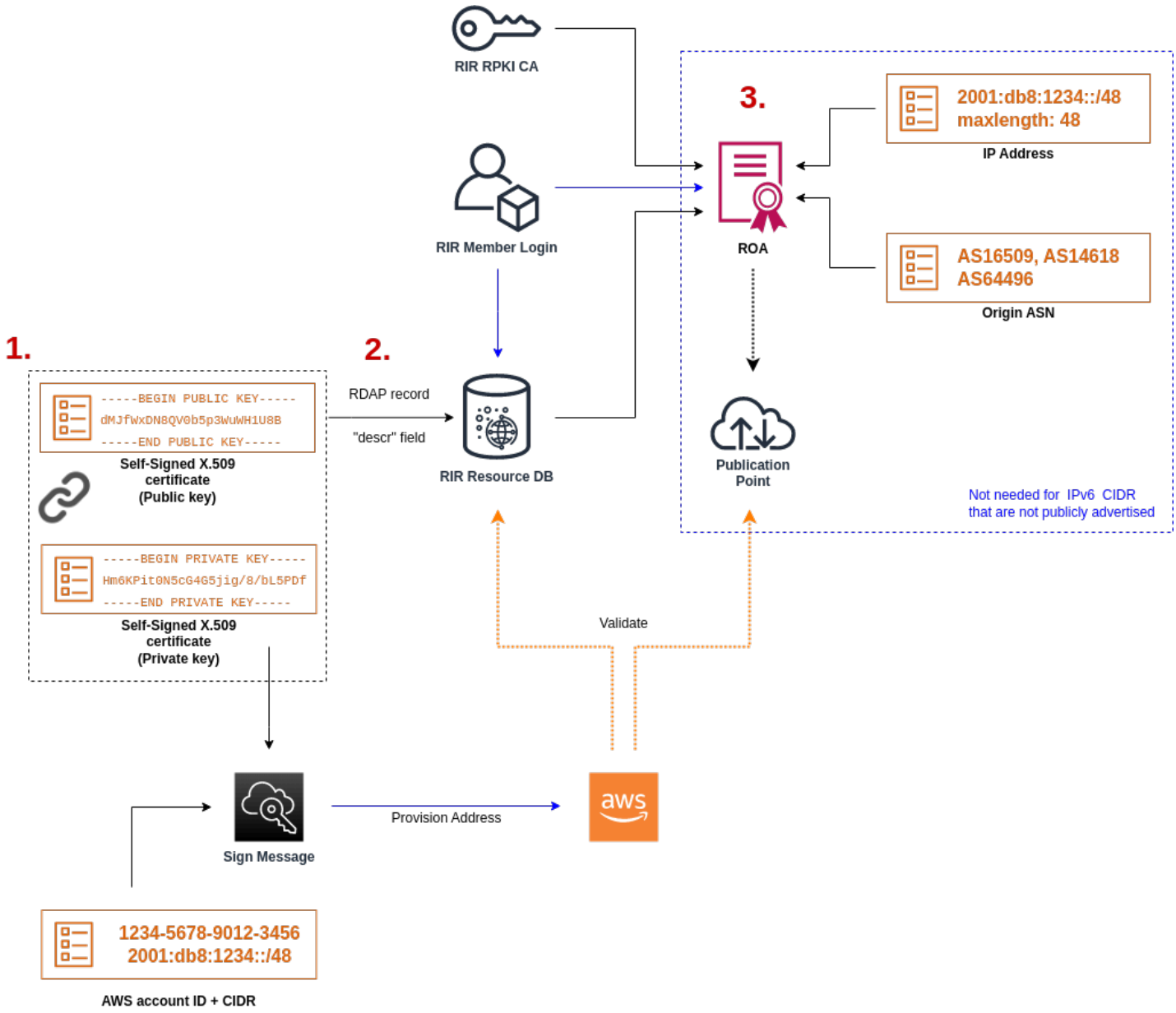
[1] 建立私密金鑰，並使用它來產生自我簽署的 X.509 憑證，以供驗證之用。此憑證僅在佈建階段使用。

RIR配置階段

[2] 將自簽名證書上傳到您的RDAP記錄評論中。

[3] 在ROA您的RIR. 定ROA義了所需的地址範圍，自治系統編號（ASNs）允許通告地址範圍，以及到期日期註冊的資源公鑰基礎結構（RPKI）您的RIR.

> **Note**
>
> 非公開廣告IPv6位址空間不需要 A ROA。



如要引入多個不連續的地址範圍，您必須針對每個地址範圍重複此程序。但是，如果跨多個不同區域分割連續 AWS 區塊，則不需要重複準備和RIR設定步驟。

引入地址範圍不會影響您先前引入的任何地址範圍。

建立私密金鑰並產生 X.509 憑證

使用下列程序建立自我簽署的 X.509 憑證，並將其新增至您的RDAP記錄中。RIR此 key pair 可用來驗證位址範圍RIR。這些指openssl令需要開啟SSL版本 1.0.2 或更高版本。

複製下列命令，並只取代預留位置值 (以彩色斜體文字表示)。

此程序遵循加密私密金鑰並要求密碼才能存取私密RSA金鑰的最佳作法。

1. 產生一個 RSA 2048 位元的私密金鑰，如下所示。

```
$ openssl genpkey -aes256 -algorithm RSA -pkeyopt rsa_keygen_bits:2048 -out
 private-key.pem
```

`-aes256` 參數會指定用來加密私有金鑰的演算法。此命令會傳回下列輸出，包括設定密碼短語的提示：

```
......+++
.+++
Enter PEM pass phrase: xxxxxxx
Verifying - Enter PEM pass phrase: xxxxxxx
```

您可以使用下列命令來檢查金鑰：

```
$ openssl pkey -in private-key.pem -text
```

這會傳回密碼短語提示和金鑰內容，應類似於以下內容：

```
Enter pass phrase for private-key.pem: xxxxxxx
-----BEGIN PRIVATE KEY-----
MIIEvgIBADANBgkqhkiG9w0BAQEFAASCBKgwggSkAgEAAoIBAQDFBXHRI4HVKAhh
3seiciooizCRTbJe1+YsxNTja4XyKypVGIFWDGhZs44FCHlPOOSVJ+NqP74w96oM
7DPS3xo9kaQyZBFn2YEp2EBq5vf307KHNRmZZUmkn0zHOSEpNmY2fMxISBxewlxR
FAniwmSd/8TDvHJMY9FvAIvWuTsv5l0tJKk+a91K4+tO3UdDR7Sno5WXExfsBrW3
g1ydo3TBsx8i5/YiVOcNApy7ge2/FiwY3aCXJB6r6nuF6H8mRgI4r4vkMRsOlAhJ
DnZPNeweboo+K3Q3lwbgbmOKD/z9svk8N/+hUTBtIX0fRtbG+PLIw3xWRHGrMSn2
BzsPVuDLAgMBAAECggEACiJUj2hfJkKv47Dc3es3Zex67A5uDVjXmxfox2Xhdupn
fAcNqAptV6fXt0SPUNbhUxbBKNbshoJGufFwXPli1SXnpzvkdU4Hyco4zgbhXFsE
RNYjYfOGzTPwdBLpNMB6k3Tp4RHse6dNrlH0jDhpioL8cQEBdBJyVF5X0wymEbmV
mC0jgH/MxsBAPWW6ZKicg9ULMlWiAZ3MRAZPjHHgpYkAAsUWKAbCBwVQcVjGO59W
jfZjzTX5pQtVVH68ruciH88DTZCwjCkjBhxg+OIkJBLE5wkh82jIHSivZ63flwLw
z+E0+HhELSZJrn2MY6Jxmik3qNNUOF/Z+3msdj2luQKBgQDjwlC/3jxp8zJy6P8o
JQKv7TdvMwUj4VSWOHZBHLv4evJaaia0uQjIo1UDa8AYitqhX1NmCCehGH8yuXj/
v6V3CzMKDkmRr1NrONnSz5QsndQ04Z6ihAQlPmJ96g4wKtgoC7AYpyP0g1a+4/sj
b1+o3YQI4pD/F71c+qaztH7PRwKBgQDdc23yNmT3+Jyptf0fKjEvONK+xwUKzi9c
L/OzBq5yOIC1Pz2T85gOe1i8kwZws+xlpG6uBT6lmIJELd0k59FyupNu4dPvX5SD
6GGqdx4jk9KvI74usGeOBohmF0phTHkrWKBxXiyT0oS8zjnJlEn8ysIpGgO28jjr
LpaHNZ/MXQKBgQDfLNcnS0LzpsS2aK0tzyZU8SMyqVHOGMxj7quhneBq2T6FbiLD
```

```
T9TVlYaGNZ0j71vQaLI19qOubWymbautH0Op5KV8owdf4+bf1/NJaPIOzhDUSIjD
Qo01WW31Z9XDSRhKFTnWzmCjBdeIcajyzf10YKsycaAW9lItu8aBrMndnQKBgQDb
nNp/JyRwqjOrNljk7DHEs+SD39kHQzzCfqd+dnTPv2sc06+cpym3yulQcbokULpy
fmRo3bin/pvJQ3aZX/Bdh9woTXqhXDdrrSwWInVYMQPyPk8f/D9mIOJp5FUWMwHD
U+whIZSxsEeE+jtixlWtheKRYkQmzQZXbWdIhYyI3QKBgD+F/6wcZ85QW8nAUykA
3WrSIx/3cwDGdm4NRGct8ZOZjTHjiy9ojMOD1L7iMhRQ/3k3hUsin5LDMp/ryWGG
x4uIaLat40kiC7T4I66DM7P59euqdz3w0PD+VU+h7GSivvsFDdySUt7bNK0AUVLh
dMJfWxDN8QV0b5p3WuWH1U8B
-----END PRIVATE KEY-----
Private-Key: (2048 bit)
modulus:
    00:c5:05:71:d1:23:81:d5:28:08:61:de:c7:a2:72:
    2a:28:8b:30:91:4d:b2:5e:d7:e6:2c:c4:d4:e3:6b:
    85:f2:2b:2a:55:18:81:56:0c:68:59:b3:8e:05:08:
    79:4f:38:e4:95:27:e3:6a:3f:be:30:f7:aa:0c:ec:
    33:d2:df:1a:3d:91:a4:32:64:11:67:d9:81:29:d8:
    40:6a:e6:f7:f7:d3:b2:87:35:19:99:65:49:a4:9f:
    4c:c7:39:21:29:36:66:36:7c:cc:48:48:1c:5e:c2:
    5c:51:14:09:e2:c2:64:9d:ff:c4:c3:bc:72:4c:63:
    d1:6f:00:8b:d6:b9:3b:2f:e6:5d:2d:24:a9:3e:6b:
    dd:4a:e3:eb:4e:dd:47:43:47:b4:a7:a3:95:97:13:
    17:ec:06:b5:b7:83:5c:9d:a3:74:c1:b3:1f:22:e7:
    f6:22:54:e7:0d:02:9c:bb:81:ed:bf:16:2c:18:dd:
    a0:97:24:1e:ab:ea:7b:85:e8:7f:26:46:02:38:af:
    8b:e4:31:1b:0e:94:08:49:0e:76:4f:35:ec:1e:6e:
    8a:3e:2b:74:37:97:06:e0:6e:63:8a:0f:fc:fd:b2:
    f9:3c:37:ff:a1:51:30:6d:21:7d:1f:46:d6:c6:f8:
    f2:c8:c3:7c:56:44:71:ab:31:29:f6:07:3b:0f:56:
    e0:cb
publicExponent: 65537 (0x10001)
privateExponent:
    0a:22:54:8f:68:5f:26:42:af:e3:b0:dc:dd:eb:37:
    65:ec:7a:ec:0e:6e:0d:58:d7:9b:17:e8:c7:65:e1:
    76:ea:67:7c:07:0d:a8:0a:6d:57:a7:d7:b7:44:8f:
    50:d6:e1:53:16:c1:28:d6:ec:86:82:46:b9:f1:70:
    5c:f9:62:d5:25:e7:a7:3b:e4:75:4e:07:c9:ca:38:
    ce:06:e1:5c:5b:04:44:d6:23:61:f3:86:cd:33:f0:
    74:12:e9:34:c0:7a:93:74:e9:e1:11:ec:7b:a7:4d:
    ae:51:f4:8c:38:69:8a:82:fc:71:01:01:74:12:72:
    54:5e:57:d3:0c:a6:11:b9:95:98:2d:23:80:7f:cc:
    c6:c0:40:3d:65:ba:64:a8:9c:83:d5:0b:32:55:a2:
    01:9d:cc:44:06:4f:8c:71:e0:a5:89:00:02:c5:16:
    28:06:c2:07:05:50:71:58:c6:3b:9f:56:8d:f6:63:
    cd:35:f9:a5:0b:55:54:7e:bc:ae:e7:22:1f:cf:03:
```

```
    4d:90:b0:8c:29:23:06:1c:60:f8:e2:24:24:12:c4:
    e7:09:21:f3:68:c8:1d:28:af:67:ad:df:97:02:f0:
    cf:e1:34:f8:78:44:2d:26:49:ae:7d:8c:63:a2:71:
    9a:29:37:a8:d3:54:38:5f:d9:fb:79:ac:76:3d:a5:
    b9
prime1:
    00:e3:c2:50:bf:de:3c:69:f3:32:72:e8:ff:28:25:
    02:af:ed:37:6f:33:05:23:e1:54:96:38:76:41:1c:
    bb:f8:7a:f2:5a:6a:26:b4:b9:08:c8:a3:55:03:6b:
    c0:18:8a:da:a1:5f:53:66:08:27:a1:18:7f:32:b9:
    78:ff:bf:a5:77:0b:33:0a:0e:49:91:af:53:6b:38:
    d9:d2:cf:94:2c:9d:d4:34:e1:9e:a2:84:04:25:3e:
    62:7d:ea:0e:30:2a:d8:28:0b:b0:18:a7:23:f4:83:
    56:be:e3:fb:23:6f:5f:a8:dd:84:08:e2:90:ff:17:
    bd:5c:fa:a6:b3:b4:7e:cf:47
prime2:
    00:dd:73:6d:f2:36:64:f7:f8:9c:a9:b5:fd:1f:2a:
    31:2f:38:d2:be:c7:05:0a:ce:2f:5c:2f:f3:b3:06:
    ae:72:38:80:b5:3f:3d:93:f3:98:0e:7b:58:bc:93:
    06:70:b3:ec:65:a4:6e:ae:05:3e:a5:98:82:44:2d:
    dd:24:e7:d1:72:ba:93:6e:e1:d3:ef:5f:94:83:e8:
    61:aa:77:1e:23:93:d2:af:23:be:2e:b0:67:8e:06:
    88:66:17:4a:61:4c:79:2b:58:a0:71:5e:2c:93:d2:
    84:bc:ce:39:c9:94:49:fc:ca:c2:29:1a:03:b6:f2:
    38:eb:2e:96:87:35:9f:cc:5d
exponent1:
    00:df:2c:d7:27:4b:42:f3:a6:c4:b6:68:ad:2d:cf:
    26:54:f1:23:32:a9:51:ce:18:cc:63:ee:ab:a1:9d:
    e0:6a:d9:3e:85:6e:22:c3:4f:d4:d5:95:86:86:35:
    9d:23:ef:5b:d0:68:b2:35:f6:a3:ae:6d:6c:a6:6d:
    ab:ad:1f:43:a9:e4:a5:7c:a3:07:5f:e3:e6:df:d7:
    f3:49:68:f2:0e:ce:10:d4:48:88:c3:42:8d:35:59:
    6d:f5:67:d5:c3:49:18:4a:15:39:d6:ce:60:a3:05:
    d7:88:71:a8:f2:cd:fd:74:60:ab:32:71:a0:16:f6:
    52:2d:bb:c6:81:ac:c9:dd:9d
exponent2:
    00:db:9c:da:7f:27:24:70:aa:33:ab:36:58:e4:ec:
    31:c4:b3:e4:83:df:d9:07:43:3c:c2:7e:a7:7e:76:
    74:cf:bf:6b:1c:d3:af:9c:a7:29:b7:ca:e9:50:71:
    ba:24:50:ba:72:7e:64:68:dd:b8:a7:fe:9b:c9:43:
    76:99:5f:f0:5d:87:dc:28:4d:7a:a1:5c:37:6b:ad:
    2c:16:22:75:58:31:03:f2:3e:4f:1f:fc:3f:66:20:
    e2:69:e4:55:16:33:01:c3:53:ec:21:21:94:b1:b0:
    47:84:fa:3b:62:c6:55:ad:85:e2:91:62:44:26:cd:
```

```
    06:57:6d:67:48:85:8c:88:dd
coefficient:
    3f:85:ff:ac:1c:67:ce:50:5b:c9:c0:53:29:00:dd:
    6a:d2:23:1f:f7:73:00:c6:76:6e:0d:44:67:2d:f1:
    93:99:8d:31:e3:8b:2f:68:8c:c3:83:d4:be:e2:32:
    14:50:ff:79:37:85:4b:22:9f:92:c3:32:9f:eb:c9:
    61:86:c7:8b:88:68:b6:ad:e3:49:22:0b:b4:f8:23:
    ae:83:33:b3:f9:f5:eb:aa:77:3d:f0:d0:f0:fe:55:
    4f:a1:ec:64:a2:be:fb:05:0d:dc:92:52:de:db:34:
    ad:00:51:52:e1:74:c2:5f:5b:10:cd:f1:05:74:6f:
    9a:77:5a:e5:87:d5:4f:01
```

不使用私有金鑰時，請將它保管在安全位置。

2. 使用在上一個步驟中建立的私密金鑰產生 X.509 憑證。在此範例中，憑證會在 365 天後過期；在這段時間之後，就無法再信任此憑證。務必設定適當的過期期限。憑證只能在佈建程序期間有效。佈建完成後，您RIR可以從記錄中移除憑證。`tr -d "\n"` 命令從輸出中刪除換行符號字元 (換行符)。出現提示時，您需要提供 Common Name (通用名稱)，但其他欄位可以保留空白。

```
$ openssl req -new -x509 -key private-key.pem -days 365 | tr -d "\n" >
 certificate.pem
```

此輸出結果類似以下內容：

```
Enter pass phrase for private-key.pem: xxxxxxx
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) []:
State or Province Name (full name) []:
Locality Name (eg, city) []:
Organization Name (eg, company) []:
Organizational Unit Name (eg, section) []:
Common Name (eg, fully qualified host name) []:example.com
Email Address []:
```

> **Note**
>
> AWS 佈建時不需要「一般名稱」。其可為任何內部或公有網域名稱。

您可以使用下列命令來檢查憑證：

```
$ cat certificate.pem
```

輸出應該是一個沒有換行符的長PEM編碼字符串，前面加上`-----BEGIN CERTIFICATE-----`和後跟。`-----END CERTIFICATE-----`

將 X.509 憑證上傳至您的RDAP記錄 RIR

將您先前建立的憑證新增至您的RDAP記錄RIR。請務必包含 `-----BEGIN CERTIFICATE-----` 和 `-----END CERTIFICATE-----` 字串之前和之後的編碼部分。所有這些內容都必須在單獨的長線上。更新程序RDAP取決於您的RIR：

- 針對ARIN，請使用客戶管理員入口網站，在代表您位址範圍的「網路資訊」物件的「公開註解」區段中新增憑證。請勿將其新增至您組織的評論區段。
- 對於RIPE，將證書作為新的「描述」字段添加到代表您的地址範圍的「inetnum」或「inet6num」對象中。這些通常可以在RIPE數據庫門戶的「我的資源」部分找到。請勿將其新增至組織的註解區段，或上述物件的「備註」欄位。
- 對於APNIC，將憑證以電子郵件傳送至 helpdesk@apnic.net，以手動將其新增至您地址範圍的「備註」欄位。使用 IP 位址的APNIC授權聯絡人傳送電子郵件。

您可以在下列佈建階段完成後，從記錄中移除憑證。RIR

在您的中創建一個ROA對象 RIR

建立ROA物件以授權 Amazon ASNs 16509 和 14618 通告您的地址範圍，以及目前授權可通告ASNs該地址範圍的物件。對於 AWS GovCloud (US) Regions，請授權 ASN 8987 而非 16509 及 14618。您必須將最大長度設定為您CIDR要引入的大小。您可以帶來的最具體的IPv4前綴是 /24。您可以提供CIDRs的最具體的IPv6地址範圍是 /48 公開廣告，而 /56 則不可公開廣CIDRs告。

> **Important**
>
> 如果您要為 Amazon VPC IP 地址管理員 (IPAM) 建立ROA物件，則在建立物件時ROAs，IPv4CIDRs必須將 IP 位址前綴的最大長度設定為/24。針對 IPv6CIDRs，如果您要將它們新增至可廣告集區，則 IP 位址前置詞的最大長度必須/48為。這樣可確保您有充分的靈活性，可以跨 AWS 區域劃分您的公有 IP 地址。IPAM強制執行您設定的最大長度。如需有關目標地BYOIP址的詳細資訊IPAM，請參閱 Amazon VPC IPAM 使用者指南IPAM中的教學課程：BYOIP地址CIDRs到。

Amazon 可能需要長達 24 小時ROA才能使用。有關更多信息，請諮詢您的RIR：

- ARIN— ROA請求
- RIPE— 管理 ROAs
- APNIC— 路線管理

將廣告從現場部署工作負載遷移到時 AWS，必須ASN先ROA為現有工作負載建立廣告，然後才能ROAs為 Amazon 建立ASNs。否則，可能會對現有路由和廣告造成影響。

> **Important**
>
> 為了讓 Amazon 宣傳並繼續廣告您的 IP 地址範圍，您ROAs的 Amazon ASNs 必須符合上述準則。如ROAs果您無效或不符合上述準則，Amazon 保留停止廣告您的 IP 位址範圍的權利。

> **Note**
>
> 非公開廣告IPv6位址空間不需要執行此步驟。

在 Amazon 上使用您的地址範圍 EC2

根據您的需求，的入職程序BYOIP包括下列工作。

任務

- 提供可公開廣告的地址範圍 AWS
- 提供不可公開廣告的IPv6地址範圍

- 通告位址範圍 AWS
- 解除佈建地址範圍
- 驗證您的 BYOIP

提供可公開廣告的地址範圍 AWS

佈建要搭配使用的地址範圍時 AWS，您確認自己控制了地址範圍，並授權 Amazon 宣傳該地址範圍。我們也會透過簽署的授權訊息驗證您確實控制該地址範圍。此郵件會使用您在使用 X.509 憑證更新RDAP記錄時所使用的自我簽署 X.509 key pair 簽署。 AWS 需要一個密碼編譯簽名的RIR授權消息，它呈現給. 會針對您新增的憑RIR證驗證簽章RDAP，並根據檢查授權詳細資料。ROA

佈建地址範圍

1. 編寫訊息

   撰寫純文字授權訊息。訊息格式如下所示，其中的日期是訊息的過期日期：

   ```
   1|aws|account|cidr|YYYYMMDD|SHA256|RSAPSS
   ```

   用您自己的值取代帳戶號碼、地址範圍和到期日，建立類似下列內容的訊息：

   ```
   text_message="1|aws|0123456789AB|198.51.100.0/24|20211231|SHA256|RSAPSS"
   ```

   這不要與具有類似外觀的ROA消息混淆。

2. 簽署訊息

   使用您先前建立的私有金鑰來簽署純文字訊息。此命令傳回的簽章是一個長字串，您需要在下一個步驟中使用它。

   > **Important**
   > 建議您複製並貼上此命令。除訊息內容外，請勿修改或取代任何值。

   ```
   signed_message=$( echo -n $text_message | openssl dgst -sha256 -sigopt
    rsa_padding_mode:pss -sigopt rsa_pss_saltlen:-1 -sign private-key.pem -keyform PEM
    | openssl base64 | tr -- '+=/' '-_~' | tr -d "\n")
   ```

3. 佈建地址

   使用 AWS CLI provision-byoip-cidr指令佈建位址範圍。`--cidr-authorization-context` 選項會使用您先前建立的訊息和簽章字串。

   > **Important**
   >
   > 如果BYOIP範圍與您的配置不同，則必須指定應佈建範圍的 AWS 區域 AWS CLI`Default region name`。

   ```
   aws ec2 provision-byoip-cidr --cidr address-range --cidr-authorization-context
    Message="$text_message",Signature="$signed_message" --region us-east-1
   ```

   佈建地址範圍是一種非同步操作；系統會立即傳回呼叫，但地址範圍要等到其狀態從 `pending-provision` 變更為 `provisioned` 時，才能開始使用。

4. 監控進度

   雖然大多數佈建將在兩小時內完成，但可能需要長達一週的時間才能完成可公開廣告範圍的佈建程序。使用指describe-byoip-cidrs令監視進度，如下列範例所示：

   ```
   aws ec2 describe-byoip-cidrs --max-results 5 --region us-east-1
   ```

   如果在佈建期間發生問題且狀態變成 `failed-provision`，您必須在問題解決後再次執行 `provision-byoip-cidr` 命令。

提供不可公開廣告的IPv6地址範圍

根據預設，可向網際網路公開公告所佈建的地址範圍。您可以佈建不可公開廣告的IPv6地址範圍。對於不可公開公告的路由，佈建程序通常會在幾分鐘內完成。將來自非公用地址範圍的IPv6CIDR區塊與 a 建立關聯時VPC，只IPv6CIDR能透過支援IPv6的混合式連線選項 (例如AWS Direct ConnectAWS網站對站台VPN或 Amazon VPC Transit Gateway) 存取。

A ROA 不需要提供非公共地址範圍。

> **Important**
>
> - 您只能指定在佈建期間可否公開公告地址範圍。您稍後無法變更可公告狀態。
> - Amazon VPC 不支持唯一的本地地址（ULA）CIDRs。所有VPCs必須具有唯一的IPv6CIDRs。兩個不VPCs能有相同的IPv6CIDR範圍。

若要佈建不可公開廣告的IPv6位址範圍，請使用下列provision-byoip-cidr指令。

```
aws ec2 provision-byoip-cidr --cidr address-range --cidr-authorization-context
 Message="$text_message",Signature="$signed_message" --no-publicly-advertisable --
region us-east-1
```

通告位址範圍 AWS

佈建好地址範圍之後，即可將其公告。佈建好地址範圍之後，您必須公布。對於已經佈建好的地址範圍，您不能必須只公布一部分。

如果您佈建的IPv6位址範圍不會公開廣告，則不需要完成此步驟。

我們建議您在通過廣告之前，停止在其他位置廣告該地址範圍或範圍的任何部分 AWS。如果您持續從其他位置宣傳您的 IP 位址範圍或其任何部分，我們將無法可靠地支援或疑難排解問題。具體而言，我們無法保證到達位址範圍或部分範圍的流量會進入我們的網路。

為了盡量減少停機時間，您可以將 AWS 資源設定為在廣告位址集區之前使用位址集區中的位址，然後同時停止從目前位置發佈廣告，並開始通過 AWS廣告。如需從您的地址集區配置彈性 IP 地址的詳細資訊，請參閱配置彈性 IP 地址。

限制

- 即使您每次都指定不同的地址範圍，最多仍只能每 10 秒執行一次 advertise-byoip-cidr 命令。
- 即使您每次都指定不同的地址範圍，最多仍只能每 10 秒執行一次 withdraw-byoip-cidr 命令。

若要通告位址範圍，請使用下列advertise-byoip-cidr命令。

```
aws ec2 advertise-byoip-cidr --cidr address-range --region us-east-1
```

要停止廣告地址範圍，請使用以下withdraw-byoip-cidr命令。

```
aws ec2 withdraw-byoip-cidr --cidr address-range --region us-east-1
```

解除佈建地址範圍

若要停止使用您的位址範圍 AWS，請先釋出任何彈性 IP 位址，並取消仍從位址集區配置的任何IPv6CIDR區塊的關聯。然後停止公告位址範圍，最後取消佈建位址範圍。

您無法取消佈建一部分的位址範圍。如果您想要使用更特定的位址範圍 AWS，請取消佈建整個位址範圍，並佈建更特定的位址範圍。

(IPv4) 若要釋放每個彈性 IP 位址，請使用下列釋放位址指令。

```
aws ec2 release-address --allocation-id eipalloc-12345678abcabcabc --region us-east-1
```

(IPv6) 若要取消IPv6CIDR圖塊的關聯，請使用下列disassociate-vpc-cidr-block指令。

```
aws ec2 disassociate-vpc-cidr-block --association-id vpc-cidr-assoc-12345abcd1234abc1
 --region us-east-1
```

要停止廣告地址範圍，請使用以下withdraw-byoip-cidr命令。

```
aws ec2 withdraw-byoip-cidr --cidr address-range --region us-east-1
```

若要取消佈建位址範圍，請使用下列deprovision-byoip-cidr命令。

```
aws ec2 deprovision-byoip-cidr --cidr address-range --region us-east-1
```

取消佈建位址範圍最多可能需要一天的時間。

驗證您的 BYOIP

1. 驗證自我簽署的 x.509 金鑰對

   透過 whois 命令，驗證憑證是否已上傳且有效。

   對於ARIN，用`whois -h whois.arin.net r + 2001:0DB8:6172::/48`於查詢地址範圍的RDAP記錄。檢查命令輸出中 `NetRange` (網路範圍)的 `Public Comments` 部分。該憑證應新增至地址範圍的 `Public Comments` 部分中。

   您可使用下列命令來檢查包含憑證的 `Public Comments`：

```
whois -h whois.arin.net r + 2001:0DB8:6172::/48 | grep Comments | grep BEGIN
```

此會傳回含有金鑰內容的輸出，應類似於下列內容：


```
Public Comments:
-----BEGIN CERTIFICATE-----
MIID1zCCAr+gAwIBAgIUBkRPNSLrPqbRAFP8RDAHSP+I1TowDQYJKoZIhvcNAQE
LBQAwezELMAkGA1UEBhMCTloxETAPBgNVBAgMCEF1Y2tsYW5kMREwDwYDVQQHDA
hBdWNrbGFuZDEcMBoGA1UECgwTQW1hem9uIFdlYiBTZXJ2aWNlczETMBEGA1UEC
wwKQllPSVAgRGVtbzETMBEGA1UEAwwKQllPSVAgRGVtbzAeFw0yMTEyMDcyMDI0
NTRaFw0yMjEyMDcyMDI0NTRaMHsxCzAJBgNVBAYTAk5aMREwDwYDVQQIDAhBdWN
rbGFuZDERMA8GA1UEBwwIQXVja2xhbmQxHDAaBgNVBAoME0FtYXpvbiBXZWIgU2
VydmljZXMxEzARBgNVBAsMCkJZT0lQIERlbW8xEzARBgNVBAMMCkJZT0lQIERlb
W8wggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQCfmacvDp0wZ0ceiXXc
R/q27mHI/U5HKt7SST4X2eAqufR9wXkfNanAEskgAseyFypwEEQr4CJijI/5hp9
prh+jsWHWwkFRoBRR9FBtwcU/45XDXLga7D3stsI5QesHVRwOaXUdprAnndaTug
mDPkD0vrl475JWDSIm+PUxGWLy+60aBqiaZq35wU/x+wXlAqBXg4MZK2KoUu27k
Yt2zhmy0S7Ky+oRfRJ9QbAiSu/RwhQbh5Mkp1ZnVIc7NqnhdeIW48QaYjhMlUEf
xdaqYUinzz8KpjfADZ4Hvqj9jWZ/eXo/9b2rGlHWkJsbhr0VEUyAGu1bwkgcdww
3A7NjOxQbAgMBAAGjUzBRMB0GA1UdDgQWBBStFyujN6SYBr2glHpGt0XGF7GbGT
AfBgNVHSMEGDAWgBStFyujN6SYBr2glHpGt0XGF7GbGTAPBgNVHRMBAf8EBTADA
QH/MA0GCSqGSIb3DQEBCwUAA4IBAQBX6nn6YLhz521lfyVfxY0t6o3410bQAeAF
08ud+ICtmQ4IO4A4B7zV3zIVYr0clrOOaFyLxngwMYN0XY5tVhDQqk4/gmDNEKS
Zy2QkX4Eg0YUWVzOyt6fPzjOvJLcsqc1hcF9wySL507XQz76Uk5cFypBOzbnk35
UkWrzA9KK97cXckfIESgK/k1N4ecwxwG6VQ8mBGqVpPpey+dXpzzzv1iBKN/VY4
ydjgH/LBfdTsVarmmy2vtWBxwrqkFvpdhSGCvRDl/qdO/GIDJi77dmZWkh/ic90
MNk1f38gs1jrCj8lThoar17Uo9y/Q5qJIsoNPyQrJRzqFU9F3FBjiPJF
-----END CERTIFICATE-----
```


對於RIPE，用`whois -r -h whois.ripe.net` *2001:0DB8:7269::/48*於查詢地址範圍的RDAP記錄。檢查命令輸出中 `inetnum` 物件 (網路範圍)的 `descr` 部分。該憑證應新增為地址範圍的新 `descr` 欄位。

您可使用下列命令來檢查包含憑證的 `descr`：

```
whois -r -h whois.ripe.net 2001:0DB8:7269::/48 | grep descr | grep BEGIN
```

此會傳回含有金鑰內容的輸出，應類似於下列內容：

```
descr:
```

```
-----BEGIN CERTIFICATE-----MIID1zCCAr+gAwIBAgIUBkRPNSLrPqbRAFP8
RDAHSP+I1TowDQYJKoZIhvcNAQELBQAwezELMAkGA1UEBhMCTloxETAPBgNVBAg
MCEF1Y2tsYW5kMREwDwYDVQQHDAhBdWNrbGFuZDEcMBoGA1UECgwTQW1hem9uIF
dlYiBTZXJ2aWNlczETMBEGA1UECwwKQllPSVAgRGVtbzETMBEGA1UEAwwKQllPS
VAgRGVtbzAeFw0yMTEyMDcyMDI0NTRaFw0yMjEyMDcyMDI0NTRaMHsxCzAJBgNV
BAYTAk5aMREwDwYDVQQIDAhBdWNrbGFuZDERMA8GA1UEBwwIQXVja2xhbmQxHDA
aBgNVBAoME0FtYXpvbiBXZWIgU2VydmljZXMxEzARBgNVBAsMCkJZT0lQIERlbW
8xEzARBgNVBAMMCkJZT0lQIERlbW8wggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwg
gEKAoIBAQCfmacvDp0wZ0ceiXXcR/q27mHI/U5HKt7SST4X2eAqufR9wXkfNanA
EskgAseyFypwEEQr4CJijI/5hp9prh+jsWHWwkFRoBRR9FBtwcU/45XDXLga7D3
stsI5QesHVRwOaXUdprAnndaTugmDPkD0vrl475JWDSIm+PUxGWLy+60aBqiaZq
35wU/x+wXlAqBXg4MZK2KoUu27kYt2zhmy0S7Ky+oRfRJ9QbAiSu/RwhQbh5Mkp
1ZnVIc7NqnhdeIW48QaYjhMlUEfxdaqYUinzz8KpjfADZ4Hvqj9jWZ/eXo/9b2r
GlHWkJsbhr0VEUyAGu1bwkgcdww3A7NjOxQbAgMBAAGjUzBRMB0GA1UdDgQWBBS
tFyujN6SYBr2glHpGt0XGF7GbGTAfBgNVHSMEGDAWgBStFyujN6SYBr2glHpGt0
XGF7GbGTAPBgNVHRMBAf8EBTADAQH/MA0GCSqGSIb3DQEBCwUAA4IBAQBX6nn6Y
Lhz521lfyVfxY0t6o3410bQAeAF08ud+ICtmQ4IO4A4B7zV3zIVYr0clrOOaFyL
xngwMYN0XY5tVhDQqk4/gmDNEKSZy2QkX4Eg0YUWVzOyt6fPzjOvJLcsqc1hcF9
wySL507XQz76Uk5cFypBOzbnk35UkWrzA9KK97cXckfIESgK/k1N4ecwxwG6VQ8
mBGqVpPpey+dXpzzzv1iBKN/VY4ydjgH/LBfdTsVarmmy2vtWBxwrqkFvpdhSGC
vRDl/qdO/GIDJi77dmZWkh/ic90MNk1f38gs1jrCj8lThoar17Uo9y/Q5qJIsoN
PyQrJRzqFU9F3FBjiPJF
-----END CERTIFICATE-----
```

對於APNIC，用`whois -h whois.apnic.net` *2001:0DB8:6170::/48*於查詢BYOIP地址範圍的RDAP記錄。檢查命令輸出中 `inetnum` 物件 (網路範圍)的 `remarks` 部分。該憑證應新增為地址範圍的新 `remarks` 欄位。

您可使用下列命令來檢查包含憑證的 `remarks`：

```
whois -h whois.apnic.net 2001:0DB8:6170::/48 | grep remarks | grep BEGIN
```

此會傳回含有金鑰內容的輸出，應類似於下列內容：

```
remarks:
-----BEGIN CERTIFICATE-----
MIID1zCCAr+gAwIBAgIUBkRPNSLrPqbRAFP8RDAHSP+I1TowDQYJKoZIhvcNAQE
LBQAwezELMAkGA1UEBhMCTloxETAPBgNVBAgMCEF1Y2tsYW5kMREwDwYDVQQHDA
hBdWNrbGFuZDEcMBoGA1UECgwTQW1hem9uIFdlYiBTZXJ2aWNlczETMBEGA1UEC
wwKQllPSVAgRGVtbzETMBEGA1UEAwwKQllPSVAgRGVtbzAeFw0yMTEyMDcyMDI0
NTRaFw0yMjEyMDcyMDI0NTRaMHsxCzAJBgNVBAYTAk5aMREwDwYDVQQIDAhBdWN
rbGFuZDERMA8GA1UEBwwIQXVja2xhbmQxHDAaBgNVBAoME0FtYXpvbiBXZWIgU2
```

```
VydmljZXMxEzARBgNVBAsMCkJZT0lQIERlbW8xEzARBgNVBAMMCkJZT0lQIERlb
W8wggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQCfmacvDp0wZ0ceiXXc
R/q27mHI/U5HKt7SST4X2eAqufR9wXkfNanAEskgAseyFypwEEQr4CJijI/5hp9
prh+jsWHWwkFRoBRR9FBtwcU/45XDXLga7D3stsI5QesHVRwOaXUdprAnndaTug
mDPkD0vrl475JWDSIm+PUxGWLy+60aBqiaZq35wU/x+wXlAqBXg4MZK2KoUu27k
Yt2zhmy0S7Ky+oRfRJ9QbAiSu/RwhQbh5Mkp1ZnVIc7NqnhdeIW48QaYjhMlUEf
xdaqYUinzz8KpjfADZ4Hvqj9jWZ/eXo/9b2rGlHWkJsbhr0VEUyAGu1bwkgcdww
3A7NjOxQbAgMBAAGjUzBRMB0GA1UdDgQWBBStFyujN6SYBr2glHpGt0XGF7GbGT
AfBgNVHSMEGDAWgBStFyujN6SYBr2glHpGt0XGF7GbGTAPBgNVHRMBAf8EBTADA
QH/MA0GCSqGSIb3DQEBCwUAA4IBAQBX6nn6YLhz521lfyVfxY0t6o3410bQAeAF
08ud+ICtmQ4IO4A4B7zV3zIVYr0clrOOaFyLxngwMYN0XY5tVhDQqk4/gmDNEKS
Zy2QkX4Eg0YUWVzOyt6fPzjOvJLcsqc1hcF9wySL507XQz76Uk5cFypBOzbnk35
UkWrzA9KK97cXckfIESgK/k1N4ecwxwG6VQ8mBGqVpPpey+dXpzzzv1iBKN/VY4
ydjgH/LBfdTsVarmmy2vtWBxwrqkFvpdhSGCvRDl/qdO/GIDJi77dmZWkh/ic90
MNk1f38gs1jrCj8lThoar17Uo9y/Q5qJIsoNPyQrJRzqFU9F3FBjiPJF
-----END CERTIFICATE-----
```


2. 驗證對ROA象的創建

   驗證使用RIPEstat數據的對ROA象的成功創建API。請務必根據 Amazon ASNs 16509 和 14618 測試您的地址範圍，以及目前獲得通告地址範圍授權的地址範圍。ASNs

   您可以使用以下命令使用您ASNs的地址範圍檢查來自不同 Amazon 的ROA對象：

```
curl --location --request GET "https://stat.ripe.net/data/rpki-validation/data.json?
resource=ASN&prefix=CIDR
```

   在此範例輸出中，回應的結果`"status": "valid"`為 Amazon ASN 16509。這表示已成功建立位址範圍的ROA物件：


```
{
    "messages": [],
    "see_also": [],
    "version": "0.3",
    "data_call_name": "rpki-validation",
    "data_call_status": "supported",
    "cached": false,
    "data": {
        "validating_roas": [
            {
                "origin": "16509",
                "prefix": "2001:0DB8::/32",
```

```
                "max_length": 48,
                "validity": "valid"
            },
            {
                "origin": "14618",
                "prefix": "2001:0DB8::/32",
                "max_length": 48,
                "validity": "invalid_asn"
            },
            {
                "origin": "64496",
                "prefix": "2001:0DB8::/32",
                "max_length": 48,
                "validity": "invalid_asn"
            }
        ],
        "status": "valid",
        "validator": "routinator",
        "resource": "16509",
        "prefix": "2001:0DB8::/32"
    },
    "query_id": "20230224152430-81e6384e-21ba-4a86-852a-31850787105f",
    "process_time": 58,
    "server_id": "app116",
    "build_version": "live.2023.2.1.142",
    "status": "ok",
    "status_code": 200,
    "time": "2023-02-24T15:24:30.773654"
}
```


的狀態“unknown”表示尚未建立位址範圍的ROA物件。的狀態“invalid_asn”表示未成功建立位址範圍的ROA物件。

在 Amazon 使用您的BYOIP地址範圍 EC2

您可以檢視和使用您在帳戶中佈建的IPv4和IPv6位址範圍。如需詳細資訊，請參閱the section called “建立您的地址範圍”。

IPv4位址範圍

您可以從位址集區建立彈性 IP 位IPv4址，並將其與資 AWS 源 (例如EC2執行個體、NAT閘道和網路負載平衡器) 搭配使用。

若要檢視您在帳戶中佈建之IPv4位址集區的相關資訊，請使用下列 describe-public-ipv4 pool 命令。

```
aws ec2 describe-public-ipv4-pools --region us-east-1
```

要從地址池創建彈性 IP IPv4 地址，請使用分配地址命令。您可以使用 `--public-ipv4-pool` 選項來指定 `describe-byoip-cidrs` 傳回的位址集區 ID。或者，您可以使用 `--address` 選項來指定您佈建的位址範圍。

IPv6位址範圍

若要檢視您在帳戶中佈建之IPv6位址集區的相關資訊，請使用下列 describe- ipv6 池命令。

```
aws ec2 describe-ipv6-pools --region us-east-1
```

若要IPv6CIDR從您的IPv6位址集區建立VPC並指定，請使用下列 create- vpc 指令。要讓 Amazon IPv6 CIDR 從您的IPv6地址池中選擇，請省略該`--ipv6-cidr-block`選項。

```
aws ec2 create-vpc --cidr-block 10.0.0.0/16 --ipv6-cidr-block ipv6-cidr --ipv6-
pool pool-id --region us-east-1
```

若要將IPv6位址集IPv6CIDR區中的區塊與 a 建立關聯VPC，請使用下列associate-vpc-cidr-block指令。要讓 Amazon IPv6 CIDR 從您的IPv6地址池中選擇，請省略該`--ipv6-cidr-block`選項。

```
aws ec2 associate-vpc-cidr-block --vpc-id vpc-123456789abc123ab --ipv6-cidr-block ipv6-
cidr --ipv6-pool pool-id --region us-east-1
```

若要檢視與相關聯的VPCsIPv6位址集區資訊，請使用 describe- vpcs 指令。若要從特定IPv6位址集區檢視關聯IPv6CIDR區塊的資訊，請使用下列 get-associated-ipv6-pool-cidrs 命令。

```
aws ec2 get-associated-ipv6-pool-cidrs --pool-id pool-id --region us-east-1
```

如果您取消與您的IPv6CIDR區塊的關聯VPC，它就會釋放回您的IPv6位址集區。

彈性 IP 地址

彈性 IP 位址是專為動態雲端運算所設計的靜態位IPv4址。彈性 IP 位址會分配給您的 AWS 帳戶，在您釋放之前是您的。透過彈性 IP 地址，您可以快速地將地址重新映射至帳戶中的另一個執行個體，以遮

罩執行個體或軟體的故障。或者，您也可以在網域的DNS記錄中指定彈性 IP 位址，讓網域指向您的執行個體。如需詳細資訊，請參閱您網域註冊商的說明文件。

彈性 IP 地址是可從互聯網訪問的公共IPv4地址。如果您需要連線至沒有公開位址的執行個體，可以將彈性 IP 位IPv4址與執行個體建立關聯，以啟用與網際網路的通訊。

目錄

- 彈性 IP 地址定價
- 彈性 IP 地址基本概念
- 彈性 IP 地址配額
- 將彈性 IP 地址與執行個體建立關聯
- 在之間傳輸彈性 IP 地址 AWS 帳戶
- 釋出彈性 IP 地址
- 在 Amazon 上為電子郵件創建反向DNS記錄 EC2

彈性 IP 地址定價

AWS 所有公共IPv4地址的費用，包括與執行中執行個體和彈性 IP IPv4 地址相關聯的公共地址。如需詳細資訊，請參閱 Amazon VPC 定價頁面上的公開IPv4位址索引標籤。

彈性 IP 地址基本概念

以下是彈性 IP 地址的基本特性：

- 彈性 IP 地址是靜態的地址；不會隨著時間而變更。
- 彈性 IP 地址僅適用於特定區域，無法移至不同區域。
- 彈性 IP 地址來自亞馬遜的地IPv4址池，或者來自您帶到您的自定義IPv4地址池 AWS 帳戶。我們不支援的彈性 IP 位址IPv6。
- 若要使用彈性 IP 地址，您可以先將一個地址配置給帳戶，再將其與您的執行個體或網路介面建立關聯。
- 當您將彈性 IP 地址與執行個體建立關聯時，它也會與執行個體的主要網路介面相關聯。當您將彈性 IP 地址與連接到執行個體的網路介面建立關聯時，它也會與執行個體相關聯。
- 當您將彈性 IP 地址與執行個體或其主要網路界面建立關聯時，如果該執行個體已有相關聯的公共地IPv4址，則該公共IPv4地址會釋放回 Amazon 的公有地IPv4址集區，而彈性 IP 地址會改為與執行個

體相關聯。您無法重複使用先前與執行個體相關聯的公用位IPv4址，也無法將該公用位IPv4址轉換為彈性 IP 位址。如需詳細資訊，請參閱公共IPv4地址。

- 您可以取消彈性 IP 地址與資源的關聯，並將其與不同的資源重新關聯。若要避免非預期的行為，請確保在變更前，已關閉與現有關聯中命名之資源的所有啟用連線。將彈性 IP 地址關聯至不同資源之後，您可以重新開啟與新關聯資源的連線。
- 在您明確將其釋放之前，取消關聯的彈性 IP 地址會維持配置給您的帳戶。無論帳戶中的所有彈性 IP 地址是否與執行個體相關聯或取消關聯，都需要向您收取費用。如需詳細資訊，請參閱 Amazon VPC 定價頁面上的公開IPv4位址索引標籤。
- 當您將彈性 IP 地址與之前具有公共地IPv4址的執行個體相關聯時，執行個體的公共DNS主機名稱會變更為符合彈性 IP 地址。
- 我們會將公開DNS主機名稱解析為執行個體網路外部執行個體的公用位IPv4址或彈性 IP 位IPv4址，以及從執行個體網路內部解析為執行個體的私有位址。
- 當您從帶入 AWS 帳戶的 IP 位址集區配置彈性 IP 位址時，該位址不會計入您的彈性 IP 位址限制。如需詳細資訊，請參閱彈性 IP 地址配額。
- 當您配置彈性 IP 地址時，可以將彈性 IP 地址與網路邊界群組建立關聯。這是我們通告CIDR塊的位置。設定網路邊界群組會將CIDR區塊限制在此群組。如果您未指定網路邊界群組，我們會設定包含區域中所有可用區域的邊界群組 (例如，`us-west-2`)。
- 彈性 IP 地址只能用於特定網路邊界群組。

彈性 IP 地址配額

根據預設，每個區域都 AWS 帳戶 有五 (5) 個彈性 IP 位址的配額，因為 public (IPv4) 網際網路位址是稀缺的公用資源。我們強烈建議您使用彈性 IP 位址，主要是因為它們能夠在執行個體失敗時將位址重新對應至其他執行個體，以及將DNS主機名稱用於所有其他節點間通訊。

如果您認為自己的架構需要更多的彈性 IP 地址，您可以直接從 Service Quotas 主控台請求提高配額。若要請求增加配額，請選擇在帳戶層級請求增加。如需詳細資訊，請參閱Amazon EC2 服務配額。

將彈性 IP 地址與執行個體建立關聯

配置彈性 IP 位址後，您可以將其與 AWS 資源 (例如EC2執行個體、NAT閘道或 Network Load Balancer) 建立關聯。若要稍後將彈性 IP 位址與其他 AWS 資源建立關聯，您可以將其與目前資源取消關聯，然後將其與新資源建立關聯。

完成下列工作，將彈性 IP 位址與EC2執行個體建立關聯。

任務

- 配置彈性 IP 地址
- 建立彈性 IP 地址的關聯
- 取消彈性 IP 地址的關聯

配置彈性 IP 地址

您可以從 Amazon 的公IPv4有地址集區，或從您帶到 AWS 帳戶的自訂 IP 地址集區分配彈性 IP 地址。如需將自己的 IP 位址範圍帶入 AWS 帳戶的詳細資訊，請參閱將您自己的 IP 地址（BYOIP）帶到 Amazon EC2。

Console

配置彈性 IP 地址

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在功能窗格中，選擇 [網路與安全性]、[彈性] IPs。
3. 選擇 Allocate Elastic IP address (配置彈性 IP 地址)。
4. (選擇性) 當您配置彈性 IP 位址 (EIP) 時，您可以選擇要在其中配置EIP. 網路邊界群組是可用區域 (AZs)、Local Zones 或 Wavelength 區域的集合，用於 AWS 通告公用 IP 位址。Local Zones 和 Wavelength 區域可能具有不同的網絡邊界群組，以確保 AWS 網絡與訪問這些區域中資源的客戶之間的最小延遲或物理距離。AZs

> **Important**
>
> 您必須EIP在相同的網路邊界群組中配置作為將與相關聯的 AWS 資源EIP。EIP在一個網路邊界群組中只能在該網路邊界群組中的區域中進行通告，而不能在由其他網路邊界群組表示的任何其他區域中進行通告。

如果您啟用了局部區域或 Wavelength 區域（如需詳細資訊，請參閱啟用 Local Z ones 或啟用 Wavelength 區域），您可以為AZs、局部區域或 Wavelength 區域選擇網路邊界群組。請謹慎選擇網路邊界群組，因為與其關聯的 AWS 資源必須位於相同的網路邊界群組中。EIP您可以使用主EC2控台來檢視可用區域、本機區域或 Wavelength 區域所在的網路邊界群組。一般而言，區域中的所有可用區域屬於相同的網路邊界群組，而 Local Zones 或 Wavelength Zones 則屬於其各自的網路邊界群組。

如果您沒有啟用「Local Zones」或「Wavelength 區域」EIP，則當您分配時，代表該區域所有的AZs網路邊界群組 (例如`us-west-2`) 會為您預先定義，且您無法變更它。這表示您分配給此網路邊界群組的資訊將會在您所AZs在地區的所有廣告中。EIP

5. 針對 [公用IPv4位址集區]，選擇下列其中一項：
   - 亞馬遜的地IPv4址池 — 如果你想從亞馬遜的IPv4地址池分配一個地IPv4址。
   - 您帶到 AWS 帳戶的公用位IPv4址 — 如果您想要從您帶入帳 AWS 戶的 IP 位址集區配置位址。IPv4如果您沒有任何 IP 地址集區，則會停用此選項。
   - 客戶擁有的地IPv4址集區 — 如果您想要從內部部署網路建立的集區配置IPv4地址，以搭配 AWS Outpost 使用。如果您沒有 AWS 前哨，則會停用此選項。
6. (選擇性) 若要新增標籤，請選擇「新增標籤」，然後輸入標籤關鍵字和標籤值。

AWS CLI

配置彈性 IP 地址

使用 allocate-address AWS CLI 命令。

```
aws ec2 allocate-address
```

PowerShell

配置彈性 IP 地址

使用指New-EC2Address AWS Tools for Windows PowerShell 令。

```
New-EC2Address -Domain Vpc
```

建立彈性 IP 地址的關聯

如果您想要將彈性 IP 地址與執行個體建立關聯來啟用與網際網路通訊的功能，您也必須確定執行個體位於公有子網中。如需詳細資訊，請參閱 Amazon 使用VPC者指南中的使用網際網路閘道啟用網際網路存取。

Console

建立彈性 IP 地址與執行個體的關聯

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 [彈性] IPs。
3. 選取要關聯的彈性 IP 地址並依序選取 Actions (動作)、Associate Elastic IP address (關聯彈性 IP 地址)。
4. 對於 Resource type (資源類型)，選擇 Instance (執行個體)。
5. 例如，選擇要與彈性 IP 地址建立關聯的執行個體。您也可以輸入文字來搜尋特定執行個體。
6. (選用) 對於 Private IP address (私人 IP 地址)，指定要與彈性 IP 地址建立關聯的私人 IP 地址。
7. 選擇 Associate (關聯)。

建立彈性 IP 地址與網路介面的關聯。

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 [彈性] IPs。
3. 選取要關聯的彈性 IP 地址並依序選取 Actions (動作)、Associate Elastic IP address (關聯彈性 IP 地址)。
4. 對於資源類型，選擇 Network interface (網路介面)。
5. 在 Network interface (網路介面) 中，選擇要與彈性 IP 地址建立關聯的網路介面。您也可以輸入文字來搜尋特定網路介面。
6. (選用) 對於 Private IP address (私人 IP 地址)，指定要與彈性 IP 地址建立關聯的私人 IP 地址。
7. 選擇 Associate (關聯)。

AWS CLI

建立彈性 IP 地址的關聯

使用關聯位址 AWS CLI 指令。

```
aws ec2 associate-address --instance-id i-0b263919b6498b123 --allocation-
id eipalloc-64d5890a
```

PowerShell

建立彈性 IP 地址的關聯

使用指Register-EC2Address AWS Tools for Windows PowerShell 令。

```
Register-EC2Address -InstanceId i-0b263919b6498b123 -AllocationId eipalloc-64d5890a
```

取消彈性 IP 地址的關聯

您可以隨時解除彈性 IP 地址與執行個體或網路介面的關聯。解除與彈性 IP 地址的關聯後，您可以將它與其他資源重新建立關聯。

Console

解除關聯並重新關聯彈性 IP 地址

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 [彈性] IPs。
3. 選取要解除彈性 IP 地址的關聯，請依序選取 Actions (動作)、Disassociate Elastic IP address (解除彈性 IP 地址的關聯)。
4. 選擇 Disassociate (取消關聯)。

AWS CLI

取消與彈性 IP 地址的關聯

使用取消關聯 AWS CLI 地址命令。

```
aws ec2 disassociate-address --association-id eipassoc-12345678
```

PowerShell

取消與彈性 IP 地址的關聯

使用指Unregister-EC2Address AWS Tools for Windows PowerShell 令。

```
Unregister-EC2Address -AssociationId eipassoc-12345678
```

在之間傳輸彈性 IP 地址 AWS 帳戶

您可以將彈性 IP 地址從一個地址轉移 AWS 帳戶 到另一個地址。在以下情況下，這可能會有所幫助：

- 災難復原 — 在緊急事件期間，快速重新對應 IP 位址，以供公用網際網路工作負載使用。
- 組織重組 — 快速將工作負載從一個移 AWS 帳戶 至另一個工作負載。位址傳輸可避免等待安全性群組和網路允許新的彈性 IP 位址ACLs。
- 集中式安全性管理 — 使用集中式 AWS 安全性帳戶來追蹤和傳輸已針對安全性合規審查的彈性 IP 位址。

定價

轉移彈性 IP 地址無需付費。

任務

- 啟用彈性 IP 地址轉移
- 接受轉移後的彈性 IP 地址
- 停用彈性 IP 地址轉移

啟用彈性 IP 地址轉移

本節旨在說明如何接受轉移後的彈性 IP 地址。請注意下列與啟用彈性 IP 地址轉移的相關限制：

- 您可以將彈性 IP 地址從任何 AWS 帳戶 （源帳戶）轉移到同一 AWS 區域中的任何其他 AWS 帳戶 （轉移帳戶）。
- 當您轉移彈性 IP 地址時， AWS 帳戶之間會發生兩步驟交握。來源帳戶開始轉移時，轉移帳戶有七天時間可以接受彈性 IP 地址轉移。在這七天內，來源帳戶可以檢視擱置中的移轉作業 (例如在 AWS 主控台中或使用describe-address-transfers AWS CLI 命令)。七天後轉移將到期，屆時彈性 IP 地址的擁有權會回到來源帳戶。
- 在接受傳輸後 14 天內，來源帳戶 (例如在 AWS 主控台或使用describe-address-transfers AWS CLI 命令) 可以看到已接受的轉移作業。
- AWS 不會通知轉移帳戶有關待處理的彈性 IP 地址轉移請求。來源帳戶的擁有者必須通知轉移帳戶的擁有者，有必須接受的彈性 IP 地址轉移的請求。
- 轉移完成時，會重設與要轉移之彈性 IP 地址相關的所有標籤。
- 您無法轉移從您 AWS 帳戶 帶到自己的公用位IPv4址集區配置的彈性 IP 位址，通常稱為「使用您自己的 IP (BYOIP) 位址集區」。

- 如果您嘗試轉移具有相關聯的反向DNS記錄的彈性 IP 位址，您可以開始移轉程序，但移轉帳戶將無法接受移轉，直到移除相關聯的DNS記錄。
- 如果您已啟用並設定 AWS Outposts，則可能已從客戶擁有的 IP 位址集區 (CoIP) 配置彈性 IP 位址。您無法轉移從 CoIP 配置的彈性 IP 地址。但是，您可以使 AWS RAM 用與另一個帳戶共享 CoIP。如需詳細資訊，請參閱 AWS Outposts 使用者指南中的客戶擁有的 IP 地址。
- 您可以使用 Amazon VPC IPAM 追蹤彈性 IP 地址傳輸到組織中帳戶的來源 AWS Organizations。如需詳細資訊，請參閱 View IP address history (檢視 IP 地址歷程記錄)。如果將彈性 IP 位址傳輸到組織 AWS 帳戶 外部，彈性 IP 位址的IPAM稽核歷程記錄將會遺失。

這些步驟必須由來源帳戶完成。

Console

啟用彈性 IP 地址轉移

1. 確定您使用的是來源 AWS 帳戶。
2. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
3. 在導覽窗格中，選擇 [彈性] IPs。
4. 選取要啟用轉移的一或多個彈性 IP 地址，然後選擇 Actions (動作)、Enable transfer (啟用轉移)。
5. 如果要轉移多個彈性 IP 地址，您會看到 Transfer type (轉移類型) 選項。請選擇下列其中一個選項：
   - 如果要將彈性 IP 地址傳輸到單個帳戶，請選擇單個 AWS 帳戶。
   - 如果要將彈性 IP 地址傳輸到多個帳戶，請選擇「多個 AWS 帳戶」。
6. 在「轉移帳戶 ID」下，輸入您要將彈性 IP 地址轉移到IDs的 AWS 帳戶。
7. 在文字方塊中輸入 **enable**，以便確認轉移。
8. 選擇提交。
9. 若要接受轉移，請參閱 接受轉移後的彈性 IP 地址。若要停用轉移，請參閱 停用彈性 IP 地址轉移。

AWS CLI

若要啟用彈性 IP 地址轉移

使用指enable-address-transfer令。

```
aws ec2 enable-address-transfer \
    --allocation-id eipalloc-09ad461b0d03f6aaf \
    --transfer-account-id 123456789012
```

PowerShell

若要啟用彈性 IP 地址轉移

使用指Enable-EC2AddressTransfer令。

```
Enable-EC2AddressTransfer -AllocationId eipalloc-09ad461b0d03f6aaf -
TransferAccountId 123456789012
```

接受轉移後的彈性 IP 地址

本節旨在說明如何接受轉移後的彈性 IP 地址。

當您轉移彈性 IP 地址時， AWS 帳戶之間會發生兩步驟交握。來源帳戶開始轉移時，轉移帳戶有七天時間可以接受彈性 IP 地址轉移。在這七天內，來源帳戶可以檢視擱置中的移轉作業 (例如在 AWS 主控台中或使用describe-address-transfers AWS CLI 命令)。七天後轉移將到期，屆時彈性 IP 地址的擁有權會回到來源帳戶。

接受轉移時，請注意下列可能發生的異常情況，以及如何解決這些異常問題：

- AddressLimitExceeded: 如果您的移轉帳戶已超過彈性 IP 位址配額，則來源帳戶可以啟用彈性 IP 位址傳輸，但是當移轉帳戶嘗試接受移轉時，就會發生此例外狀況。根據預設，每個區域的所有 AWS 帳戶限制為 5 個彈性 IP 位址。如需提高限制的指示，請參閱 彈性 IP 地址配額。
- InvalidTransfer。 AddressCustomPtrSet：如果您或組織中的某人已設定您嘗試轉移的彈性 IP 位址以使用反向DNS查詢，則來源帳戶可以啟用彈性 IP 位址的移轉，但是當移轉帳戶嘗試接受移轉時，就會發生此例外狀況。若要解決這個問題，來源帳戶必須移除彈性 IP 位址的DNS記錄。如需詳細資訊，請參閱在 Amazon 上為電子郵件創建反向DNS記錄 EC2。
- InvalidTransfer。 AddressAssociated：如果彈性 IP 位址與ENI或EC2執行個體相關聯，則來源帳戶可以啟用彈性 IP 位址的傳輸，但是當移轉帳戶嘗試接受移轉時，就會發生此例外狀況。若要解決這個問題，來源帳戶必須取消與該彈性 IP 地址的關聯。如需詳細資訊，請參閱取消彈性 IP 地址的關聯。

如有任何其他異常情況，請聯絡 AWS Support。

這些步驟必須由轉移帳戶完成。

Console

接受彈性 IP 地址轉移

1. 確保您使用的是轉移帳戶。
2. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
3. 在導覽窗格中，選擇 [彈性] IPs。
4. 選擇 Actions (動作)、Accept transfer (接受轉移)。
5. 當您接受轉移時，與該彈性 IP 地址有關的所有標籤都不會隨著該彈性 IP 地址轉移。如果您要為接受的彈性 IP 地址定義 Name (名稱) 標籤，請選取 Create a tag with a key of 'Name' and a value that you specify (建立索引鍵為 'Name' 並具有您指定值的標籤)。
6. 輸入您要轉移的彈性 IP 地址。
7. 如果您要接受多個轉移的彈性 IP 地址，請選擇 Add address (新增位址)，以輸入其他彈性 IP 地址。
8. 選擇提交。

AWS CLI

若要接受彈性 IP 地址轉移

使用指accept-address-transfer令。

```
aws ec2 accept-address-transfer --address 100.21.184.216
```

PowerShell

若要接受彈性 IP 地址轉移

使用指Approve-EC2AddressTransfer令。

```
Approve-EC2AddressTransfer -Address 100.21.184.216
```

停用彈性 IP 地址轉移

本節旨在說明如何在啟用轉移後停用彈性 IP 轉移。

這些步驟必須由啟用轉移的來源帳戶完成。

Console

停用彈性 IP 地址轉移

1. 確保您正在使用源 AWS 帳戶。
2. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
3. 在導覽窗格中，選擇 [彈性] IPs。
4. 在 Elastic 的資源清單中IPs，確定您已啟用顯示「傳輸」狀態欄的屬性。
5. 選取 Transfer status (轉移狀態) 為 Pending (待定) 的一或多個彈性 IP 地址，然後選擇 Actions (動作)、Disable transfer (停用轉移)。
6. 在文字方塊中輸入 **disable**，以便進行確認。
7. 選擇提交。

AWS CLI

若要停用彈性 IP 地址轉移

使用指disable-address-transfer令。

```
aws ec2 disable-address-transfer --allocation-id eipalloc-09ad461b0d03f6aaf
```

PowerShell

若要停用彈性 IP 地址轉移

使用指Disable-EC2AddressTransfer令。

```
Disable-EC2AddressTransfer -AllocationId eipalloc-09ad461b0d03f6aaf
```

釋出彈性 IP 地址

如果您不再需要彈性 IP 位址，我們建議您將其釋出。要釋放的彈性 IP 位址目前不能與 AWS 資源相關聯。

Console

釋出彈性 IP 地址

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 [彈性] IPs。
3. 選取要釋出的彈性 IP 地址，然後依序選取 Actions (動作)、Release Elastic IP addresses (釋出彈性 IP 地址)。
4. 選擇 Release (釋出)。

AWS CLI

釋出彈性 IP 地址

使用發布地址命令 AWS CLI 。

```
aws ec2 release-address --allocation-id eipalloc-64d5890a
```

PowerShell

釋出彈性 IP 地址

使用指Remove-EC2Address AWS Tools for Windows PowerShell 令。

```
Remove-EC2Address -AllocationId eipalloc-64d5890a
```

釋放彈性 IP 地址後，您可能可以恢復。適用的規定如下：

- 如果彈性 IP 地址已分配給其他 AWS 帳戶，或者它會導致您超出彈性 IP 地址限制，則無法恢復該地址。
- 您無法復原與彈性 IP 位址相關聯的標籤。

AWS CLI

復原彈性 IP 地址

使用分配地址 AWS CLI 命令，並使用`--address`參數指定 IP 地址，如下所示。

```
aws ec2 allocate-address --domain vpc --address 203.0.113.3
```

PowerShell

復原彈性 IP 地址

使用命New-EC2Address AWS Tools for Windows PowerShell 令並使用`-Address`參數指定 IP 位址，如下所示。

```
New-EC2Address -Address 203.0.113.3 -Domain vpc -Region us-east-1
```

在 Amazon 上為電子郵件創建反向DNS記錄 EC2

如果您打算從EC2執行個體傳送電子郵件給第三方，建議您佈建一或多個彈性 IP 位址，並將靜態反向DNS記錄指派給用於傳送電子郵件的彈性 IP 位址。這有助於避免某些反垃圾郵件組織將您的電子郵件標記為垃圾郵件。 AWS ISPs與互聯網反垃圾郵件組織合作，以減少從這些地址發送的電子郵件被標記為垃圾郵件的機會。

考量事項

- 在建立反向DNS記錄之前，您必須設定指向彈性 IP 位址的對應正向記DNS錄 (記錄類型 A)。
- 如果反向DNS記錄與彈性 IP 地址相關聯，則彈性 IP 地址將鎖定到您的帳戶，並且在刪除記錄之前無法從您的帳戶釋放。
- 如果您聯繫了反 AWS Support 向設置彈性 IP 地址，則可以刪除反向 IP 地址DNS，但是由DNS於彈性 IP 地址已鎖定，因此無法釋放彈性 IP 地址 AWS Support。若要解除彈性 IP 地址的鎖定，請聯絡 AWS Support。解鎖彈性 IP 地址後，您可以釋放它。
- [AWS GovCloud (US) Region] 您無法創建反向DNS記錄。 AWS 必須為您分配靜態反向DNS記錄。打開請求刪除反向DNS和電子郵件發送限制，並向我們提供您的彈性 IP 地址和反向DNS記錄。

創建一個反向DNS記錄

您可以按如下方式為彈性 IP 地址創建反向DNS記錄。

Console

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 [彈性] IPs。

3. 選取彈性 IP 位址，然後選擇 [動作]、[更新反向] DNS。
4. 在反向DNS網域名稱中，輸入網域名稱。
5. 輸入 **update** 以確認。
6. 選擇 Update (更新)。

AWS CLI

使用中的modify-address-attribute指令 AWS CLI，如下列範例所示。

```
aws ec2 modify-address-attribute --allocation-id eipalloc-abcdef01234567890  --
domain-name example.com
```

以下是示例輸出

```
{
    "Addresses": [
        {
            "PublicIp": "192.0.2.0",
            "AllocationId": "eipalloc-abcdef01234567890",
            "PtrRecord": "example.net.",
            "PtrRecordUpdate": {
                "Value": "example.com.",
                "Status": "PENDING"
        }
    ]
}
```

移除反向DNS記錄

您可以按如下方式從彈性 IP 地址中刪除反向DNS記錄。

如果您收到下列錯誤，您可以提交要求以移除電子郵件傳送限制以 AWS Support 尋求協助。

```
The address cannot be released because it is locked to your account.
```

Console

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。

2. 在導覽窗格中，選擇 [彈性] IPs。
3. 選取彈性 IP 位址，然後選擇 [動作]、[更新反向] DNS。
4. 對於反向DNS網域名稱，請清除網域名稱。
5. 輸入 **update** 以確認。
6. 選擇 Update (更新)。

AWS CLI

使用中的reset-address-attribute指令 AWS CLI，如下列範例所示。

```
aws ec2 reset-address-attribute --allocation-id eipalloc-abcdef01234567890 --
attribute domain-name
```

以下是示例輸出

```
{
    "Addresses": [
        {
            "PublicIp": "192.0.2.0",
            "AllocationId": "eipalloc-abcdef01234567890",
            "PtrRecord": "example.com.",
            "PtrRecordUpdate": {
                "Value": "example.net.",
                "Status": "PENDING"
        }
    ]
}
```

彈性網絡介面

elastic network interface 是代表虛擬網路卡的邏輯網路元件。VPC您可以建立和設定網路介面，並將它們附加到您在相同可用區域中啟動的執行個體。網路介面的屬性在它連接到執行個體或與執行個體分離，以及重新連接到另一個執行個體時，會一直跟著它。當您將網路介面從一個執行個體移至另一個執行個體時，網路流量會從原始執行個體重新導向至新執行個體。

請注意，此 AWS 資源在 AWS Management Console 和 Amazon 被稱為網絡接口EC2API。因此，我們在本文件中使用「網路介面」而非「彈性網絡介面」。本文件中的「網路介面」一詞一律表示「彈性網絡介面」。

網路介面屬性

網路界面可包含下列屬性：

- 子網路IPv4位IPv4址範圍中的主要私人位址
- 子網路IPv6位IPv6址範圍中的主要位址
- 子網路位IPv4址範圍中的次要私人IPv4位址
- 每個私人位址都有一個彈性 IP 位IPv4址 (IPv4)
- 一個公共IPv4地址
- 次要IPv6位址
- 安全群組
- 一個MAC地址
- 一個來源/目標檢查標記
- 一項描述

監控流量

您可以在網路介面上啟用VPC流量記錄，以擷取進出網路介面之流量的相關資訊。建立流程日誌後，您可以在 Amazon CloudWatch 日誌中檢視和擷取其資料。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的VPC流程日誌。

目錄

- 網路介面概念
- 網路卡
- 每個網路介面的 IP 位址上限
- 為您的 Amazon EC2 執行個體建立網路界面
- 管理網路介面的 IP 位址
- 修改網路介面屬性
- 適用於 Amazon EC2 執行個體的多個網路界面
- 申請者受管的網路介面
- Amazon EC2 網路界面的前綴委派
- 刪除網路介面

網路介面概念

以下是開始使用網路介面時需要瞭解的重要概念。

主要網路介面

每個執行個體都有預設的網路介面，稱為主要網路介面。您無法從執行個體中分離主要網路介面。

次要網路介面

您可以建立次要網路介面，並將其連接至執行個體。網路介面的最大數目會因執行個體類型而有所不同。如需詳細資訊，請參閱每個網路介面的 IP 位址上限。

IPv4網路介面的位址

當您將EC2執行個體啟動到IPv4唯一堆疊或雙重堆疊子網路時，執行個體會從子網路的位IPv4址範圍接收主要私有 IP 位址。您也可以指定其他私人IPv4位址，稱為次要私人IPv4位址。與主要私有 IP 地址不同，輔助私有 IP 地址可從原執行個體重新指派至別的執行個體。

網路介面的公用IPv4位址

所有子網路都有一個可修改的屬性，可決定在該子網路中建立的網路介面 (以及因此啟動到該子網路的執行個體) 是否要指派公IPv4用位址。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的子網路設定。當您啟動執行個體時，IP 位址會指派給主要網路介面。如果您在啟動執行個體時將現有的網路介面指定為主要網路介面，則公用IPv4位址將由此網路介面決定。

當您建立網路介面時，它會從子網路繼承公用位IPv4址屬性。如果您稍後修改子網路的公用位IPv4址屬性，網路介面會保留建立時生效的設定。

IPv6網路介面的位址

如果您將IPv6CIDR區塊與VPC和子網路相關聯，則可以將子網路範圍中的IPv6位址指派給網路介面。每個IPv6位址都可以指派給一個網路介面。

所有子網路都有一個可修改的屬性，可決定在該子網路中建立的網路介面 (以及因此啟動到該子網路的執行個體) 是否會自動指派子網路範圍內的IPv6位址。當您啟動執行個體時，該IPv6位址會指派給主要網路介面。

網路介面的彈性 IP 位址

您可以將彈性 IP 位址與網路介面的其中一個私人位IPv4址建立關聯。您可以將一個彈性 IP 位址與每個私人位IPv4址建立關聯。如果您取消彈性 IP 位址與網路介面的關聯，您可以將其釋放，或將其與其他執行個體建立關聯。

終止行為

您可設定連接至執行個體之網路介面的終止行為。您可指定網路介面是否應該在您終止它所連接的執行個體時自動刪除。

來源/目標檢查

您可以啟用或停用來源/目的地檢查，這可確保執行個體是其接收之任何流量的來源或目的地。來源/目的地檢查預設為啟用。如果執行個體執行網路位置轉譯、路由或防火牆等服務，則必須停用來源/目標檢查。

申請者受管的網路介面

這些網路介面由建立和管理， AWS 服務 以便您使用某些資源和服務。您無法自行管理這些網路介面。如需詳細資訊，請參閱申請者受管的網路介面。

前綴委派

前置詞是保留的專用IPv4或IPv6CIDR範圍，您可以為自動或手動指派給與執行個體相關聯的網路介面配置。透過使用委派字首，您就可以將 IP 地址範圍指派為單一字首，以更快速地啟動服務。

網路卡

大多數執行個體類型支援一張網路卡。支援多張網路卡的執行個體類型可提供更高的網路效能，包括 100 Gbps 以上的頻寬功能和改善的封包速率效能。將網路介面連接至支援多張網路卡的執行個體時，您可以為網路介面選取網路卡。主要網路介面必須指派給網路卡索引 0。

如果您在啟動支援多張網路卡的執行個體時啟用彈性網狀架構介面卡 (EFA)，則所有網路卡皆可使用。每張網路卡最多可指EFA派一個。EFA計為網路介面。

下列執行個體類型支援多張網路卡。

執行個體類型	網路卡數量
c6in.32xlarge	2
c6in.metal	2
dl1.24xlarge	4
g6e.24xlarge	2

執行個體類型	網路卡數量
g6e.48xlarge	4
hpc6id.32xlarge	2
hpc7a.12xlarge	2
hpc7a.24xlarge	2
hpc7a.48xlarge	2
hpc7a.96xlarge	2
m6idn.32xlarge	2
m6idn.metal	2
m6in.32xlarge	2
m6in.metal	2
p4d.24xlarge	4
p4de.24xlarge	4
p5.48xlarge	32
r6idn.32xlarge	2
r6idn.metal	2
r6in.32xlarge	2
r6in.metal	2
trn1.32xlarge	8
trn1n.32xlarge	16
u7in-16tb.224xlarge	2

執行個體類型	網路卡數量
u7in-24tb.224xlarge	2
u7in-32tb.224xlarge	2

每個網路介面的 IP 位址上限

每種執行個體類型都支援網路介面數目上限、每個網路介面的最大私人IPv4位址數目，以及每個網路介面的IPv6位址數目上限。IPv6位址的限制與每個網路介面的私人IPv4位址限制不同。並非所有執行個體類型都支援IPv6定址

可用的網路介面

Amazon EC2 執行個體類型指南提供每個執行個體類型可用網路界面的相關資訊。如需詳細資訊，請參閱下列內容：

- 網路規格 — 一般用途
- 網路規格 — 運算最佳化
- 網路規格 — 記憶體最佳化
- 網路規格 — 儲存最佳化
- 網路規格 — 加速運算
- 網路規格 — 高效能運算
- 網路規格 — 前一代

若要使用擷取網路介面資訊 AWS CLI

您可以使用describe-instance-types AWS CLI 指令來顯示執行個體類型的相關資訊，例如其支援的網路介面和每個介面的 IP 位址。下列範例會顯示所有 C5 執行個體的此資訊。

```
aws ec2 describe-instance-types \
    --filters "Name=instance-type,Values=c5.*" \
    --query "InstanceTypes[].{ \
        Type: InstanceType, \
        MaxENI: NetworkInfo.MaximumNetworkInterfaces, \
        IPv4addr: NetworkInfo.Ipv4AddressesPerInterface}" \
```

```
    --output table
```

下列為範例輸出。

```
----------------------------------------
|        DescribeInstanceTypes         |
+----------+----------+---------------+
| IPv4addr | MaxENI   |     Type      |
+----------+----------+---------------+
|  30      |  8       |  c5.4xlarge   |
|  50      |  15      |  c5.24xlarge  |
|  15      |  4       |  c5.xlarge    |
|  30      |  8       |  c5.12xlarge  |
|  10      |  3       |  c5.large     |
|  15      |  4       |  c5.2xlarge   |
|  50      |  15      |  c5.metal     |
|  30      |  8       |  c5.9xlarge   |
|  50      |  15      |  c5.18xlarge  |
+----------+----------+---------------+
```

若要使用擷取網路介面資訊 AWS Tools for PowerShell

您可以使用Get-EC2InstanceType PowerShell 指令來顯示執行個體類型的相關資訊，例如其支援的網路介面和每個介面的 IP 位址。下列範例會顯示所有 C5 執行個體的此資訊。

```
Get-EC2InstanceType -Filter @{Name = "instance-type"; Values = "c5.*" } | `
Select-Object `
    @{Name = 'Ipv4AddressesPerInterface'; Expression =
 {($_.Networkinfo.Ipv4AddressesPerInterface)}},
    @{Name = 'MaximumNetworkInterfaces'; Expression =
 {($_.Networkinfo.MaximumNetworkInterfaces)}},
    InstanceType | `
Format-Table -AutoSize
```

下列為範例輸出。

```
Ipv4AddressesPerInterface MaximumNetworkInterfaces InstanceType
------------------------- ------------------------ ------------
                       30                        8 c5.4xlarge
                       15                        4 c5.xlarge
                       30                        8 c5.12xlarge
                       50                       15 c5.24xlarge
```

```
                        30                        8 c5.9xlarge
                        50                       15 c5.metal
                        15                        4 c5.2xlarge
                        10                        3 c5.large
                        50                       15 c5.18xlarge
```

為您的 Amazon EC2 執行個體建立網路界面

當您建立網路介面時，您可以指定為其建立該介面的子網路。建立網路介面後，您無法將其移至其他子網路。建立網路介面後，您可以將其連接至相同可用區域中的執行個體。如果從執行個體中分離次要網路介面，您可以將其附加至不同的例證。

考量事項

- 您可在執行個體執行時 (熱連接)、停止時 (暖連接) 或正在啟動時 (冷連接) 將網路介面連接到執行個體。
- 您可在執行個體執行時或停止時分離輔助網路介面。但是您不能分離主要網路介面。
- 如果執行個體位於相同的可用區域，VPC但位於不同的子網路中，您可以將次要網路介面從一個執行個體移至另一個執行個體。
- 使用、或啟動執行個體時 CLI APISDK，您可以指定主要網路介面和其他網路介面。
- 啟動具有多個網路界面的 Amazon Linux 或 Windows 伺服器執行個體，可在執行個體的作業系統上自動設定界面、私有IPv4地址和路由表。
- 額外網路介面的暖連接或熱連接可能需要您手動啟動第二個介面、設定私人IPv4位址，以及相應地修改路由表。執行 Amazon Linux 或 Windows Server 的執行個體會自動辨識暖或熱連接並自行設定。
- 您無法將另一個網路介面連接至執行個體 (例如，NIC成組組態)，以增加或增加雙重主目錄執行個體的網路頻寬。
- 如果您將相同子網中兩個或以上的網路介面連接到執行個體，您可能會發生聯網問題，例如非對稱路由。如果可能的話，請改用主要網路介面上的次要私人IPv4位址。
- 對於IPv6僅子網路中的EC2執行個體，如果您連接次要網路介面，次要網路介面的私人DNS主機名稱會解析為主要網路介面的主要IPv6位址。
- Windows 執行個體 — 如果您使用多個網路介面，則必須將網路介面配置為使用靜態路由。

建立網路介面

您可以建立供執行個EC2體使用的網路介面。您選擇的子網路會決定您可以在其中使用網路介面的可用區域。

使用主控台建立網路介面

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選擇 Create network interface (建立網路介面)。
4. (選用) 針對 Description (描述)，輸入描述性名稱。
5. 針對 Subnet (子網)，選取子網。後續步驟中的可用選項會根據您選取的子網路類型而有所不同 (IPv4IPv6僅限、僅限或雙堆疊 (IPv4和IPv6))。
6. 針對私人IPv4位址，請執行下列其中一項作業：
   - 選擇自動指派以允許 Amazon EC2 從子網路選取IPv4位址。
   - 選擇 [自訂]，然後輸入您從子網路選取的IPv4位址。
7. (僅限具有IPv6位址的子網路) 若為位IPv6址，請執行下列其中一項作業：
   - 如果您不想為網路介面指定IPv6位址，請選擇「無」。
   - 選擇自動指派以允許 Amazon EC2 從子網路選取IPv6位址。
   - 選擇 [自訂]，然後輸入您從子網路選取的IPv6位址。
8. (選擇性) 如果您要在雙堆疊或IPv6唯一子網路中建立網路介面，您可以選擇指派主要 IPv6 IP。這會將主要的IPv6全域單點傳送位址 (GUA) 指派給網路介面。指派主要IPv6位址可讓您避免中斷執行個體或ENIs的流量。如果要連接此執行個體的IPv6位址不ENI會變更，請選擇啟用。 AWS 會自動將與執行個體相關聯的IPv6位址指派為主要IPv6位址。ENI將地址啟用為主要IPv6GUA地址後IPv6，就無法停用該地址。當您啟用某個IPv6GUA位址作為主要位址時IPv6，第一個位址IPv6GUA會成為主要IPv6位址，直到執行個體終止或網路介面中斷連線為止。如果您有多個與執行個ENI體相關聯的地址，並且啟用了主要位IPv6址，則與之相關聯的第一個IPv6位IPv6GUA址ENI就會成為主要位IPv6址。
9. (選用) 若要建立 Elastic Fabric Adapter，選擇Elastic Fabric Adapter、Enable (啟用)。
10. (選用) 在進階設定下的閒置連線追蹤逾時，修改預設閒置連線逾時。如需關於這些選項的詳細資訊，請參閱 閒置連線追蹤逾時。
    - TCP已建立逾時：處於已建立狀態的閒置TCP連線逾時 (以秒為單位)。最小值：60 秒。最大值：432000 秒 (5 天)。預設：432000 秒。建議使用：432000 秒以下。
    - UDPtimeout：閒置UDP流量僅在單一方向或單一要求-回應交易中看到流量的逾時 (以秒為單位)。最小值：30 秒。最大值：60 秒。預設：30 秒。
    - UDP串流逾時：分類為已看到多個要求-回應交易之串UDP流的閒置流程逾時 (以秒為單位)。最小值：60 秒。最大值：180 秒 (3 分鐘) 預設：180 秒。
11. 針對 Security groups (安全群組)，選取一個或多個安全群組。

12. (選用) 針對每個標籤，請選擇 Add new tag (新增標籤)，然後輸入標籤金鑰和選用標籤值。
13. 選擇 Create network interface (建立網路介面)。

使用命令列建立網路介面

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱訪問 Amazon EC2。

- create-network-interface (AWS CLI)
- New-EC2NetworkInterface (AWS Tools for Windows PowerShell)

連接網路介面

您可以使用 Amazon EC2 主控台的執行個體或網路界面頁面，將網路界面附加到與網路界面相同的可用區域中的任何執行個體。或者，可以在啟動執行個體時指定現有網路介面。

> **Note**
>
> 您可以使用attach-network-interface AWS CLI 指令將位於另一個 VPC (但位於相同可用區域) 的網路介面連接至執行個體。您無法使用 AWS Management Console.

如果執行個體上的公共IPv4位址已釋出，如果執行個體連接了一個以上的網路介面，則不會收到新的網路介面。如需有關公用IPv4位址行為的詳細資訊，請參閱公共IPv4地址。

Instances page

使用 Instances (執行個體) 頁面將網路介面連接到執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體的核取方塊。
4. 選擇 Actions (動作)、Networking (網路)、Attach network interface (連接網路介面)。
5. 選擇一個VPC。如果您要將次要網路介面連接至執行個體，則網路介面可以位於與您的執行個體VPC相同，也可VPC以位於您擁有的不同位置 (只要網路介面位於與執行個體位於相同可用區域的子網路中)。這可讓您在不同的網路和安全性VPCs設定之間建立多重主目錄執行個體。
6. 選取網路介面。如果執行個體支援多張網路卡，您可以選擇網路卡。
7. 選擇 Attach (連接)。

Network Interfaces page

使用 Network Interfaces (網路介面) 頁面將網路介面連接到執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面的核取方塊。
4. 選擇 Actions (動作)、Attach (連接)。
5. 選擇執行個體。如果執行個體支援多張網路卡，您可以選擇網路卡。
6. 選擇 Attach (連接)。

使用命令列將網路介面連接到執行個體

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱訪問 Amazon EC2。

- attach-network-interface (AWS CLI)
- Add-EC2NetworkInterface (AWS Tools for Windows PowerShell)

卸離網路介面

您可以使用 Amazon EC2 主控台的「EC2執行個體」或「網路界面」頁面，隨時卸離連接到執行個體的次要網路界面。

如果您嘗試從其他服務 (例如 Elastic Load Balancing 負載平衡器、Lambda 函數、或NAT閘道) 中斷連接至資源的網路介面，您會收到錯誤訊息，指出您沒有存取資源的權限。 WorkSpace若要找出哪個服務建立了連接至網路介面的資源，請檢查網路介面的描述。如果您刪除資源，則會刪除其網路介面。

Instances page

使用 Instances (執行個體) 頁面從執行個體分離網路介面

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體的核取方塊。檢查 Networking (網路) 標籤的 Network interfaces (網路介面) 區段，確認網路介面已連接至執行個體作為次要網路介面。
4. 選擇 Actions (動作)、Networking (網路)、Detach network interface (分離網路介面)。
5. 選取網路介面，然後選取 Detach (分離)。

Network Interfaces page

使用 Network Interfaces (網路介面) 頁面從執行個體分離網路介面

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面的核取方塊。檢查 Networking (網路) 標籤的 Instance details (執行個體詳細資訊) 區段，確認網路介面已連接至執行個體作為次要網路介面。
4. 選擇 Actions (動作)、Detach (分離)。
5. 當出現確認提示時，選擇 [分離]。
6. 如果網路介面無法從執行個體分離，請選擇 Force detachment (強制分離)、Enable (啟用)，然後再試一次。我們建議強制分離只作為最後的手段。強制分離可以防止您在相同索引上連接不同的網路介面，直到重新啟動執行個體為止。它也可以防止執行個體中繼資料反映網路介面已分離，直到您重新啟動執行個體為止。

使用命令列分離網路介面

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱訪問 Amazon EC2。

- detach-network-interface (AWS CLI)
- Dismount-EC2NetworkInterface (AWS Tools for Windows PowerShell)

管理網路介面的 IP 位址

您可以管理以下網路介面的 IP 地址：

- 彈性 IP 位址 (每個私人位IPv4址一個)
- IPv4地址
- IPv6地址

管理主控台網路介面的彈性 IP 地址

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面的核取方塊。

4. 若要建立彈性 IP 地址的關聯，請執行下列操作：
   a. 選擇 Actions (動作)、Associate address (建立與地址的關聯)。
   b. 針對 Elastic IP address (彈性 IP 地址)，選取彈性 IP 地址。
   c. 對於私人位IPv4址，請選取要與彈性 IP 位IPv4址建立關聯的私人位址。
   d. (選用) 如果網路介面目前與另一個執行個體或網路介面建立關聯，選擇 Allow the Elastic IP address to be reassociated (允許與彈性 IP 地址重新建立關聯)。
   e. 選擇 Associate (關聯)。
5. 若要取消彈性 IP 地址的關聯，請執行下列操作：
   a. 選擇 Actions (動作)、Disassociate address (取消與地址的關聯)。
   b. 針對 Public IP address (公有 IP 地址)，選取彈性 IP 地址。
   c. 選擇 Disassociate (取消關聯)。

使用主控台管理網路介面的IPv4和IPv6位址

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面。
4. 選擇 Actions (動作)、Manage IP addresses (管理 IP 地址)。
5. 展開網路介面。
6. 對於位IPv4址，請視需要修改 IP 位址。若要指派位IPv4址，請選擇 [指派新 IP 位址]，然後從子網路範圍指定位IPv4址，或讓您 AWS 選擇一個位址。若要取消指派IPv4地址，請選擇地址旁邊的「取消指派」。
7. 若要指派或取消指派公用IPv4位址給網路介面，請選擇「自動指派公用 IP」。您可以為任何網路介面啟用或停用此選項，但僅適用於主要網路介面 (eth0)。
8. 對於位IPv6址，請視需要修改 IP 位址。若要指派位IPv6址，請選擇 [指派新 IP 位址]，然後從子網路範圍指定位IPv6址，或讓您 AWS 選擇一個位址。若要取消指派IPv6地址，請選擇地址旁邊的「取消指派」。
9. (選擇性) 如果您修改雙堆疊或IPv6僅子網路中的網路介面，您可以選擇指派主要 IPv6 IP。指派主要IPv6位址可讓您避免中斷執行個體或ENIs的流量。如果要連接此執行個體的IPv6位址不ENI會變更，請選擇啟用。 AWS 會自動將與執行個體相關聯的IPv6位址指派為主要IPv6位址。ENI將地址啟用為主要IPv6GUA地址後IPv6，就無法停用該地址。當您啟用某個IPv6GUA位址作為主要位

址時IPv6，第一個位址IPv6GUA會成為主要IPv6位址，直到執行個體終止或網路介面中斷連線為止。如果您有多個與執行個ENI體相關聯的地址，並且啟用了主要位IPv6址，則與之相關聯的第一個IPv6位IPv6GUA址ENI就會成為主要位IPv6址。

10. 選擇 Save (儲存)。

管理網路介面的 IP 位址 AWS CLI

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 訪問 Amazon EC2。

- assign-ipv6-addresses
- associate-address
- disassociate-address
- unassign-ipv6-addresses

若要使用 Windows 的工具來管理網路介面的 IP 位址 PowerShell

您可以使用下列其中一個命令。

- Register-EC2Address
- Register-EC2Ipv6 AddressList
- Unregister-EC2Address
- Unregister-EC2Ipv6 AddressList

修改網路介面屬性

您可以變更下列網路介面屬性：

- Description
- 安全群組
- 在終止時刪除
- 來源/目標檢查

如何使用主控台變更網路介面的描述

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。

2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面的核取方塊。
4. 選擇 Actions (動作)、Change description (變更描述)。
5. 針對 Description (描述)，請輸入網路介面的描述。
6. 選擇 Save (儲存)。

如何使用主控台變更網路介面的安全群組

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面的核取方塊。
4. 選擇 Actions (動作)、Change security groups (變更安全群組)。
5. 針對 Associated security groups (關聯的安全群組)，選取要使用的安全群組，然後選取 Save (儲存)。

   必須為相同的群組和網路介面建立安全性群組和網路介面VPC。若要變更 Elastic Load Balancing 等其他服務所擁有的執行個體的安全群組，請透過該服務執行操作。

如何使用主控台變更網路介面的終止行為

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面的核取方塊。
4. 選擇 Actions (動作)、Change termination behavior (變更終止行為)。
5. 視需選取或清除 Delete on termination (終止時刪除)、Enable (啟用)，然後選取 Save (儲存)。

使用主控台變更網路介面的來源/目標檢查

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面的核取方塊。
4. 選擇 Actions (動作)、Change source/dest check (變更來源/目的地檢查)。

5. 視需選取或清除 Source/destination check (來源/目的地檢查)、Enable (啟用)，然後選取 Save (儲存)。

變更閒置連線追蹤逾時的方法如下：

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面的核取方塊。
4. 選擇動作、修改連線逾時。
5. 修改閒置連線追蹤逾時。如需關於這些選項的詳細資訊，請參閱 閒置連線追蹤逾時。
   - TCP已建立逾時：處於已建立狀態的閒置TCP連線逾時 (以秒為單位)。最小值：60 秒。最大值：432000 秒 (5 天)。預設：432000 秒。建議使用：432000 秒以下。
   - UDPtimeout：閒置UDP流量僅在單一方向或單一要求-回應交易中看到流量的逾時 (以秒為單位)。最小值：30 秒。最大值：60 秒。預設：30 秒。
   - UDP串流逾時：分類為已看到多個要求-回應交易之串UDP流的閒置流程逾時 (以秒為單位)。最小值：60 秒。最大值：180 秒 (3 分鐘) 預設值：180 秒。
6. 選擇 Save (儲存)。

如何使用命令列修改網路介面屬性

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱訪問 Amazon EC2。

- modify-network-interface-attribute (AWS CLI)
- Edit-EC2NetworkInterfaceAttribute (AWS Tools for Windows PowerShell)

適用於 Amazon EC2 執行個體的多個網路界面

當您需要下列項目時，將多個網路介面連接至執行個體非常有用：

- 管理網路。
- 網路和安全設備。
- 工作負載位於不同子網路或. VPCs
- 低預算、高可用性的解決方案。

管理網

下列概觀說明使用多個網路介面建立的管理網路。

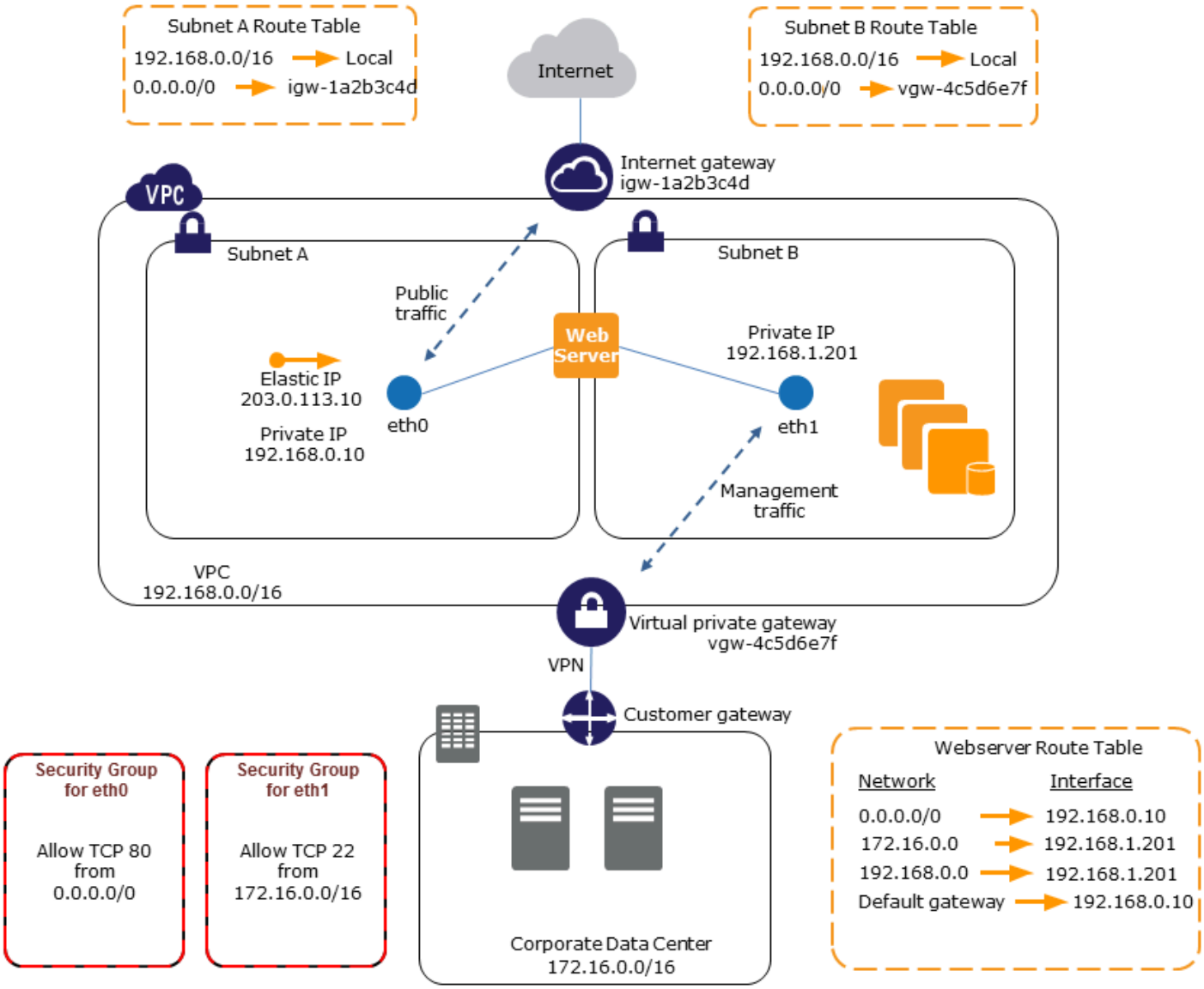
條件

- 執行個體 (eth0) 上的主要網路介面會處理公用流量。
- 執行個體 (eth1) 上的次要網路介面可處理後端管理流量。它連線到另一個子網路，具有更嚴格的存取控制，並與主要網路界面位於同一個可用區域 (AZ)。

設定

- 主要網路介面可能在負載平衡器後面，也可能不在負載平衡器後面，具有關聯的安全群組，可允許從網際網路存取伺服器。例如，允許TCP連接埠 80 和 443 從 0.0.0/0 或來自負載平衡器。
- 次要網路介面有一個關聯的安全性群組，該群組僅允許SSH存取，從下列其中一個位置啟動：
  - 允許的 IP 位址範圍，無論是在VPC或來自網際網路。
  - 與主要網路介面位於相同 AZ 中的私有子網路。
  - 虛擬私有閘道。

> **Note**
>
> 若要確保容錯移轉功能，請考慮IPv4針對網路介面上的傳入流量使用次要專用。萬一發生執行個體故障，您可以將介面和/或次要私人IPv4位址移至待命執行個體。



網路和安全設備

某些網路和安全性應用裝置，例如負載平衡器、網路位址轉譯 (NAT) 伺服器和 Proxy 伺服器偏好設定多個網路介面。您可建立輔助網路介面，並將它們連接到正在執行這些應用程式類型的執行個體，然後使用它們自己的公有和私有 IP 地址、安全群組和來源/目標檢查來設定其他介面。

工作負載位於不同子網路的雙主目錄執行個體

您可在連線至應用程式伺服器所在之中間層網路的每一個 Web 伺服器上放置網路介面。應用程式伺服器也可以是資料庫伺服器所在之後端網路 (子網) 的雙目錄。每個雙目錄執行個體都會在前端收到及處

理請求、起始後端連線，然後將請求傳送至後端網路的伺服器，而非透過雙目錄執行個體路由網路套件。

在同一帳戶中具有不同工作負載的雙主目錄執VPCs行個體

您可以在其中一個EC2執行個體中ENI啟動執行個體，VPC然後將另一個執行個體 VPC (但位於相同可用區域) 的次要執行個體附加至 這可讓您在不同的網路和安全性VPCs設定之間建立多重主目錄執行個體。您無法跨不同帳戶建立多重主目錄執行VPCs個 AWS 體。

您可以VPCs在下列使用案例中使用雙主目錄執行個體：

- 克服兩者之間無法對等VPCs的重CIDR疊：您可以利用一個中的次要CIDR執行個體，VPC並允許執行個體跨兩個非重疊 IP 範圍進行通訊。
- 在單一帳號VPCs內 Connect 多個：啟用通常會以VPC邊界分隔的個別資源之間的通訊。

低預算、高可用性解決方案

如果提供特定功能的其中一個執行個體故障，其網路介面可連接到為相同角色預先設定之替代或熱待命的執行個體，以迅速復原服務。例如，您可以使用網路介面作為重要服務 (例如資料庫執行個體或執行個體) 的主要或次要網路介面。NAT如果執行個體故障，您 (或更有可能是代您執行的程式碼) 可將網路介面連接到熱待命執行個體。由於介面會維護其私有 IP 位址、彈性 IP 位址和位MAC址，因此只要您將網路介面連接到替代執行個體，網路流量就會開始流向待命執行個體。在執行個體失敗到網路介面連接到待命執行個體的時間之間，使用者會遇到短暫的連線中斷，但不需要變更路由表或DNS伺服器。

申請者受管的網路介面

請求者管理的網路介面是代表您 AWS 服務 建立的VPC網路介面。網路界面與其他服務的資源相關聯，例如來自 Amazon 的資料庫執行個體RDS、NAT閘道或來自的介面VPC端點 AWS PrivateLink。

考量事項

- 您可以檢視您帳戶中請求者受管的網路介面。您可以新增或移除標籤，但不能變更請求者受管網路介面的其他屬性。
- 您無法分離請求者受管網路介面。
- 當您刪除與請求者管理的網路介面相關聯的資源時，會 AWS 服務 卸離網路介面並將其刪除。如果服務分離網路介面，但沒有將其刪除，則可以刪除分離的網路介面。

Console

使用主控台檢視申請者受管的網路介面

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，依序選擇 Network & Security (網路與安全) 和 Network Interfaces (網路介面)。
3. 選取網路介面 ID，開啟其詳細資訊頁面。
4. 下列為可用來確定網路介面用途的重點欄位：
   - Description (說明)：由建立介面的 AWS 服務提供的說明。例如，"VPC端點介面
   - 要求者管理：指出網路介面是否由管理。 AWS
   - 請求者識別碼：建立網路介面之主體或服務的別名或 AWS 帳戶識別碼。如果您建立了網路介面，這就是您的 AWS 帳戶 ID。否則，其他主體或服務將會加以建立。

AWS CLI

若要檢視請求者管理的網路介面，請使用 AWS CLI

使用 describe-network-interfaces 命令，如下所示。

```
aws ec2 describe-network-interfaces --filters Name=requester-managed,Values=true
```

下列為顯示可用來確定網路介面用途之重點欄位的範例輸出：`Description` 和 `InterfaceType`。

```
{
     ...
    "Description": "VPC Endpoint Interface vpce-089f2123488812123",
    ...
    "InterfaceType": "vpc_endpoint",
    ...
    "NetworkInterfaceId": "eni-0d11e3ccd2c0e6c57",
    ...
    "RequesterId": "727180483921",
    "RequesterManaged": true,
    ...
}
```

PowerShell

使用 Windows 的工具檢視請求者管理的網路介面 PowerShell

使用Get-EC2NetworkInterface指令程式，如下所示。

```
Get-EC2NetworkInterface -Filter @{ Name="requester-managed"; Values="true" }
```

下列為顯示可用來確定網路介面用途之重點欄位的範例輸出：`Description` 和 `InterfaceType`。

```
Description        : VPC Endpoint Interface vpce-089f2123488812123
...
InterfaceType      : vpc_endpoint
...
NetworkInterfaceId : eni-0d11e3ccd2c0e6c57
...
RequesterId        : 727180483921
RequesterManaged   : True
...
```

Amazon EC2 網路界面的前綴委派

您可以自動IPv4或手動將私有或IPv6CIDR範圍指派給您的網路介面。透過指派字首，您就可以擴展和簡化應用程式的管理，包括需要在執行個體上有多個 IP 地址的容器和聯網應用程式。如需IPv4和IPv6位址的更多資訊，請參閱Amazon EC2 實例 IP 定址。

以下為可用的指派選項：

- 自動指派 — 從子網VPC路IPv4或IPv6CIDR區塊中 AWS 選擇前置詞，並將其指派給您的網路介面。
- 手動指派 — 您可以從子網VPC路IPv4或IPv6CIDR區塊中指定首碼，並在將前置詞指定給網路介面之前，先 AWS 確認該首碼是否尚未指定給其他資源。

指派字首有下列優點：

- 增加網路介面上的 IP 地址 – 當您使用字首時，會指派 IP 地址區塊，而非個別 IP 地址。這會增加網路介面的 IP 地址數量。

- 簡化容器VPC管理 — 在容器應用程式中，每個容器都需要唯一的 IP 位址。為執行個體指派前置詞可簡化對執行個體的管理VPCs，因為您可以啟動和終止容器，而無需呼叫 Amazon 進EC2APIs行個別 IP 指派。

目錄

- 基本概念
- 考量事項
- 管理網路介面的前置字元

基本概念

- 您可以為新的或現有的網路介面指派字首。
- 若要使用字首，請先為網路介面指派字首，然後將網路介面連接至執行個體，接著設定作業系統。
- 當您選擇指定字首的選項時，該字首必須符合下列需求：
  - 您可以指定的IPv4前置詞是/28。
  - 您可以指定的IPv6前置詞是/80。
  - 前置詞位於網路介面CIDR的子網路中，且不會與指派給子網路中現有資源的其他前置字元或 IP 位址重疊。
- 您可以將字首指派給主要或次要網路介面。
- 您可以將彈性 IP 地址指派給具有指派字首的網路介面。
- 您也可以將彈性 IP 地址指派給已指派字首的 IP 地址部分。
- 我們將執行個體的私人DNS主機名稱解析為主要私人IPv4位址。
- 我們使用以下格式為網絡接口分配每個私有IPv4地址，包括前綴中的那些私有地址：
  - us-east-1 區域

    ```
    ip-private-ipv4-address.ec2.internal
    ```

  - 所有其他區域

    ```
    ip-private-ipv4-address.region.compute.internal
    ```

考量事項

使用字首時，請考慮下列事項：

- AWS Nitro 系統上建置的執行個體支援具有前置字元的網路介面。
- 網路介面的前置字元僅限於IPv6位址和私人IPv4位址。
- 您可以指派給網路介面的 IP 地址數量上限會因執行個體類型而異。指派給網路介面的每個字首都計為一個 IP 地址。例如，`c5.large`執行個體對每個網路介面的`10`IPv4位址有限制。此執行個體的每個網路介面都有一個主要IPv4位址。如果網路介面沒有次要IPv4位址，您最多可以為網路介面指定 9 個字首。對於您指派給網路介面的每個額外IPv4位址，您可以為網路介面指定一個較少的前置詞。如需詳細資訊，請參閱每個網路介面的 IP 位址上限。
- 來源/目標檢查中包含字首。
- 您必須配置操作系統以使用帶有前綴的網絡接口。帶有前綴的接口。注意下列事項：
  - 一些 Amazon Linux AMIs 包含由安裝的其他腳本 AWS，稱為`ec2-net-utils`. 這些指令碼會選擇性自動化您網路介面的組態。它們僅在 Amazon Linux 上使用。
  - 對於容器，您可以針對 Kubernetes 外掛程式使用容器網路介面 (CNI)，或者`dockerd`如果您使用 Docker 來管理您的容器。

管理網路介面的前置字元

您可以使用網路介面管理前置詞，如下所示。

任務

- 在建立網路介面期間指派字首
- 為現有的網路介面指定首碼
- 從網路介面移除字首

在建立網路介面期間指派字首

您可以在建立網路介面時指定自動或自訂前置詞。

Console

在建立網路介面期間指派自動字首

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。

2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選擇 Create network interface (建立網路介面)。
4. 輸入網路介面的說明、選取要在其中建立網路介面的子網路，然後設定專用IPv4和IPv6位址。
5. 展開 Advanced settings (進階設定)。
6. 對於IPv4前綴委派，請執行以下操作之一：
   - 若要自動指定IPv4首碼，請選擇「自動指定」。在「IPv4首碼數目」中，輸入要指定的首碼數目。
   - 若要指定特定IPv4首碼，請選擇「自訂」。選擇「新增首碼」，然後輸入首碼。
7. 對於IPv4前綴委派，請執行以下操作之一：
   - 若要自動指定IPv6首碼，請選擇「自動指定」。在「IPv6首碼數目」中，輸入要指定的首碼數目。
   - 若要指定特定IPv6首碼，請選擇「自訂」。選擇「新增首碼」，然後輸入首碼。

   > **Note**
   >
   > IPv6只有在啟用選取的子網路時，才會顯示首碼委派IPv6。

8. 選取要與網路介面建立關聯的安全群組，並視需要指派資源標籤。
9. 選擇 Create network interface (建立網路介面)。

AWS CLI

在網路介面建立期間指定自動IPv4字首

使用指 create-network-interface令並設定`--ipv4-prefix-count`為您要 AWS 指定的前置字元數目。在下列範例中， AWS 指定一個前置詞。

```
$ C:\> aws ec2 create-network-interface \
--subnet-id subnet-047cfed18eEXAMPLE \
--description "IPv4 automatic example" \
--ipv4-prefix-count 1
```

在網路介面建立期間指定特定IPv4前置字元

使用指 create-network-interface令並設定`--ipv4-prefixes`為前置字元。 AWS 選取此範圍內的 IP 位址。在下列範例中，字首CIDR為`10.0.0.208/28`。

```
$ C:\> aws ec2 create-network-interface \
    --subnet-id subnet-047cfed18eEXAMPLE \
    --description "IPv4 manual example" \
    --ipv4-prefixes Ipv4Prefix=10.0.0.208/28
```

在網路介面建立期間指定自動IPv6字首

使用指 create-network-interface令並設定`--ipv6-prefix-count`為您要 AWS 指定的前置字元數目。在下列範例中， AWS 指定一個前置詞。

```
$ C:\> aws ec2 create-network-interface \
--subnet-id subnet-047cfed18eEXAMPLE \
--description "IPv6 automatic example" \
--ipv6-prefix-count 1
```

在網路介面建立期間指定特定IPv6前置字元

使用指 create-network-interface令並設定`--ipv6-prefixes`為前置字元。 AWS 選取此範圍內的 IP 位址。在下列範例中，字首CIDR為`2600:1f13:fc2:a700:1768::/80`。

```
$ C:\> aws ec2 create-network-interface \
    --subnet-id subnet-047cfed18eEXAMPLE \
    --description "IPv6 manual example" \
    --ipv6-prefixes Ipv6Prefix=2600:1f13:fc2:a700:1768::/80
```

為現有的網路介面指定首碼

您可以為現有的網路介面指定自動或自訂首碼。

Console

將自動字首指派給現有網路介面

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取要指派字首的網路介面，然後選取 Actions (動作)、Manage prefixes (管理字首)。
4. 對於IPv4前綴委派，請執行以下操作之一：
   - 若要自動指定IPv4首碼，請選擇「自動指定」。在「IPv4首碼數目」中，輸入要指定的首碼數目。

- 若要指定特定IPv4首碼，請選擇「自訂」。選擇「新增首碼」，然後輸入首碼。

5. 對於IPv6前綴委派，請執行以下操作之一：
   - 若要自動指定IPv6首碼，請選擇「自動指定」。在「IPv6首碼數目」中，輸入要指定的首碼數目。
   - 若要指定特定IPv6首碼，請選擇「自訂」。選擇「新增首碼」，然後輸入首碼。

> **Note**
>
> IPv6只有在啟用選取的子網路時，才會顯示首碼委派IPv6。

6. 選擇 Save (儲存)。

AWS CLI

您可以使用指定 -ipv6 位址指令來指定IPv6前置字元，並使用指令將字首指IPv4定給現有的網路介面assign-private-ip-addresses。

指定現有網路IPv4介面的自動字首

使用指 assign-private-ip-addresses令並設定`--ipv4-prefix-count`為您要 AWS 指定的前置字元數目。在下列範例中， AWS 指定一個IPv4前置詞。

```
aws ec2 assign-private-ip-addresses \
--network-interface-id eni-081fbb4095EXAMPLE \
--ipv4-prefix-count 1
```

指定特定IPv4首碼給現有的網路介面

使用指 assign-private-ip-addresses令並設定`--ipv4-prefixes`為前置字元。 AWS 選取此範圍內的IPv4位址。在下列範例中，字首CIDR為`10.0.0.208/28`。

```
aws ec2 assign-private-ip-addresses \
--network-interface-id eni-081fbb4095EXAMPLE \
--ipv4-prefixes 10.0.0.208/28
```

指定現有網路IPv6介面的自動字首

使用指定 -ipv6 位址指令，並設定`--ipv6-prefix-count`為您要指派的前置字元數目。 AWS 在下列範例中， AWS 指定一個IPv6前置詞。

```
aws ec2 assign-ipv6-addresses \
--network-interface-id eni-00d577338cEXAMPLE \
--ipv6-prefix-count 1
```

指定特定IPv6首碼給現有的網路介面

使用分配 ipv6 地址命令並設置為前綴。`--ipv6-prefixes` AWS 選取此範圍內的IPv6位址。在下列範例中，字首CIDR為`2600:1f13:fc2:a700:18bb::/80`。

```
aws ec2 assign-ipv6-addresses \
--network-interface-id eni-00d577338cEXAMPLE \
--ipv6-prefixes 2600:1f13:fc2:a700:18bb::/80
```

從網路介面移除字首

您可以從現有的網路介面移除前置詞。

Console

從網路介面移除字首

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面。
4. 選擇 [動作]、[管理首碼]。
5. 對於IPv4前置碼委派，若要移除特定首碼，請選擇要移除的首碼旁邊的「取消指定」。若要移除所有前置字元，請選擇「不指派」。
6. 對於IPv6前置碼委派，若要移除特定首碼，請選擇要移除的首碼旁邊的「取消指定」。若要移除所有前置字元，請選擇「不指派」。

   > **Note**
   >
   > IPv6只有在啟用選取的子網路時，才會顯示首碼委派IPv6。

7. 選擇 Save (儲存)。

AWS CLI

您可以使用未unassign-private-ip-addresses指定-ipv6-address 指令移除字首，並使用指令移除現有網路介面中的IPv6字首。IPv4

若要從網路IPv4介面移除首碼

使用unassign-private-ip-addresses命令並設置`--ipv4-prefix`為要刪除的地址。

```
aws ec2 unassign-private-ip-addresses \
--network-interface-id eni-081fbb4095EXAMPLE \
--ipv4-prefixes 10.0.0.176/28
```

若要從網路IPv6介面移除首碼

使用 unassign-ipv6-addresses 命令，並將 `--ipv6-prefix` 設定為您要移除的位址。

```
aws ec2 unassign-ipv6-addresses \
--network-interface-id eni-00d577338cEXAMPLE \
--ipv6-prefix 2600:1f13:fc2:a700:18bb::/80
```

刪除網路介面

刪除網路介面會釋放所有與介面關聯的屬性，並釋放其他執行個體要使用的所有私有 IP 地址或彈性 IP 地址。

您無法刪除使用中的網路介面。首先，您必須分開網路介面。

使用主控台刪除網路介面

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面的核取方塊，然後選取 Actions (動作)、Delete (刪除)。
4. 出現確認提示時，請選擇 Delete (刪除)。

使用命令列刪除網路介面

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱訪問 Amazon EC2。

- delete-network-interface (AWS CLI)
- Remove-EC2NetworkInterface (AWS Tools for Windows PowerShell)

Amazon EC2 實例網絡帶寬

執行個體頻寬規格適用於執行個體的傳入和傳出流量。舉例來說，如果某個執行個體指定的頻寬最多為 10 Gbps，這意味著傳入流量最高可達 10 Gbps 的頻寬，而傳出流量最高可達 10 Gbps 的頻寬。執行個體可用的網路頻寬EC2取決於幾個因素，如下所示。

多流流量

對於通過網際網路閘道或本機閘道的流量，多流量流量的基準頻寬限制為可用頻寬的 50%。

單一流量

當執行個體不在同一個集群放置群組中時，單一流量的流量基準頻寬限制為 5 Gbps。若要降低延遲並增加單一流量頻寬，請嘗試下列其中一項：

- 使用集群放置群組可為相同置放群組中的執行個體實現最高 10 Gbps 的頻寬。
- 使用 Multipath TCP (MPTCP) 設定兩個端點之間的多個路徑，以達到更高的頻寬。
- 為相同可用區域內的合格執行個體設定 ENA Express，以便在這些執行個體之間達到高達 25 Gbps 的效能。

> **Note**
>
> 單一流程被視為唯一的 5 個元組TCP或流程。UDP對於 IP 標頭之後的其他通訊協定，例如`GRE`或`IPsec`，來源 IP、目的地 IP 和下一個通訊協定的 3 個元組用於定義流程。

可用執行個體頻寬

執行個體的可用網路頻寬取決於執行個體擁 vCPUs 有的數量。例如，`m5.8xlarge`執行個體擁有 32 vCPUs 和 10 Gbps 的網路頻寬，而`m5.16xlarge`執行個體則有 64 vCPUs 和 20 Gbps 的網路頻寬。然而，如果執行個體超過執行個體層級的網路限額 (例如每秒封包數或追蹤連線數)，則執行個體可能無法達到此頻寬。流量可以利用多少可用頻寬取決於數量 vCPUs 和目的地。例如，`m5.16xlarge`執行個體擁有 64 個vCPUs，因此到區域中另一個執行個體的流量可以使用可用的完整頻寬 (20 Gbps)。不過，經過網際網路閘道或本機閘道的流量只能使用 50% 的可用頻寬 (10 Gbps)。

一般而言，16 vCPUs 或以下 (大小`4xlarge`且較小) 的執行個體會記錄為具有「最高」指定頻寬；例如，「最高 10 Gbps」。這些執行個體具有基準頻寬。若要滿足其他需求，他們可使用網路輸入/輸出額度機制，以實現超越基準頻寬的高載頻寬。執行個體可以在有限的時間內使用高載頻寬 (通常是 5 到 60 分鐘)，具體取決於執行個體大小。

執行個體在啟動時會收到最大的網路輸入/輸出額度數。如果執行個體耗盡其網路輸入/輸出額度，則會返回其基準頻寬。每當執行中的執行個體使用的網路頻寬比其基準頻寬少，就會獲得網路輸入/輸出額度。已停止的執行個體不會獲得網路輸入/輸出額度。由於高載頻寬是共用資源，所以即使執行個體具有可用額度，執行個體高載仍只會是盡量實現。

傳入和傳出流量有單獨的網路 I/O 額度儲存貯體。

基本和高載網路效能

Amazon EC2 執行個體類型指南說明每個執行個體類型的網路效能，以及可使用高載頻寬的執行個體可用的基準網路頻寬。如需詳細資訊，請參閱下列內容：

- 網路規格 — 一般用途
- 網路規格 — 運算最佳化
- 網路規格 — 記憶體最佳化
- 網路規格 — 儲存最佳化
- 網路規格 — 加速運算
- 網路規格 — 高效能運算
- 網路規格 — 前一代

或者，您可以使用命令行工具來獲取此信息。

AWS CLI

您可以使用指describe-instance-types AWS CLI 令來顯示例證類型的相關資訊。下列範例會顯示所有 C5 執行個體的網路效能資訊。

```
aws ec2 describe-instance-types \
    --filters "Name=instance-type,Values=c5.*" \
    --query "InstanceTypes[].[InstanceType, NetworkInfo.NetworkPerformance,
 NetworkInfo.NetworkCards[0].BaselineBandwidthInGbps] | sort_by(@,&[2])" \
    --output table
```

下列為範例輸出。

```
---------------------------------------------
|           DescribeInstanceTypes           |
+--------------+--------------------+-------+
|  c5.large    |  Up to 10 Gigabit  |  0.75 |
|  c5.xlarge   |  Up to 10 Gigabit  |  1.25 |
|  c5.2xlarge  |  Up to 10 Gigabit  |  2.5  |
|  c5.4xlarge  |  Up to 10 Gigabit  |  5.0  |
|  c5.9xlarge  |  12 Gigabit        |  12.0 |
|  c5.12xlarge |  12 Gigabit        |  12.0 |
|  c5.18xlarge |  25 Gigabit        |  25.0 |
|  c5.24xlarge |  25 Gigabit        |  25.0 |
|  c5.metal    |  25 Gigabit        |  25.0 |
+--------------+--------------------+-------+
```

PowerShell

您可以使用指Get-EC2InstanceType PowerShell 令來顯示例證類型的相關資訊。下列範例會顯示所有 C5 執行個體的網路效能資訊。

```
Get-EC2InstanceType -Filter @{Name = "instance-type"; Values = "c5.*" } | `
    Select-Object `
    InstanceType,
    @{Name = 'NetworkPerformance'; Expression =
 {($_.Networkinfo.NetworkCards.NetworkPerformance)}},
    @{Name = 'BaselineBandwidthInGbps'; Expression =
 {($_.Networkinfo.NetworkCards.BaselineBandwidthInGbps)}} | `
Format-Table -AutoSize
```

下列為範例輸出。

```
InstanceType NetworkPerformance BaselineBandwidthInGbps
------------ ------------------ -----------------------
c5.4xlarge   Up to 10 Gigabit                      5.00
c5.xlarge    Up to 10 Gigabit                      1.25
c5.12xlarge  12 Gigabit                           12.00
c5.9xlarge   12 Gigabit                           12.00
c5.24xlarge  25 Gigabit                           25.00
c5.metal     25 Gigabit                           25.00
c5.2xlarge   Up to 10 Gigabit                      2.50
c5.large     Up to 10 Gigabit                      0.75
c5.18xlarge  25 Gigabit                           25.00
```

監控執行個體頻寬

您可以使用 CloudWatch 指標來監控執行個體網路頻寬以及傳送和接收的封包。您可以使用彈性網路介面卡 (ENA) 驅動程式提供的網路效能指標來監控流量何時超過 Amazon 在執行個體層級EC2定義的網路限額。

您可以設定 Amazon 是 CloudWatch 使用一分鐘期間還是五分鐘的期間EC2傳送執行個體的指標資料。網路效能測量結果可能會顯示超過限額且封包已被捨棄，而 CloudWatch 執行個體指標則沒有。當執行個體對網路資源的需求 (稱為微突發) 短暫激增時，就可能會發生這種情況，但 CloudWatch 指標的精細度不足以反映這些微秒尖峰。

進一步了解

- 執行個體指標
- 監控網路效能

Amazon EC2 執行個體上的增強聯網

增強型網路使用單根 I/O 虛擬化 (SR-IOV)，針對支援的執行個體類型提供高效能的網路功能。SR-IOV 是一種裝置虛擬化方法，相較於傳統的虛擬化網路介面，可提供更高的 I/O 效能和較低的CPU使用率。增強型聯網可提供更高的頻寬、更高的每秒封包 (PPS) 效能，並持續降低執行個體之間的延遲。使用增強型聯網無需額外收費。

如需每種執行個體類型支援網路速度的詳細資訊，請參閱 Amazon EC2 執行個體類型。

除了 T2 執行個體外，所有目前世代的執行個體類型都支援增強型網路功能。

您可以使用下列其中一種機制來啟用增強型網路：

彈性網路介面卡 (ENA)

針對支援的執行個體類型，彈性網路介面卡 (ENA) 支援高達 100 Gbps 的網路速度。

建立在 AWS Nitro 系統上的所有執行個體都會用ENA於增強型聯網。此外，下列 Xen 執行個體類型也支援ENA：H1、I3、G3、P2、P3 `m4.16xlarge`、P3dn 和 R4。

如需詳細資訊，請參閱在EC2執行個體ENA上啟用增強型聯網。

Intel 82599 虛擬功能 (VF) 介面

Intel 82599 虛擬功能介面在支援的執行個體類型上支援高達 10 Gbps 的網路速度。

下列執行個體類型使用 Intel 82599 VF 介面來增強網路功能：C3、C4、D2、I2、M4 (不包括 m4.16xlarge) 和 R3。

如需詳細資訊，請參閱使用英特爾 82599 VF 接口的增強型網絡。

目錄

- 在EC2執行個體ENA上啟用增強型聯網
- 使用 ENA Express 改善EC2執行個體間的網路效能
- 使用英特爾 82599 VF 接口的增強型網絡
- 監控EC2執行個體ENA設定的網路效能
- 疑難排解 Linux 上的ENA核心驅動程式
- 排除彈性網路介面卡 Windows 驅動程式
- 改善 Linux EC2 執行個體的網路延遲
- 對於性能調整的硝基系統考量
- 最佳化 EC2 Windows 執行個體的網路效能

在EC2執行個體ENA上啟用增強型聯網

Amazon EC2 透過彈性網路介面卡 (ENA) 提供增強的聯網功能。若要使用增強型網路功能，您必須使用包AMI含所需ENA驅動程式的驅動程式，或手動安裝。然後，您可以在執行個體上啟用ENA支援。

若要檢閱版本說明或ENA驅動程式的安裝說明，請參閱與您執行個體作業系統平台相符的索引標籤。

Linux

您可以檢閱下列文件 GitHub：

- 請參閱上的 ENALinux 核心驅動程式版本說明 GitHub。
- 如需包含安裝指示的 ENA Linux 核心驅動程式的概觀，請參閱上的彈性網路介面卡 (ENA) 系列的 Linux 核心驅動程式 GitHub。

Windows

您可以從本指南的「管理裝置驅動程式」一節檢閱下列文件：

- 追蹤ENA驅動程式版本.

- 在 EC2 Windows 執行個體上安裝ENA驅動程式.

對於 Nitro 型執行個體，增強型聯網功能會因執行個體類型實作的 Nitro 版本而有所不同。

若要檢閱執行個體的網路規格，請選擇執行個體類型的執行個體系列連結。如果您不確定哪個執行個體系列適用，請參閱 Amazon EC2 執行個體類型指南中的命名慣例。

- 加速運算執行個體的網路規格
- 運算優化執行個體的網路規格
- 一般用途執行個體的網路規格
- 高效能運算執行個體網路規格
- 記憶體最佳化執行個體的網路
- 儲存優化執行個體的網路規格

目錄

- 增強型網路的先決條件 ENA
- 測試是否已啟用增強型網路
- 在執行個體上啟用增強型網路

增強型網路的先決條件 ENA

若要使用進行增強型聯網準備ENA，請按如下方式設定您的執行個體：

- 啟動在AWS 硝基系統上構建的實例。
- 確定該執行個體具有網際網路連線能力。
- 如果您有要保留的執行個體上的重要資料，則應立即從執行個體建立AMI來備份該資料。更新ENA核心驅動程式並啟用`enaSupport`屬性可能會導致不相容的執行個體或作業系統無法存取。如果您有較新的備份，發生這些狀況時資料便能獲得保留。
- Linux 執行個體 — 使用受支援的 Linux 核心版本和受支援的發行版本啟動執行個體，以便自動為您的執行個體啟用ENA增強型聯網。如需詳細資訊，請參閱 ENALinux 核心驅動程式版本說明。
- 視窗執行個體 — 如果執行個體執行的是 Windows 伺服器 2008 R2SP1，請確定具有 SHA-2 個程式碼簽章支援更新。
- 在您選擇的任何電腦AWS Tools for Windows PowerShell上使用 AWS CloudShell AWS Management Console，AWS CLI或安裝並設定或，最好是您的本機桌上型電腦或筆記型電腦。如需

詳細資訊，請參閱 訪問 Amazon EC2 和 AWS CloudShell 使用者指南。增強型聯網無法從 Amazon 主EC2控台管理。

測試是否已啟用增強型網路

您可以測試執行個體或您的AMIs.

實例屬性

若要查看執行個體是否設定增強型聯網的 `enaSupport` 屬性，請使用下列其中一項命令。如果屬性已設定，回應為 true。

- describe-instances (AWS CLI/AWS CloudShell)

```
aws ec2 describe-instances --instance-ids instance_id --query
 "Reservations[].Instances[].EnaSupport"
```

- Get-EC2Instance(視窗工具 PowerShell)

```
(Get-EC2Instance -InstanceId instance-id).Instances.EnaSupport
```

圖片屬性

若要檢查是否AMI已設定增強型網路`enaSupport`屬性，請使用下列其中一個指令。如果屬性已設定，回應為 true。

- describe-images (AWS CLI/AWS CloudShell)

```
aws ec2 describe-images --image-id ami_id --query "Images[].EnaSupport"
```

- Get-EC2Image(視窗工具 PowerShell)

```
(Get-EC2Image -ImageId ami_id).EnaSupport
```

Linux 網路介面驅動程式

使用下列指令來驗證`ena`核心驅動程式是否正在特定介面上使用，取代您要檢查的介面名稱。如果您使用單一介面 (預設)，介面為 `eth0`。如果作業系統支援可預測的網路名稱，則名稱可能像是 `ens5`。

在下列範例中，不會載入ena核心驅動程式，因為列出的驅動程式為vif。

```
[ec2-user ~]$ ethtool -i eth0
driver: vif
version:
firmware-version:
bus-info: vif-0
supports-statistics: yes
supports-test: no
supports-eeprom-access: no
supports-register-dump: no
supports-priv-flags: no
```

在此範例中，ena核心驅動程式會以最低建議版本載入。此執行個體已正確設定增強型網路。

```
[ec2-user ~]$ ethtool -i eth0
driver: ena
version: 1.5.0g
firmware-version:
expansion-rom-version:
bus-info: 0000:00:05.0
supports-statistics: yes
supports-test: no
supports-eeprom-access: no
supports-register-dump: no
supports-priv-flags: no
```

在執行個體上啟用增強型網路

您使用的程序視執行個體的作業系統而定。

Amazon Linux

Amazon Linux 2 和最新版本的 Amazon Linux AMI 包含增強型聯網所需的核心驅動程式，且已ENA安裝且已啟用ENA支援。因此，如果您在受支援的執行個體類型上啟動具有 Amazon Linux HVM 版本的執行個體，則您的執行個體已啟用增強型聯網功能。如需詳細資訊，請參閱 測試是否已啟用增強型網路。

如果您使用舊版 Amazon Linux 啟動執行個體AMI，但尚未啟用增強型聯網，請使用下列程序啟用增強型聯網。

在 Amazon Linux 上啟用增強型聯網 AMI

1. 連線到您的執行個體。
2. 從執行個體執行下列命令，以最新的核心驅動程式更新執行個體，包括ena：

```
[ec2-user ~]$ sudo yum update
```

3. 從本機電腦，使用 Amazon EC2 主控台或下列其中一個命令重新啟動執行個體：reboot-instances(AWS CLI)、Restart-EC2Instance(AWS Tools for Windows PowerShell)。
4. 再次 Connect 至您的執行個體，並使用modinfo ena指令確認ena核心驅動程式已安裝，以及建議的最低版本測試是否已啟用增強型網路。
5. [EBS-backed 執行個體] 在本機電腦上，使用 Amazon EC2 主控台或下列其中一個命令停止執行個體：stop-instances(AWS CLI)、Stop-EC2Instance(AWS Tools for Windows PowerShell)。如果您的執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中停止執行個體，讓執行個體狀態保持同步。

   [執行個體存放區後端執行個體] 您無法停止執行個體來修改屬性。請改為執行此程序：在 Amazon Linux 上啟用增強型聯網 AMI (執行個體存放區支援執行個體)。
6. 在本機電腦上使用下列其中一個命令啟用增強型聯網屬性。
   - modify-instance-attribute (AWS CLI)

     ```
     aws ec2 modify-instance-attribute --instance-id instance_id --ena-support
     ```

   - Edit-EC2InstanceAttribute(視窗工具 PowerShell)

     ```
     Edit-EC2InstanceAttribute -InstanceId instance-id -EnaSupport $true
     ```

7. (選擇性) AMI 從執行個體建立，如中所述創建一個 Amazon 支EBS持 AMI。會AMI繼承執行個體的增強型聯網enaSupport屬性。因此，您可以使用此AMI功能啟動另一個預設啟用增強型聯網的執行個體。
8. 從您的本機電腦，使用 Amazon EC2 主控台或下列其中一個命令啟動執行個體：start-instances(AWS CLI)、Start-EC2Instance(AWS Tools for Windows PowerShell)。如果您的執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中啟動執行個體，讓執行個體狀態保持同步。
9. 使用ethtool -i eth*n*指令 Connect 至您的執行個體，並確認ena核心驅動程式是否已安裝並載入到網路介面上測試是否已啟用增強型網路。

如果啟用增強型聯網後無法連接至您的執行個體，請參閱 疑難排解 Linux 上的ENA核心驅動程式。

在 Amazon Linux 上啟用增強型聯網 AMI (執行個體存放區支援執行個體)

遵循先前的程序，一直進行到您停止執行個體的那個步驟。AMI如中所述建立新的建立執行個體存放區支援 AMI，請務必在註冊時啟用增強型網路屬性AMI。

- register-image (AWS CLI)

```
aws ec2 register-image --ena-support ...
```

- Register-EC2Image (AWS Tools for Windows PowerShell)

```
Register-EC2Image -EnaSupport $true ...
```

Ubuntu

最新的 Ubuntu HVM AMIs 包括已ENA安裝並啟用ENA支持的增強網絡所需的內核驅動程序。因此，如果您在受支援的執行個體類型HVMAMI上使用最新 Ubuntu 啟動執行個體，則您的執行個體已啟用增強型聯網功能。如需詳細資訊，請參閱 測試是否已啟用增強型網路。

如果您使用舊版AMI執行個體啟動執行個體，但尚未啟用增強型網路功能，您可以安裝`linux-aws`核心套件以取得最新的增強型網路驅動程式，並更新必要的屬性。

安裝 **`linux-aws`** 核心套件 (Ubuntu 16.04 或更新版本)

Ubuntu 16.04 和 18.04 隨附 Ubuntu 自訂核心 (`linux-aws` 核心套件)。若要使用不同的核心，請聯絡 AWS Support。

安裝 **`linux-aws`** 核心套件 (Ubuntu Trusty 14.04)

1. 連線到您的 執行個體。
2. 更新套件快取和套件。

```
ubuntu:~$ sudo apt-get update && sudo apt-get upgrade -y linux-aws
```

> **Important**
>
> 如果更新程序期間提示您安裝 `grub`，請用 `/dev/xvda` 安裝 `grub`，然後選擇保留目前版本的 `/boot/grub/menu.lst`。

3. [EBS-backed 執行個體] 在本機電腦上，使用 Amazon EC2 主控台或下列其中一個命令停止執行個體：stop-instances(AWS CLI)、Stop-EC2Instance(AWS Tools for Windows PowerShell)。如果您的執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中停止執行個體，讓執行個體狀態保持同步。

   [執行個體存放區後端執行個體] 您無法停止執行個體來修改屬性。請改為執行此程序：在 Ubuntu 上啟用增強型聯網 (執行個體後端執行個體)。

4. 在本機電腦上使用下列其中一個命令啟用增強型聯網屬性。

   - modify-instance-attribute (AWS CLI)

     ```
     aws ec2 modify-instance-attribute --instance-id instance_id --ena-support
     ```

   - Edit-EC2InstanceAttribute(視窗工具 PowerShell)

     ```
     Edit-EC2InstanceAttribute -InstanceId instance-id -EnaSupport $true
     ```

5. (選擇性) AMI 從執行個體建立，如中所述創建一個 Amazon 支EBS持 AMI。會AMI繼承執行個體的增強型聯網`enaSupport`屬性。因此，您可以使用此AMI功能啟動另一個預設啟用增強型聯網的執行個體。

6. 從您的本機電腦，使用 Amazon EC2 主控台或下列其中一個命令啟動執行個體：start-instances(AWS CLI)、Start-EC2Instance(AWS Tools for Windows PowerShell)。如果您的執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中啟動執行個體，讓執行個體狀態保持同步。

在 Ubuntu 上啟用增強型聯網 (執行個體後端執行個體)

遵循先前的程序，一直進行到您停止執行個體的那個步驟。AMI如中所述建立新的建立執行個體存放區支援 AMI，請務必在註冊時啟用增強型網路屬性AMI。

- register-image (AWS CLI)

```
aws ec2 register-image --ena-support ...
```

- Register-EC2Image (AWS Tools for Windows PowerShell)

```
Register-EC2Image -EnaSupport $true ...
```

RHEL,SUSE, CentOS

RHEL、SUSE Linux 企業伺服器和 CentOS 的最新AMIs版本包括增強網路所需的核心驅動程式，ENA並且已啟用ENA支援。因此，如果您使用支援執行個體類型的最新執AMI行個體啟動執行個體，則您的執行個體已啟用增強型聯網功能。如需詳細資訊，請參閱 測試是否已啟用增強型網路。

下列程序提供在 Amazon Linux AMI 或 Ubuntu 以外的 Linux 發行版上啟用增強型聯網的一般步驟。如需像是詳細命令語法、檔案位置或套件和工具支援等詳細資訊，請參閱 Linux 發行版本的文件。

在 Linux 上啟用增強型聯網

1. 連線到您的執行個體。
2. 從 GitHub at 複製執行個體上ena核心驅動程式的原始程式碼https://github.com/amzn/amzn-drivers。SUSELinux 企業伺服器 12 SP2 及更新版本預設包含 ENA 2.02，因此您不需要下載和編譯ENA驅動程式。 對於 SUSE Linux 企業服務器 12 SP2 及更高版本，您應該提出一個請求，以將所需的驅動程序版本添加到庫存內核中）。

   ```
   git clone https://github.com/amzn/amzn-drivers
   ```

3. 在執行個體上編譯並安裝ena核心驅動程式。這些步驟視 Linux 發行版本而定。如需有關在 RHEL 上編譯核心驅動程式的詳細資訊，請參閱如何在執行的 Amazon EC2 執行個體上安裝最新的ENS驅動程式以獲得增強的網路支援RHEL？
4. 執行命sudo depmod令以更新核心驅動程式相依性。
5. 更新您`initramfs`的執行個體，以確保新的核心驅動程式會在開機時載入。例如，如果您的發行版本支援 dracut，您可以使用下列命令。

   ```
   dracut -f -v
   ```

6. 判斷系統是否依預設使用可預測的網路介面名稱。使用 systemd 或 udev 197 或以上版本的系統可重新命名乙太網路裝置，但不保證單一網路介面會命名為 `eth0`。該行為會造成連線至執行個體時

發生問題。如需詳細資訊及查看其他的組態選項，請參閱 freedesktop.org 網站上的可預測的網路介面名稱。

a. 您可以使用以下命令檢查RPM基於系統的systemd或udev版本。

```
rpm -qa | grep -e '^systemd-[0-9]\+\|^udev-[0-9]\+'
systemd-208-11.el7_0.2.x86_64
```

在上述的 Red Hat Enterprise Linux 7 範例中，systemd 版本為 208，因此可預測的網路介面名稱已停用。

b. 若要停用可預測網路介面名稱，請將 `net.ifnames=0` 選項新增至 `GRUB_CMDLINE_LINUX` 的 `/etc/default/grub` 行中。

```
sudo sed -i '/^GRUB\_CMDLINE\_LINUX/s/\"$/\ net\.ifnames\=0\"/' /etc/default/
grub
```

c. 重建 grub 組態檔案。

```
sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

7. [EBS-backed 執行個體] 在本機電腦上，使用 Amazon EC2 主控台或下列其中一個命令停止執行個體：stop-instances(AWS CLI)、Stop-EC2Instance(AWS Tools for Windows PowerShell)。如果您的執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中停止執行個體，讓執行個體狀態保持同步。

   [執行個體存放區後端執行個體] 您無法停止執行個體來修改屬性。請改為執行此程序：在 Linux 上啟用增強型聯網 (執行個體存放區後端執行個體)。

8. 在本機電腦上使用下列其中一個命令啟用增強型聯網的 `enaSupport` 屬性：

   - modify-instance-attribute (AWS CLI)

     ```
     aws ec2 modify-instance-attribute --instance-id instance_id --ena-support
     ```

   - Edit-EC2InstanceAttribute(視窗工具 PowerShell)

     ```
     Edit-EC2InstanceAttribute -InstanceId instance-id -EnaSupport $true
     ```

9. (選擇性) AMI 從執行個體建立，如中所述創建一個 Amazon 支EBS持 AMI。會AMI繼承執行個體的增強型聯網`enaSupport`屬性。因此，您可以使用此AMI功能啟動另一個預設啟用增強型聯網的執行個體。

   如果您的執行個體作業系統包含`/etc/udev/rules.d/70-persistent-net.rules`檔案，您必須先刪除該檔案，然後再建立AMI. 此檔案包含原始執行個體的乙太網路介面卡MAC位址。如果其他執行個體使用此檔案開機，作業系統將無法找到設備，且 `eth0` 可能會失敗，因而造成開機問題。這個檔案會在下次開機週期重新產生，而任何從此啟動的執行個體都會AMI建立自己的檔案版本。

10. 從您的本機電腦，使用 Amazon EC2 主控台或下列其中一個命令啟動執行個體：start-instances(AWS CLI)、Start-EC2Instance(AWS Tools for Windows PowerShell)。如果您的執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中啟動執行個體，讓執行個體狀態保持同步。

11. (選擇性) Connect 至執行個體，並確認已安裝核心驅動程式。

    如果啟用增強型聯網後無法連接至您的執行個體，請參閱 疑難排解 Linux 上的ENA核心驅動程式。

在 Linux 上啟用增強型聯網 (執行個體存放區後端執行個體)

遵循先前的程序，一直進行到您停止執行個體的那個步驟。AMI如中所述建立新的建立執行個體存放區支援 AMI，請務必在註冊時啟用增強型網路屬性AMI。

- register-image (AWS CLI)

```
aws ec2 register-image --ena-support ...
```

- Register-EC2Image (AWS Tools for Windows PowerShell)

```
Register-EC2Image -EnaSupport ...
```

使用 Ubuntu DKMS

此方法僅用於測試和意見回饋用途。不適用於生產部署。如需生產部署，請參閱Ubuntu。

> **Important**
>
> 使用無DKMS效訂閱的支援合約。它不應該用於生產部署。

若要ENA在 Ubuntu (EBS支援的執行個體) 上啟用增強型網路功能

1. 請遵循Ubuntu中的步驟 1 和 2。
2. 安裝`build-essential`套件以編譯核心驅動程式和`dkms`套件，以便在每次更新`ena`核心時重建核心驅動程式。

```
ubuntu:~$ sudo apt-get install -y build-essential dkms
```

3. 從 GitHub 位置複製執行個體上`ena`核心驅動程式的原始碼https://github.com/amzn/amzn-drivers。

```
ubuntu:~$ git clone https://github.com/amzn/amzn-drivers
```

4. 將`amzn-drivers`套件移至`/usr/src/`目錄，DKMS以便在每個核心更新中尋找並建置套件。附加原始碼版本編號 (目前的版本編號可在版本備註中找到) 到目錄名稱。例如，下列範例顯示`1.0.0` 版本。

```
ubuntu:~$ sudo mv amzn-drivers /usr/src/amzn-drivers-1.0.0
```

5. 使用下列值建立DKMS組態檔案，取代您的版本`ena`。

   建立檔案。

```
ubuntu:~$ sudo touch /usr/src/amzn-drivers-1.0.0/dkms.conf
```

   編輯檔案並新增下列值。

```
ubuntu:~$ sudo vim /usr/src/amzn-drivers-1.0.0/dkms.conf
PACKAGE_NAME="ena"
PACKAGE_VERSION="1.0.0"
CLEAN="make -C kernel/linux/ena clean"
MAKE="make -C kernel/linux/ena/ BUILD_KERNEL=${kernelver}"
BUILT_MODULE_NAME[0]="ena"
BUILT_MODULE_LOCATION="kernel/linux/ena"
DEST_MODULE_LOCATION[0]="/updates"
```

```
DEST_MODULE_NAME[0]="ena"
AUTOINSTALL="yes"
```

6. 使用. 在執行個體上新增、建置及安裝ena核心驅動程式DKMS。

   將核心驅動程式新增至DKMS。

```
ubuntu:~$ sudo dkms add -m amzn-drivers -v 1.0.0
```

   使用dkms指令建置核心驅動程式。

```
ubuntu:~$ sudo dkms build -m amzn-drivers -v 1.0.0
```

   使用安裝核心驅動程式dkms。

```
ubuntu:~$ sudo dkms install -m amzn-drivers -v 1.0.0
```

7. 重建`initramfs`以便在開機時載入正確的核心驅動程式。

```
ubuntu:~$ sudo update-initramfs -u -k all
```

8. 請使用中測試是否已啟用增強型網路的 modinfo ena 指令，確認ena核心驅動程式是否已安裝。

```
ubuntu:~$ modinfo ena
filename:    /lib/modules/3.13.0-74-generic/updates/dkms/ena.ko
version:  1.0.0
license:  GPL
description: Elastic Network Adapter (ENA)
author:   Amazon.com, Inc. or its affiliates
srcversion:  9693C876C54CA64AE48F0CA
alias:    pci:v00001D0Fd0000EC21sv*sd*bc*sc*i*
alias:    pci:v00001D0Fd0000EC20sv*sd*bc*sc*i*
alias:    pci:v00001D0Fd00001EC2sv*sd*bc*sc*i*
alias:    pci:v00001D0Fd00000EC2sv*sd*bc*sc*i*
depends:
vermagic:    3.13.0-74-generic SMP mod_unload modversions
parm:     debug:Debug level (0=none,...,16=all) (int)
parm:     push_mode:Descriptor / header push mode (0=automatic,1=disable,3=enable)
    0 - Automatically choose according to device capability (default)
    1 - Don't push anything to device memory
    3 - Push descriptors and header buffer to device memory (int)
parm:     enable_wd:Enable keepalive watchdog (0=disable,1=enable,default=1) (int)
```

```
parm:     enable_missing_tx_detection:Enable missing Tx completions. (default=1)
 (int)
parm:     numa_node_override_array:Numa node override map
(array of int)
parm:     numa_node_override:Enable/Disable numa node override (0=disable)
(int)
```

9. 繼續執行Ubuntu中的步驟 3。

啟用 Windows 上的增強型網路

如果您啟動執行個體，且其尚未啟用增強型聯網，您必須在執行個體上下載並安裝必要的網路轉接器驅動程式，然後設定 `enaSupport` 執行個體屬性，以啟動增強型聯網。

啟用增強型聯網

1. 連線到您的執行個體，並以本機管理員登入。
2. [僅限 Windows 伺服器 2016 年和 2019 年] 安裝驅動程式後，執行下列EC2LaunchPowerShell 指令碼以設定執行個體。

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 -
Schedule
```

3. 在執行個體上，安裝驅動程式，如下所示：
   a. 將最新的驅動程式下載到執行個體。
   b. 將 zip 封存檔解壓縮。
   c. 通過運行腳本安裝驅動程`install.ps1`PowerShell 序。

> **Note**
> 如果您收到執行政策錯誤，請將政策設定為 `Unrestricted` (預設設定為 `Restricted` 或 `RemoteSigned`)。在命令列中執行`Set-ExecutionPolicy -ExecutionPolicy Unrestricted`，然後再次執行指令`install.ps1` PowerShell 碼。

4. 在本機電腦上，使用 Amazon EC2 主控台或下列其中一個命令停止執行個體：stop-instances(AWS CLI/AWS CloudShell)、Stop-EC2Instance(AWS Tools for Windows

PowerShell)。如果您的執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中停止執行個體，讓執行個體狀態保持同步。

5. 在執行個體上啟用ENA支援，如下所示：

   a. 在您的本機電腦上，EC2執行下列其中一個命令，以檢查執行個體上的執行個體ENA支援屬性。如果此屬性未啟用，輸出會是「[]」或空白。`EnaSupport` 預設會設定為 `false`。

      - describe-instances (AWS CLI/AWS CloudShell)

        ```
        aws ec2 describe-instances --instance-ids instance_id --query
         "Reservations[].Instances[].EnaSupport"
        ```

      - Get-EC2Instance(視窗工具 PowerShell)

        ```
        (Get-EC2Instance -InstanceId instance-id).Instances.EnaSupport
        ```

   b. 若要啟用ENA支援，請執行下列其中一個指令：

      - modify-instance-attribute (AWS CLI/AWS CloudShell)

        ```
        aws ec2 modify-instance-attribute --instance-id instance_id --ena-support
        ```

      - Edit-EC2InstanceAttribute (AWS Tools for Windows PowerShell)

        ```
        Edit-EC2InstanceAttribute -InstanceId instance_id -EnaSupport $true
        ```

      如果您在重新啟動執行個體時遇到問題，也可以使用下列其中一個指令停用ENA支援：

      - modify-instance-attribute (AWS CLI/AWS CloudShell)

        ```
        aws ec2 modify-instance-attribute --instance-id instance_id --no-ena-support
        ```

      - Edit-EC2InstanceAttribute (AWS Tools for Windows PowerShell)

        ```
        Edit-EC2InstanceAttribute -InstanceId instance_id -EnaSupport $false
        ```

   c. 如先前所示，使用 `true` 或 describe-instances，確認屬性已設定為 Get-EC2Instance。您現在應該會看到下列輸出：

      ```
      [
      ```

```
 true
]
```

6. 從您的本機電腦，使用 Amazon EC2 主控台或下列其中一個命令啟動執行個體：start-instances(AWS CLI/AWS CloudShell)、Start-EC2Instance(AWS Tools for Windows PowerShell)。如果您的執行個體由管理 AWS OpsWorks，您應該使用 AWS OpsWorks 主控台啟動執行個體，讓執行個體狀態保持同步。
7. 在執行個體上，驗證ENA驅動程式是否已安裝並啟用，如下所示：
   a. 用滑鼠右鍵按一下網路圖示，然後選擇 Open Network and Sharing Center (開啟網路和共用中心)。
   b. 選擇乙太網路卡 (例如，Ethernet 2)。
   c. 請選擇 Details (詳細資訊)。針對 Network Connection Details (網路連線詳細資料)，檢查 Description (描述) 是否為 Amazon Elastic Network Adapter (Amazon 彈性網路轉接器)。
8. (選擇性) AMI 從執行個體建立。會AMI繼承執行個體的`enaSupport`屬性。因此，您可以使用它AMI來啟動另一個預設為ENA啟用的執行個體。

使用 ENA Express 改善EC2執行個體間的網路效能

ENAExpress 由可 AWS 擴展的可靠數據報（SRD）技術提供支持。SRD是一種高效能網路傳輸通訊協定，使用動態路由來增加輸送量並將尾端延遲降至最低。使用 ENA Express，您可以在相同可用區域中的兩個EC2執行個體之間進行通訊。

ENA快遞的好處

- 將可用區域內單一流程可使用的最大頻寬從 5 Gbps 增加到 25 Gbps，最高可達彙總執行個體限制。
- 減少EC2執行個體之間網路流量的尾部延遲，特別是在高網路負載期間。
- 偵測並避免擁塞的網路路徑。
- 直接在網路層中處理某些任務，例如接收端的封包重新排序，以及大多數需要的重新傳輸。這會釋放應用程式層進行其他工作。

> **Note**
>
> - 如果應用程式每秒傳送或接收大量封包，而且大部分時間需要最佳化延遲，尤其是在網路上沒有擁塞的期間，則 增強型網路 可能更適合您的網路。

- ENA快速流量無法跨本機區域中的子網路傳送。

在執行個體上為網路介面附件啟用 ENA Express 之後，傳送執行個體會起始與接收執行個體的通訊，並SRD偵測 ENA Express 是否同時在傳送執行個體和接收執行個體上運作。如果 ENA Express 正在運行，則通信可以使用SRD傳輸。如果 ENA Express 未操作，則通信會退回到標準ENA傳輸。

在網路流量較輕的期間，您可能會注意到封包使用 ENA Express 時，封包延遲會略有增加 (數十微秒)。在這段時間內，優先考慮特定網路效能特性的應用程式可以從 ENA Express 中受益，如下所示：

- 在相同可用區域內，將單一流量頻寬上限從 5 Gbps 增加到 25 Gbps，最高可達彙總執行個體限制，程序可受益。例如，如果特定執行個體類型最高支援 12.5 Gbps，則單一流量頻寬也會限制為 12.5 Gbps。
- 在網路擁塞期間，執行時間較長的程序會減少尾延遲。
- 程序可以從更順暢且更標準的網路回應時間分佈中獲益。

主題

- ENA快遞的工作原理
- ENA快速支援的執行個體類型
- Linux 執行個體的先決條
- 調整 Linux 執行個體上ENA快速設定的效能
- 執行個體的ENA快速檢閱工具設EC2定
- 設定EC2執行個體的ENA快速設定

ENA快遞的工作原理

ENAExpress 由可 AWS 擴展的可靠數據報（SRD）技術提供支持。它會將每個網路流量的封包分配到不同的 AWS 網路路徑，並在偵測到擁塞跡象時動態調整分佈。它還管理接收端的封包重新排序。

為確保 ENA Express 能夠按預期管理網路流量，傳送和接收執行個體以及它們之間的通訊必須符合下列所有需求：

- 支援傳送和接收執行個體類型。如需詳細資訊，請參閱 ENA快速支援的執行個體類型 資料表。
- 傳送和接收執行個體都必須設定 ENA Express。如果組態有所差異，您可能會遇到流量預設為標準ENA傳輸的情況。下列情境說明會發生的情況。

情況：組態的差異

執行個體	ENA快速啟用	UDP使用ENA快遞
執行個體 1	是	是
執行個體 2	是	否

在此情況下，兩個執行個體之間的TCP流量可以使用 ENA Express，因為這兩個執行個體都已啟用 Express。但是，由於其中一個執行個體不使用 ENA Express 進行UDP流量，因此這兩個執行個體之間的通訊會UDP使用標準ENA傳輸。

- 傳送和接收執行個體必須在相同的可用區域中執行。
- 執行個體之間的網路路徑不得包含中介軟體方塊。ENA快遞目前不支持中間件框。
- (僅限 Linux 執行個體) 若要充分利用可能的頻寬，請使用 2.2.9 或更新版本的驅動程式。
- (僅限 Linux 執行個體) 若要產生指標，請使用 2.8 或更新版本的驅動程式。

如果未滿足任何要求，則執行個體會使用標準TCP/通UDP訊協定，但不會SRD進行通訊。

若要確保您的執行個體網路驅動程式設定為最佳效能，請檢閱建議的ENA驅動程式最佳做法。這些最佳作法也適用於 ENA Express。如需詳細資訊，請參閱 GitHub 網站上的《ENALinux 驅動程式最佳作法和效能最佳化指南》。

> **Note**
>
> Amazon 是EC2指執行個體與以附件形式連接到執行個體的網路界面之間的關係。ENA快速設定會套用至貼附。如果網路介面從執行個體中斷連結，則附件將不再存在，套用至該介面的 ENA Express 設定將不再有效。執行個體終止時也是如此，即使網路介面仍然存在。

在傳送執行個體和接收執行個體上啟用網路介面附件的 ENA Express 之後，您可以使用 ENA Express 指標來協助確保執行個體能夠充分利用SRD技術提供的效能改進。如需ENA快速測量結果的詳細資訊，請參閱ENA快速的測量結果。

ENA快速支援的執行個體類型

下列索引標籤顯示支援 ENA Express 的執行個體類型。

General purpose

執行個體類型	架構
m6a.12xlarge	x86_64
m6a.16xlarge	x86_64
m6a.24xlarge	x86_64
m6a.32xlarge	x86_64
m6a.48xlarge	x86_64
m6a.metal	x86_64
m6i.8xlarge	x86_64
m6i.12xlarge	x86_64
m6i.16xlarge	x86_64
m6i.24xlarge	x86_64
m6i.32xlarge	x86_64
m6i.metal	x86_64
m6id.8xlarge	x86_64
m6id.12xlarge	x86_64
m6id.16xlarge	x86_64
m6id.24xlarge	x86_64
m6id.32xlarge	x86_64
m6id.metal	x86_64
m7a.12xlarge	x86_64

執行個體類型	架構
m7a.16xlarge	x86_64
m7a.24xlarge	x86_64
m7a.32xlarge	x86_64
m7a.48xlarge	x86_64
m7a.metal-48xl	x86_64
m7g.12xlarge	arm64
m7g.16xlarge	arm64
m7g.metal	arm64
m7gd.12xlarge	arm64
m7gd.16xlarge	arm64
m7gd.metal	arm64
m7i.12xlarge	x86_64
m7i.16xlarge	x86_64
m7i.24xlarge	x86_64
m7i.48xlarge	x86_64
m7i.metal-24xl	x86_64
m7i.metal-48xl	x86_64

Compute optimized

執行個體類型	架構
c6a.12xlarge	x86_64
c6a.16xlarge	x86_64
c6a.24xlarge	x86_64
c6a.32xlarge	x86_64
c6a.48xlarge	x86_64
c6a.metal	x86_64
c6gn.16xlarge	arm64
c6i.8xlarge	x86_64
c6i.12xlarge	x86_64
c6i.16xlarge	x86_64
c6i.24xlarge	x86_64
c6i.32xlarge	x86_64
c6i.metal	x86_64
c6id.8xlarge	x86_64
c6id.12xlarge	x86_64
c6id.16xlarge	x86_64
c6id.24xlarge	x86_64
c6id.32xlarge	x86_64
c6id.metal	x86_64

執行個體類型	架構
c7a.12xlarge	x86_64
c7a.16xlarge	x86_64
c7a.24xlarge	x86_64
c7a.32xlarge	x86_64
c7a.48xlarge	x86_64
c7a.metal-48xl	x86_64
c7g.12xlarge	arm64
c7g.16xlarge	arm64
c7g.metal	arm64
c7gd.12xlarge	arm64
c7gd.16xlarge	arm64
c7gd.metal	arm64
c7i.12xlarge	x86_64
c7i.16xlarge	x86_64
c7i.24xlarge	x86_64
c7i.48xlarge	x86_64
c7i.metal-24xl	x86_64
c7i.metal-48xl	x86_64

Memory optimized

執行個體類型	架構
r6a.12xlarge	x86_64
r6a.16xlarge	x86_64
r6a.24xlarge	x86_64
r6a.32xlarge	x86_64
r6a.48xlarge	x86_64
r6a.metal	x86_64
r6i.8xlarge	x86_64
r6i.12xlarge	x86_64
r6i.16xlarge	x86_64
r6i.24xlarge	x86_64
r6i.32xlarge	x86_64
r6i.metal	x86_64
r6id.8xlarge	x86_64
r6id.12xlarge	x86_64
r6id.16xlarge	x86_64
r6id.24xlarge	x86_64
r6id.32xlarge	x86_64
r6id.metal	x86_64
r7a.12xlarge	x86_64

執行個體類型	架構
r7a.16xlarge	x86_64
r7a.24xlarge	x86_64
r7a.32xlarge	x86_64
r7a.48xlarge	x86_64
r7a.metal-48xl	x86_64
r7g.12xlarge	arm64
r7g.16xlarge	arm64
r7g.metal	arm64
r7gd.12xlarge	arm64
r7gd.16xlarge	arm64
r7gd.metal	arm64
r7i.12xlarge	x86_64
r7i.16xlarge	x86_64
r7i.24xlarge	x86_64
r7i.48xlarge	x86_64
r7i.metal-24xl	x86_64
r7i.metal-48xl	x86_64
r8g.12xlarge	arm64
r8g.16xlarge	arm64
r8g.24xlarge	arm64

執行個體類型	架構
r8g.48xlarge	arm64
r8g.metal-24xl	arm64
r8g.metal-48xl	arm64
u7i-12tb.224xlarge	x86_64
u7in-16tb.224xlarge	x86_64
u7in-24tb.224xlarge	x86_64
u7in-32tb.224xlarge	x86_64
x2idn.16xlarge	x86_64
x2idn.24xlarge	x86_64
x2idn.32xlarge	x86_64
x2idn.metal	x86_64
x2iedn.8xlarge	x86_64
x2iedn.16xlarge	x86_64
x2iedn.24xlarge	x86_64
x2iedn.32xlarge	x86_64
x2iedn.metal	x86_64

Accelerated computing

執行個體類型	架構
g6.48xlarge	x86_64

執行個體類型	架構
g6e.24xlarge	x86_64
g6e.48xlarge	x86_64

Storage optimized

執行個體類型	架構
i4g.4xlarge	arm64
i4g.8xlarge	arm64
i4g.16xlarge	arm64
i4i.8xlarge	x86_64
i4i.12xlarge	x86_64
i4i.16xlarge	x86_64
i4i.24xlarge	x86_64
i4i.32xlarge	x86_64
i4i.metal	x86_64
im4gn.4xlarge	arm64
im4gn.8xlarge	arm64
im4gn.16xlarge	arm64

Linux 執行個體的先決條

若要確保 ENA Express 可以有效運作，請依照下列方式更新 Linux 執行個體的設定。

- 如果您的執行個體使用 Jumbo 框架，請執行下列命令，將最大傳輸單位 (MTU) 設定為8900。

```
[ec2-user ~]$ sudo ip link set dev eth0 mtu 8900
```

- 增加接收器 (Rx) 環的大小，如下所示：

```
[ec2-user ~]$ ethtool -G device rx 8192
```

- 若要最大化 ENA Express 頻寬，請依照下列方式設定TCP佇列限制：

  1. 將TCP小型佇列限制設定為 1MB 或更高。這會讓在通訊端上排入佇列並等待傳輸的資料量增多。

     ```
     sudo sh -c 'echo 1048576 > /proc/sys/net/ipv4/tcp_limit_output_bytes'
     ```

  2. 如果 Linux 發行版本已啟用 eth 裝置上的位元組佇列限制，請將其停用。對裝置佇列來說，這也會讓排入佇列並等待傳輸的資料增多。

     ```
     sudo sh -c 'for txq in /sys/class/net/eth0/queues/tx-*; do echo max > ${txq}/
     byte_queue_limits/limit_min; done'
     ```

     > **Note**
     >
     > Amazon Linux 散發的ENA驅動程式預設會停用位元組佇列限制。

調整 Linux 執行個體上ENA快速設定的效能

若要檢查 Linux 執行個體組態以獲得最佳的 ENA Express 效能，您可以執行 Amazon GitHub 儲存庫上提供的以下指令碼：

https://github.com/amzn/AMZN-EC2-ENA-實用程序/BLOB /主/ENA-快遞 /.sh check-ena-express-settings

該腳本運行一系列測試，並建議建議和必要的配置更改。

執行個體的ENA快速檢閱工具設EC2定

本節介紹如何從 AWS Management Console 或中檢視 ENA Express 資訊 AWS CLI。如需詳細資訊，請選擇與您將使用的方法相符的標籤。

Console

此索引標籤說明如何在中尋找目前 ENA Express 設定的相關資訊 AWS Management Console。

從網路介面清單檢視設定

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取網路介面以查看該執行個體的詳細資訊。您可以選擇 Network interface ID (網路介面 ID) 連結，以開啟詳細資訊頁面；或選取清單左側的核取方塊，以在頁面底部的詳細資訊窗格中檢視詳細資訊。
4. 在 [詳細資料] 索引標籤或詳細資料頁面的 [網路介面附件] 區段中，檢閱ENA快速和ENA快速的設定UDP。

檢視執行個體清單中的設定

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側導覽窗格中，選擇執行個體。
3. 選取執行個體以查看該執行個體的詳細資訊。您可以選擇 Instance ID (執行個體 ID) 連結，以開啟詳細資訊頁面；或選取清單左側的核取方塊，以在頁面底部的詳細資訊窗格中檢視詳細資訊。
4. 在 [網路功能] 索引標籤的 [網路介面] 區段中，向右捲動以檢閱 ENAExpress 和 ENAExpress 的設定UDP。

AWS CLI

此索引標籤說明如何在中尋找目前 ENA Express 設定的相關資訊 AWS CLI。

描述執行個體

如需有關指定執行處理之 ENA Express 組態的資訊，請執行中的describe-instances命令 AWS CLI，如下所示。此命令範例會傳回附加至`--instance-ids`參數所指定之每個執行中執行處理之網路介面的 ENA Express 組態清單。

```
[ec2-user ~]$ aws ec2 describe-instances --instance-
ids i-1234567890abcdef0 i-0598c7d356eba48d7 --query 'Reservations[*].Instances[*].
[InstanceId, NetworkInterfaces[*].Attachment.EnaSrdSpecification]'[
 [
```

```
  [
   "i-1234567890abcdef0",
   [
    {
     "EnaSrdEnabled": true,
     "EnaSrdUdpSpecification": {
      "EnaSrdUdpEnabled": false
     }
    }
   ]
  ]
 ],
 [
  [
   "i-0598c7d356eba48d7",
   [
    {
     "EnaSrdEnabled": true,
     "EnaSrdUdpSpecification": {
      "EnaSrdUdpEnabled": false
     }
    }
   ]
  ]
 ]
]
```

描述網路介面

如需有關網路介面之 ENA Express 設定的資訊，請執行下列describe-network-interfaces命令：

AWS CLI

```
[ec2-user ~]$ aws ec2 describe-network-interfaces
{
"NetworkInterfaces": [
{
 "Association": {
  ....IPs, DNS...
 },
 "Attachment": {
  "AttachTime": "2022-11-17T09:04:28+00:00",
  "AttachmentId": "eni-attach-0ab1c23456d78e9f0",
  "DeleteOnTermination": true,
```

```
  "DeviceIndex": 0,
  "NetworkCardIndex": 0,
  "InstanceId": "i-1234567890abcdef0",
  "InstanceOwnerId": "111122223333",
  "Status": "attached",
  "EnaSrdSpecification": {
   "EnaSrdEnabled": true,
   "EnaSrdUdpSpecification": {
    "EnaSrdUdpEnabled": true
   }
  }
 },
 ...
 "NetworkInterfaceId": "eni-1234567890abcdef0",
 "OwnerId": "111122223333",
 ...
}
]
}
```

PowerShell

此索引標籤說明如何使用尋找目前 ENA Express 設定的相關資訊 PowerShell。

描述網路介面

如需網路介面的 ENA Express 設定的相關資訊，請Get-EC2NetworkInterface Cmdlet使用「工具」執行， PowerShell 如下所示：

```
PS C:\> Get-EC2NetworkInterface -NetworkInterfaceId eni-0d1234e5f6a78901b | `
Select-Object `
    Association,
    NetworkInterfaceId,
    OwnerId,
    @{Name = 'AttachTime'; Expression = { $_.Attachment.AttachTime } },
    @{Name = 'AttachmentId'; Expression = { $_.Attachment.AttachmentId } },
    @{Name = 'DeleteOnTermination'; Expression =
 { $_.Attachment.DeleteOnTermination } },
    @{Name = 'NetworkCardIndex'; Expression = { $_.Attachment.NetworkCardIndex } },
    @{Name = 'InstanceId'; Expression = { $_.Attachment.InstanceId } },
    @{Name = 'InstanceOwnerId'; Expression = { $_.Attachment.InstanceOwnerId } },
    @{Name = 'Status'; Expression = { $_.Attachment.Status } },
    @{Name = 'EnaSrdEnabled'; Expression =
 { $_.Attachment.EnaSrdSpecification.EnaSrdEnabled } },
```

```
    @{Name = 'EnaSrdUdpEnabled'; Expression =
 { $_.Attachment.EnaSrdSpecification.EnaSrdUdpSpecification.EnaSrdUdpEnabled } }

Association         :
NetworkInterfaceId  : eni-0d1234e5f6a78901b
OwnerId             : 111122223333
AttachTime          : 6/11/2022 1:13:11 AM
AttachmentId        : eni-attach-0d1234e5f6a78901b
DeleteOnTermination : True
NetworkCardIndex    : 0
InstanceId          : i-0d1234e5f6a78901b
InstanceOwnerId     : 111122223333
Status              : attached
EnaSrdEnabled       : True
EnaSrdUdpEnabled    : False
```

設定EC2執行個體的ENA快速設定

您可以為支援的EC2執行個體類型設定 ENA Express，而不需要安裝任何其他軟體。

本節介紹如何從 AWS Management Console 或設定 ENA Express AWS CLI。如需詳細資訊，請選擇與您將使用的方法相符的標籤。

Console

此索引標籤說明如何管理連接至執行個體之網路介面的 ENA Express 設定。

從網路介面清單管理ENA快速

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取要連接到執行個體的網路介面。您可以選擇 Network interface ID (網路介面 ID) 連結，以開啟詳細資訊頁面；或選取清單左側的核取方塊。
4. 從頁面右上角的「動作」功能表選擇「管理ENA快速」。這會開啟「管理ENA快速」對話方塊，並顯示選取的網路介面 ID 和目前設定。

> **Note**
> 如果您選取的網路介面未連接至執行個體，則此動作不會顯示在功能表中。

5. 若要使用ENA快速，請選取 [啟用] 核取方塊。
6. 啟用ENA快速時，您可以進行UDP設定。若要使用ENA快速 UDP，請選取 [啟用] 核取方塊。
7. 若要儲存設定，請選擇 Save (儲存)。

從執行個體清單管理ENA快速

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側導覽窗格中，選擇執行個體。
3. 選取您要管理的執行個體。您可以選擇 Instance ID (執行個體 ID) 連結，以開啟詳細資訊頁面；或選取清單左側的核取方塊。
4. 選取要為您的執行個體設定的 Network interface (網路介面)。
5. 從頁面右上角的「動作」功能表選擇「管理ENA快速」。
6. 若要為連接至執行個體的網路介面設定 ENA Express，請從 [網路介面] 清單中選取它。
7. 若要將 ENAExpress 用於選取的網路介面附件，請選取啟用核取方塊。
8. 啟用ENA快速時，您可以進行UDP設定。若要使用ENA快速 UDP，請選取 [啟用] 核取方塊。
9. 若要儲存設定，請選擇 Save (儲存)。

將網路介面連接至EC2執行個體時設定 ENA Express

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選取未連接到執行個體的網路介面 (Status (狀態) 為 Available (可用)。您可以選擇 Network interface ID (網路介面 ID) 連結，以開啟詳細資訊頁面；或選取清單左側的核取方塊。
4. 選取您要連接的 Instance (執行個體)。
5. 若要在將網路介面連接至執行個體之後使用 ENAExpress，請選取「啟用」(Enable) 核取方塊。
6. 啟用ENA快速時，您可以進行UDP設定。若要使用ENA快速 UDP，請選取 [啟用] 核取方塊。
7. 若要將網路介面連接至執行個體並儲存 ENA Express 設定，請選擇 [連接]。

AWS CLI

此索引標籤說明如何在中設定ENA快速設定 AWS CLI。

在連接網路介面時設定ENA快速

若要在將網路介面連接至執行個體時設定 ENA Express，請執行中的attach-network-interface命令 AWS CLI，如下列範例所示：

範例 1：使用ENA快速TCP傳輸流量，但不適用於UDP流量

在此範例中，我們會將 `EnaSrdEnabled` 設定為 true，並且我們允許將 `EnaSrdUdpEnabled` 預設為 false。

```
[ec2-user ~]$ aws ec2 attach-network-interface --network-interface-
id eni-0123f4567890a1b23 --instance-id i-0f1a234b5cd67e890 --device-index 1 --ena-
srd-specification 'EnaSrdEnabled=true'
{
"AttachmentId": "eni-attach-012c3d45e678f9012"
}
```

範例 2：同時針對TCP流量和UDP流量使用ENA快速

在此範例中，我們將 `EnaSrdEnabled` 和 `EnaSrdUdpEnabled` 都設定為 true。

```
[ec2-user ~]$ aws ec2 attach-network-interface --network-interface-
id eni-0123f4567890a1b23 --instance-id i-0f1a234b5cd67e890 --device-index 1 --ena-
srd-specification
 'EnaSrdEnabled=true,EnaSrdUdpSpecification={EnaSrdUdpEnabled=true}'
{
"AttachmentId": "eni-attach-012c3d45e678f9012"
}
```

更新網路介面附件的ENA快速設定

若要更新連接至執行個體之網路介面的 ENA Express 設定，請執行中的modify-network-interface-attribute命令 AWS CLI，如下列範例所示：

範例 1：使用ENA快速TCP傳輸流量，但不適用於UDP流量

在此範例中，我們會將 `EnaSrdEnabled` 設定為 true，並且我們允許將 `EnaSrdUdpEnabled` 預設為 false，如果之前並未設定。

```
[ec2-user ~]$ aws ec2 modify-network-interface-attribute --network-interface-
id eni-0123f4567890a1b23 --ena-srd-specification 'EnaSrdEnabled=true'
```

範例 2：同時針對TCP流量和UDP流量使用ENA快速

在此範例中，我們將 `EnaSrdEnabled` 和 `EnaSrdUdpEnabled` 都設定為 true。

```
[ec2-user ~]$ aws ec2 modify-network-interface-attribute --
network-interface-id eni-0123f4567890a1b23 --ena-srd-specification
 'EnaSrdEnabled=true,EnaSrdUdpSpecification={EnaSrdUdpEnabled=true}'
```

範例 3：停止使用ENA快速交UDP通

在此範例中，我們會將 `EnaSrdUdpEnabled` 設定為 false。

```
[ec2-user ~]$ aws ec2 modify-network-interface-attribute --
network-interface-id eni-0123f4567890a1b23 --ena-srd-specification
 'EnaSrdUdpSpecification={EnaSrdUdpEnabled=false}'
```

PowerShell

此索引標籤說明如何使用來設定ENA快速設定 PowerShell。

在連接網路介面時設定ENA快速

若要設定網路介面的 ENA Express 設定，請Add-EC2NetworkInterface Cmdlet使用的工具執行，PowerShell 如下列範例所示：

範例 1：使用ENA快速TCP傳輸流量，但不適用於UDP流量

在此範例中，我們會將 `EnaSrdEnabled` 設定為 true，並且我們允許將 `EnaSrdUdpEnabled` 預設為 false。

```
PS C:\> Add-EC2NetworkInterface `
-NetworkInterfaceId eni-0123f4567890a1b23 `
-InstanceId i-0f1a234b5cd67e890 `
-DeviceIndex 1 `
-EnaSrdSpecification_EnaSrdEnabled $true

eni-attach-012c3d45e678f9012
```

範例 2：同時針對TCP流量和UDP流量使用ENA快速

在此範例中，我們將 `EnaSrdEnabled` 和 `EnaSrdUdpEnabled` 都設定為 true。

```
PS C:\> Add-EC2NetworkInterface `
-NetworkInterfaceId eni-0123f4567890a1b23 `
-InstanceId i-0f1a234b5cd67e890 `
-DeviceIndex 1 `
-EnaSrdSpecification_EnaSrdEnabled $true `
-EnaSrdUdpSpecification_EnaSrdUdpEnabled $true

eni-attach-012c3d45e678f9012
```

更新網路介面附件的ENA快速設定

若要更新連接至執行個體之網路介面的 ENA Express 設定，請執行 Tools 中的Add-EC2NetworkInterface Cmdlet命令 PowerShell，如下列範例所示：

範例 1：使用ENA快速TCP傳輸流量，但不適用於UDP流量

在此範例中，我們會將 `EnaSrdEnabled` 設定為 true，並且我們允許將 `EnaSrdUdpEnabled` 預設為 false，如果之前並未設定。

```
PS C:\> Edit-EC2NetworkInterfaceAttribute `
-NetworkInterfaceId eni-0123f4567890a1b23 `
-EnaSrdSpecification_EnaSrdEnabled $true ;
Get-EC2NetworkInterface -NetworkInterfaceId eni-0123f4567890a1b23 | `
Select-Object `
    NetworkInterfaceId,
    @{Name = 'EnaSrdEnabled'; Expression =
 { $_.Attachment.EnaSrdSpecification.EnaSrdEnabled }},
    @{Name = 'EnaSrdUdpEnabled'; Expression =
 { $_.Attachment.EnaSrdSpecification.EnaSrdUdpSpecification.EnaSrdUdpEnabled }} | `
Format-List

NetworkInterfaceId : eni-0123f4567890a1b23
EnaSrdEnabled      : True
EnaSrdUdpEnabled   : False
```

範例 2：同時針對TCP流量和UDP流量使用ENA快速

在此範例中，我們將 `EnaSrdEnabled` 和 `EnaSrdUdpEnabled` 都設定為 true。

```
PS C:\> Edit-EC2NetworkInterfaceAttribute `
-NetworkInterfaceId eni-0123f4567890a1b23 `
```

```
-EnaSrdSpecification_EnaSrdEnabled $true `
-EnaSrdSpecification_EnaSrdUdpSpecification_EnaSrdUdpEnabled $true ;
Get-EC2NetworkInterface -NetworkInterfaceId eni-0123f4567890a1b23 | `
Select-Object `
    NetworkInterfaceId,
    @{Name = 'EnaSrdEnabled'; Expression =
 { $_.Attachment.EnaSrdSpecification.EnaSrdEnabled }},
    @{Name = 'EnaSrdUdpEnabled'; Expression =
 { $_.Attachment.EnaSrdSpecification.EnaSrdUdpSpecification.EnaSrdUdpEnabled }} | `
Format-List

NetworkInterfaceId : eni-0123f4567890a1b23
EnaSrdEnabled      : True
EnaSrdUdpEnabled   : True
```

範例 3：停止使用ENA快速交UDP通

在此範例中，我們會將 `EnaSrdUdpEnabled` 設定為 false。

```
PS C:\> Edit-EC2NetworkInterfaceAttribute `
-NetworkInterfaceId eni-0123f4567890a1b23 `
-EnaSrdSpecification_EnaSrdUdpSpecification_EnaSrdUdpEnabled $false ;
Get-EC2NetworkInterface -NetworkInterfaceId eni-0123f4567890a1b23 | `
Select-Object `
    NetworkInterfaceId,
    @{Name = 'EnaSrdEnabled'; Expression =
 { $_.Attachment.EnaSrdSpecification.EnaSrdEnabled }},
    @{Name = 'EnaSrdUdpEnabled'; Expression =
 { $_.Attachment.EnaSrdSpecification.EnaSrdUdpSpecification.EnaSrdUdpEnabled }} | `
Format-List

NetworkInterfaceId : eni-0123f4567890a1b23
EnaSrdEnabled      : True
EnaSrdUdpEnabled   : False
```

啟動時設定ENA快速

啟動執行個體時，您可以使用下列其中一種方法直接設定 ENA Express。指定的連結會引 AWS Management Console 導您參考這些方法的指示。

- 啟動執行個體精靈：您可以在啟動時使用啟動執行個體精靈來設定 ENA Express。如需詳細資訊，請參閱啟動執行個體精靈中Network settings (網路設定)的進階網路組態。

- 啟動範本：當您使用啟動範本時，您可以在啟動時設定 ENA Express。如需詳細資訊，請參閱創建一個 Amazon EC2 啟動模板頁面，然後展開 [網路設定] 區段，並檢閱進階網路組態。

使用英特爾 82599 VF 接口的增強型網絡

對於非 AWS Nitro 系統之上建置的EC2執行個體類型，Intel 82599 虛擬功能 (VF) 介面可提供增強的網路功能。該接口使用英特爾`ixgbevf`驅動程序。

下列索引標籤顯示如何驗證針對執行個體作業系統安裝的網路介面卡驅動程式。

Linux

Linux 網路介面驅動程式

請用下列命令確認模組使用於特定介面，並將介面名稱換成您想要檢查的介面。如果您使用單一介面 (預設)，則這會是 `eth0`。如果作業系統支援可預測的網路名稱，則名稱可能像是 `ens5`。

在下列範例中，不會載入 `ixgbevf` 模組，因為列出的驅動程式為 `vif`。

```
[ec2-user ~]$ ethtool -i eth0
driver: vif
version:
firmware-version:
bus-info: vif-0
supports-statistics: yes
supports-test: no
supports-eeprom-access: no
supports-register-dump: no
supports-priv-flags: no
```

在此範例中，將載入 `ixgbevf` 模組。此執行個體已正確設定增強型網路。

```
[ec2-user ~]$ ethtool -i eth0
driver: ixgbevf
version: 4.0.3
firmware-version: N/A
bus-info: 0000:00:03.0
supports-statistics: yes
supports-test: yes
supports-eeprom-access: no
supports-register-dump: yes
supports-priv-flags: no
```

Windows

視窗網路介面卡

若要確定已安裝驅動程式，請連線到執行個體，並開啟裝置管理員。您應該會在網路介面卡下看到`Intel(R) 82599 Virtual Function`列出。

目錄

- 為增強型聯網準備執行個體
- 測試是否已啟用增強型網路
- 在執行個體上啟用增強型網路
- 疑難排解連線問題

為增強型聯網準備執行個體

若要準備使用 Intel 82599 VF 介面的增強型聯網，請依下列方式設定您的執行個體：

- 選取下列支援的執行個體類型：C3、C4、D2、I2、M4 (不含 `m4.16xlarge`) 和 R3。
- 確定該執行個體具有網際網路連線能力。
- 如果您有要保留的執行個體上的重要資料，則應立即從執行個體建立AMI來備份該資料。更新核心與核心模組，以及啟用 `sriovNetSupport` 屬性，可能使執行個體變得不相容或使作業系統變得無法連線。如果您有較新的備份，發生這些狀況時資料便能獲得保留。
- Linux 執行個體 — 從HVMAMI使用的 Linux 核心版本 2.6.32 或更新版本啟動執行個體。最新的 Amazon Linux HVM AMIs 具有安裝增強型聯網所需的模組，並已設定必要的屬性。因此，如果您使用目前的 Amazon EBS Linux 啟動 Amazon 支援的增強型聯網執行個體 HVMAMI，則您的執行個體已啟用增強型聯網功能。

> **Warning**
>
> 僅HVM執行個體支援增強型聯網。啟用 PV 執行個體的增強型聯網可能使其變為無法連線。在沒有正確的模組或模組版本下設定此屬性，也可能使執行個體變為無法連線。

- 視窗執行個體 — 從 64 位元啟動執行個體HVMAMI。您無法在視窗伺服器 2008 上啟用增強型網路功能。增強型網路已啟用視窗伺服器 2012 R2 和視窗伺服器 2016 及更新版本AMIs。Windows Server 2012 R2 內含 Intel 驅動程式 1.0.15.3，因此我們建議您使用 Pnputil.exe 公用程式將驅動程式升級至最新版本。

- 在您選擇的任何電腦AWS Tools for Windows PowerShell上使用 AWS CloudShell AWS Management Console，AWS CLI或安裝並設定或，最好是您的本機桌上型電腦或筆記型電腦。如需詳細資訊，請參閱 訪問 Amazon EC2 和 AWS CloudShell 使用者指南。增強型聯網無法從 Amazon 主EC2控台管理。

測試是否已啟用增強型網路

確認`sriovNetSupport`屬性已設定。

實例屬性 (sriovNetSupport)

若要查看執行個體是否設定增強型聯網的 `sriovNetSupport` 屬性，請使用下列其中一項命令。如果已設定屬性，則值為`simple`。

- describe-instance-attribute (AWS CLI) (AWS CLI/AWS CloudShell)

```
aws ec2 describe-instance-attribute --instance-id instance_id --attribute
 sriovNetSupport
```

- Get-EC2InstanceAttribute (AWS Tools for Windows PowerShell)

```
Get-EC2InstanceAttribute -InstanceId instance-id -Attribute sriovNetSupport
```

映像屬性 (sriovNetSupport)

若要檢查是否AMI已設定增強型網路`sriovNetSupport`屬性，請使用下列其中一個指令。如果已設定屬性，則值為`simple`。

- describe-images (AWS CLI)

```
aws ec2 describe-images --image-id ami_id --query "Images[].SriovNetSupport"
```

- Get-EC2Image (AWS Tools for Windows PowerShell)

```
(Get-EC2Image -ImageId ami-id).SriovNetSupport
```

在執行個體上啟用增強型網路

您使用的程序視執行個體的作業系統而定。

> **Warning**
>
> 增強型聯網屬性一經啟用後，便無法停用。

Amazon Linux

最新的 Amazon Linux HVM AMIs 具有安裝增強型聯網所需的`ixgbevf`模組，並已設定必要的`sriovNetSupport`屬性。因此，如果您使用目前的 Amazon Linux 啟動執行個體類型 HVMAMI，則您的執行個體已啟用增強型聯網。如需詳細資訊，請參閱測試是否已啟用增強型網路。

如果您使用舊版 Amazon Linux 啟動執行個體AMI，但尚未啟用增強型聯網，請使用下列程序啟用增強型聯網。

啟用增強型聯網

1. 連線到您的執行個體。
2. 在執行個體上，執行下列命令，將您的執行個體更新為最新的核心與核心模組，包括 `ixgbevf`：

   ```
   [ec2-user ~]$ sudo yum update
   ```

3. 從本機電腦使用 Amazon EC2 主控台或下列命令之一重新啟動執行個體：reboot-instances(AWS CLI)、Restart-EC2Instance(AWS Tools for Windows PowerShell)。
4. 再次連接至執行個體，並使用 `ixgbevf` 中的 modinfo ixgbevf 命令確認 測試是否已啟用增強型網路 模組已安裝且為最低建議版本。
5. [EBS-backed 執行個體] 在本機電腦上，使用 Amazon EC2 主控台或下列其中一個命令停止執行個體：stop-instances(AWS CLI)、Stop-EC2Instance(AWS Tools for Windows PowerShell)。如果您的執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中停止執行個體，讓執行個體狀態保持同步。

   [執行個體存放區後端執行個體] 您無法停止執行個體來修改屬性。請跳至下一個程序。
6. 在本機電腦上使用下列其中一個命令啟用增強型聯網屬性。

   AWS CLI

   modify-instance-attribute (AWS CLI)

   ```
   aws ec2 modify-instance-attribute --instance-id instance_id --sriov-net-support
    simple
   ```

PowerShell

Edit-EC2InstanceAttribute (AWS Tools for Windows PowerShell)

```
Edit-EC2InstanceAttribute -InstanceId instance_id -SriovNetSupport "simple"
```

7. (選擇性) AMI 從執行個體建立，如中所述創建一個 Amazon 支EBS持 AMI。會AMI繼承執行個體的增強型聯網屬性。因此，您可以使用此AMI功能啟動另一個預設啟用增強型聯網的執行個體。
8. 從您的本機電腦，使用 Amazon EC2 主控台或下列其中一個命令啟動執行個體：start-instances(AWS CLI)、Start-EC2Instance(AWS Tools for Windows PowerShell)。如果您的執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中啟動執行個體，讓執行個體狀態保持同步。
9. 連接至執行個體，並使用 `ixgbevf` 中的 ethtool -i eth*n* 命令確認 測試是否已啟用增強型網路 模組已在網路介面上安裝及載入。

啟用增強型聯網 (執行個體後端執行個體)

遵循先前的程序，一直進行到您停止執行個體的那個步驟。AMI如中所述建立新的建立執行個體存放區支援 AMI，請務必在註冊時啟用增強型網路屬性AMI。

AWS CLI

register-image (AWS CLI/AWS CloudShell)

```
aws ec2 register-image --sriov-net-support simple ...
```

PowerShell

Register-EC2Image (AWS Tools for Windows PowerShell)

```
Register-EC2Image -SriovNetSupport "simple" ...
```

Ubuntu

開始之前，請先檢查您的執行個體是否啟用增強型聯網。

快速入門 Ubuntu HVM AMIs 包括增強網絡的必要驅動程序。如果您有 2.16.4 版以前的 `ixgbevf`，則可以安裝 `linux-aws` 核心套件來取得最新的增強型聯網驅動程式。

以下程序提供在 Ubuntu 執行個體上編譯 `ixgbevf` 模組的一般步驟。

安裝 **`linux-aws`** 核心套件

1. 連線到您的 執行個體。
2. 更新套件快取和套件。

```
ubuntu:~$ sudo apt-get update && sudo apt-get upgrade -y linux-aws
```

> **Important**
>
> 如果更新程序期間提示您安裝 `grub`，請用 `/dev/xvda` 安裝 `grub`，然後選擇保留目前版本的 `/boot/grub/menu.lst`。

其他 Linux 發行版本

開始之前，請先檢查您的執行個體是否啟用增強型聯網。最新的快速入門HVMAMIs包括增強型網路的必要驅動程式，因此您不需要執行其他步驟。

如果您需要在 Amazon Linux 或 Ubuntu 以外的其他 Linux 發行版本上使用 Intel 82599 VF 介面啟用增強型聯網，以下程序提供一般步驟。如需像是詳細命令語法、檔案位置或套件和工具支援等詳細資訊，請參閱 Linux 發行版本的特定文件。

在 Linux 上啟用增強型聯網

1. 連線到您的 執行個體。
2. 在您的執行個體上從 Sourceforge (https://sourceforge.net/projects/e1000/files/ixgbevf%20stable/) 下載 `ixgbevf` 模組來源。

   2.16.4 之前版本的 `ixgbevf`，包括 2.14.2 版，無法在某些 Linux 發行版本上正確建置，包括特定版本的 Ubuntu。
3. 在您的執行個體上編譯及安裝 `ixgbevf` 模組。

> **Warning**
>
> 如果您為目前核心編譯 `ixgbevf` 模組，接著升級核心，但未針對新核心重建驅動程式，系統可能在下次重新啟動時還原為發行版本專屬的 `ixgbevf` 模組。如果發行版本專屬版本與增強型聯網不相容，則系統可能變得無法連線。

4. 執行 sudo depmod 命令更新模組相依性。
5. 在執行個體上更新 `initramfs`，確定開機時載入新模組。
6. 判斷系統是否依預設使用可預測的網路介面名稱。使用 systemd 或 udev 197 或以上版本的系統可重新命名乙太網路裝置，但不保證單一網路介面會命名為 `eth0`。該行為會造成連線至執行個體時發生問題。如需詳細資訊及查看其他的組態選項，請參閱 freedesktop.org 網站上的可預測的網路介面名稱。

   a. 您可以使用以下命令檢查RPM基於系統的systemd或udev版本：

   ```
   [ec2-user ~]$ rpm -qa | grep -e '^systemd-[0-9]\+\|^udev-[0-9]\+'
   systemd-208-11.el7_0.2.x86_64
   ```

   在上述的 Red Hat Enterprise Linux 7 範例中，systemd 版本為 208，因此可預測的網路介面名稱已停用。

   b. 若要停用可預測網路介面名稱，請將 `net.ifnames=0` 選項新增至 `GRUB_CMDLINE_LINUX` 的 `/etc/default/grub` 行中。

   ```
   [ec2-user ~]$ sudo sed -i '/^GRUB\_CMDLINE\_LINUX/s/\"$/\ net\.ifnames\=0\"/' /
   etc/default/grub
   ```

   c. 重建 grub 組態檔案。

   ```
   [ec2-user ~]$ sudo grub2-mkconfig -o /boot/grub2/grub.cfg
   ```

7. [EBS-backed 執行個體] 在本機電腦上，使用 Amazon EC2 主控台或下列其中一個命令停止執行個體：停止執行個體 (AWS CLI/AWS CloudShell)、Stop-EC2Instance()。AWS Tools for Windows PowerShell如果您的執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中停止執行個體，讓執行個體狀態保持同步。

   [執行個體存放區後端執行個體] 您無法停止執行個體來修改屬性。請跳至下一個程序。

8. 在本機電腦上使用下列其中一個命令啟用增強型聯網屬性。

AWS CLI

modify-instance-attribute (AWS CLI/AWS CloudShell)

```
aws ec2 modify-instance-attribute --instance-id instance_id --sriov-net-support
 simple
```

PowerShell

Edit-EC2InstanceAttribute (AWS Tools for Windows PowerShell)

```
Edit-EC2InstanceAttribute -InstanceId instance_id -SriovNetSupport "simple"
```

9. (選擇性) AMI 從執行個體建立，如中所述創建一個 Amazon 支EBS持 AMI。會AMI繼承執行個體的增強型聯網屬性。因此，您可以使用此AMI功能啟動另一個預設啟用增強型聯網的執行個體。

   如果您的執行個體作業系統包含`/etc/udev/rules.d/70-persistent-net.rules`檔案，您必須先刪除該檔案，然後再建立AMI. 此檔案包含原始執行個體的乙太網路介面卡MAC位址。如果其他執行個體使用此檔案開機，作業系統將無法找到設備，且 `eth0` 可能會失敗，因而造成開機問題。這個檔案會在下次開機週期重新產生，而任何從此啟動的執行個體都會AMI建立自己的檔案版本。

10. 從您的本機電腦，使用 Amazon EC2 主控台或下列其中一個命令啟動執行個體：start-instances(AWS CLI)、Start-EC2Instance(AWS Tools for Windows PowerShell)。如果您的執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中啟動執行個體，讓執行個體狀態保持同步。

11. (選擇性) 連線到執行個體，並確認模組已安裝。

啟用增強型聯網 (執行個體存放區支援的執行個體)

遵循先前的程序，一直進行到您停止執行個體的那個步驟。AMI如中所述建立新的建立執行個體存放區支援 AMI，請務必在註冊時啟用增強型網路屬性AMI。

AWS CLI

register-image (AWS CLI/AWS CloudShell)

```
aws ec2 register-image --sriov-net-support simple ...
```

PowerShell

Register-EC2Image (AWS Tools for Windows PowerShell)

```
Register-EC2Image -SriovNetSupport "simple" ...
```

Windows

如果您啟動執行個體，且其尚未啟用增強型聯網，您必須在執行個體上下載並安裝必要的網路轉接器驅動程式，然後設定 `sriovNetSupport` 執行個體屬性，以啟動增強型聯網。您只能在支援的執行個體類型上啟用此屬性。如需詳細資訊，請參閱Amazon EC2 執行個體上的增強聯網。

> **Important**
>
> 若要在 Windows 中檢視最新的驅動程式更新AMIs，請參閱 Windows AMI參考中的AWS Windows AMI 版本歷程記錄。

啟用增強型聯網

1. 連線到您的執行個體，並以本機管理員登入。
2. [Windows 伺服器 2016 及更新版本] 在安裝驅動程式之後，執行下列EC2啟動 PowerShell指令碼以設定執行個體。

   ```
   PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 -
   Schedule
   ```

   > **Important**
   >
   > 當您啟用初始化執行個體 EC2 Launch 指令碼時，管理員密碼將會重設。您可以修改組態檔來停用管理員密碼重設，方法為在初始化任務的設定中指定它。

3. 從執行個體中，下載適用於您的作業系統的 Intel 網路轉接器驅動程式：

   - Windows Server 2022

     請造訪下載頁面並下載`Wired_driver_`*`version`*`_x64.zip`。

   - Windows Server 2019 (包括 Server 版本 1809 和更新版本*)

請造訪下載頁面並下載Wired_driver_*version*_x64.zip。

- Windows Server 2016 (包括 Server 版本 1803 和更早版本*)

  請造訪下載頁面並下載Wired_driver_*version*_x64.zip。

- Windows Server 2012 R2

  請造訪下載頁面並下載Wired_driver_*version*_x64.zip。

- Windows Server 2012

  請造訪下載頁面並下載Wired_driver_*version*_x64.zip。

- Windows Server 2008 R2

  請造訪下載頁面並下載PROWinx64Legacy.exe。

* Intel 驅動程式與軟體頁面並未特別說明伺服器版本 1803 和更早版本，以及 1809 和更新版本。

4. 安裝適用於您的作業系統的 Intel 網路轉接器驅動程式。

   - Windows Server 2008 R2

     1. 在 Downloads (下載) 資料夾中，找出 PROWinx64Legacy.exe 檔案並將其重新命名為 PROWinx64Legacy.zip。
     2. 解壓縮 PROWinx64Legacy.zip 檔案的內容。
     3. 開啟命令列，導覽至解壓縮的資料夾，然後執行下列命令，以使用公用pnputil程式在驅動程式存放區中新增和安裝INF檔案。

        ```
        C:\> pnputil -a PROXGB\Winx64\NDIS62\vxn62x64.inf
        ```

   - Windows Server 2022、Windows Server 2019、Windows Server 2016、Windows Server 2012 R2 和 Windows Server 2012

     1. 在 Downloads (下載) 資料夾中，解壓縮 Wired_driver_*version*_x64.zip 檔案的內容。
     2. 在解壓縮的資料夾中，找出 Wired_driver_*version*_x64.exe 檔案並將其重新命名為 Wired_driver_*version*_x64.zip。
     3. 解壓縮 Wired_driver_*version*_x64.zip 檔案的內容。
     4. 開啟命令列，導覽至解壓縮的資料夾，然後執行下列其中一個命令，以使用公用pnputil程式在驅動程式存放區中新增和安裝INF檔案。

        - Windows Server 2022

```
C:\> pnputil -i -a PROXGB\Winx64\WS2022\vxs.inf
```

- Windows Server 2019

```
C:\> pnputil -i -a PROXGB\Winx64\NDIS68\vxn68x64.inf
```

- Windows Server 2016

```
C:\> pnputil -i -a PROXGB\Winx64\NDIS65\vxn65x64.inf
```

- Windows Server 2012 R2

```
C:\> pnputil -i -a PROXGB\Winx64\NDIS64\vxn64x64.inf
```

- Windows Server 2012

```
C:\> pnputil -i -a PROXGB\Winx64\NDIS63\vxn63x64.inf
```

5. 在本機電腦上使用下列其中一個命令啟用增強型聯網屬性。

   AWS CLI

   modify-instance-attribute (AWS CLI/AWS CloudShell)

```
aws ec2 modify-instance-attribute --instance-id instance_id --sriov-net-support
 simple
```

   PowerShell

   Edit-EC2InstanceAttribute (AWS Tools for Windows PowerShell)

```
Edit-EC2InstanceAttribute -InstanceId instance_id -SriovNetSupport "simple"
```

6. (選擇性) AMI 從執行個體建立，如中所述創建一個 Amazon 支EBS持 AMI。會AMI繼承執行個體的增強型聯網屬性。因此，您可以使用此AMI功能啟動另一個預設啟用增強型聯網的執行個體。
7. 從您的本機電腦，使用 Amazon EC2 主控台或下列其中一個命令啟動執行個體：start-instances(AWS CLI)、Start-EC2Instance(AWS Tools for Windows PowerShell)。如果您的執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中啟動執行個體，讓執行個體狀態保持同步。

疑難排解連線問題

如果您在啟用增強型聯網時失去連線，則 `ixgbevf` 模組可能與核心不相容。請嘗試安裝您執行個體適用之 Linux 發行版本隨附的 `ixgbevf` 模組版本。

如果您為 PV 執行個體啟用增強型聯網AMI，或者，這可能會導致您的執行個體無法連線。

如需詳細資訊，請參閱如何在EC2執行個體上開啟和設定增強型聯網？

監控EC2執行個體ENA設定的網路效能

彈性網路介面卡 (ENA) 驅動程式會從啟用它們的執行個體發佈網路效能指標。您可以使用這些指標來進行執行個體效能的故障診對、為工作負載選擇正確的執行個體大小、主動計劃擴展活動，以及基準化分析應用程式，以判斷它們是否能夠最大化執行個體上的可用效能。

Amazon 在執行個體層級EC2定義了網路最大值，以確保高品質的聯網體驗，包括跨執行個體大小的一致網路效能。 AWS 為每個執行個體提供下列項目的上限：

- 頻寬功能 — 根據EC2執行個體類型和大小，每個執行個體都有最大頻寬來彙總傳入和輸出流量。一些執行個體使用網路 I/O 額度機制，根據平均頻寬使用率來配置網路頻寬。Amazon EC2 也有流量和互聯網 AWS Direct Connect 的最大帶寬。如需詳細資訊，請參閱Amazon EC2 實例網絡帶寬。
- P acket-per-second (PPS) 效能 — 根據EC2執行個體類型和大小，每個執行個體都有最大PPS效能。
- 追蹤的連線 - 安全性群組會追蹤每個建立的連線，以確保傳回封包如預期般交付。每個執行個體可追蹤的連線數目上限。如需詳細資訊，請參閱 Amazon EC2 安全組連接跟踪
- 連結本機服務存取 — Amazon EC2 為服務、執行PPS個體中繼資料服務和 Amazon 時間同步服務等DNS服務的流量提供每個網路界面的最大數量。

當執行個體的網路流量超過上限時，會排入佇列，然後卸除網路封包，以 AWS 塑造超過上限的流量。您可以使用網路效能指標監控流量何時超過上限。這些指標可即時通知您網路流量的影響，以及可能的網路效能問題。

目錄

- 要求
- ENA驅動程式的指標
- 檢視 執行個體的網路效能指標
- ENA快速的測量結果

- 帶有DPDK驅動程序的網絡性能指標 ENA
- 執行免費執行個體的指標 BSD

要求

Linux 執行個體

- 安裝ENA驅動程式版本 2.2.10 或更新版本。若要確認已安裝的版本，請使用 ethtool 命令。在下列範例中，版本符合最低需求。

```
[ec2-user ~]$ ethtool -i eth0 | grep version
version: 2.2.10
```

  若要升級ENA驅動程式，請參閱增強型網路。
- 若要將這些指標匯入 Amazon CloudWatch，請安裝 CloudWatch 代理程式。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的收集網路效能指標。
- 若要支援`conntrack_allowance_available`指標，請安裝ENA驅動程式版本 2.8.1。

Windows 執行個體

- 安裝ENA驅動程式 2.2.2 或更新版本。若要確認已安裝的版本，請使用如下所示的 Device Manager (裝置管理員)。
  1. 執行 `devmgmt.msc` 以開啟 Device Manager (裝置管理員)。
  2. 展開 Network Adapters (網路介面卡)。
  3. 選擇 Amazon Elastic Network Adapter (Amazon 彈性網絡介面卡)、Properties (屬性) 。
  4. 在 Driver (驅動程式)索引標籤上，找出 Driver Version (驅動程式版本)

  若要升級ENA驅動程式，請參閱增強型網路。
- 若要將這些指標匯入 Amazon CloudWatch，請安裝 CloudWatch 代理程式。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的「收集進階網路指標」。

ENA驅動程式的指標

ENA驅動程式會即時提供下列指標給執行個體。它們提供自上次重設驅動程式後，每個網路介面上排入佇列或丟棄的累積封包數目。

指標	描述	支援於
bw_in_allowance_exceeded	因傳入的彙總頻寬超過執行個體的上限而排入佇列或丟棄的封包數目。	所有執行個體類型
bw_out_allowance_exceeded	因傳出的彙總頻寬超過執行個體的上限而排入佇列或丟棄的封包數目。	所有執行個體類型
conntrack_allowance_exceeded	因為連線追蹤超過執行個體的上限且無法建立新的連線，而丟棄的封包數目。這可能會導致傳送或傳回執行個體流量的封包遺失。	所有執行個體類型
conntrack_allowance_available	在達到該執行個體類型的追蹤連線限額之前，執行個體可建立的追蹤連線數目。	僅在 AWS Nitro 系統上建置的執行個體
linklocal_allowance_exceeded	由於本機 Proxy 服務PPS的流量超過網路介面的上限，所以捨棄的封包數目。這會影響到DNS服務、執行個體中繼資料服務和 Amazon 時間同步服務的流量。	所有執行個體類型
pps_allowance_exceeded	因為雙向PPS超過執行個體的上限，所以佇列或捨棄的封包數目。此限制也會計算每個超過 1024 PPS 次ENI的出口片段掉落數。	所有執行個體類型

檢視 執行個體的網路效能指標

您使用的程序視執行個體的作業系統而定。

Linux 執行個體

您可以將指標發佈至您最愛的工具，以視覺化方式呈現指標資料。例如，您可以 CloudWatch 使用 CloudWatch 代理程式將指標發佈到 Amazon。代理程式可讓您選取個別指標並控制發佈。

您也可以使用 ethtool 以擷取每個網路介面的指標，例如下方所示的 eth0。

```
[ec2-user ~]$ ethtool -S eth0
     bw_in_allowance_exceeded: 0
     bw_out_allowance_exceeded: 0
     pps_allowance_exceeded: 0
     conntrack_allowance_exceeded: 0
     linklocal_allowance_exceeded: 0
     conntrack_allowance_available: 136812
```

Windows 執行個體

您可以使用任何 Windows 效能計數器的取用者來檢視指標。數據可以根據 EnaPerfCounters 清單進行解析。這是定義效能計數器提供者及其計數器的XML檔案。

若要安裝資訊清單

如果您使用包含ENA驅動程式 2.2.2 或更新版本的執行個AMI體啟動，或是使用驅動程式套件 2.2.2 的安裝指令碼，則資訊清單已經安裝。ENA若要手動安裝資訊清單，請使用下列步驟：

1. 使用下列命令移除現有的資訊清單：

   ```
   unlodctr /m:EnaPerfCounters.man
   ```

2. 將資訊清單檔案 `EnaPerfCounters.man` 從驅動程式安裝套件複製到 `%SystemRoot%\System32\drivers`。
3. 使用下列命令安裝新的資訊清單：

   ```
   lodctr /m:EnaPerfCounters.man
   ```

使用效能監視器檢視測量結果

1. 開啟效能監控器。
2. 按 Ctrl+N 以新增計數器。
3. 從清單中選擇「ENA封包設定」。

4. 選取要監控的執行個體，然後選取 Add (新增)。
5. 選擇 OK (確定)。

ENA快速的測量結果

ENAExpress 由可 AWS 擴展的可靠數據報（SRD）技術提供支持。SRD是一種高效能網路傳輸通訊協定，使用動態路由來增加輸送量並將尾端延遲降至最低。如果您已針對傳送執行個體和接收執行個體上的網路介面附件啟用 ENA Express，則可以使用 ENA Express 指標來協助確保執行個體充分利用SRD技術提供的效能改善功能。例如：

- 評估您的資源，以確保它們有足夠的能力來建立更多的SRD連接。
- 識別存在阻止合格傳出封包使用的潛在問題的位置SRD。
- 計算用SRD於執行個體的傳出流量百分比。
- 計算用SRD於執行個體的傳入流量百分比。

> **Note**
>
> 若要產生指標，請使用 2.8 或更新版本的驅動程式。

若要查看已針對 ENA Express 篩選的 Linux 執行個體指標清單，請針對您的網路介面執行下列ethtool命令 (如下所示`eth0`)。記下量`ena_srd_mode`度的值。

```
[ec2-user ~]$ ethtool -S eth0 | grep ena_srd
NIC statistics:
 ena_srd_mode: 1
 ena_srd_tx_pkts: 0
 ena_srd_eligible_tx_pkts: 0
 ena_srd_rx_pkts: 0
 ena_srd_resource_utilization: 0
```

下列測量結果適用於所有啟用 ENA Express 的執行處理。

管理模式

說明已啟用哪些ENA快速功能。相關值如下：

- `0`= ENA 快速關閉，UDP關閉

- 1= ENA 快速開啟、UDP關閉
- 2= ENA 快速關閉，UDP開啟

> **Note**
>
> 這只會在 ENA Express 最初啟用且設定UDP為使用它時才會發生。先前的值會保留供UDP流量使用。

- 3= ENA 快速開啟，UDP開啟

使用符合資格的方法

網絡數量如下：

- 支援傳送和接收執行個體類型。如需詳細資訊，請參閱 ENA快速支援的執行個體類型 資料表。
- 傳送和接收執行個體都必須設定 ENA Express。
- 傳送和接收執行個體必須在相同的可用區域中執行。
- 執行個體之間的網路路徑不得包含中介軟體方塊。ENA快遞目前不支持中間件框。

> **Note**
>
> ENAExpress 資格量度涵蓋來源和目的地需求，以及兩個端點之間的網路。在計算後，合格封包仍可能會被取消資格。例如，如果一個合格的封包超過最大傳輸單位 (MTU) 限制，它會回落到標準ENA傳輸，雖然封包仍會反映為計數器中的資格。

埃納德 _ 晶片 _pkts

指定時間範圍內傳輸的SRD封包數目。

ena_srd_rx_pkts

指定時間段內接收到的SRD封包數目。

資源使用率

執行處理使用的並行SRD連線允許的最大記憶體使用率百分比。

若要確認封包傳輸是否正在使用SRD，您可以比較指定期間內的合格封包數目 (`ena_srd_eligible_tx_pkts`測量結果) 與傳輸的SRD封包數目 (`ena_srd_tx_pkts`測量結果)。

出口流量 (傳出封包)

若要確保您的出口流量SRD如預期使用，請比較指定時段內的SRD合格SRD封包數 (`ena_srd_tx_pkts`) 與傳送的封包數 ()。`ena_srd_eligible_tx_pkts`

合格封包數目與傳送封包數目之間的SRD顯著差異通常是資源使用率問題所造成的。當連接到執行個體的網路卡已耗盡其最大資源，或封包超過MTU限制時，合格的封包將無法透過傳輸SRD，而且必須回復至標準ENA傳輸。在即時移轉或即時伺服器更新期間，封包也可能會落入此間隙。需要額外的疑難排解，才能判斷根本原因。

> **Note**
>
> 您可以忽略符合條件的封包數量和封包數目之間偶爾的SRD細微差異。例如，當您的執行個體為SRD流量建立與另一個執行個體的連線時，就會發生這種情況。

若要瞭解指定時間期間內的總輸出流量佔多少百分比SRD，請將 sent (`ena_srd_tx_pkts`) SRD 封包數與該執行個體在該時間內傳送的封包總數 (`NetworkPacketOut`) 進行比較。

輸入流量 (傳入封包)

若要瞭解輸入流量使用的百分比SRD，請比較指定時段內收到的SRD封包數 (`ena_srd_rx_pkts`) 與該執行個體在該時間內接收到的封包總數 (`NetworkPacketIn`)。

資源使用率

資源使用率是根據單一執行環境在指定時間可保留的並行SRD連線數目而定。資源使用率指標 (`ena_srd_resource_utilization`) 會追蹤您目前執行個體的使用率。當使用率接近 100% 時，您預期將會看到效能問題。ENAExpress 從標準傳輸退回SRD到標準ENA傳輸，並且丟棄數據包的可能性增加。高資源使用率表示是時候擴展執行個體，以改善網路效能。

> **Note**
>
> 當執行個體的網路流量超過上限時，會排入佇列，然後卸除網路封包，以 AWS 塑造超過上限的流量。

Persistence

在為執行個體啟用「ENA快速」時，會累積輸出和輸入指標。如果 ENA Express 停用，量度就會停止累積，但只要執行個體仍在執行，就會持續存在。如果執行個體重新啟動或終止，或者如果網路介面已從執行個體中斷連接，就會重設指標。

帶有DPDK驅動程序的網絡性能指標 ENA

ENA驅動程式 2.2.0 版及更新版本支援網路度量報告。DPDK20.11 包括ENA驅動程式 2.2.0，是支援此功能的第一個DPDK版本。

您可以使用範例應用程式來檢視DPDK統計資料。若要啟動範例應用程式的互動式版本，請執行下列命令。

```
./app/dpdk-testpmd -- -i
```

在此互動式工作階段中，您可以輸入命令來擷取連接埠的延伸統計資料。下列範例命令會擷取連接埠 0 的統計資料。

```
show port xstats 0
```

以下是使用範例應用程式的互動式工作階段DPDK範例。

```
[root@ip-192.0.2.0 build]# ./app/dpdk-testpmd -- -i
        EAL: Detected 4 lcore(s)
        EAL: Detected 1 NUMA nodes
        EAL: Multi-process socket /var/run/dpdk/rte/mp_socket
        EAL: Selected IOVA mode 'PA'
        EAL: Probing VFIO support...
        EAL:   Invalid NUMA socket, default to 0
        EAL:   Invalid NUMA socket, default to 0
        EAL: Probe PCI driver: net_ena (1d0f:ec20) device: 0000:00:06.0
(socket 0)
        EAL: No legacy callbacks, legacy socket not created
        Interactive-mode selected

        Port 0: link state change event
        testpmd: create a new mbuf pool <mb_pool_0>: n=171456,
size=2176, socket=0
        testpmd: preferred mempool ops selected: ring_mp_mc

        Warning! port-topology=paired and odd forward ports number, the
last port will pair with itself.

        Configuring Port 0 (socket 0)
        Port 0: 02:C7:17:A2:60:B1
        Checking link statuses...
        Done
```

```
        Error during enabling promiscuous mode for port 0: Operation
not supported - ignore
        testpmd> show port xstats 0
        ###### NIC extended statistics for port 0
        rx_good_packets: 0
        tx_good_packets: 0
        rx_good_bytes: 0
        tx_good_bytes: 0
        rx_missed_errors: 0
        rx_errors: 0
        tx_errors: 0
        rx_mbuf_allocation_errors: 0
        rx_q0_packets: 0
        rx_q0_bytes: 0
        rx_q0_errors: 0
        tx_q0_packets: 0
        tx_q0_bytes: 0
        wd_expired: 0
        dev_start: 1
        dev_stop: 0
        tx_drops: 0
        bw_in_allowance_exceeded: 0
        bw_out_allowance_exceeded: 0
        pps_allowance_exceeded: 0
        conntrack_allowance_exceeded: 0
        linklocal_allowance_exceeded: 0
        rx_q0_cnt: 0
        rx_q0_bytes: 0
        rx_q0_refill_partial: 0
        rx_q0_bad_csum: 0
        rx_q0_mbuf_alloc_fail: 0
        rx_q0_bad_desc_num: 0
        rx_q0_bad_req_id: 0
        tx_q0_cnt: 0
        tx_q0_bytes: 0
        tx_q0_prepare_ctx_err: 0
        tx_q0_linearize: 0
        tx_q0_linearize_failed: 0
        tx_q0_tx_poll: 0
        tx_q0_doorbells: 0
        tx_q0_bad_req_id: 0
        tx_q0_available_desc: 1023
        testpmd>
```

如需有關範例應用程式以及使用它來擷取延伸統計資料的詳細資訊。請參閱文件中的 Testpmd 應用程式使用者指南。DPDK

執行免費執行個體的指標 BSD

從 2.3.0 版開始，ENA免費BSD驅動程式支援在執行 Free 的執行個體上收集網路效能指標。BSD若要啟用「免費」BSD 量度的集合，請輸入下列命令並設定 *interval* 轉換成介於 1 到 3600 之間的值。這會指定收集免費BSD量度的頻率 (以秒為單位)。

```
sysctl dev.ena.network_interface.eni_metrics.sample_interval=interval
```

例如，下列命令會將驅動程式設定為每 10 秒收集網路介面 1 上的 Free BSD 度量：

```
sysctl dev.ena.1.eni_metrics.sample_interval=10
```

若要關閉 Free BSD 量度的集合，您可以執行上述命令並指定`0`為 *interval*.

啟用收集「免費」BSD 測量結果之後，您可以執行下列命令來擷取最新一組收集的測量結果。

```
sysctl dev.ena.network_interface.eni_metrics
```

疑難排解 Linux 上的ENA核心驅動程式

彈性網路介面卡 (ENA) 旨在改善作業系統的健康狀況，並減少因意外硬體行為或故障而導致長期中斷的機會。該ENA架構使設備或驅動程序故障對系統盡可能透明。本主題提供的疑難排解資訊ENA。

如果您無法連線至您的執行個體，請由 疑難排解連線問題 章節開始。

如果您在移轉至第六代執行個體類型後遇到效能降低的情況，請參閱將執行個體遷移到第六代EC2執行個體之前，我需要執行哪些動作，以確保獲得最大的網路效能？

如果您能夠連線至您的執行個體，則可以使用本主題後面章節中說明的故障診斷和復原機制來收集診斷資訊。

內容

- 疑難排解連線問題
- 保持有效機制
- 註冊讀取逾時
- 統計資料

- syslog 中的驅動程式錯誤日誌
- 非最佳化組態通知

疑難排解連線問題

如果您在啟用增強聯網時失去連線，則 ena 模組可能與您的執行個體目前執行之核心不相容。如果您為特定核心版本 (不含 dkms 或具有設定不正確的 dkms.conf 檔案) 安裝模組並更新執行個體核心，則可能會發生此情況。如果在開機階段載入的執行個體核心沒有正確安裝 ena 模組，則您的執行個體將無法識別網路轉接器，且將無法連接到您的執行個體。

如果您為 PV 執行個體啟用增強型聯網AMI，或者這也可能導致您的執行個體無法連線。

如果您的執行個體在啟用增強型聯網功能後變成無法連線ENA，您可以停用執行個體的`enaSupport`屬性，它會回復到庫存網路介面卡。

使用 (EBS支援的執行個體ENA) 停用增強型聯網

1. 在本機電腦上，使用 Amazon EC2 主控台或下列其中一個命令停止執行個體：停止執行個體 (AWS CLI)、Stop-EC2Instance(AWS Tools for Windows PowerShell)。如果您的執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中停止執行個體，讓執行個體狀態保持同步。

   > **Tip**
   > 如果您使用執行個體後端執行個體，則無法停止該執行個體。請跳至若要停用增強型聯網 ENA (執行個體商店烘焙執行個體)。

2. 在本機電腦上使用下列命令停用增強聯網屬性。

   - modify-instance-attribute (AWS CLI)

   ```
   $ C:\> aws ec2 modify-instance-attribute --instance-id instance_id --no-ena-support
   ```

3. 從您的本機電腦，使用 Amazon EC2 主控台或下列其中一個命令啟動執行個體：啟動執行個體 (AWS CLI)、Start-EC2Instance(AWS Tools for Windows PowerShell)。如果您的執行個體由管理 AWS OpsWorks，您應該在 AWS OpsWorks 主控台中啟動執行個體，讓執行個體狀態保持同步。
4. (選用) 請連線到您的執行個體，並嘗試遵循 ena 中的步驟使用目前核心版本來重新安裝 在EC2執行個體ENA上啟用增強型聯網 模組。

使用 (執行個體存放區支援執行個體ENA) 停用增強型聯網

如果您的執行個體是執行個體商店支援的執行個體，請依照中所述建立新執AMI行個體。建立執行個體存放區支援 AMI當您註冊時，請務必停用增強型網路enaSupport屬性AMI。

- register-image (AWS CLI)

```
$ C:\> aws ec2 register-image --no-ena-support ...
```

- Register-EC2Image (AWS Tools for Windows PowerShell)

```
C:\> Register-EC2Image -EnaSupport $false ...
```

保持有效機制

該ENA設備以固定速率（通常每秒一次）發布保持活動事件。該ENA驅動程序實現了一個看門狗機制，該機制檢查這些保持活動消息是否存在。如果有訊息存在，監視程式會重新啟動，否則驅動程式會認為裝置出現故障並執行下列作業：

- 傾印目前統計資料至 syslog
- 重ENA設裝置
- 重設ENA驅動程式狀態

上述重置過程可能會導致短時間內的某些流量損失（TCP連接應該能夠恢復），但不應以其他方式影響用戶。

ENA裝置也可能會間接要求裝置重設程序，方法是不傳送 keep-alive 通知，例如，如果ENA裝置在載入無法復原的組態之後達到未知狀態。

以下是重設程序的範例：

```
[18509.800135] ena 0000:00:07.0 eth1: Keep alive watchdog timeout. // The watchdog
 process initiates a reset
[18509.815244] ena 0000:00:07.0 eth1: Trigger reset is on
[18509.825589] ena 0000:00:07.0 eth1: tx_timeout: 0 // The driver logs the current
 statistics
[18509.834253] ena 0000:00:07.0 eth1: io_suspend: 0
[18509.842674] ena 0000:00:07.0 eth1: io_resume: 0
[18509.850275] ena 0000:00:07.0 eth1: wd_expired: 1
```

```
[18509.857855] ena 0000:00:07.0 eth1: interface_up: 1
[18509.865415] ena 0000:00:07.0 eth1: interface_down: 0
[18509.873468] ena 0000:00:07.0 eth1: admin_q_pause: 0
[18509.881075] ena 0000:00:07.0 eth1: queue_0_tx_cnt: 0
[18509.888629] ena 0000:00:07.0 eth1: queue_0_tx_bytes: 0
[18509.895286] ena 0000:00:07.0 eth1: queue_0_tx_queue_stop: 0
.......
........
[18511.280972] ena 0000:00:07.0 eth1: free uncompleted tx skb qid 3 idx 0x7 // At the
 end of the down process, the driver discards incomplete packets.
[18511.420112] [ENA_COM: ena_com_validate_version] ena device version: 0.10 //The
 driver begins its up process
[18511.420119] [ENA_COM: ena_com_validate_version] ena controller version: 0.0.1
 implementation version 1
[18511.420127] [ENA_COM: ena_com_admin_init] ena_defs : Version:[b9692e8] Build date
 [Wed Apr  6 09:54:21 IDT 2016]
[18512.252108] ena 0000:00:07.0: Device watchdog is Enabled
[18512.674877] ena 0000:00:07.0: irq 46 for MSI/MSI-X
[18512.674933] ena 0000:00:07.0: irq 47 for MSI/MSI-X
[18512.674990] ena 0000:00:07.0: irq 48 for MSI/MSI-X
[18512.675037] ena 0000:00:07.0: irq 49 for MSI/MSI-X
[18512.675085] ena 0000:00:07.0: irq 50 for MSI/MSI-X
[18512.675141] ena 0000:00:07.0: irq 51 for MSI/MSI-X
[18512.675188] ena 0000:00:07.0: irq 52 for MSI/MSI-X
[18512.675233] ena 0000:00:07.0: irq 53 for MSI/MSI-X
[18512.675279] ena 0000:00:07.0: irq 54 for MSI/MSI-X
[18512.772641] [ENA_COM: ena_com_set_hash_function] Feature 10 isn't supported
[18512.772647] [ENA_COM: ena_com_set_hash_ctrl] Feature 18 isn't supported
[18512.775945] ena 0000:00:07.0: Device reset completed successfully // The reset
 process is complete
```

註冊讀取逾時

該ENA架構建議內存映射 I/O（MMIO）讀取操作的使用有限。MMIO寄存器僅在其初始化過程中由ENA設備驅動程序訪問。

如果驅動程式日誌 (在 dmesg 輸出中可用) 指出讀取操作失敗，則可能是由不相容或編譯錯誤的驅動程式、忙碌的硬體裝置或硬體故障引起的。

表示讀取操作失敗的間歇日誌項目不應視為問題；驅動程式會在此情況下將其重試。但是，包含讀取失敗的一系列日誌項目則會指出驅動程式或硬體問題。

以下是驅動程序日誌項目的範例，指出由於逾時而導致的讀取操作失敗：

```
[ 47.113698] [ENA_COM: ena_com_reg_bar_read32] reading reg failed for timeout.
 expected: req id[1] offset[88] actual: req id[57006] offset[0]
[ 47.333715] [ENA_COM: ena_com_reg_bar_read32] reading reg failed for timeout.
 expected: req id[2] offset[8] actual: req id[57007] offset[0]
[ 47.346221] [ENA_COM: ena_com_dev_reset] Reg read32 timeout occurred
```

統計資料

如果您遇到網路效能不足或延遲問題，則應擷取並檢查裝置統計資料。您可以使用 ethtool 來獲得這些統計資料，如下所示。

```
[ec2-user ~]$ ethtool -S ethN
NIC statistics:
 tx_timeout: 0
 suspend: 0
 resume: 0
 wd_expired: 0
 interface_up: 1
 interface_down: 0
 admin_q_pause: 0
 bw_in_allowance_exceeded: 0
 bw_out_allowance_exceeded: 0
 pps_allowance_exceeded: 0
 conntrack_allowance_available: 450878
 conntrack_allowance_exceeded: 0
 linklocal_allowance_exceeded: 0
 queue_0_tx_cnt: 4329
 queue_0_tx_bytes: 1075749
 queue_0_tx_queue_stop: 0
 ...
```

以下說明下列命令輸出參數：

`tx_timeout`: *N*

Netdev 監視程式啟用的次數。

`suspend`: *N*

驅動程式執行暫停操作的次數。

`resume`: *N*

驅動程式執行繼續操作的次數。

`wd_expired`: *N*

驅動程式在前 3 秒內未收到保持有效事件的次數。

`interface_up`: *N*

ENA介面啟動的次數。

`interface_down`: *N*

ENA介面關閉的次數。

`admin_q_pause`: *N*

在執行中狀態下找不到管理員佇列的次數。

`bw_in_allowance_exceeded`: *N*

因傳入的彙總頻寬超過執行個體的上限而排入佇列或丟棄的封包數目。

`bw_out_allowance_exceeded`: *N*

因傳出的彙總頻寬超過執行個體的上限而排入佇列或丟棄的封包數目。

`pps_allowance_exceeded`: *N*

因為雙向PPS超過執行個體的上限，所以佇列或捨棄的封包數目。此限制也會計算每個超過 1024 PPS 次ENI的出口片段掉落數。

`conntrack_allowance_available`: *N*

在達到該執行個體類型的追蹤連線限額之前，執行個體可建立的追蹤連線數目。僅適用於 Nitro 型執行個體。免費BSD執行個體或DPDK環境不支援。

`conntrack_allowance_exceeded`: *N*

因為連線追蹤超過執行個體的上限且無法建立新的連線，而丟棄的封包數目。這可能會導致傳送或傳回執行個體流量的封包遺失。

`linklocal_allowance_exceeded`: *N*

由於本機 Proxy 服務PPS的流量超過網路介面的上限，所以捨棄的封包數目。這會影響到DNS服務、執行個體中繼資料服務和 Amazon 時間同步服務的流量。

`queue_`*N*`_tx_cnt`: *N*

此佇列的已傳輸封包數目。

queue_*N*_tx_bytes: *N*

此佇列的已傳輸位元組數目。

queue_*N*_tx_queue_stop: *N*

排入佇列的次數 *N* 已滿並停止。

queue_*N*_tx_queue_wakeup: *N*

排入佇列的次數 *N* 停止後恢復。

queue_*N*_tx_dma_mapping_err: *N*

直接記憶體存取錯誤計數。如果此值為 0，則指出系統資源不足。

queue_*N*_tx_linearize: *N*

此佇列嘗試SKB線性化的次數。

queue_*N*_tx_linearize_failed: *N*

此佇列SKB線性化失敗的次數。

queue_*N*_tx_napi_comp: *N*

napi 處理常式為此佇列呼叫 napi_complete 的次數。

queue_*N*_tx_tx_poll: *N*

napi 處理常式為此佇列排程的次數。

queue_*N*_tx_doorbells: *N*

此佇列的傳輸門鈴數目。

queue_*N*_tx_prepare_ctx_err: *N*

此佇列的 ena_com_prepare_tx 失敗次數。

queue_*N*_tx_bad_req_id: *N*

此佇列的無效 req_id。有效的 req_id 為零，減去 queue_size，減去 1。

queue_*N*_tx_llq_buffer_copy: *N*

其標頭大小大於此佇列 llq 項目的封包數目。

queue_*N*_tx_missed_tx: *N*

此佇列未完成的封包數目。

queue_*N*_tx_unmask_interrupt: *N*

此佇列 tx 中斷取消遮罩的次數。

queue_*N*_rx_cnt: *N*

此佇列接收的封包數目。

queue_*N*_rx_bytes: *N*

此佇列接收的位元組數目。

queue_*N*_rx_rx_copybreak_pkt: *N*

此佇列的 rx 佇列接收到小於 rx_copybreak 封包大小之封包的次數。

queue_*N*_rx_csum_good: *N*

此佇列的 rx 佇列接收到檢查總和已檢查且正確之封包的次數。

queue_*N*_rx_refil_partial: *N*

驅動程式未成功以此佇列緩衝區重新填滿 rx 佇列之空白部分的次數。如果值不為 0，則指出記憶體資源不足。

queue_*N*_rx_bad_csum: *N*

rx 佇列對此佇列有不良檢查總和的次數 (僅當支援 rx 檢查總和卸載時)。

queue_*N*_rx_page_alloc_fail: *N*

此佇列頁面配置失敗的次數。如果值不為 0，則指出記憶體資源不足。

queue_*N*_rx_skb_alloc_fail: *N*

此佇列SKB配置失敗的時間數。如果值不為 0，則指出系統資源不足。

queue_*N*_rx_dma_mapping_err: *N*

直接記憶體存取錯誤計數。如果此值為 0，則指出系統資源不足。

queue_*N*_rx_bad_desc_num: *N*

每個封包的緩衝區過多。如果值不為 0，則表示使用非常小的緩衝區。

queue_*N*_rx_bad_req_id: *N*

此佇列的 req_id 無效。有效的 req_id 是從 [0, queue_size - 1] 開始。

queue_*N*_rx_empty_rx_ring: *N*

此佇列的 rx 佇列為空白的次數。

queue_*N*_rx_csum_unchecked: *N*

此佇列的 rx 佇列接收到檢查總和尚未檢查之封包的次數。

queue_*N*_rx_xdp_aborted: *N*

XDP封包被歸類為 XDP _ 的次數ABORT。

queue_*N*_rx_xdp_drop: *N*

XDP封包被歸類為 XDP _ 的次數DROP。

queue_*N*_rx_xdp_pass: *N*

XDP封包被歸類為 XDP _ 的次數PASS。

queue_*N*_rx_xdp_tx: *N*

XDP封包被分類為 XDP _TX 的次數。

queue_*N*_rx_xdp_invalid: *N*

封包XDP傳回碼無效的次數。

queue_*N*_rx_xdp_redirect: *N*

XDP封包被歸類為 XDP _ 的次數REDIRECT。

queue_*N*_xdp_tx_cnt: *N*

此佇列的已傳輸封包數目。

queue_*N*_xdp_tx_bytes: *N*

此佇列的已傳輸位元組數目。

queue_*N*_xdp_tx_queue_stop: *N*

此佇列已滿並停止的次數。

queue_*N*_xdp_tx_queue_wakeup: *N*

此佇列在停止後恢復的次數。

queue_*N*_xdp_tx_dma_mapping_err: *N*

直接記憶體存取錯誤計數。如果此值為 0，則指出系統資源不足。

queue_*N*_xdp_tx_linearize: *N*

此佇列嘗試XDP緩衝區線性化的次數。

queue_*N*_xdp_tx_linearize_failed: *N*

此佇列的XDP緩衝區線性化失敗次數。

queue_*N*_xdp_tx_napi_comp: *N*

napi 處理常式為此佇列呼叫 napi_complete 的次數。

queue_*N*_xdp_tx_tx_poll: *N*

napi 處理常式為此佇列排程的次數。

queue_*N*_xdp_tx_doorbells: *N*

此佇列的傳輸門鈴數目。

queue_*N*_xdp_tx_prepare_ctx_err: *N*

此佇列的 ena_com_prepare_tx 失敗次數。此值應始終為 0；如不是，請查看驅動程式日誌。

queue_*N*_xdp_tx_bad_req_id: *N*

此佇列的 req_id 無效。有效的 req_id 是從 [0, queue_size - 1] 開始。

queue_*N*_xdp_tx_llq_buffer_copy: *N*

此佇列使用 llq 緩衝區複本複製其標頭的封包數目。

queue_*N*_xdp_tx_missed_tx: *N*

tx 佇列項目錯過此佇列完成逾時的次數。

queue_*N*_xdp_tx_unmask_interrupt: *N*

此佇列 tx 中斷取消遮罩的次數。

ena_admin_q_aborted_cmd: *N*

管理命令中止次數。此情況通常發生在自動復原程序中。

`ena_admin_q_submitted_cmd`: *N*

管理佇列門鈴數目。

`ena_admin_q_completed_cmd`: *N*

管理佇列完成數目。

`ena_admin_q_out_of_space`: *N*

驅動程式嘗試提交新管理命令但佇列已滿的次數。

`ena_admin_q_no_completion`: *N*

驅動程式未取得命令的管理完成之次數。

syslog 中的驅動程式錯誤日誌

ENA驅動程式會syslog在系統開機期間寫入記錄訊息。如果您遇到問題，可以檢查這些日誌以尋找錯誤。以下是ENA驅動程序在系統啟動syslog期間記錄的信息的示例，以及選擇消息的一些註釋。

```
Jun  3 22:37:46 ip-172-31-2-186 kernel: [  478.416939] [ENA_COM:
 ena_com_validate_version] ena device version: 0.10
Jun  3 22:37:46 ip-172-31-2-186 kernel: [  478.420915] [ENA_COM:
 ena_com_validate_version] ena controller version: 0.0.1 implementation version 1
Jun  3 22:37:46 ip-172-31-2-186 kernel: [  479.256831] ena 0000:00:03.0: Device
 watchdog is Enabled
Jun  3 22:37:46 ip-172-31-2-186 kernel: [  479.672947] ena 0000:00:03.0: creating 8 io
 queues. queue size: 1024
Jun  3 22:37:46 ip-172-31-2-186 kernel: [  479.680885] [ENA_COM:
 ena_com_init_interrupt_moderation] Feature 20 isn't supported  // Interrupt moderation
 is not supported by the device
Jun  3 22:37:46 ip-172-31-2-186 kernel: [  479.691609] [ENA_COM:
 ena_com_get_feature_ex] Feature 10 isn't supported // RSS HASH function configuration
 is not supported by the
 device
Jun  3 22:37:46 ip-172-31-2-186 kernel: [  479.694583] [ENA_COM:
 ena_com_get_feature_ex] Feature 18 isn't supported //RSS HASH input source
 configuration is not supported by the device
Jun  3 22:37:46 ip-172-31-2-186 kernel: [  479.697433] [ENA_COM:
 ena_com_set_host_attributes] Set host attribute isn't supported
Jun  3 22:37:46 ip-172-31-2-186 kernel: [  479.701064] ena 0000:00:03.0 (unnamed
 net_device) (uninitialized): Cannot set host attributes
Jun  3 22:37:46 ip-172-31-2-186 kernel: [  479.704917] ena 0000:00:03.0: Elastic
 Network Adapter (ENA) found at mem f3000000, mac addr 02:8a:3c:1e:13:b5 Queues 8
```

```
Jun  3 22:37:46 ip-172-31-2-186 kernel: [  480.805037] EXT4-fs (xvda1): re-mounted.
 Opts: (null)
Jun  3 22:37:46 ip-172-31-2-186 kernel: [  481.025842] NET: Registered protocol family
 10
```

我可以忽略哪些錯誤？

彈性網路轉接器可以忽略下列可能出現在系統錯誤日誌中的警告：

Set host attribute isn't supported

此裝置不支援主機屬性。

failed to alloc buffer for rx queue

此為可復原的錯誤，指出在錯誤拋出時可能存在記憶體壓力問題。

功能 *X* 不支援

參考的功能不受彈性網路轉接器支援。可能的值 *X* 包括：

- **10**：此設備不支持RSS哈希函數配置。
- **12**：此裝置不支援間RSS接表格組態。
- **18**：此裝置不支援RSS雜湊輸入組態。
- **20**：此裝置不支援中斷管制。
- **27**：彈性網路轉接器驅動程式不支援從 snmpd 中輪詢乙太網路功能。

設定失敗 AENQ

彈性網路介面卡不支援AENQ組態。

嘗試設定不支援的AENQ事件

此錯誤表示嘗試設定彈性網路介面卡不支援的AENQ事件群組。

非最佳化組態通知

ENA裝置會偵測驅動程式中您可以變更的次最佳組態設定。裝置會通知ENA驅動程式，並將警告記錄到主控台。下列範例顯示警告訊息的格式。

```
Sub-optimal configuration notification code: 1. Refer to AWS ENA documentation for
 additional details and mitigation options.
```

下列清單顯示通知程式碼詳細資訊，以及非最佳化組態調查結果的建議動作。

- 代碼**1**：不建議使用廣泛的LLQ配置ENA快遞

  ENA快ENI遞配置寬LLQ。此組態不是最佳化的，可能會影響 ENA Express 的效能。我們建議您在使用 ENA Express 時停用廣泛LLQ設定，ENIs如下所示。

  ```
  sudo rmmod ena && sudo modprobe ena force_large_llq_header=0
  ```

  如需 ENA Express 之最佳組態的詳細資訊，請參閱使用 ENA Express 改善EC2執行個體間的網路效能。

- 代碼**2**：不建ENI議使用次優 Tx 隊列深度ENA快遞

  ENAExpress 配ENI置了次最佳的 Tx 隊列深度。此組態可能會影響「ENA快速」的效能。當您使用 ENA Express 時，我們建議您將所有 Tx 佇列放大到網路介面的最大值，ENIs如下所示。

  您可以執行下列ethtool指令來調整LLQ大小。若要進一步了解如何控制、查詢和啟用廣泛LLQ，請參閱 Linux 核心驅動程式的大型低延遲佇列 (大型LLQ) 主題，以取得 Amazon Drivers GitHub 儲存庫中的ENA文件。

  ```
  ethtool -g interface
  ```

  將您的 Tx 隊列設置為最大深度：

  ```
  ethtool -G interface tx depth
  ```

  如需 ENA Express 之最佳組態的詳細資訊，請參閱使用 ENA Express 改善EC2執行個體間的網路效能。

- 代碼**3**：ENA具有常規大LLQ小和 Tx 數據包流量超過標頭支持的最大大小

  默認情況下，ENALLQ支持 Tx 數據包頭大小最大為 96 字節。如果封包標頭大小超過 96 個位元組，則會捨棄封包。為了緩解此問題，建議您啟用 wide-LLQ，這樣會將支援的 Tx 封包標頭大小增加到最多 224 個位元組。

  但是，當您啟用寬廣時LLQ，Tx 環的最大尺寸從 1000 減少到 512 個項目。所有 Nitro v4 及更新版本的執行個體類型預設為啟用寬廣功能。LLQ

  - Nitro v4 實例類型具有 512 個項目的默認最大寬 LLQ Tx 環大小，無法更改。
  - Nitro v5 執行個體類型的預設寬 LLQ Tx 環形大小為 512 個項目，最多可增加 1000 個項目。

您可以執行下列ethtool指令來調整LLQ大小。若要進一步了解如何控制、查詢和啟用廣泛LLQ，請參閱 Linux 核心驅動程式的大型低延遲佇列 (大型LLQ) 主題，以取得 Amazon Drivers GitHub 儲存庫中的ENA文件。

查找 Tx 隊列的最大深度：

```
ethtool -g interface
```

將您的 Tx 隊列設置為最大深度：

```
ethtool -G interface tx depth
```

排除彈性網路介面卡 Windows 驅動程式

彈性網路介面卡 (ENA) 旨在改善作業系統健康狀況，並減少意外的硬體行為或可能中斷 Windows 執行個體作業的故障。該ENA架構使設備或驅動程序故障對操作系統盡可能透明。

收集有關執行個體的診斷資訊

開啟 Windows 作業系統 (OS) 工具的步驟會有所不同，具體取決於您執行個體上安裝的作業系統版本。在以下各部分，我們使用 Run (執行) 對話方塊開啟工具，該工具在所有作業系統版本的操作方式均相同。但是，您可以使用偏好的任何方法來存取這些工具。

存取 Run (執行) 對話方塊

- 使用 Windows 標誌鍵組合：`Windows + R`
- 使用搜尋列：
  - 在搜尋列中，輸入 `run`。
  - 從搜尋結果中選擇 Run (執行) 應用程式。

某些步驟需要內容功能表才能存取屬性或內容相關動作。根據您的作業系統版本和硬體，可透過多種方式執行此作業。

存取內容功能表

- 使用滑鼠：在項目上按一下滑鼠右鍵以顯示其內容功能表。
- 使用鍵盤：

- 根據您的作業系統版本，使用 `Shift + F10` 或 `Ctrl + Shift + F10`。
- 如果鍵盤上有內容鍵 (方塊中帶有三條水平線)，請選擇所需項目，然後按內容鍵。

如果可以連接到自己的執行個體，請使用以下技術收集診斷資訊以進行故障診斷。

檢查ENA裝置狀態

若要使用 Windows 裝置管理員檢查 ENA Windows 驅動程式的狀態，請依照下列步驟執行：

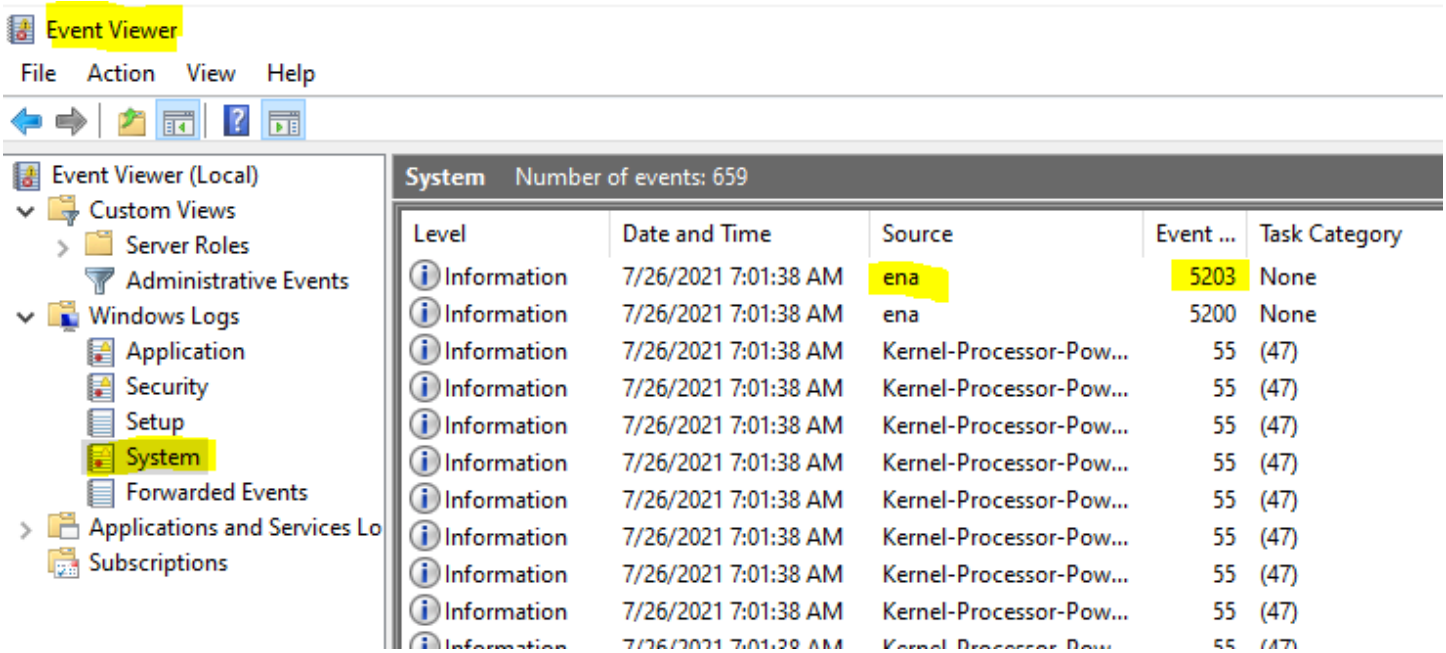
1. 使用上一部分中描述的其中一種方法開啟 Run (執行) 對話方塊。
2. 若要開啟 Windows Device Manager (Windows 裝置管理員)，請在 Run (執行) 對話方塊中輸入 `devmgmt.msc`。
3. 選擇確定。這將開啟 Device Manager (裝置管理員) 視窗。
4. 選取 Network adapters (網路轉接器) 左側的箭頭以展開清單。
5. 選取名稱，或開啟 Amazon Elastic Network Adapter (Amazon 彈性網路轉接器) 的內容功能表，然後選擇 Properties (屬性)。這會開啟 Amazon 彈性網路介面卡屬性對話方塊。
6. 確認一般索引標籤中的訊息顯示「此裝置運作正常」。

調查驅動程式事件訊息

若要使用 ENA Windows 事件檢視器檢視 Windows 驅動程式事件記錄檔，請依照下列步驟執行：

1. 使用上一部分中描述的其中一種方法開啟 Run (執行) 對話方塊。
2. 若要開啟 Windows 事件檢視器，請在 Run (執行) 對話方塊中輸入 `eventvwr.msc`。
3. 選擇確定。Event Viewer (事件檢視器) 視窗隨即開啟。
4. 展開 Windows Logs (Windows 日誌) 功能表，然後選擇 System (系統)。
5. 在右上方面板中的 Action (動作) 下，選擇 Filter Current Log (篩選當前日誌)。此時將顯示篩選對話方塊。
6. 在 Event sources (事件來源) 方塊中，輸入 `ena`。這會將結果限制為 ENA Windows 驅動程式所產生的事件。
7. 選擇確定。此時將在視窗的詳細資訊部分顯示篩選的事件日誌結果。
8. 要深入查看詳細資訊，請從清單中選擇一條事件訊息。

下列範例顯示 Windows 事件檢視器系統事件清單中的ENA驅動程式事件：



事件訊息摘要

下表顯示 ENA Windows 驅動程式所產生的事件訊息。

輸入

事件 ID	ENA驅動程式事件說	Type
5001	硬體資源不足	錯誤
5002	轉接器偵測到硬體錯誤	錯誤
5005	介面卡的NDIS作業逾時，但未及時完成	錯誤
5032	轉接器未能重設裝置	錯誤
5200	已初始化轉接器	資訊
5201	已停止轉接器	資訊
5202	已暫停轉接器	資訊
5203	已重新啟動轉接器	資訊
5204	已關閉轉接器	資訊

事件 ID	ENA驅動程式事件說	Type
5205	已重設轉接器	錯誤
5206	已意外刪除轉接器	錯誤
5208	轉接器初始化常式失敗	錯誤
5210	轉接器遇到內部問題並已成功恢復	錯誤

檢閱效能指標

ENAWindows 驅動程式會從啟用測量結果的執行個體發佈網路效能測量結果。您可以使用本機效能監控應用程式檢視和啟用執行個體上的指標。如需 ENA Windows 驅動程式產生之度量的詳細資訊，請參閱〈〉監控EC2執行個體ENA設定的網路效能。

在已啟用ENA指標且已安裝 Amazon CloudWatch 代理程式的執行個體上， CloudWatch會收集與 Windows 效能監視器中計數器相關聯的指標，以及的一些進階指標ENA。除了EC2執行處理預設啟用的測量結果外，還會收集這些測量結果。如需有關指標的詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的 CloudWatch 代理程式收集的指標。

> **Note**
>
> 效能指標適用於 2.4.0 及更新版本的ENA驅動程式 (也適用於版本 2.2.3)。ENA驅動程式版本 2.2.4 已回復，因為第六代EC2執行個體可能會降低效能。建議您升級至驅動程式的目前版本，確保您擁有最新的更新項目。

使用效能指標的一些方式包括：

- 對執行個體效能問題執行故障診斷。
- 為工作負載選擇合適的執行個體大小。
- 主動規劃擴展活動。
- 基準化應用程式，以確定它們是否最大限度地提高了執行個體上可用的效能。

重新整理速率

根據預設，驅動程式使用 1 秒的間隔重新整理指標。但是，擷取指標的應用程式可能會使用不同的間隔進行輪詢。您可以使用驅動程式的進階屬性，在 Device Manager (裝置管理員)中變更重新整理間隔。

若要變更 ENA Windows 驅動程式的指標重新整理間隔，請依照下列步驟執行：

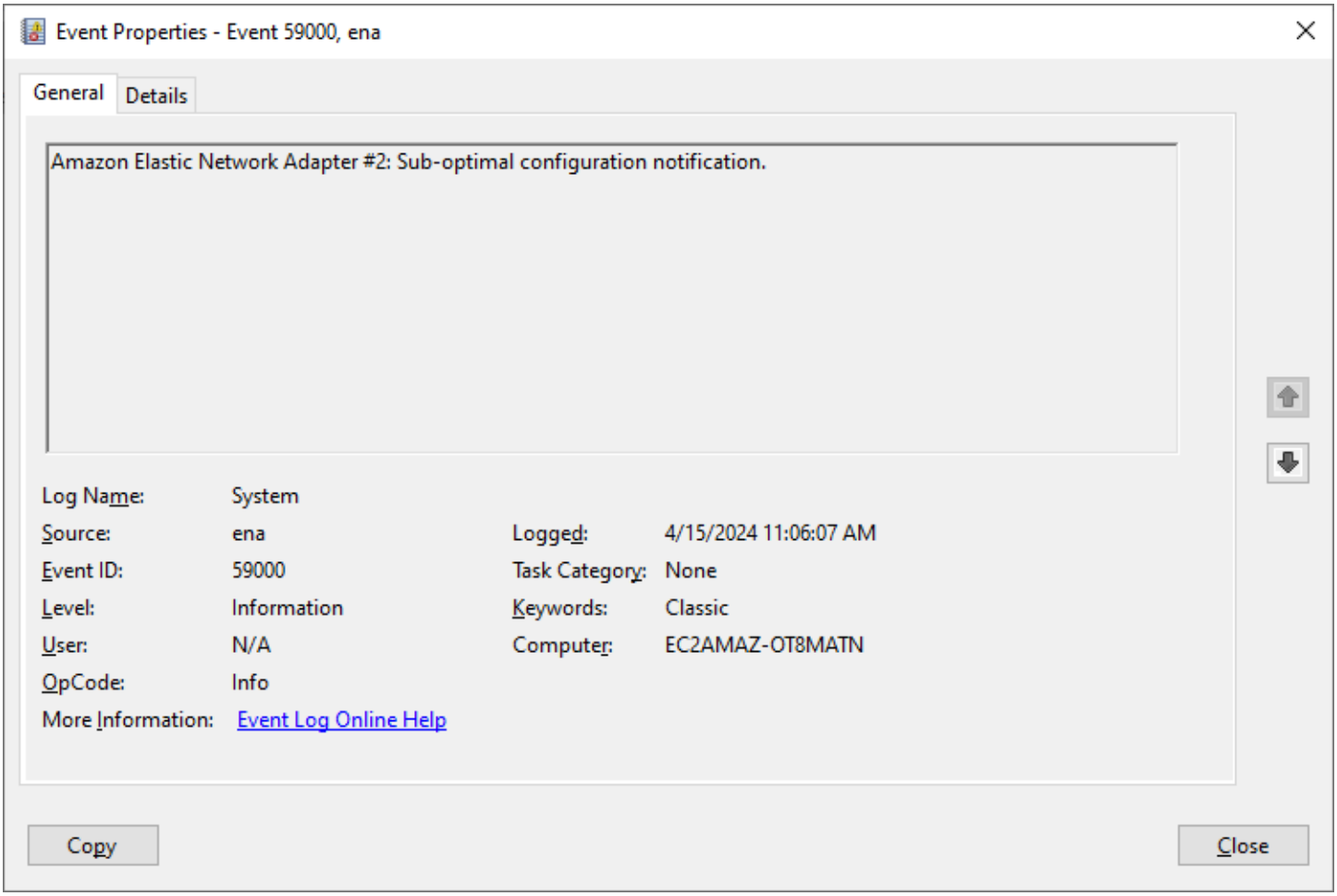
1. 使用上一部分中描述的其中一種方法開啟 Run (執行) 對話方塊。
2. 若要開啟 Windows Device Manager (Windows 裝置管理員)，請在 Run (執行) 對話方塊中輸入 `devmgmt.msc`。
3. 選擇確定。這將開啟 Device Manager (裝置管理員) 視窗。
4. 選取 Network adapters (網路轉接器) 左側的箭頭以展開清單。
5. 選取名稱，或開啟 Amazon Elastic Network Adapter (Amazon 彈性網路轉接器) 的內容功能表，然後選擇 Properties (屬性)。這會開啟 Amazon 彈性網路介面卡屬性對話方塊。
6. 在彈出式視窗中開啟 Advanced (進階) 索引標籤。
7. 從 Property (屬性) 清單中，選擇 Metrics Refresh Interval (指標重新整理間隔) 以變更值。
8. 完成後，選擇 OK (確定)。

調查次最佳組態通知

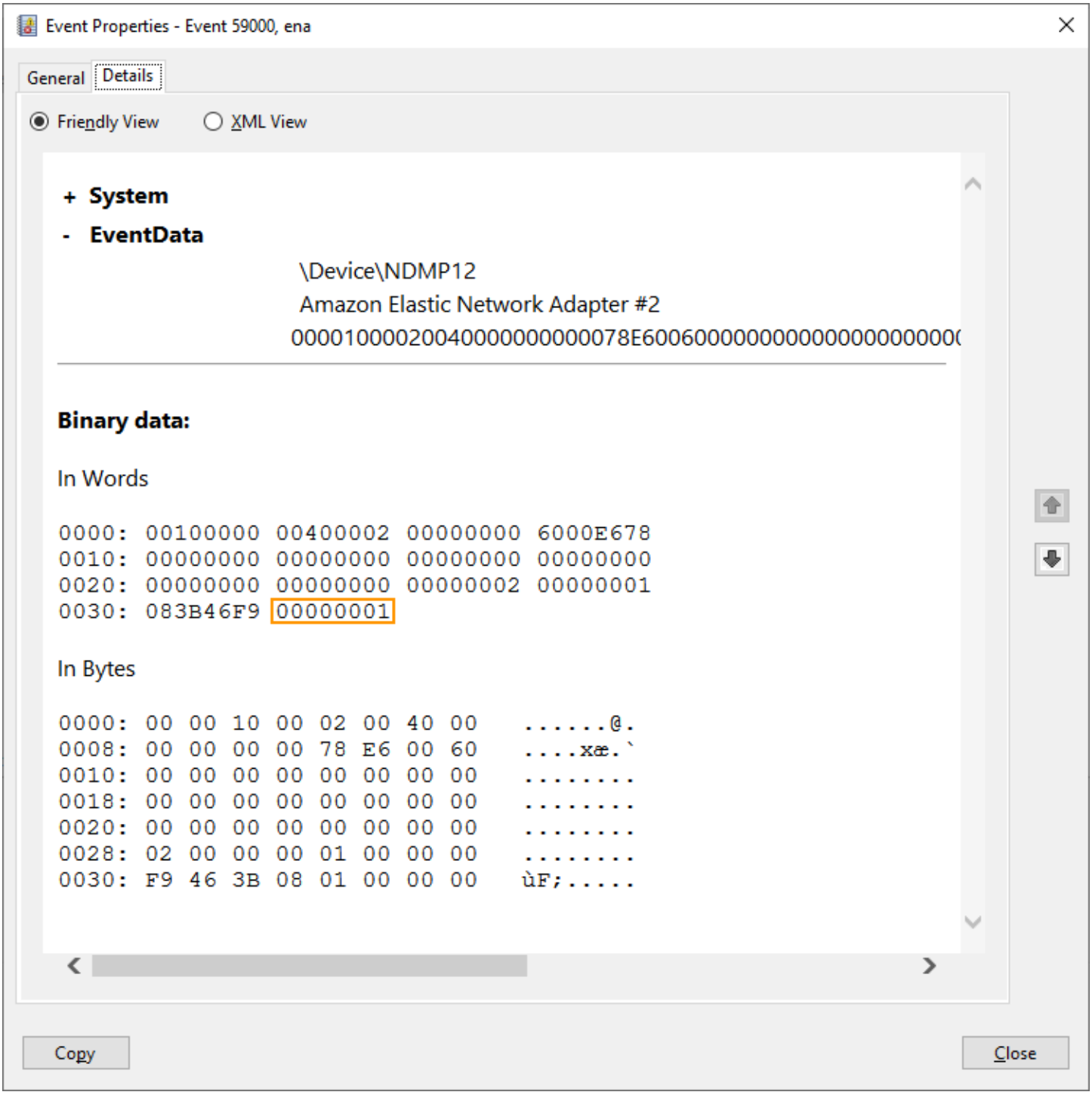
ENA裝置會偵測驅動程式中您可以變更的次最佳組態設定。裝置會通知ENA驅動程式並記錄事件通知。若要在 Windows 事件檢視器中檢閱次最佳化事件

1. 使用上一部分中描述的其中一種方法開啟 Run (執行) 對話方塊。
2. 若要開啟 Windows 事件檢視器，請在 Run (執行) 對話方塊中輸入 `eventvwr.msc`。
3. 選擇確定。Event Viewer (事件檢視器) 視窗隨即開啟。
4. 展開 Windows Logs (Windows 日誌) 功能表，然後選擇 System (系統)。
5. 在右上方面板中的 Action (動作) 下，選擇 Filter Current Log (篩選當前日誌)。此時將顯示篩選對話方塊。
6. 在 Event sources (事件來源) 方塊中，輸入 `ena`。這會將結果限制為 ENA Windows 驅動程式所產生的事件。
7. 選擇確定。此時將在視窗的詳細資訊部分顯示篩選的事件日誌結果。

具有 ID 的事件會`59000`通知您有次最佳組態發現項目。在事件上按一下滑鼠右鍵，然後選擇「事件內容」以開啟詳細檢視，或從「檢視」功能表選取「預覽窗格」 以查看相同的詳細



開啟 [詳細資料] 索引標籤以查看事件代碼。在二進制數據：在單詞部分，最後一個單詞是代碼。



下列清單顯示通知程式碼詳細資訊，以及非最佳化組態調查結果的建議動作。

- 代碼**1**：不建議使用廣泛的LLQ配置ENA快遞

  ENA快ENI遞配置寬LLQ。此組態不是最佳化的，可能會影響 ENA Express 的效能。我們建議您在使用 ENA Express 時停用廣泛LLQ設定，ENIs如下所示。

1. 若要開啟 Windows Device Manager (Windows 裝置管理員)，請在 Run (執行) 對話方塊中輸入 `devmgmt.msc`。
2. 選擇確定。這將開啟 Device Manager (裝置管理員) 視窗。
3. 選取 Network adapters (網路轉接器) 左側的箭頭以展開清單。
4. 開啟的裝置內容`Amazon Elastic Network Adapter`。
5. 從那裡開啟 [進階] 索引標籤以進行變更。
6. 選取「LLQ標頭大小策略」屬性，並將其值設定為`Normal (128 Bytes)`。
7. 選擇 OK (確定) 以儲存變更。

- 代碼**2**：不建ENI議使用次優 Tx 隊列深度ENA快遞

  ENAExpress 配ENI置了次最佳的 Tx 隊列深度。此組態可能會影響「ENA快速」的效能。當您使用 ENA Express 時，我們建議您將所有 Tx 佇列放大到網路介面的最大值，ENIs如下所示。

  請按照以下步驟將 Tx 隊列擴大到最大深度：

  1. 若要開啟 Windows Device Manager (Windows 裝置管理員)，請在 Run (執行) 對話方塊中輸入 `devmgmt.msc`。
  2. 選擇確定。這將開啟 Device Manager (裝置管理員) 視窗。
  3. 選取 Network adapters (網路轉接器) 左側的箭頭以展開清單。
  4. 開啟的裝置內容`Amazon Elastic Network Adapter`。
  5. 從那裡開啟 [進階] 索引標籤以進行變更。
  6. 選取「傳輸緩衝區」內容，並將其值設定為支援的最大值。
  7. 選擇 OK (確定) 以儲存變更。

ENA介面卡重設

當 ENA Windows 驅動程式偵測到介面卡上的錯誤，並將介面卡標示為狀態不良時，重設程序便會啟動。驅動程式無法自行重設，因此需視作業系統來檢查介面卡健康狀態，並呼叫 ENA Windows 驅動程式的重設控點。重設程序可能會導致短時間內出現流量遺失。但是，TCP連接應該能夠恢復。

ENA介面卡也可能會間接要求裝置重設程序，因為無法傳送保持連線通知。例如，如果ENA介面卡在載入無法復原的組態之後到達未知狀態，它可能會停止傳送持續使用中通知。

ENA介面卡重設的常見原因

- 缺少保持有效訊息

  ENA介面卡會以固定速率 (通常每秒一次) 張貼保持活動的事件。ENAWindows 驅動程式會實作監視程式機制，該機制會定期檢查這些保持活動訊息是否存在。如果自上次檢查以來偵測到一條或多條新訊息，它就會記錄成功結果。否則，驅動程式得出裝置出現故障的結論，並啟動重設序列。

- 封包卡在傳輸佇列中

  ENA介面卡會確認封包是否如預期流經傳輸佇列。ENAWindows 驅動程式會偵測封包是否卡住，並啟動重設順序 (如果是)。

- 記憶體對應 I/O (MMIO) 暫存器的讀取逾時

  若要限制記憶體對應 I/O (MMIO) 讀取作業，ENAWindows 驅動程式只會在初始化和重設程序期間存取MMIO暫存器。如果驅動程式偵測到逾時，它將執行以下動作之一，具體取決於正在執行的程序：

  - 如果在初始化期間偵測到逾時，流程會失敗，導致驅動程式在 Windows 裝置管理員中的ENA介面卡顯示黃色驚嘆號。
  - 如果在重設過程中偵測到逾時，則會讓流程失敗。然後，作業系統會啟動ENA介面卡的意外移除，並透過停止並啟動已移除的介面卡來復原它。如需有關意外移除網路介面卡 (NIC) 的詳細資訊，請參閱 Microsoft Windows 硬體開發人員說明文件NIC中的處理意外移除。

故障診斷方案

下列案例可協助您疑難排解 ENA Windows 驅動程式可能遇到的問題。如果您沒有最新版本，我們建議您先升級ENA驅動程式。若要尋找 Windows 作業系統版本的最新驅動程式，請參閱 追蹤ENA驅動程式版本。

未預期的ENA驅動程式版

描述

完成安裝特定版本ENA驅動程式的步驟之後，Windows 裝置管理員會顯示 Windows 安裝了不同版本的ENA驅動程式。

原因

當您執行驅動程式套件的安裝時，Windows 會在開始安裝之前，對本機驅動程式存放區中指定裝置的全部有效驅動程式套件進行排名。然後其會選取最低等級值的套件作為最佳相符項目。這可能與您要

安裝的套件不同。如需有關裝置驅動程式套件選取程序的詳細資訊，請參閱 Microsoft 文件網站上的 Windows 如何為裝置選取驅動程式套件。

解決方案

若要確保 Windows 安裝您選擇的驅動程式套件版本，您可以使用 P nPUtil 命令列工具從驅動程式存放區移除排名較低的驅動程式套件。

請依照下列步驟更新ENA驅動程式：

1. 連線到您的執行個體，並以本機管理員登入。
2. 開啟 Device Manager (裝置管理員) 屬性視窗，如 檢查ENA裝置狀態 部分中所述。這會開啟 Amazon 彈性網路介面卡屬性視窗的一般索引標籤。
3. 開啟 Driver (驅動程式) 索引標籤。
4. 選擇 Update Driver (更新驅動程式)。這會開啟更新驅動程式軟體 – Amazon 彈性網路介面卡對話方塊。
   a. 在您要如何搜尋驅動程式軟體？區段中，選擇瀏覽電腦上的驅動程式軟體。
   b. 在瀏覽電腦上的驅動程式軟體頁面中，選擇位於搜尋欄下方的讓我從電腦上的裝置驅動程式清單挑選。
   c. 在選取您要為此硬體安裝的裝置驅動程式頁面中，選擇擁有磁碟...。
   d. 在從磁碟安裝視窗中，選擇下拉式清單中檔案位置旁邊的瀏覽...。
   e. 瀏覽至您下載目標ENA驅動程式套件的位置。選取名為 `ena.inf` 的檔案，然後選擇開啟。
   f. 若要開始安裝，請選擇確定，然後選擇下一步。
5. 如果安裝程式未自動重新啟動執行個體，請執行指Restart-Computer PowerShell 令程式。

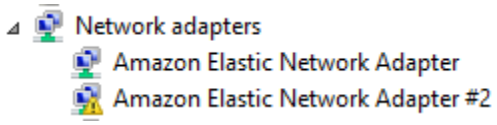
```
PS C:\> Restart-Computer
```

ENA驅動程序的設備警告

描述

裝置管理員網路介面卡區段中的介面卡圖示會顯示警告標誌 (內部有驚嘆號的黃色三角形)。ENA

下列範例會顯示 Windows 裝置管理員中含有警告圖示的ENA介面卡：



原因

此裝置警告通常由環境問題引起，此問題可能需要進行更多的研究，並且通常需要執行消除程序來確定根本原因。如需裝置錯誤的完整清單，請參閱 Microsoft Windows 硬體開發人員文件中的 Device Manager (裝置管理員) 錯誤訊息。

解決方案

此裝置警告的解決方案取決於根本原因。此處介紹的消除程序包括幾個基本步驟，用於幫助確定和解決可能具有簡單解決方案的最常見問題。如果這些步驟無法解決問題，則需要進行其他根本原因分析。

請按照以下步驟幫助確定和解決常見問題：

1. 停止並啟動裝置。

   開啟 Device Manager (裝置管理員) 屬性視窗，如 檢查ENA裝置狀態 部分中所述。這將開啟 Amazon Elastic Network Adapter Properties (Amazon 彈性網路轉接器屬性) 視窗中的 General (常規) 索引標籤，其中 Device status (裝置狀態) 顯示錯誤代碼和一條簡短訊息。

   a. 開啟 Driver (驅動程式) 索引標籤。
   b. 選擇 Disable Device (停用裝置)，在顯示的警告訊息中回應 Yes (是)。
   c. 選擇 Enable Device (啟用裝置)。
2. 停止並啟動EC2執行個體

   如果介面卡在裝置管理員中仍顯示警告圖示，下一個步驟就是停止並啟動EC2執行個體。在大多數情況下，這會在不同硬體上重新啟動執行個體。
3. 調查可能的執行個體資源問題

   如果您已停止並啟動EC2執行個體，但問題仍然存在，這可能表示執行個體出現資源問題，例如記憶體不足。

轉接器重設時連接逾時 (錯誤代碼 5007、5205)

描述

Windows 事件檢視器會顯示介面卡逾時和重設事件組合發生的ENA介面卡。消息類似以下範例：

- 事件 ID 5007：Amazon 彈性網路轉接器：操作過程中逾時。
- 事件 ID 5205：Amazon 彈性網路轉接器：轉接器重設已啟動。

轉接器重設會導致最小規模的流量中斷。即使執行多個重設，它們也很少會導致任何嚴重的網路中斷。

原因

此事件順序表示 ENA Windows 驅動程式對沒有回應的ENA介面卡啟動了重設。但是，設備驅動程序用於檢測此問題的機制可能會導致 CPU 0 飢餓導致的誤報。

解決方案

如果這種錯誤組合頻繁發生，請檢查您的資源分配，以了解調整可能發揮作用的位置。

1. 使用上一部分中描述的其中一種方法開啟 Run (執行) 對話方塊。
2. 若要開啟 Windows Resource Monitor (資源監視器)，請在 Run (執行) 對話方塊中輸入 `resmon`。
3. 選擇確定。這將開啟 Resource Monitor (資源監視器) 視窗。
4. 開啟索CPU引標籤。「資源監視器」視窗右側會顯示每個CPU使用情況圖表。
5. 檢查 CPU 0 的使用層級，看看它們是否太高。

建議您設定為針RSS對較大執行個體類型 (超過 16 vCPU) 的ENA介面卡排除 CPU 0。對於較小的執行個體類型，設定RSS可能會改善使用體驗，但由於可用核心數量較少，因此需要進行測試，以確保限制CPU核心不會對效能造成負面影響。

使用指Set-NetAdapterRss令來設定您RSS的ENA介面卡，如下列範例所示。

```
Set-NetAdapterRss -name (Get-NetAdapter | Where-Object {$_.InterfaceDescription -like
 "*Elastic*"}).Name -Baseprocessorgroup 0 -BaseProcessorNumber 1
```

遷移到第六代執行個體基礎設施會影響效能或配件

描述

如果您移轉至第六代EC2執行個體，如果尚未更新 ENA Windows 驅動程式版本，則可能會遇到效能降低或ENA附件失敗的情況。

原因

根據EC2執行個體作業系統 (OS)，第六代執行個體類型需要以下最低版本的 ENA Windows 驅動程式。

最低版本

Windows Server 版本	ENA 驅動程式版本
Windows Server 2008 R2	2.2.3 或 2.4.0
Windows Server 2012 及更新版本	2.2.3 和更新版本
Windows 工作站	2.2.3 和更新版本

解決方案

升級至第六代EC2執行個體之前，請確定AMI您從中啟動的執行個體具有以執行個體作業系統為基礎的相容驅動程式，如上表所示。如需詳細資訊，請參閱將執行個體遷移到第六代EC2執行個體之前，我需要執行哪些動作，以確保我獲得最大的網路效能？ 在AWS re:Post 知識中心。

彈性網路介面效能不佳

描述

ENA介面未如預期般執行。

原因

效能問題的根本原因分析是一種消除程序。其中涉及太多的變數，無法指定常見原因。

解決方案

根本原因分析的第一步是檢閱未按預期發揮效能的執行個體的診斷資訊，以確定是否存在可能導致問題的錯誤。如需詳細資訊，請參閱 收集有關執行個體的診斷資訊 一節。

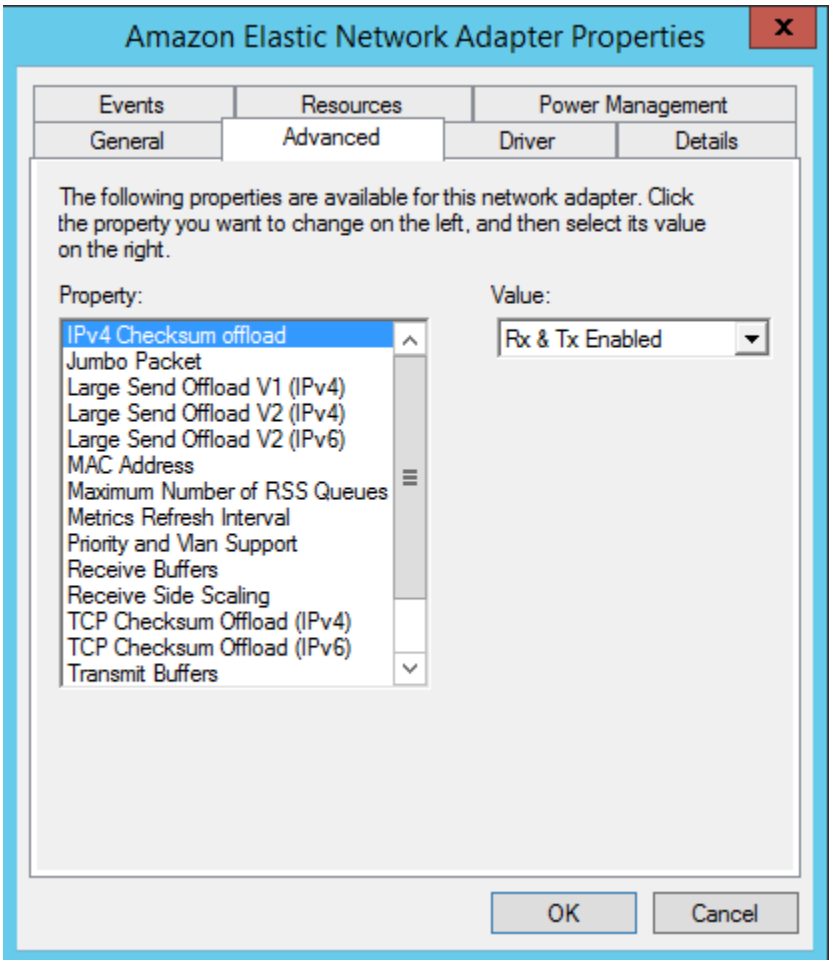
為了在執行個體上達到使用增強型聯網的最大網路效能，您可能需要修改預設的作業系統組態。某些最佳化，例如開啟總和檢查碼卸載和啟用RSS，預設會在官方 Windows 中設定。AMIs如需可套用至ENA介面卡的其他最佳化，請參閱中ENA介面卡效能調整顯示的效能調整。

我們建議您謹慎進行，並將裝置內容調整限制為本節所列的內容，或是 AWS 支援團隊建議的特定變更。

若要變更ENA介面卡內容，請遵循下列步驟：

1. 使用上一部分中描述的其中一種方法開啟 Run (執行) 對話方塊。
2. 若要開啟 Windows Device Manager (Windows 裝置管理員)，請在 Run (執行) 對話方塊中輸入 `devmgmt.msc`。
3. 選擇確定。這將開啟 Device Manager (裝置管理員) 視窗。
4. 選取 Network adapters (網路轉接器) 左側的箭頭以展開清單。
5. 選取名稱，或開啟 Amazon Elastic Network Adapter (Amazon 彈性網路轉接器) 的內容功能表，然後選擇 Properties (屬性)。這會開啟 Amazon 彈性網路介面卡屬性對話方塊。
6. 若要進行變更，請開啟進階索引標籤。
7. 完成後，請選擇確定以儲存變更。

下列範例會顯示 Windows 裝置管理員中的ENA介面卡內容：



ENA介面卡效能調整

下表包含可調整以改善ENA介面效能的屬性。

輸入

屬性	描述	預設值	調整
接收緩衝區	控制軟體接收佇列中的項目數目。	1024	可提高至上限 8192。
接收端調整比例 (RSS)	CPUs在多處理器系統中，能夠有效率地將網路接收處理分配到多重處理器。	已啟用	您可以將負載分散到多個處理器中。如需進一步了解，請參閱最佳化 EC2 Windows 執行個體的網路效能。
RSS佇列數目上限	設定啟用時RSS允許的最大RSS佇列數目。	32	RSS佇列數目是在驅動程式初始化期間決定的，包括下列限制 (除其他外)： • RSS此屬性設定的佇列限制 • 執行個體限制 (v CPU 計數) • 硬體產生限制 (最多 8 個RSS佇列輸入ENAv1，最多 32 個RSS佇列中ENAv2) 根據執行個體和硬體世代限制，您可以設定 1-32 的值。如需進一步了解，請參閱最

屬性	描述	預設值	調整
			佳化 EC2 Windows 執行個體的網路效能。
巨型封包	允許使用巨型乙太網路訊框 (超過 1500 個位元組的承載)。	已停用 (這將承載限制為 1500 個位元組或更少)	值可以設定為最高 9015，即轉換為 9001 個位元組的承載。這是巨型乙太網路訊框的最大承載。請參閱使用巨型乙太網路訊框的考量事項。

使用巨型乙太網路訊框的考量事項

巨型訊框可增加單一封包的承載大小，這會增加不屬於封包成本的封包比例，藉此允許超過 1500 位元組的資料。傳送相同數量的可用資料，所需的封包數即會減少。但是，在以下情況下，流量限制為最多 MTU 1500：

- EC2傳統版特定 AWS 區域以外的流量。
- 單一之外的流量VPC。
- 通過區域間對VPC等連接的流量。
- 通過VPN連接的流量。
- 網際網路閘道的流量。

> **Note**
>
> 超過 1500 個位元組的封包會分段。如果 IP 標頭設有 `Don't Fragment` 旗標，這些封包將被捨棄。
>
> Jumbo 框架應謹慎用於連結網際網路的流量，或任何離開. VPC 封包由中介系統切割，因此拖累流量速度。若要在不影響離開的輸出流量的VPC情況下使用 Jumbo 框架VPC，請嘗試下列其中一個選項：
>
> - 按路由配置MTU大小。
> - 使用具有不同MTU大小和不同路由的多個網絡接口。

巨型訊框的建議使用案例

Jumbo 框架對於內部和之間的流量非常有用VPCs。我們建議在以下使用案例中使用巨型訊框：

- 以配置於叢集置放群組內的執行個體而言，巨型訊框有助實現最大網路傳輸量。如需詳細資訊，請參閱 Amazon EC2 執行個體的放置群組。
- 您可以使用 Jumbo 框架來處理您VPCs與內部部署網路之間的流量 AWS Direct Connect。如需有關使用 AWS Direct Connect和驗證 Jumbo 框架功能的詳細資訊，請參閱《使MTU用指南》中的〈設定私有虛擬介面的網路或傳輸虛擬介面〉。AWS Direct Connect
- 如需傳輸閘道支援MTU大小的詳細資訊，請參閱 Amazon Transit 閘道中傳VPC輸閘道的配額。

改善 Linux EC2 執行個體的網路延遲

網路延遲是指資料封包從其來源傳送到目的地所需的時間。透過網路傳送資料的應用程式仰賴及時回應為使用者提供正面體驗。高網路延遲可能會導致各種問題，例如：

- 網頁載入時間緩慢
- 影片串流延遲
- 難以存取線上資源

本節概述了在 Linux 上執行的 Amazon EC2 執行個體上改善網路延遲時可採取的步驟。若要達到最佳延遲，請依照下列步驟設定執行個體、核心和ENA驅動程式設定。如需其他組態指引，請參閱上的《ENALinux 驅動程式最佳作法和效能最佳化指南》GitHub。

> **Note**
>
> 視您的特定網路硬體、啟動執行個體的來源以及應用程式使用案例而定，步驟和設定可能會略有不同。AMI變更之前，請徹底測試並監控網路效能，確保可獲得理想結果。

減少資料封包的網路躍點數

資料封包在從路由器之間移動時所經過的每個躍點都會增加網路延遲。通常，流量必須經過多個躍點才能到達目的地。有兩種方法可以減少 Amazon EC2 執行個體的網路躍點，如下所示：

- 叢集放置群組 — 當您指定叢集置放群組時，Amazon 會以更緊密的封裝方式EC2啟動彼此相近、實際位於相同可用區域 (AZ) 內的執行個體。群組中執行個體的實體接近性可讓其充分利用高速連線，進而實現低延遲和高單流輸送量。

- 專用執行個體–專用執行個體是專供您使用的實體伺服器。使用專用執行個體，可以啟動執行個體在相同實體伺服器上執行。在相同專用執行個體上執行的執行個體之間的通訊可以不需要額外的躍點。

Linux 內核配置如何影響延遲

Linux 核心組態可以增加或減少網路延遲。若要實現延遲最佳化目標，請務必根據工作負載的特定需求對 Linux 核心組態進行微調。

Linux 核心的許多組態選項可能有助於降低網路延遲。最具影響力的選項如下所示。

- 啟用忙碌輪詢模式 — 忙碌輪詢模式可減少網路接收路徑的延遲。若啟用忙碌輪詢模式，通訊端層代碼可以直接輪詢網路裝置的接收佇列。忙碌輪詢的缺點是在緊密循環中輪詢新數據的主機中CPU使用率更高。有兩個全域設定可控制等待所有介面封包的微秒數。

  `busy_read`

  通訊端讀取的低延遲忙碌輪詢逾時。這可控制等待通訊端層讀取裝置佇列上封包的微秒數。若要使用 sysctl 命令全域啟用此功能，Linux 核心組織建議使用 50 微秒的值。如需詳細資訊，請參閱 Linux 核心使用者和管理員指南中的 busy_read。

  ```
  $ C:\> sudo sysctl -w net.core.busy_read=50
  ```

  `busy_poll`

  輪詢和選取的低延遲忙碌輪詢逾時。這可控制等待事件的微秒數。建議值介於 50-100 微秒之間，具體取決於要輪詢的通訊端數目。新增的通訊端越多，數值越高。

  ```
  $ C:\> sudo sysctl -w net.core.busy_poll=50
  ```

- 配置CPU電源狀態（C 狀態）— C 狀態控制核心處於非活動狀態時可能進入的睡眠級別。建議您控制 C-state 來微調系統的延遲與效能。在更深層的 C 狀態中，基本上CPU是「睡眠」，並且在喚醒並轉換回使用中狀態之前無法回應要求。核心進入休眠狀態需要時間，而且雖然休眠核心讓其他核心更有餘裕加速至更高頻率，但是欲喚醒該休眠核心並開始執行工作也需要時間。

  例如，若負責處理網路封包中斷的核心處於休眠狀態，則處理此類中斷的服務就可能受到延遲。您可以設定系統，使其不會使用更深層的 C-state。不過，雖然此組態可能會減少處理器反應延遲，但同時也減少了 Turbo Boost 的其他核心可用預留空間。

若要減少處理器反應延遲，您可以限制更深層的 C-state。如需詳細資訊，請參閱 Amazon Linux 2 使用者指南中的限制更深 C 狀態的高效能和低延遲。

ENA網路驅動程式組

ENA網路驅動程式可啟用執行個體與網路之間的通訊。驅動程式會處理網路封包，並將它們傳遞至網路堆疊或 Nitro 卡。當網路封包進入時，Nitro 卡會產生中斷，CPU以通知軟體發生事件。

插斷

中斷是裝置或應用程式傳送給處理器的訊號。中斷會通知處理器已發生事件或已滿足需要立即注意的條件。中斷可以處理時間敏感任務，例如從網絡介面接收資料、處理硬體事件或服務來自其他裝置的請求。

插斷仲裁

中斷調節是一種技術，可以透過彙總或延遲來減少裝置產生的中斷次數。中斷調節的目的是透過減少處理大量中斷所產生的額外負荷來改善系統效能。中斷次數過多會增加CPU使用量，對輸送量造成不利影響，而中斷次數太少會增加延遲。

動態中斷調節

動態中斷調節是一種增強型的中斷調節形式，可根據目前的系統負載和流量模式動態調整中斷速率。其目標是在減少中斷負荷和每秒封包數或頻寬之間取得平衡。

> **Note**
>
> 動態岔斷協調在某些情況下默認啟用AMIs（但可以啟用或禁用所有AMIs）。

為了將網路延遲降到最低，可能需要停用中斷調節。但是這也可能會增加插斷處理的開銷。在減少延遲和盡可能減少開銷之間找到適當的平衡點非常重要。`ethtool` 命令可協助您設定插斷仲裁。依預設，`rx-usecs` 設定為 `20`，`tx-usecs` 設定為 `64`。

若要取得目前的插斷仲裁設定，請使用下列命令。

```
$ C:\> ethtool -c interface | egrep "rx-usecs:|tx-usecs:|Adaptive RX"
Adaptive RX: on  TX: off
rx-usecs: 20
tx-usecs: 64
```

若要停用中斷修改和動態中斷調節，請使用下列命令。

```
$ C:\> sudo ethtool -C interface adaptive-rx off rx-usecs 0 tx-usecs 0
```

對於性能調整的硝基系統考量

Nitro 系統結合一系列由 AWS 建立的硬體和軟體元件，能保障高效能、高可用性和高安全性。Nitro System 提供類似裸機的功能，可消除虛擬化額外負荷，並支援需要完全存取主機硬體的工作負載。有關更多詳細信息，請參閱AWS 硝基系統。

目前所有的執行EC2個體類型都會在 EC2 Nitro 卡上執行網路封包處理。本主題涵蓋 Nitro 卡上的高階封包處理、影響封包處理效能的網路架構和組態的常見方面，以及您可以採取哪些動作來達到 Nitro 型執行個體的峰值效能。

Nitro 卡處理所有輸入和輸出（I/O）接口，例如虛擬私有雲（VPCs）所需的接口。對於透過網路傳送或接收資訊的所有元件，Nitro 卡會充當 I/O 流量的獨立運算裝置，實際上與執行客戶工作負載的系統主機板分開。

硝基卡上的網絡數據包流

EC2建立在 Nitro 系統上的執行個體具有硬體加速功能，可實現更快的封包處理速度，這些功能是以每秒封包數 (PPS) 傳輸率來衡量的。當 Nitro 卡對新流程執行初始評估時，它會儲存流程中所有封包的相同資訊，例如安全群組、存取控制清單和路由表項目。當它處理相同流程的其他封包時，可以使用儲存的資訊來減少這些封包的額外負荷。

您的連線速率是以每秒連線數 (CPS) 量度來測量。每個新連線都需要額外的處理額外負荷，這些負荷必須納入工作負載能力的估計。在設計工作負載時，請務必同時考慮CPS和PPS指標。

如何建立連線

在 Nitro 型執行個體與另一個端點之間建立連線時，Nitro 卡會評估兩個端點之間傳送或接收的第一個封包的完整流程。對於相同流程的後續封包，通常不需要完全重新評估。但是，也有例外。如需例外狀況的詳細資訊，請參閱不使用硬體加速的封包。

下列內容定義兩個端點及它們之間的封包流量。這五個屬性一起被稱為 5 元組流程。

- 來源 IP
- 來源連接埠
- 目標 IP

- 目標連接埠
- 通訊協定

封包流程的方向稱為入口 (入埠) 和出口 (出埠)。下列高階說明摘要說明端對端網路封包流量。

- Ingress — 當 Nitro 卡處理輸入網路封包時，會根據狀態防火牆規則和存取控制清單來評估封包。它會追蹤連線、測量連線，並視情況執行其他動作。然後它會將封包轉送到主機CPU上的目的地。
- 輸出 — 當 Nitro 卡處理輸出網路封包時，它會查詢遠端介面目的地、評估各種VPC功能、套用速率限制，以及執行其他適用的動作。然後它會將封包轉送到網路上的下一個躍點目的地。

設計您的網路以獲得最佳效能

要利用 Nitro 系統的性能功能，您必須了解網絡處理需求以及這些需求如何影響 Nitro 資源的工作負載。然後，您可以針對網路環境設計最佳效能。您的基礎結構設定以及應用程式工作負載的設計和組態可能會影響封包處理和連線速率。例如，如果您的應用程式建立連線速率很高，例如DNS服務、防火牆或虛擬路由器，則只有在建立連線之後才會發生的硬體加速，就會有較少的機會。

您可以設定應用程式和基礎架構設定，以簡化工作負載並改善網路效能 不過，並非所有封包都符合加速的資格。Nitro 系統將完整的網路流程用於新連線，以及不符合加速資格的封包。

本節的其餘部分將著重於應用程式和基礎結構設計考量，以協助確保封包盡可能在加速路徑內流通。

硝基系統的網路設計考量

當您為執行個體設定網路流量時，有許多方面需要考量，可能會影響PPS效能。建立流程之後，大多數定期進出的封包都有資格加速。但是，存在例外情況，以確保基礎架構設計和封包流程持續符合通訊協定標準。

為了從 Nitro 卡獲得最佳性能，您應該仔細考慮基礎架構和應用程序的以下配置詳細信息的優缺點。

基礎架構考

您的基礎架構組態可能會影響您的封包流量和處理效率。下列清單包含一些重要的考量。

具有不對稱性的網絡接口配置

安全性群組會使用連線追蹤來追蹤進出執行個體的流量相關資訊。非對稱路由是流量透過一個網路介面進入執行個體，並透過不同的網路介面離開，可降低執行個體在追蹤流程時所能達到的尖峰

效能。如需有關安全性群組連線追蹤、未追蹤連線以及自動追蹤連線的詳細資訊，請參閱Amazon EC2 安全組連接跟踪。

網絡驅動

網絡驅動程序會定期更新和發布。如果您的驅動程序過時，可能會嚴重損害性能。讓驅動程式保持在最新狀態，以確保您擁有最新的修補程式，並可利用效能改進的優勢，例如僅適用於最新一代驅動程式的加速路徑功能。較早的驅動程序不支持加速路徑功能。

若要利用加速路徑功能，建議您在執行個體上安裝最新的ENA驅動程式。

Linux 執行個體 — ENA Linux 驅動程式 2.2.9 或更新版本。若要從 Amazon 驅動程式 GitHub 儲存庫安裝或更新 ENA Linux 驅動程式，請參閱讀我檔案的驅動程式編譯一節。

視窗執行個體 — ENA 視窗驅動程式 2.0.0 或更新版本。若要安裝或更新 ENA Windows 驅動程式，請參閱在 EC2 Windows 執行個體上安裝ENA驅動程式。

端點之間的距離

由於在應用程式層進行TCP視窗化，相同可用區域中的兩個執行個體之間的連線每秒處理的封包數量會比跨區域的連線多，這會決定在任何指定時間可以處理傳輸中的資料量。執行個體之間的長距離會增加延遲，並減少端點可處理的封包數目。

應用程式的設計

應用程式設計和組態的某些方面可能會影響您的處理效率。下列清單包含一些重要的考量。

封包大小

較大的封包大小可增加執行個體在網路上傳送和接收的資料輸送量。較小的封包大小可以增加封包處理速率，但是當封包數量超過PPS容量時，這樣可以減少達到的最大頻寬。

如果封包的大小超過網路躍點的最大傳輸單位 (MTU)，路徑上的路由器可能會將其分段。產生的封包片段會被視為例外狀況，並以標準速率處理 (非加速)。這可能會導致您的表現變化。Amazon EC2 支持 9001 字節的巨型框架，但並非所有服務都支持它。我們建議您在設定時評估拓撲MTU。

協議權衡

可靠的協議，例TCP如比不可靠的協議具有更多的UDP開銷。較低的額外負荷和簡化的UDP傳輸通訊協定的網路處理可能會導致較高的PPS速率，但會犧牲可靠的封包傳遞。如果可靠的數據包傳遞對您的應用程序並不重要，那麼可UDP能是一個不錯的選擇。

微爆破

當流量在短時間內超過限額而非均勻分佈時，就會發生微爆炸。這通常發生在微秒的範圍內。

例如，假設您有一個最多可傳送 10 Gbps 的執行個體，而您的應用程式會在半秒內傳送完整的 10 Gb。這種微爆在上半年第二次超過了津貼，並且在第二個的剩餘部分沒有留下任何東西。即使您在 1 秒的時間範圍內傳送 10Gb，前半秒的限額也可能導致封包排入佇列或捨棄。

您可以使用網路排程器 (例如 Linux 流量控制) 來協助調整輸送量，並避免因為微量爆發造成佇列或丟棄封包。

流量數

單一流程的限制為 5 Gbps，除非它位於支援高達 10 Gbps 的叢集置放群組內，或使用支援高達 25 Gbps 的 ENA Express。

同樣地，與使用單一流程相比，Nitro 卡可以跨多個流程處理更多封包。為了達到每個執行個體的尖峰封包處理速率，我們建議在 100 Gbps 或更高彙總頻寬的執行個體上至少 100 個流量。隨著彙總頻寬功能的增加，達到尖峰處理速率所需的流量數也會增加。基準測試可協助您判斷網路上達到最高速率所需的組態。

彈性網路介面卡 (ENA) 佇列數

根據預設，ENA佇列數目上限會根據您的執行個體大小和類型分配給網路介面。減少佇列計數可降低可達到的最大PPS速率。我們建議使用預設佇列配置以獲得最佳效能。

對於 Linux，網絡接口默認配置為最大值。對於以「資料平面開發套件」(DPDK) 為基礎的應用程式，建議您設定可用的佇列數目上限。

功能過程開銷

流量鏡像和 ENA Express 等功能可以增加更多的處理額外負荷，從而降低絕對封包處理效能。您可以限制功能使用或停用功能，以提高封包處理速率。

連線追蹤以維持狀態

您的安全性群組會使用連線追蹤來儲存進出執行個體的流量相關資訊。連線追蹤會針對每個個別網路流量套用規則，以判斷流量是否允許或拒絕。Nitro 卡使用流量追蹤來維持流量的狀態。隨著套用的安全性群組規則越多，評估流程需要更多工作。

> **Note**
>
> 並非所有網路流量都會追蹤。如果使用設定安全性群組規則未追蹤的連線，則除了自動追蹤的連線以確保在有多個有效回覆路徑時對稱路由，則不需要其他工作。

不使用硬體加速的封包

並非所有封包都能利用硬體加速。處理這些例外狀況涉及一些處理額外負荷，這是確保網路流程健全狀況所必需的。網路流程必須可靠地符合通訊協定標準、符合VPC設計中的變更，並且只將封包路由到允許的目的地。但是，開銷會降低您的性能。

封包片段

如應用程式考量所述，超過網路封包所產生的封包片段MTU會當做例外狀況處理，而且無法利用硬體加速的優勢。

閒置連線

當連線有一段時間沒有活動時，即使連線尚未達到其逾時限制，系統也可以取消其優先順序。然後，如果在取消連接優先級之後傳入數據，則系統需要將其作為異常處理才能重新連接。

若要管理連線，您可以使用連線追蹤逾時來關閉閒置連線。您也可以使用 TCP Keepalives 來保持閒置連線的開啟狀態。如需詳細資訊，請參閱閒置連線追蹤逾時。

VPC突變

安全性群組、路由表和存取控制清單的更新都需要在處理路徑中重新評估，以確保路由項目和安全性群組規則仍如預期般套用。

ICMP流

網際網路控制訊息通訊協定 (ICMP) 是網路裝置用來診斷網路通訊問題的網路層通訊協定。這些封包一律會使用完整流程。

最大限度地提高 Nitro 系統的網絡性能

在您對執行個體做出任何設計決策或調整任何網路設定之前，我們建議您採取下列步驟，以協助確保您獲得最佳結果：

1. 透過檢閱，瞭解您可以採取的行動的優缺點硝基系統的網路設計考量。

   如需有關 Linux 執行個體組態的更多考量和最佳做法，請參閱上的《ENALinux 驅動程式最佳做法和效能最佳化指南》 GitHub。

2. 使用尖峰作用中流量計數對工作負載進行基準測試，以決定應用程式效能的基準 使用效能基準，您可以測試設定或應用程式設計中的變化，以瞭解哪些考量會產生最大的影響，尤其是在您打算擴充或向外擴充時。

下列清單包含您可以根據系統需求調整PPS效能的動作。

- 減少兩個例證之間的物理距離。當傳送和接收執行個體位於相同的可用區域或使用叢集置放群組時，您可以減少封包從一個端點移動到另一個端點所需採取的躍點數目。
- 請使用 未追蹤的連線。
- 使用通UDP訊協定進行網路流量。
- 對於彙總頻寬為 100 Gbps 或更高的EC2執行個體，請將工作負載分配超過 100 個或更多個別流程，以將工作平均分散到 Nitro 卡上。

監控 Linux 執行個體的效能

您可以在 Linux 執行個體上使用 Ethtool 指標來監控執行個體網路效能指標，例如頻寬、封包速率和連線追蹤。如需詳細資訊，請參閱監控EC2執行個體ENA設定的網路效能。

最佳化 EC2 Windows 執行個體的網路效能

若要在具有增強型聯網功能的 Windows 執行個體上達到最大的網路效能，您可能需要修改預設作業系統組態。建議您針對需要高網路效能的應用程式進行下列組態變更。其他優化（例如RSS，打開校驗和卸載和啟用）已在官方 Windows 上配置。AMIs

> **Note**
>
> TCP煙囪卸載應該在大多數使用情況下被禁用，並已被棄用從 Windows 服務器 2016 年。

除了這些作業系統最佳化之外，您還應該考慮網路流量的最大傳輸單位 (MTU)，並根據您的工作負載和網路架構進行調整。如需詳細資訊，請參閱 EC2執行個體的網路最大傳輸單元 (MTU)。

AWS 定期測量在 50us 叢集置放群組中啟動的執行個體之間的平均往返延遲，以及 99.9 百分位數為 200us 的尾部延遲。如果您的應用程式需要持續較低的延遲，建議您在 Nitro System 上建置的固定效能執行個體上使用最新版本的ENA驅動程式。

設定接收端調整調整CPU相似

「接收端調整」(RSS) 是用來將網路流量CPU負載分配到多個處理器之間。預設情況下，官方的 Amazon 視窗AMIs設定為RSS啟用。ENA彈性網路介面最多可提供八個RSS佇列。透過定義RSS佇列和其他系統處理序的CPU相似性，可以將CPU負載分散到多核心系統上，以便處理更多網路流量。在超過 16 個的執行個體類型上vCPUs，建議您使用指令程式，該`Set-NetAdapterRSS` PowerShell 指

令程式會從所有彈性網路介面的RSS組態中手動排除開機處理器 (啟用超執行緒時，邏輯處理器 0 和 1)，以避免與各種系統元件發生爭用。

Windows 具有超執行緒感知功能，可確保單一網路介面卡 (NIC) 的RSS佇列永遠位於不同的實體核心上。因此，除非停用超執行緒，否則為了完全防止與其他產生爭用NICs，請在 16 個邏輯處理器範圍內散佈每個NIC執行緒的RSS組態。指`Set-NetAdapterRss`令程式可讓您定義、、、和 NumaNode (選用) 的值 BaseProcessorGroup， BaseProcessorNumber以定義每個NIC範圍的有效邏輯處理器。MaxProcessingGroup MaxProcessorNumber如果沒有足夠的實體核心無法完全消除間NIC爭用，請根據介面的預期工作負載，將重疊的範圍縮到最小，或減少 elastic network interface 範圍中的邏輯處理器數目 (換句話說，低容量管理網路介面可能不需要指派那麼多RSS佇列)。此外，如先前所述，各種元件必須在 CPU 0 上執行，因此我們建議您在有足夠 vCPUs 的可用時將其從所有RSS組態中排除。

例如，當一個 72 v 執行個CPU體上有三個彈性網路介面，且 2 個NUMA節點已啟用超執行緒時，下列指令會將兩者之間的網路負載分散CPUs而不會重疊，並防止完全使用核心 0。

```
Set-NetAdapterRss -Name NIC1 -BaseProcessorGroup 0 -BaseProcessorNumber 2 -
MaxProcessorNumber 16
Set-NetAdapterRss -Name NIC2 -BaseProcessorGroup 1 -BaseProcessorNumber 0 -
MaxProcessorNumber 14
Set-NetAdapterRss -Name NIC3 -BaseProcessorGroup 1 -BaseProcessorNumber 16 -
MaxProcessorNumber 30
```

注意，這些設定對於每個網路轉接器都是持續有效。如果將執行個體的大小調整為具有不同數目的執行個體vCPUs，您應該重新評估每個已啟用的 elastic network interface 的RSS組態。您可以在這裡找到`Set-NetAdapterRss`指令程式的完整 Microsoft 文件：https://docs.microsoft.com/en-us/PowerShell /模組/網路適配器/設定網路適配器。

SQL工作負載的特別注意事項：我們也建議您檢閱 I/O 執行緒相CPUs似性設定以及 elastic network interface RSS 組態，以將 I/O 和網路爭用降到最低。請參閱 親和性遮罩伺服器組態選項。

適用於 Amazon 上的機器學習HPC工作負載的彈性光纖 EC2

彈性網狀架構介面卡 (EFA) 是一種網路裝置，您可以將其連接至 Amazon EC2 執行個體，以加速高效能運算 (HPC) 和機器學習應用程式。EFA透過 AWS 雲端提供的延展性、彈性和彈性，可讓您達到內部部署HPC叢集的應用程式效能。

EFAs與傳統在雲端HPC系統中使用的傳TCP輸相比，可提供更低且更一致的延遲和更高的輸送量。它增強了對擴展HPC和機器學習應用程序至關重要的實例間通信的性能。它經過優化，可以在現有的 AWS 網絡基礎架構上工作，並且可以根據應用程序需求進行擴展。

EFAs與 Libfabric 1.7.0 及更高版本集成，它支持開放 MPI 5 和更高版本以及英特爾 MPI 2019 年更新 5 及更高版本的HPC應用程序，以及用於機器學習應用的 Nvidia 集體通信庫（NCCL）。

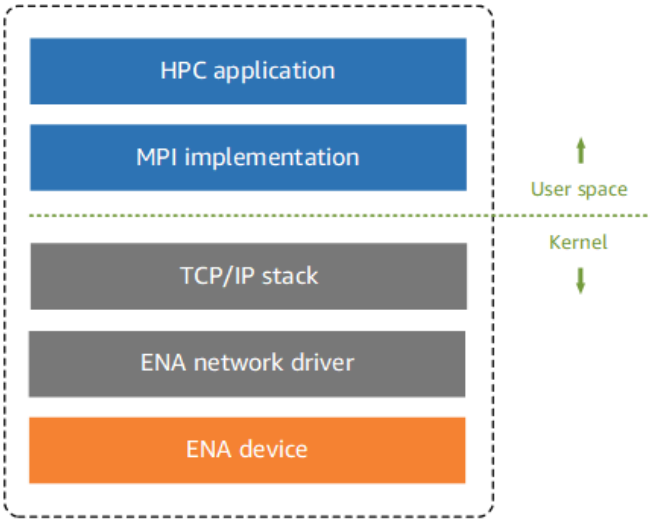
> **Note**
>
> Windows 執行個體不支援的EFAs作業系統略過功能。如果您附加EFA到 Windows 執行個體，則執行個體會以彈性網路介面卡的形式運作，而不需要新增的功EFA能。

目錄

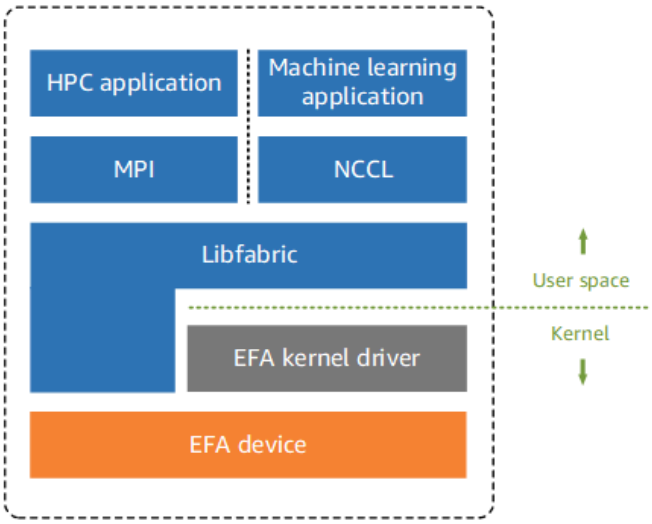
- EFA基礎
- 支援的介面和程式庫
- 支援的執行個體類型
- 支援的作業系統
- EFA限制
- EFA定價
- 在加速運算執行個體上最大化網路頻寬 EFA
- 開始使用EFA和處理 Amazon 上MPI的HPC工作負載 EC2
- 在 Amazon 上開始使用EFA和處NCCL理機器學習工作負載 EC2
- 建立彈性結構介面卡並將其連接至 Amazon EC2 執行個體
- 從 Amazon EC2 執行個體EFA中分離並刪除
- 在 Amazon 上監控彈性織物適配器 EC2
- 使用總和檢查碼驗證EFA安裝程式

EFA基礎

EFA是具有附加功能的彈性網路介面卡 (ENA)。它提供了一個的所有功能ENA，以及一個額外的操作系統旁路功能。OS-bypass 是一種存取模式，可讓HPC機器學習應用程式與網路介面硬體直接通訊，以提供低延遲、可靠的傳輸功能。



Traditional HPC software stack in EC2



HPC software stack in EC2 with EFA

傳統上，HPC應用程式會使用訊息傳遞介面 (MPI) 與系統的網路傳輸進行介面。在 AWS 雲中，這意味著應用程序與接口MPI，然後使用操作系統的 TCP /IP 堆棧和ENA設備驅動程序來啟用實例之間的網絡通信。

使用EFA, HPC 應用程式使NCCL用MPI或與 Libfabric API 介面。Libfabric 會API略過作業系統核心，並直接與EFA裝置通訊，以便將封包放在網路上。這減少了開銷，並使HPC應用程序能夠更有效地運行。

> **Note**
>
> Libfabric 是 OpenFabrics 接口（OFI）框架的核心組件，它定義和導出的用戶空間API。OFI如需詳細資訊，請參閱 LibFabric OpenFabrics 網站。

和之間EFAs的差異 ENAs

彈性網路介面卡 (ENAs) 提供支援網路所需的傳統 IP 網VPC路功能。EFAs提供與傳統 IP 網路相同的所有功能ENAs，並且還支援作業系統旁路功能。作業系統旁路可讓HPC機器學習應用程式略過作業系統核心，並直接與裝置通訊。EFA

支援的介面和程式庫

EFAs支持以下接口和庫：

- 開啟 MPI 5 及更高版本

- 開啟 MPI 4.0 或更新版本為重力子的優先選擇
- 英特尔 MPI 2019 年更新 5 及更高版本
- NVIDIA集體通訊庫 (NCCL) 2.4.2 及更高版本

支援的執行個體類型

下列執行個體類型支援 EFAs：

- 一般用途：m5dn.24xlargem5dn.metalm5n.24xlargem5n.metal| m5zn.12xlarge | m5zn.metal | m6a.48xlarge | m6a.metal | m6i.32xlarge | m6i.metal | m6id.32xlarge | m6id.metal | m6idn.32xlarge | m6idn.metal | m6in.32xlarge | m6in.metal m7a.48xlarge | m7a.metal-48xl | m7g.16xlarge | m7g.metal | m7gd.16xlarge | m7gd.metal | m7i.48xlarge | m7i.metal-48xl
- 運算最佳化：c5n.9xlargec5n.18xlargec5n.metalc6a.48xlargec6a.metalc6gn.16xlargec6i.32xla c6i.metal c6id.32xlarge | c6id.metal | c6in.32xlarge | c6in.metal c7a.48xlarge | c7a.metal-48xl | c7g.16xlarge c7g.metal | c7gd.16xlarge | c7gd.metal | c7gn.16xlarge c7gn.metal | c7i.48xlarge | c7i.metal-48xl
- 記憶體最佳化：r5dn.24xlarger5dn.metalr5n.24xlarger5n.metalr6a.48xlarger6a.metalr6i.32xlar r6id.32xlarge r6id.metal | r7a.48xlarge r7a.metal-48xl | r7g.16xlarge | r7g.metal r7gd.16xlarge | r7gd.metal r7i.48xlarge | r7i.metal-48xl r7iz.32xlarge | r7iz.metal-32xl r8g.24xlarge | r8g.48xlarge r8g.metal-24xl | r8g.metal-48xl | u7i-12tb.224xlarge u7in-16tb.224xlarge | u7in-24tb.224xlarge u7in-32tb.224xlarge | x2idn.32xlarge x2idn.metal | x2iedn.32xlarge x2iedn.metal | x2iezn.12xlarge |x2iezn.metal
- 儲存最佳化：i3en.12xlarge | i3en.24xlarge | i3en.metal | i4g.16xlarge | i4i.32xlarge | i4i.metal | im4gn.16xlarge
- 加速運算：dl1.24xlargedl2q.24xlargeg4dn.8xlargeg4dn.12xlargeg4dn.16xlargeg4dn.metalg5 p3dn.24xlarge p4d.24xlarge p4de.24xlarge p5.48xlarge | trn1.32xlarge trn1n.32xlarge | vt1.24xlarge
- 高效能運算：hpc6a.48xlargehpc6id.32xlargehpc7a.12xlargehpc7a.24xlargehpc7a.48xlargehpc7 hpc7g.4xlarge | hpc7g.8xlarge | hpc7g.16xlarge

查看特定區域支援的可用執EFAs行個體類型

可用的執行個體類型因區域而異。若要查看區域EFAs中支援的可用例證類型，請將describe-instance-types指令與`--region`參數搭配使用。包括`--filters`參數，將結果範圍定為支援的執行個體類型，以EFA及將輸出範圍限定為的`--query`參數`InstanceType`。

```
aws ec2 describe-instance-types  --region us-east-1  --filters Name=network-info.efa-supported,Values=true  --query "InstanceTypes[*].[InstanceType]"  --output text | sort
```

支援的作業系統

作業系統支援會因處理器類型而有所不同。下表顯示支援的作業系統。

作業系統	英特爾/AMD(**x86_64**) 實例類型	AWS 重力子 (**arm64**) 例證類型
Amazon Linux 2023	✓	✓
Amazon Linux 2	✓	✓
RHEL八和九	✓	✓
Debian 10 和 11	✓	✓
Rocky Linux 8 和 9	✓	✓
Ubuntu 二零四、二零四及二十四	✓	✓
SUSE企業版 15 SP2 及更新版本	✓	✓
開啟SUSE躍進 15.5 及更新版本	✓	

> **Note**
>
> Ubuntu 20.04 與 `dl1.24xlarge` 執行個體搭配使用時支援對等直接支援。

EFA限制

EFAs有以下限制：

- 所有 P4d 和 P5 執行個體類型都支援NVIDIAGPUDirect遠端直接記憶體存取 ()。RDMA
- EFA目前不支援 P4D/P4DE/ DL1 執行個體與其他執行個體類型之間的流量。
- EFA每張網路卡可以設定支援多張網路卡的執行個體類型。所有其他支援的執行個體類型僅支援EFA每個執行個體一
- 連接時 `c7g.16xlargem7g.16xlarge`，`r7g.16xlarge`不支援專用執行個EFA體和專用主機。
- EFA作業系統略過的流量無法跨越可用區域或 AWS 帳戶。VPCs換句話說，作業EFA系統略過的流量無法從一個可用區域流動 VPC (有或沒有VPC對等連線) 或 AWS 帳戶到另一個可用區域。這不適用於來自EFA.
- EFA作業系統略過的流量無法跨本機區域中的子網路傳送。
- EFA作業系統略過流量不可路由。來自的正常 IP 流量保EFA持可路由。
- EFA必須是安全性群組的成員，該群組允許進出安全性群組本身的所有輸入和輸出流量。
- EFA在視窗執行個體上不受支援。
- EFA AWS Outposts 不支持。

EFA定價

EFA提供選用的 Amazon EC2 聯網功能，您可以在任何受支援的執行個體上啟用，無需額外付費。

在加速運算執行個體上最大化網路頻寬 EFA

若要將下列加速執行個體類型的頻寬最大化，您可以使用多個彈性網狀架構介面卡 (EFA) 介面。

- P5 執行個體最高可支援 32 張網路卡，並可提供高達 3200 Gbps 的網路頻寬。
- G6e 執行個體最多支援四張網路卡，並可提供高達 400 Gbps 的網路頻寬

如需開始使用GPU加速執行個體的詳細資訊，請參閱GPU執行個體的效能加速。

我們建議您為每張EFA網路卡定義一個網路介面卡。若要在啟動時設定這些介面，我們建議您進行下列設定：

- 對於網路介面 0，指定裝置索引 0
- 對於網路介面 1 至 31，指定裝置索引 1

如果您使用 Amazon EC2 主控台，請在啟動執行個體精靈中，選擇 「網路設定」區段中的「編輯」。展開進階網路組態並選擇新增網路介面以新增所需數量的網路介面。對於每個網路介面 EFA，選取啟用。對於所有網路介面 (主要網路介面除外)，針對裝置索引指定 1。視需要進行其餘的設定。

如果您正在使用 AWS CLI，請使用 run-instance 指令，對`--network-interfaces`，指定所需的網路介面數。對於每個網路介面，針對 `InterfaceType` 指定 `efa`。對於主要網路介面，針對 `NetworkCardIndex` 和 `DeviceIndex` 指定 `0`。對於剩餘的網路介面，針對 `NetworkCardIndex` 指定從 1 至 31 的唯一值，以及針對 `DeviceIndex` 指定 1。

下列範例命令程式碼片段顯示具有 32 個EFA網路介面的要求。

```
$ aws --region $REGION ec2 run-instances \
 --instance-type p5.48xlarge \
 --count 1 \
 --key-name key_pair_name \
 --image-id ami_id \
 --network-interfaces
 "NetworkCardIndex=0,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
 \

 "NetworkCardIndex=1,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
 \

 "NetworkCardIndex=2,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
 \

 "NetworkCardIndex=3,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
 \

 "NetworkCardIndex=4,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
 \

 "NetworkCardIndex=5,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
 \
```

```
"NetworkCardIndex=6,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
\

"NetworkCardIndex=7,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
\

"NetworkCardIndex=8,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
\

"NetworkCardIndex=9,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
\

"NetworkCardIndex=10,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\

"NetworkCardIndex=11,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\

"NetworkCardIndex=12,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\

"NetworkCardIndex=13,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\

"NetworkCardIndex=14,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\

"NetworkCardIndex=15,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\

"NetworkCardIndex=16,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\

"NetworkCardIndex=17,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\

"NetworkCardIndex=18,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\

"NetworkCardIndex=19,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
\
```

```
 "NetworkCardIndex=20,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
 \

 "NetworkCardIndex=21,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
 \

 "NetworkCardIndex=22,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
 \

 "NetworkCardIndex=23,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
 \

 "NetworkCardIndex=24,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
 \

 "NetworkCardIndex=25,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
 \

 "NetworkCardIndex=26,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
 \

 "NetworkCardIndex=27,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
 \

 "NetworkCardIndex=28,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
 \

 "NetworkCardIndex=29,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
 \

 "NetworkCardIndex=30,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
 \

 "NetworkCardIndex=31,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=e
...
```

如果您使用的是啟動範本，請在啟動範本中指定所需的網路介面數量。對於每個網路介面，針對 `InterfaceType` 指定 `efa`。對於主要網路介面，針對 `NetworkCardIndex` 和 `DeviceIndex` 指定 `0`。對於剩餘的網路介面，針對 `NetworkCardIndex` 指定從 1 至 31 的唯一值，以及針對 `DeviceIndex` 指定 `1`。下面的程式碼片段顯示了一個範例，其中包含可能的 32 個網路介面中的 3 個網路介面。

```
"NetworkInterfaces":[
{
  "NetworkCardIndex":0,
  "DeviceIndex":0,
  "InterfaceType": "efa",
  "AssociatePublicIpAddress":false,
  "Groups":[
    "security_group_id"
  ],
  "DeleteOnTermination":true
},
{
  "NetworkCardIndex": 1,
  "DeviceIndex": 1,
  "InterfaceType": "efa",
  "AssociatePublicIpAddress":false,
  "Groups":[
    "security_group_id"
  ],
  "DeleteOnTermination":true
},
{
  "NetworkCardIndex": 2,
  "DeviceIndex": 1,
  "InterfaceType": "efa",
  "AssociatePublicIpAddress":false,
  "Groups":[
    "security_group_id"
  ],
  "DeleteOnTermination":true
}
...
```

啟動具有多個網路介面的 P5 執行個體時，您無法自動指派公有 IP 地址。不過，您可以在啟動網際網路連線後將彈性 IP 位址附加至主要網路介面 (NetworkCardIndex DeviceIndex=0, =0)。Ubuntu 20.04 及更新版本以及 Amazon Linux 2 及更新版本均設定為在執行個體按照本頁面上的建議啟動時，將主要網路介面用於網際網路流量。

若要在 G6e 執行個體上達到最高的網路效能，您可以使用IMDS對應連接的網路介面，並在脫離的網路執行個體上使用網路執行個體進行最佳化。 NetworkCardIndexes

下列範例指令碼會將附加的詳細資訊匯集在一起 NetworkCardIndexes。

```
$ TOKEN=$(curl -s -X PUT "http://169.254.169.254/latest/api/token" -H "X-
 aws-ec2-metadata-token-ttl-seconds: 21600")
 count=$(ls -l /sys/class/net/eth* | wc -l)

 for ((i = 0 ; i < ${count} ; i++)); do
  mac=$(cat /sys/class/net/eth$i/address)

  network_card=$(curl -s -H "X-aws-ec2-metadata-token: $TOKEN"
"http://169.254.169.254/latest/meta-data/network/interfaces/macs/$mac/
network-card")

   device_number=$(curl -s -H "X-aws-ec2-metadata-token: $TOKEN"
"http://169.254.169.254/latest/meta-data/network/interfaces/macs/$mac/
device-number")

   device_bdf=$(realpath /sys/class/net/eth${i}/device)

   echo "eth${i} ${network_card} ${device_number} ${device_bdf}"
done
```

開始使用EFA和處理 Amazon 上MPI的HPC工作負載 EC2

本教學課程可協助您針對HPC工作負載啟MPI用EFA且已啟用的執行個體叢集。

任務

- 步驟 1：準備EFA已啟用的安全性群組
- 步驟 2：啟動暫時執行個體
- 步驟 3：安裝 EFA 軟體
- 步驟 4：（可選）啟用打開 MPI 5
- 步驟 5：（可選）安裝英特爾 MPI
- 步驟 6：停用 ptrace 保護
- 步驟 7. 確認安裝
- 步驟 8：安裝您的HPC應用程式
- 步驟 9：建立EFA已啟用 AMI
- 步驟 10：在叢集置放群組中EFA啟動已啟用的執行個體
- 步驟 11：終止暫時執行個體
- 步驟 12：啟用無密碼 SSH

步驟 1：準備EFA已啟用的安全性群組

EFA需要安全性群組，該群組允許進出安全性群組本身的所有輸入和輸出流量。下列程序會建立安全性群組，允許所有入站和出站流量進出，並允許來自任何IPv4位址的輸入SSH流量進行SSH連線。

> **Important**
>
> 此安全群組僅供測試之用。對於生產環境，我們建議您建立輸入SSH規則，該規則僅允許來自您連線的 IP 位址的流量，例如電腦的 IP 位址或區域網路中的一系列 IP 位址。

如需其他案例，請參閱不同使用案例的安全群組規則。

若要建立EFA已啟用的安全性群組

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Security Groups (安全群組)，然後選擇 Create Security Group (建立安全群組)。
3. 在 Create Security Group (建立安全群組) 視窗中，執行下列動作：
   a. 對於 Security group name (安全群組名稱)，輸入安全群組的描述性名稱，例如 `EFA-enabled security group`。
   b. (選用) 對於 Description (描述)，輸入安全群組的簡短描述。
   c. 在VPC中 VPC，選取您要啟用已啟EFA用執行個體的目標。
   d. 選擇 Create Security Group (建立安全群組)。
4. 選取您建立的安全群組，在 Details (詳細資料) 索引標籤上，複製 Security group ID (安全群組 ID)。
5. 在仍然選取安全群組的情況下，依序選取 Actions (動作) 和 Edit inbound rules (編輯傳入規則)，然後執行下列動作。
   a. 選擇新增規則。
   b. 針對類型，選擇所有流量。
   c. 若為 Source type (來源類型)，選擇 Custom (自訂)，然後將您複製的安全群組 ID 貼到欄位中。
   d. 選擇新增規則。
   e. 對於「類型」，選擇SSH。

f. 針對「來源類型」，選擇「隨處-」IPv4。

g. 選擇儲存規則。

6. 在仍然選取安全群組的情況下，依序選取 Actions (動作) 和 Edit outbound rules (編輯傳出規則)，然後執行下列動作。

   a. 選擇新增規則。

   b. 針對類型，選擇所有流量。

   c. 若為 Destination type (目的地類型)，選擇 Custom (自訂)，然後將您複製的安全群組 ID 貼到欄位中。

   d. 選擇儲存規則。

步驟 2：啟動暫時執行個體

啟動可用來安裝和設定EFA軟體元件的暫時執行個體。您可以使用此執行個體建立EFA已啟用的執行個體，AMI從中啟動EFA已啟用的執行個體。

啟動暫時執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)，接著選擇 Launch Instances (啟動執行個體) 來開啟新的啟動執行個體精靈。
3. (選用) 在 Name and tags (名稱和標籤) 區段中，為執行個體命名，例如，`EFA-instance`。該名稱將指派作為執行個體的資源標籤 (Name=*`EFA-instance`*)。
4. 在「應用程式和作業系統映像」區段中，AMI為其中一個支援的作業系統選取一個。
5. 在 Instance type (執行個體類型) 區段中，選取支援的執行個體類型。
6. 在 Key pair (金鑰對) 區段中，選取要用於執行個體的金鑰對。
7. 在 Network settings (網路設定) 區段，選擇 Edit (編輯)，接著執行下列動作：

   a. 對於 Subnet (子網)，請選擇要在其中啟動執行個體的子網。如果您未選取子網路，則無法啟用的執行個體EFA。

   b. 針對 Firewall (security groups) (防火牆 (安全群組))，選取 Select existing security group (選取現有的安全群組)，接著選取您在前一個步驟中建立的安全群組。

   c. 展開 Advanced network configuration (進階網路組態) 區段，並針對 Elastic Fabric Adapter，選取 Enable (啟用)。

8. 在 儲存 區段中，根據需求設定磁碟區。

9. 在右邊的 Summary (摘要) 面板中，選擇 Launch instance (啟動執行個體)。

> **Note**
>
> 除非您已經設IMDSv2定為帳戶的預設值，否則請考慮IMDSv2針對暫存執行個體以及在步驟 9 中建立的執行個體使用。AMI如需IMDSv2組態步驟的詳細資訊，請參閱設定新執行個體的執行個體中繼資料選項。

步驟 3：安裝 EFA 軟體

安EFA裝在暫存執行個體EFA上支援所需的已啟用核心、EFA驅動程式、Libfabric 和 Open MPI 堆疊。

這些步驟會因您打算搭配「開啟」MPI、「Intel」或「EFA開放」MPI 和「IntelMPI」搭配使用而有所不同MPI。

安裝 EFA 軟體

1. 連接至您啟動的執行個體。如需詳細資訊，請參閱使用 Connect 到您的 Linux 執行個體 SSH。
2. 為確保所有軟體套件皆為最新版本，請對您的執行個體執行快速軟體更新。此程序可能需要幾分鐘的時間。

   - Amazon Linux 2023, Amazon Linux 2, RHEL 8/9, 洛基 Linux 8/9

     ```
     $ sudo yum update -y
     ```

   - Ubuntu 和 Debian

     ```
     $ sudo apt-get update && sudo apt-get upgrade -y
     ```

   - SUSELinux 企業版

     ```
     $ sudo zypper update -y
     ```

3. 重新啟動執行個體並重新連線至其中。
4. 下載 EFA 軟體安裝檔案。軟體安裝檔案已封裝成壓縮 tarball (`.tar.gz`) 檔案。若要下載最新穩定版本，請使用下列命令：

   ```
   $ curl -O https://efa-installer.amazonaws.com/aws-efa-installer-1.34.0.tar.gz
   ```

您也可以在之前的命令中以 `latest` 取代版本號碼，以取得最新版本。

5. (選擇性) 驗證 EFA tarball (`.tar.gz`) 檔案的真實性與完整性。

   我們建議您執行這項操作來確認軟體發行者的身分，並檢查檔案自發行以來並未遭到變更或損毀。如果您不想驗證 tarball 檔案，請略過此步驟。

   > **Note**
   >
   > 或者，如果您想要使用MD5或SHA256總和檢查碼來驗證 tarball 檔案，請參閱。使用總和檢查碼驗證EFA安裝程式

   a. 下載公開GPG金鑰並將其匯入您的金鑰圈。

   ```
   $ wget https://efa-installer.amazonaws.com/aws-efa-installer.key && gpg --
   import aws-efa-installer.key
   ```

   命令應傳回金鑰值。請記下金鑰的值，因為下一個步驟將需要它。

   b. 驗證GPG金鑰的指紋。執行以下命令，並指定上一步驟中的金鑰值。

   ```
   $ gpg --fingerprint key_value
   ```

   命令應傳回與 `4E90 91BC BB97 A96B 26B1 5E59 A054 80B1 DD2D 3CCC` 相同的指紋。如果指紋不匹配，請不要運行EFA安裝腳本，然後聯繫 AWS Support。

   c. 下載簽章檔案並驗證 EFA tarball 檔案的簽章。

   ```
   $ wget https://efa-installer.amazonaws.com/aws-efa-installer-1.34.0.tar.gz.sig
    && gpg --verify ./aws-efa-installer-1.34.0.tar.gz.sig
   ```

   下面顯示了範例輸出。

   ```
   gpg: Signature made Wed 29 Jul 2020 12:50:13 AM UTC using RSA key ID DD2D3CCC
   gpg: Good signature from "Amazon EC2 EFA <ec2-efa-maintainers@amazon.com>"
   gpg: WARNING: This key is not certified with a trusted signature!
   gpg:          There is no indication that the signature belongs to the owner.
   Primary key fingerprint: 4E90 91BC BB97 A96B 26B1  5E59 A054 80B1 DD2D 3CCC
   ```

如果結果包含`Good signature`，而指紋與上一個步驟中傳回的指紋相符，請繼續下一個步驟。如果沒有，請不要運行EFA安裝腳本，然後聯繫 AWS Support。

6. 從壓縮的 `.tar.gz` 檔案中解壓縮檔案，然後導覽至解壓縮的目錄。

```
$ tar -xf aws-efa-installer-1.34.0.tar.gz && cd aws-efa-installer
```

7. 安裝 EFA 軟體。根據使用案例而定，執行以下其中一個命令。

> **Note**
>
> EFA不支援NVIDIAGPUDirect使用 SUSE Linux。如果您使用的是 SUSE Linux，則必須另外指定`--skip-kmod`選項以防止 kmod 安裝。根據預設，SUSELinux 不允許 out-of-tree 核心模組。

Open MPI and Intel MPI

如果您打算EFA搭配「開放」MPI 和「Intel」使用MPI，您必須使用 Libfabric 與開放式來安裝EFA軟體MPI，而且您必須完成「步驟 5：安裝 Intel MPI」。

若要使用 Libfabric 安裝EFA軟體並開啟MPI，請執行下列命令。

> **Note**
>
> 從 EFA 1.30.0 開始，預設會安裝「開啟 MPI 4」和「開啟 MPI 5」。您可以選擇性地指定要安裝MPI的 Open 版本。要僅安裝打開 MPI 4，包括`--mpi=openmpi4`。要僅安裝打開 MPI 5，包括`--mpi=openmpi5`。若要同時安裝這兩種版本，請忽略 `--mpi` 選項。

```
$ sudo ./efa_installer.sh -y
```

Libfabric 會安裝到 `/opt/amazon/efa`。打開 MPI 4 安裝到`/opt/amazon/openmpi`。打開 MPI 5 安裝到`/opt/amazon/openmpi5`。

Open MPI only

如果您打算在「MPI僅EFA開啟」中使用，則必須使用 Libfabric 和「開啟」來安裝EFA軟體MPI，而且您可以略過「步驟 5：安裝 Intel MPI」。若要使用 Libfabric 安裝EFA軟體並開啟MPI，請執行下列命令。

> **Note**
>
> 從 EFA 1.30.0 開始，預設會安裝「開啟 MPI 4」和「開啟 MPI 5」。您可以選擇性地指定要安裝MPI的 Open 版本。要僅安裝打開 MPI 4，包括`--mpi=openmpi4`。要僅安裝打開 MPI 5，包括`--mpi=openmpi5`。若要同時安裝這兩種版本，請忽略 `--mpi` 選項。

```
$ sudo ./efa_installer.sh -y
```

Libfabric 會安裝到 `/opt/amazon/efa`。打開 MPI 4 安裝到`/opt/amazon/openmpi`。打開 MPI 5 安裝到`/opt/amazon/openmpi5`。

Intel MPI only

如果您MPI只打算EFA與 Intel 搭配使用，您可以在沒有 Libfabric 和開啟MPI的情況下安裝EFA軟體。在這種情況下，英特爾MPI使用其嵌入式利布。如果您選擇這樣做，您必須完成步驟 5：安裝 Intel MPI。

若要在不使用 Libfabric 並開啟的情況下安裝EFA軟體MPI，請執行下列命令。

```
$ sudo ./efa_installer.sh -y --minimal
```

8. 如果EFA安裝程式提示您重新啟動執行個體，請執行此動作，然後重新連線至執行個體。否則，請登出執行個體，然後重新登入以完成安裝。

步驟 4：（可選）啟用打開 MPI 5

> **Note**
>
> 只有在您打算使用「開啟 MPI 5」時才執行此步驟。

從 EFA 1.30.0 開始，預設會安裝「開啟 MPI 4」和「開啟 MPI 5」。或者，您可以選擇僅安裝「打開 MPI 4」或「打開 MPI 5」。

如果您選擇在步驟 3：安裝EFA軟體中安裝 Open MPI 5，並且打算使用它，則必須執行以下步驟來啟用它。

啟用「開啟 MPI 5」

1. 將 Open MPI 5 加入至PATH環境變數。

```
$ module load openmpi5
```

2. 確認已啟用「開啟 MPI 5」以供使用。

```
$ which mpicc
```

   該命令應返回 Open MPI 5 安裝目錄-`/opt/amazon/openmpi5`。

3. (選擇性) 若要確保每次執行個體啟動時都會將 Open MPI 5 新增至PATH環境變數，請執行下列動作：

   bash shell

   將 `module load openmpi5` 新增至 `/home/`*`username`*`/.bashrc` 和 `/home/`*`username`*`/.bash_profile`。

   csh and tcsh shells

   將 `module load openmpi5` 新增至 `/home/`*`username`*`/.cshrc`。

如果您需要從PATH環境變數中移除 Open MPI 5，請執行下列命令，並從 shell 啟動程序檔中移除該命令。

```
$ module unload openmpi5
```

步驟 5：（可選）安裝英特爾 MPI

> **Important**
>
> 只有在您打算使用 Intel 時才執行此步驟MPI。如果您只打算使用「打開」MPI，請跳過此步驟。

Intel MPI 需要額外的安裝與環境變數組態。

先決條件

確保執行下列步驟的使用者具有 sudo 許可。

若要安裝英特爾 MPI

1. 若要下載 Intel MPI 安裝指令碼，請執行下列動作
   a. 請造訪 Intel 網站。
   b. 在網頁的「Intel 程式MPI庫」區段中，選擇適用於 Linux 的 Intel 程式MPI庫離線安裝程式的連結。
2. 執行您在上一個步驟中下載的安裝指令碼。

```
$ sudo bash installation_script_name.sh
```

3. 在安裝程式中，選擇 Accept & install (接受並安裝)。
4. 閱讀 Intel 改進計畫，選擇適當的選項，然後選擇 Begin Installation (開始安裝)。
5. 完成安裝時，請選擇 Close (關閉)。
6. 根據預設，Intel MPI 會使用其內嵌式 (內部) 網格架構。您可以MPI將 Intel 設定為使用隨EFA安裝程式隨附的 Libfabric。通常情況下，EFA安裝程式隨附的 Libfabric 版本比英特爾MPI更新。在某些情況下，EFA安裝程式隨附的 Libfabric 比 Intel 更具效能。MPI若要設定 Intel MPI 使用隨EFA安裝程式隨附的 Libfabric，請根據您的殼層執行下列其中一項動作。

   bash shells

   將以下陳述式新增至 `/home/username/.bashrc` 和 `/home/username/.bash_profile`。

```
export I_MPI_OFI_LIBRARY_INTERNAL=0
```

csh and tcsh shells

將以下陳述式新增至 /home/*username*/.cshrc。

```
setenv I_MPI_OFI_LIBRARY_INTERNAL 0
```

7. 新增下列 source 命令到您的 shell 指令碼，從安裝目錄中獲取 vars.sh 的指令碼，從而在每次啟動執行個體時設定編譯器環境。根據 shell 而定，執行以下其中一個命令。

bash shells

將以下陳述式新增至 /home/*username*/.bashrc 和 /home/*username*/.bash_profile。

```
source /opt/intel/oneapi/mpi/latest/env/vars.sh
```

csh and tcsh shells

將以下陳述式新增至 /home/*username*/.cshrc。

```
source /opt/intel/oneapi/mpi/latest/env/vars.csh
```

8. 根據預設，如果EFA因MPI為設定錯誤而無法使用，Intel 會預設使用 TCP /IP 網路堆疊，這可能會導致應用程式效能變慢。透過將 I_MPI_OFI_PROVIDER 設定為 efa，可防止此情況發生。如EFA果無法使MPI用，這會導致 Intel 失敗，並出現下列錯誤：

```
Abort (XXXXXX) on node 0 (rank 0 in comm 0): Fatal error in PMPI_Init: OtherMPI
 error,
MPIR_Init_thread (XXX)........:
MPID_Init (XXXX)..............:
MPIDI_OFI_mpi_init_hook (XXXX):
open_fabric (XXXX)............:
find_provider (XXXX)..........:
OFI fi_getinfo() failed (ofi_init.c:2684:find_provider:
```

根據 shell 而定，執行以下其中一個命令。

bash shells

將以下陳述式新增至 /home/*username*/.bashrc 和 /home/*username*/.bash_profile。

```
export I_MPI_OFI_PROVIDER=efa
```

csh and tcsh shells

將以下陳述式新增至 /home/*username*/.cshrc。

```
setenv I_MPI_OFI_PROVIDER efa
```

9. 依預設，Intel MPI 不會列印除錯資訊。您可以指定不同的詳細資訊等級來控制偵錯資訊。可能的值 (依提供的詳細資訊量排序) 為：0 (預設)、1、2、3、4、5。等級 1 和更高等級會列印 `libfabric version` 和 `libfabric provider`。用`libfabric version`來檢查 Intel MPI 是否正在使用安裝程式隨附的內部 Libfabric 或是隨附於安裝程式的 Libfabric。EFA如果它使用的是內部 Libfabric，則版本字尾為 `impi`。用`libfabric provider`於檢查 Intel MPI 正在使用EFA或 TCP /IP 網路。如果它正在使用EFA，則值為`efa`。如果使用 TCP /IP，則值為`tcp;ofi_rxm`。

   若要啟用偵錯資訊，請根據您的 shell 執行以下其中一個命令。

   bash shells

   將以下陳述式新增至 /home/*username*/.bashrc 和 /home/*username*/.bash_profile。

   ```
   export I_MPI_DEBUG=value
   ```

   csh and tcsh shells

   將以下陳述式新增至 /home/*username*/.cshrc。

   ```
   setenv I_MPI_DEBUG value
   ```

10. 根據預設，Intel MPI 會使用作業系統的共用記憶體 (`shm`) 進行節點內部通訊，而且僅使用 Libfabric (`ofi`) 進行節點間通訊。通常，此組態可提供最佳效能。不過，在某些情況下，Intel MPI shm 網狀架構可能會造成某些應用程式無限期懸置。

若要解決這個問題，您可以強制 Intel 使MPI用 Libfabric 進行節點內部與節點間通訊。為此，請根據您的 shell，執行以下其中一個命令。

bash shells

將以下陳述式新增至 `/home/username/.bashrc` 和 `/home/username/.bash_profile`。

```
export I_MPI_FABRICS=ofi
```

csh and tcsh shells

將以下陳述式新增至 `/home/username/.cshrc`。

```
setenv I_MPI_FABRICS ofi
```

> **Note**
>
> EFALibfabric 提供者會使用作業系統的共用記憶體進行節點內部通訊。這表示將 `I_MPI_FABRICS` 設定為 `ofi` 可產生與預設 `shm:ofi` 組態類似的效能。

11. 登出執行個體，然後重新登入。

如果您不想再使用 IntelMPI，請從殼層啟動程序檔中移除環境變數。

步驟 6：停用 ptrace 保護

為了改善HPC應用程式的效能，當處理序在同一個執行個體上執行時，Libfabric 會使用執行個體的本機記憶體進行處理序間通訊。

共用記憶體功能使用跨記憶體連接 (CMA)，ptrace 保護不支援此功能。如果您使用的 Linux 發行套件依預設已啟用 ptrace 防護，例如 Ubuntu，您必須停用該保護。如果您的 Linux 發行套件依預設未啟用 ptrace 保護，請略過此步驟。

停用 ptrace 保護

請執行下列其中一項：

- 如果出於測試用途要暫時停用 ptrace 保護，請執行下列命令。

```
$ sudo sysctl -w kernel.yama.ptrace_scope=0
```

- 若要永久停用 ptrace 保護，請將 `kernel.yama.ptrace_scope = 0` 新增至 `/etc/sysctl.d/10-ptrace.conf`，並重新啟動執行個體。

步驟 7. 確認安裝

確認已成功安裝

1. 若要確認MPI已成功安裝，請執行下列命令：

   ```
   $ which mpicc
   ```

   - 對於 OpenMPI，返回的路徑應包括 `/opt/amazon/`
   - 對於 Intel 而言MPI，傳回的路徑應該包含`/opt/intel/`。如果您沒有得到預期的輸出，請確定您已採購 Intel 指MPI`vars.sh`令碼。
2. 若要確認EFA軟體元件和 Libfabric 已成功安裝，請執行下列命令。

   ```
   $ fi_info -p efa -t FI_EP_RDM
   ```

   該命令應該返回有關 LibFabric EFA 接口的信息。以下範例顯示命令輸出。

   ```
   provider: efa
       fabric: EFA-fe80::94:3dff:fe89:1b70
       domain: efa_0-rdm
       version: 2.0
       type: FI_EP_RDM
       protocol: FI_PROTO_EFA
   ```

步驟 8：安裝您的HPC應用程式

在臨時實例上安裝HPC應用程序。安裝程序會根據特定HPC應用程式而有所不同。如需詳細資訊，請參閱 Amazon Linux 2 使用者指南中的管理AL2執行個體上的軟體。

> **Note**
>
> 如需安裝說明，請參閱HPC應用程式的文件。

步驟 9：建立EFA已啟用 AMI

安裝必要的軟體元件之後，您可以建立可重複使用AMI以啟動已啟用執行個體的EFA執行個體。

若要AMI從暫存執行個體建立

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取您建立的暫時執行個體，然後選取 Actions (動作)、Image (映像)、Create image (建立映像)。
4. 對於 Create image (建立映像)，執行下列動作：
   a. 在「影像名稱」中，輸入的描述性名稱AMI。
   b. (選擇性) 在影像說明中，輸入用途的簡短描述AMI。
   c. 選擇 Create image (建立映像)。
5. 在導覽窗格中，選擇AMIs。
6. 找到您在列表中創建的 AMI tht。等待狀態從 `pending` 變為 `available`，再繼續進行下一個步驟。

步驟 10：在叢集置放群組中EFA啟動已啟用的執行個體

使用您在步驟 7 中建立的EFA已啟用，以及您在步驟 1 中建立的EFA已AMI啟用安全性群組，將EFA已啟用的執行個體啟動到叢集置放群組。

> **Note**
>
> - 將EFA已啟用的執行個體啟動到叢集配置群組並不是絕對需求。不過，我們建議您在叢集置放群組中執行EFA已啟用的執行個體，因為它會將執行個體啟動到單一可用區域中的低延遲群組。
> - 要確保在擴展叢集執行個體時容量可用，您可以為集群放置群組建立容量保留。如需詳細資訊，請參閱在叢集置放群組中建立容量保留區。

啟動執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)，接著選擇 Launch Instances (啟動執行個體) 來開啟新的啟動執行個體精靈。
3. (選用) 在 Name and tags (名稱和標籤) 區段中，為執行個體命名，例如，`EFA-instance`。該名稱將指派作為執行個體的資源標籤 (Name=*`EFA-instance`*)。
4. 在 [應用程式和作業系統映像] 區段中AMIs，選擇 [我的]，然後選取AMI您在上一個步驟中建立的。
5. 在 Instance type (執行個體類型) 區段中，選取支援的執行個體類型。
6. 在 Key pair (金鑰對) 區段中，選取要用於執行個體的金鑰對。
7. 在 Network settings (網路設定) 區段，選擇 Edit (編輯)，接著執行下列動作：
   a. 對於 Subnet (子網)，請選擇要在其中啟動執行個體的子網。如果您未選取子網路，則無法啟用的執行個體EFA。
   b. 針對 Firewall (security groups) (防火牆 (安全群組))，選取 Select existing security group (選取現有的安全群組)，接著選取您在前一個步驟中建立的安全群組。
   c. 展開 Advanced network configuration (進階網路組態) 區段，並針對 Elastic Fabric Adapter，選取 Enable (啟用)。
8. （選用）在 儲存 區段中，根據需求設定磁碟區。
9. 在 進階詳細資訊 區段中，針對 配置群組名稱，選取要在其中啟動執行個體的集群放置群組。如果您需要建立新的集群放置群組，請選取 建立新的配置群組。
10. 在右側的「摘要」面板中，針對「執行個體數目」，輸入您要啟動的EFA已啟用執行個體數目，然後選擇 Launch 執行個體。

步驟 11：終止暫時執行個體

此時，您不再需要在步驟 2 中啟動的執行個體。您可以終止執行個體，以停止產生費用。

終止暫時執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取您建立的暫時執行個體，然後選取動作、執行個體狀態、終止執行個體。
4. 出現確認提示時，請選擇終止。

步驟 12：啟用無密碼 SSH

若要讓應用程式能夠在叢集中的所有執行個體上執行，您必須啟用從領導節點到成員節點的無密碼SSH存取。領導節點是您在其中執行應用程式的執行個體。叢集中的其餘執行個體為成員節點。

若要在叢集中的執行個體SSH之間啟用無密碼

1. 選取叢集中的一個執行個體作為領導節點，並與它連線。
2. 在領導節點上停用 `strictHostKeyChecking` 並啟用 `ForwardAgent`。使用您偏好的文字編輯器開啟 `~/.ssh/config`，然後新增下列命令。

   ```
   Host *
       ForwardAgent yes
   Host *
       StrictHostKeyChecking no
   ```

3. 產生 RSA key pair。

   ```
   $ ssh-keygen -t rsa -N "" -f ~/.ssh/id_rsa
   ```

   在 `$HOME/.ssh/` 目錄中建立金鑰對。
4. 變更領導節點上私有金鑰的許可。

   ```
   $ chmod 600 ~/.ssh/id_rsa
   chmod 600 ~/.ssh/config
   ```

5. 使用您偏好的文字編輯器開啟 `~/.ssh/id_rsa.pub` 並複製該金鑰。
6. 對於叢集中的每個成員節點，執行以下操作：

   a. 連線到執行個體。

   b. 使用您偏好的文字編輯器開啟 `~/.ssh/authorized_keys`，然後新增您先前複製的公有金鑰。
7. 若要測試無密碼SSH是否如預期般運作，請連接至您的領導節點並執行下列命令。

   ```
   $ ssh member_node_private_ip
   ```

   您應該連線至成員節點，而不會提示您輸入金鑰或密碼。

在 Amazon 上開始使用EFA和處NCCL理機器學習工作負載 EC2

NVIDIA集體通訊程式庫 (NCCL) 是一個標準集體通訊常式程式庫，用於GPUs跨單一節點或多個節點的多個。NCCL可與 Libfabric 一起使用EFA，並MPI支援各種機器學習工作負載。如需詳細資訊，請參閱NCCL網站。

下列步驟可協助您開始NCCL使用其中一個受支援的作業系統EFA並使用基礎AMI。

> **Note**
>
> - 只支援 `p3dn.24xlarge`、`p4d.24xlarge` 和 `p5.48xlarge` 執行個體類型。
> - 僅支持 Amazon Linux 2 和 21 月 22 日基礎AMIs。
> - 僅支援 NCCL 2.4.2 及更新版本。EFA
> - 如需有關NCCL使用EFA和使用執行機器學習工作負載的詳細資訊 AWS Deep Learning AMI，請參閱AWS Deep Learning AMI 開發人員指南DLAMI中的〈使用EFA〉。

步驟

- 步驟 1：準備EFA已啟用的安全性群組
- 步驟 2：啟動暫時執行個體
- 第 3 步：安裝 Nvidia GPU 驅動程序，Nvidia CUDA 工具包和 cu DNN
- 步驟 4：安裝 GDRCopy
- 步驟 5：安裝EFA軟件
- 步驟 6：安裝 NCCL
- 步驟 7：安裝 aws-ofi-nccl 插件
- 步驟 8：安裝測NCCL試
- 步驟 9：測試您EFA的NCCL配置
- 步驟 10：安裝您的機器學習應用程式
- 第 11 步：創建一個EFA和NCCL啟用 AMI
- 步驟 12：終止暫時執行個體
- 步驟 13：在叢集置放群組中啟動EFA並啟NCCL用執行個體
- 步驟 14：啟用無密碼 SSH

步驟 1：準備EFA已啟用的安全性群組

EFA需要安全性群組，該群組允許進出安全性群組本身的所有輸入和輸出流量。下列程序會建立安全性群組，允許所有入站和出站流量進出，並允許來自任何IPv4位址的輸入SSH流量進行SSH連線。

> **Important**
>
> 此安全群組僅供測試之用。對於生產環境，我們建議您建立輸入SSH規則，該規則僅允許來自您連線的 IP 位址的流量，例如電腦的 IP 位址或區域網路中的一系列 IP 位址。

如需其他案例，請參閱不同使用案例的安全群組規則。

若要建立EFA已啟用的安全性群組

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Security Groups (安全群組)，然後選擇 Create Security Group (建立安全群組)。
3. 在 Create Security Group (建立安全群組) 視窗中，執行下列動作：
   a. 對於 Security group name (安全群組名稱)，輸入安全群組的描述性名稱，例如 `EFA-enabled security group`。
   b. (選用) 對於 Description (描述)，輸入安全群組的簡短描述。
   c. 在VPC中 VPC，選取您要啟用已啟EFA用執行個體的目標。
   d. 選擇 Create Security Group (建立安全群組)。
4. 選取您建立的安全群組，在 Details (詳細資料) 索引標籤上，複製 Security group ID (安全群組 ID)。
5. 在仍然選取安全群組的情況下，依序選取 Actions (動作) 和 Edit inbound rules (編輯傳入規則)，然後執行下列動作。
   a. 選擇新增規則。
   b. 針對類型，選擇所有流量。
   c. 若為 Source type (來源類型)，選擇 Custom (自訂)，然後將您複製的安全群組 ID 貼到欄位中。
   d. 選擇新增規則。
   e. 對於「類型」，選擇SSH。

f. 針對「來源類型」，選擇「隨處-」IPv4。

g. 選擇儲存規則。

6. 在仍然選取安全群組的情況下，依序選取 Actions (動作) 和 Edit outbound rules (編輯傳出規則)，然後執行下列動作。

   a. 選擇新增規則。

   b. 針對類型，選擇所有流量。

   c. 若為 Destination type (目的地類型)，選擇 Custom (自訂)，然後將您複製的安全群組 ID 貼到欄位中。

   d. 選擇儲存規則。

步驟 2：啟動暫時執行個體

啟動可用來安裝和設定EFA軟體元件的暫時執行個體。您可以使用此執行個體建立EFA已啟用的執行個體，AMI從中啟動EFA已啟用的執行個體。

啟動暫時執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)，接著選擇 Launch Instances (啟動執行個體) 來開啟新的啟動執行個體精靈。
3. (選用) 在 Name and tags (名稱和標籤) 區段中，為執行個體命名，例如，`EFA-instance`。該名稱將指派作為執行個體的資源標籤 (Name=*`EFA-instance`*)。
4. 在「應用程式和作業系統映像」區段中，AMI為其中一個支援的作業系統選取一個。僅支持 Amazon Linux 2，Ubuntu 20.04 和 Ubuntu 22.04。
5. 在執行個體類型區段中，選取 `p3dn.24xlarge`、`p4d.24xlarge` 或 `p5.48xlarge`。
6. 在 Key pair (金鑰對) 區段中，選取要用於執行個體的金鑰對。
7. 在 Network settings (網路設定) 區段，選擇 Edit (編輯)，接著執行下列動作：

   a. 對於 Subnet (子網)，請選擇要在其中啟動執行個體的子網。如果您未選取子網路，則無法啟用的執行個體EFA。

   b. 針對 Firewall (security groups) (防火牆 (安全群組))，選取 Select existing security group (選取現有的安全群組)，接著選取您在前一個步驟中建立的安全群組。

   c. 展開 Advanced network configuration (進階網路組態) 區段，並針對 Elastic Fabric Adapter，選取 Enable (啟用)。

8. 在 儲存 區段中，根據需求設定磁碟區。

> **Note**
>
> 您必須為 Nvidia CUDA 工具組提供額外的 10 至 20 GiB 儲存空間。如果您沒有佈建足夠的儲存空間，您會在嘗試安裝 Nvidia 驅動程式和CUDA工具組時收到`insufficient disk space`錯誤訊息。

9. 在右邊的 Summary (摘要) 面板中，選擇 Launch instance (啟動執行個體)。

第 3 步：安裝 Nvidia GPU 驅動程序，Nvidia CUDA 工具包和 cu DNN

Amazon Linux 2

要安裝 Nvidia GPU 驅動程序，Nvidia CUDA 工具包和 cu DNN

1. 為確保所有軟體套件皆為最新版本，請對您的執行個體執行快速軟體更新。

```
$ sudo yum upgrade -y && sudo reboot
```

   重新啟動執行個體後，請重新與它連線。

2. 安裝安裝 Nvidia 驅動程式和 N GPU vidia CUDA 工具組所需的公用程式。

```
$ sudo yum groupinstall 'Development Tools' -y
```

3. 停用 `nouveau` 開放原始碼驅動程式。

   a. 針對您目前執行的核心版本，安裝需要的公用程式和核心標頭套件。

```
$ sudo yum install -y wget kernel-devel-$(uname -r) kernel-headers-$(uname -
r)
```

   b. 新增 `nouveau` 至 `/etc/modprobe.d/blacklist.conf` 拒絕清單檔案。

```
$ cat << EOF | sudo tee --append /etc/modprobe.d/blacklist.conf
blacklist vga16fb
blacklist nouveau
blacklist rivafb
blacklist nvidiafb
blacklist rivatv
```

```
EOF
```

c. 將 `GRUB_CMDLINE_LINUX="rdblacklist=nouveau"` 附加至 `grub` 檔案並重建 Grub 組態。

```
$ echo 'GRUB_CMDLINE_LINUX="rdblacklist=nouveau"' | sudo tee -a /etc/
default/grub \
&& sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

4. 重新啟動執行個體並重新連線至其中。
5. 準備所需的儲存庫

a. 為您的 Linux 發行版本安裝EPEL存放庫DKMS並啟用任何選用的存放庫。

```
$ sudo yum install -y https://dl.fedoraproject.org/pub/epel/epel-release-
latest-7.noarch.rpm
```

b. 安裝CUDA儲存庫公開GPG金鑰。

```
$ distribution='rhel7'
```

c. 設定CUDA網路儲存庫並更新儲存庫快取。

```
$ ARCH=$( /bin/arch ) \
&& sudo yum-config-manager --add-repo http://developer.download.nvidia.com/
compute/cuda/repos/$distribution/${ARCH}/cuda-$distribution.repo \
&& sudo yum clean expire-cache
```

d. (僅限核心版本 5.10) 唯有當您使用 Amazon Linux 2 搭配核心版本 5.10 時，才執行下列步驟。如果您是使用 Amazon Linux 2 搭配核心版本 4.12，請跳過下列步驟。若要檢查核心版本，請執行 uname -r。

i. 建立名為 `/etc/dkms/nvidia.conf` 的 Nvidia 驅動程式組態檔案。

```
$ sudo mkdir -p /etc/dkms \
&& echo "MAKE[0]=\"'make' -j2 module SYSSRC=\${kernel_source_dir}
 IGNORE_XEN_PRESENCE=1 IGNORE_PREEMPT_RT_PRESENCE=1 IGNORE_CC_MISMATCH=1
 CC=/usr/bin/gcc10-gcc\"" | sudo tee /etc/dkms/nvidia.conf
```

ii. (僅限 `p4d.24xlarge` 和 `p5.48xlarge`) 複製 Nvidia 驅動程序組態檔案。

```
$ sudo cp /etc/dkms/nvidia.conf /etc/dkms/nvidia-open.conf
```

6. 安裝 Nvidia GPU 驅動程序，NVIDIACUDA工具包和 cu DNN。

   - `p3dn.24xlarge`

     ```
     $ sudo yum clean all \
     && sudo yum -y install kmod-nvidia-latest-dkms nvidia-driver-latest-dkms \
     && sudo yum -y install cuda-drivers-fabricmanager cuda libcudnn8-devel
     ```

   - `p4d.24xlarge` 和 `p5.48xlarge`

     ```
     $ sudo yum clean all \
     && sudo yum -y install kmod-nvidia-open-dkms nvidia-driver-latest-dkms \
     && sudo yum -y install cuda-drivers-fabricmanager cuda libcudnn8-devel
     ```

7. 重新啟動執行個體並重新連線至其中。
8. (僅限 `p4d.24xlarge` 和 `p5.48xlarge`) 啟動 Nvidia Fabric Manager 服務，並確保它會在執行個體啟動時自動啟動。NV Switch Management 需要使用 Nvidia Fabric Manager。

   ```
   $ sudo systemctl enable nvidia-fabricmanager && sudo systemctl start nvidia-
   fabricmanager
   ```

9. 確保每次執行個體啟動時都已設定CUDA路徑。

   - 針對 bash shell，新增以下陳述式到 /home/*username*/.bashrc 和 /home/*username*/.bash_profile。

     ```
     export PATH=/usr/local/cuda/bin:$PATH
     export LD_LIBRARY_PATH=/usr/local/cuda/lib64:/usr/local/cuda/extras/CUPTI/
     lib64:$LD_LIBRARY_PATH
     ```

   - 針對 tcsh shell，新增以下陳述式到 /home/*username*/.cshrc。

     ```
     setenv PATH=/usr/local/cuda/bin:$PATH
     setenv LD_LIBRARY_PATH=/usr/local/cuda/lib64:/usr/local/cuda/extras/CUPTI/
     lib64:$LD_LIBRARY_PATH
     ```

10. 要確認 Nvidia GPU 驅動程序是否正常工作，請運行以下命令。

```
$ nvidia-smi -q | head
```

該命令應該返回有關 NvidiaGPUs，Nvidia GPU 驅動程序和 Nvidia CUDA 工具包的信息。

Ubuntu 20.04/22.04

要安裝 Nvidia GPU 驅動程序，Nvidia CUDA 工具包和 cu DNN

1. 為確保所有軟體套件皆為最新版本，請對您的執行個體執行快速軟體更新。

```
$ sudo apt-get update && sudo apt-get upgrade -y
```

2. 安裝安裝 Nvidia 驅動程式和 N GPU vidia CUDA 工具組所需的公用程式。

```
$ sudo apt-get update && sudo apt-get install build-essential -y
```

3. 若要使用 Nvidia GPU 驅動程式，您必須先停用nouveau開放原始碼驅動程式。

   a. 針對您目前執行的核心版本，安裝需要的公用程式和核心標頭套件。

   ```
   $ sudo apt-get install -y gcc make linux-headers-$(uname -r)
   ```

   b. 新增 nouveau 至 `/etc/modprobe.d/blacklist.conf` 拒絕清單檔案。

   ```
   $ cat << EOF | sudo tee --append /etc/modprobe.d/blacklist.conf
   blacklist vga16fb
   blacklist nouveau
   blacklist rivafb
   blacklist nvidiafb
   blacklist rivatv
   EOF
   ```

   c. 使用您偏好的文字編輯器開啟 `/etc/default/grub`，然後新增下列命令。

   ```
   GRUB_CMDLINE_LINUX="rdblacklist=nouveau"
   ```

   d. 重建 Grub 組態。

   ```
   $ sudo update-grub
   ```

4. 重新啟動執行個體並重新連線至其中。
5. 添加存CUDA儲庫並安裝 Nvidia GPU 驅動程序，NVIDIACUDA工具包和 cu DNN。

   - `p3dn.24xlarge`

```
$ sudo apt-key adv --fetch-keys http://developer.download.nvidia.com/compute/
machine-learning/repos/ubuntu2004/x86_64/7fa2af80.pub \
&& wget -O /tmp/deeplearning.deb http://developer.download.nvidia.com/compute/
machine-learning/repos/ubuntu2004/x86_64/nvidia-machine-learning-repo-
ubuntu2004_1.0.0-1_amd64.deb \
&& sudo dpkg -i /tmp/deeplearning.deb \
&& wget -O /tmp/cuda.pin https://developer.download.nvidia.com/compute/cuda/
repos/ubuntu2004/x86_64/cuda-ubuntu2004.pin \
&& sudo mv /tmp/cuda.pin /etc/apt/preferences.d/cuda-repository-pin-600 \
&& sudo apt-key adv --fetch-keys https://developer.download.nvidia.com/
compute/cuda/repos/ubuntu2004/x86_64/3bf863cc.pub \
&& sudo add-apt-repository 'deb http://developer.download.nvidia.com/compute/
cuda/repos/ubuntu2004/x86_64/ /' \
&& sudo apt update \
&& sudo apt install nvidia-dkms-535 \
&& sudo apt install -o Dpkg::Options::='--force-overwrite' cuda-drivers-535
 cuda-toolkit-12-3 libcudnn8 libcudnn8-dev -y
```

   - `p4d.24xlarge` 和 `p5.48xlarge`

```
$ sudo apt-key adv --fetch-keys http://developer.download.nvidia.com/compute/
machine-learning/repos/ubuntu2004/x86_64/7fa2af80.pub \
&& wget -O /tmp/deeplearning.deb http://developer.download.nvidia.com/compute/
machine-learning/repos/ubuntu2004/x86_64/nvidia-machine-learning-repo-
ubuntu2004_1.0.0-1_amd64.deb \
&& sudo dpkg -i /tmp/deeplearning.deb \
&& wget -O /tmp/cuda.pin https://developer.download.nvidia.com/compute/cuda/
repos/ubuntu2004/x86_64/cuda-ubuntu2004.pin \
&& sudo mv /tmp/cuda.pin /etc/apt/preferences.d/cuda-repository-pin-600 \
&& sudo apt-key adv --fetch-keys https://developer.download.nvidia.com/
compute/cuda/repos/ubuntu2004/x86_64/3bf863cc.pub \
&& sudo add-apt-repository 'deb http://developer.download.nvidia.com/compute/
cuda/repos/ubuntu2004/x86_64/ /' \
&& sudo apt update \
&& sudo apt install nvidia-kernel-open-535 \
&& sudo apt install -o Dpkg::Options::='--force-overwrite' cuda-drivers-535
 cuda-toolkit-12-3 libcudnn8 libcudnn8-dev -y
```

6. 重新啟動執行個體並重新連線至其中。
7. (僅限 `p4d.24xlarge` 和 `p5.48xlarge`) 安裝 Nvidia Fabric Manager。

   a. 您安裝的 Nvidia Fabric Manager 版本必須與您在上一個步驟中安裝的 Nvidia 核心模組版本一致。

      執行以下命令來判斷 Nvidia 核心模組的版本。

      ```
      $ cat /proc/driver/nvidia/version | grep "Kernel Module"
      ```

      下列為範例輸出。

      ```
      NVRM version: NVIDIA UNIX x86_64 Kernel Module  450.42.01  Tue Jun 15
       21:26:37 UTC 2021
      ```

      在上述範例中，已安裝核心模組主要版本 `450`。這表示您需要安裝 Nvidia Fabric Manager 版本 `450`。

   b. 安裝 Nvidia Fabric Manager。執行以下命令，並指定在上一步驟中使用的主要版本。

      ```
      $ sudo apt install -o Dpkg::Options::='--force-overwrite' nvidia-
      fabricmanager-major_version_number
      ```

      例如，如果已安裝核心模組的主要版本 `450`，請使用下列命令來安裝同樣版本的 Nvidia Fabric Manager。

      ```
      $ sudo apt install -o Dpkg::Options::='--force-overwrite' nvidia-
      fabricmanager-450
      ```

   c. 啟動服務，並確保服務會在執行個體啟動時自動啟動。NV Switch Management 需要使用 Nvidia Fabric Manager。

      ```
      $ sudo systemctl start nvidia-fabricmanager && sudo systemctl enable nvidia-
      fabricmanager
      ```

8. 確保每次執行個體啟動時都已設定CUDA路徑。

   - 針對 bash shell，新增以下陳述式到 `/home/`*`username`*`/.bashrc` 和 `/home/`*`username`*`/.bash_profile`。

```
export PATH=/usr/local/cuda/bin:$PATH
export LD_LIBRARY_PATH=/usr/local/cuda/lib64:/usr/local/cuda/extras/CUPTI/
lib64:$LD_LIBRARY_PATH
```

- 針對 tcsh shell，新增以下陳述式到 /home/*username*/.cshrc。

```
setenv PATH=/usr/local/cuda/bin:$PATH
setenv LD_LIBRARY_PATH=/usr/local/cuda/lib64:/usr/local/cuda/extras/CUPTI/
lib64:$LD_LIBRARY_PATH
```

9. 要確認 Nvidia GPU 驅動程序是否正常工作，請運行以下命令。

```
$ nvidia-smi -q | head
```

該命令應該返回有關 NvidiaGPUs，Nvidia GPU 驅動程序和 Nvidia CUDA 工具包的信息。

步驟 4：安裝 GDRCopy

安裝GDRCopy以提高利布的性能。如需有關的詳細資訊GDRCopy，請參閱GDRCopy存放庫。

Amazon Linux 2

若要安裝 GDRCopy

1. 安裝所需的依存項目。

```
$ sudo yum -y install dkms rpm-build make check check-devel subunit subunit-
devel
```

2. 下載並解壓縮GDRCopy套件。

```
$ wget https://github.com/NVIDIA/gdrcopy/archive/refs/tags/v2.4.tar.gz \
&& tar xf v2.4.tar.gz ; cd gdrcopy-2.4/packages
```

3. 建置GDRCopyRPM套件。

```
$ CUDA=/usr/local/cuda ./build-rpm-packages.sh
```

4. 安裝GDRCopyRPM套件。

```
$ sudo rpm -Uvh gdrcopy-kmod-2.4-1dkms.noarch*.rpm \
&& sudo rpm -Uvh gdrcopy-2.4-1.x86_64*.rpm \
&& sudo rpm -Uvh gdrcopy-devel-2.4-1.noarch*.rpm
```

Ubuntu 20.04/22.04

若要安裝 GDRCopy

1. 安裝所需的依存項目。

```
$ sudo apt -y install build-essential devscripts debhelper check libsubunit-dev
 fakeroot pkg-config dkms
```

2. 下載並解壓縮GDRCopy套件。

```
$ wget https://github.com/NVIDIA/gdrcopy/archive/refs/tags/v2.4.tar.gz \
&& tar xf v2.4.tar.gz \
&& cd gdrcopy-2.4/packages
```

3. 建置GDRCopyRPM套件。

```
$ CUDA=/usr/local/cuda ./build-deb-packages.sh
```

4. 安裝GDRCopyRPM套件。

```
$ sudo dpkg -i gdrdrv-dkms_2.4-1_amd64.*.deb \
&& sudo dpkg -i libgdrapi_2.4-1_amd64.*.deb \
&& sudo dpkg -i gdrcopy-tests_2.4-1_amd64.*.deb \
&& sudo dpkg -i gdrcopy_2.4-1_amd64.*.deb
```

步驟 5：安裝EFA軟件

安EFA裝在暫存執行個體EFA上支援所需的已啟用核心、EFA驅動程式、Libfabric 和 Open MPI 堆疊。

安裝 EFA 軟體

1. 連接至您啟動的執行個體。如需詳細資訊，請參閱使用 Connect 到您的 Linux 執行個體 SSH。

2. 下載 EFA 軟體安裝檔案。軟體安裝檔案已封裝成壓縮 tarball (`.tar.gz`) 檔案。若要下載最新穩定版本，請使用下列命令：

```
$ curl -O https://efa-installer.amazonaws.com/aws-efa-installer-1.34.0.tar.gz
```

   您也可以在之前的命令中以 `latest` 取代版本號碼，以取得最新版本。

3. (選擇性) 驗證 EFA tarball (`.tar.gz`) 檔案的真實性與完整性。

   我們建議您執行這項操作來確認軟體發行者的身分，並檢查檔案自發行以來並未遭到變更或損毀。如果您不想驗證 tarball 檔案，請略過此步驟。

   > **Note**
   >
   > 或者，如果您想要使用MD5或SHA256總和檢查碼來驗證 tarball 檔案，請參閱。使用總和檢查碼驗證EFA安裝程式

   a. 下載公開GPG金鑰並將其匯入您的金鑰圈。

```
$ wget https://efa-installer.amazonaws.com/aws-efa-installer.key && gpg --
import aws-efa-installer.key
```

      命令應傳回金鑰值。請記下金鑰的值，因為下一個步驟將需要它。

   b. 驗證GPG金鑰的指紋。執行以下命令，並指定上一步驟中的金鑰值。

```
$ gpg --fingerprint key_value
```

      命令應傳回與 `4E90 91BC BB97 A96B 26B1 5E59 A054 80B1 DD2D 3CCC` 相同的指紋。如果指紋不匹配，請不要運行EFA安裝腳本，然後聯繫 AWS Support。

   c. 下載簽章檔案並驗證 EFA tarball 檔案的簽章。

```
$ wget https://efa-installer.amazonaws.com/aws-efa-installer-1.34.0.tar.gz.sig
 && gpg --verify ./aws-efa-installer-1.34.0.tar.gz.sig
```

      下面顯示了範例輸出。

```
gpg: Signature made Wed 29 Jul 2020 12:50:13 AM UTC using RSA key ID DD2D3CCC
```

```
gpg: Good signature from "Amazon EC2 EFA <ec2-efa-maintainers@amazon.com>"
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the owner.
Primary key fingerprint: 4E90 91BC BB97 A96B 26B1  5E59 A054 80B1 DD2D 3CCC
```

如果結果包含`Good signature`，而指紋與上一個步驟中傳回的指紋相符，請繼續下一個步驟。如果沒有，請不要運行EFA安裝腳本，然後聯繫 AWS Support。

4. 從壓縮的 `.tar.gz` 檔案中解壓縮檔案，然後導覽至解壓縮的目錄。

```
$ tar -xf aws-efa-installer-1.34.0.tar.gz && cd aws-efa-installer
```

5. 執行 EFA 軟體安裝指令碼。

> **Note**
>
> 從 EFA 1.30.0 開始，預設會安裝「開啟 MPI 4」和「開啟 MPI 5」。除非您需要 Open MPI 5，否則我們建議您僅安裝 Open MPI 4。下列指令只會安裝 Open MPI 4。如果要安裝「打開 MPI 4」和「打開 MPI 5」，請刪除`--mpi=openmpi4`。

```
$ sudo ./efa_installer.sh -y --mpi=openmpi4
```

Libfabric 安裝在`/opt/amazon/efa`目錄中，而打開安裝MPI在`/opt/amazon/openmpi`目錄中。

6. 如果EFA安裝程式提示您重新啟動執行個體，請執行此動作，然後重新連線至執行個體。否則，請登出執行個體，然後重新登入以完成安裝。
7. 確認EFA軟體元件已成功安裝。

```
$ fi_info -p efa -t FI_EP_RDM
```

該命令應該返回有關 LibFabric EFA 接口的信息。以下範例顯示命令輸出。

- `p3dn.24xlarge` 具有單一網路介面

```
provider: efa
fabric: EFA-fe80::94:3dff:fe89:1b70
domain: efa_0-rdm
version: 2.0
```

```
type: FI_EP_RDM
protocol: FI_PROTO_EFA
```

- 具有多個網路介面的 `p4d.24xlarge` 和 `p5.48xlarge`

```
provider: efa
fabric: EFA-fe80::c6e:8fff:fef6:e7ff
domain: efa_0-rdm
version: 111.0
type: FI_EP_RDM
protocol: FI_PROTO_EFA
provider: efa
fabric: EFA-fe80::c34:3eff:feb2:3c35
domain: efa_1-rdm
version: 111.0
type: FI_EP_RDM
protocol: FI_PROTO_EFA
provider: efa
fabric: EFA-fe80::c0f:7bff:fe68:a775
domain: efa_2-rdm
version: 111.0
type: FI_EP_RDM
protocol: FI_PROTO_EFA
provider: efa
fabric: EFA-fe80::ca7:b0ff:fea6:5e99
domain: efa_3-rdm
version: 111.0
type: FI_EP_RDM
protocol: FI_PROTO_EFA
```

步驟 6：安裝 NCCL

安裝NCCL。如需有關的詳細資訊NCCL，請參閱NCCL存放庫。

若要安裝 NCCL

1. 導覽至 `/opt` 目錄。

```
$ cd /opt
```

2. 將官方NCCL存儲庫克隆到實例，然後導航到本地克隆的存儲庫。

```
$ sudo git clone https://github.com/NVIDIA/nccl.git && cd nccl
```

3. 建置並安裝NCCL並指定CUDA安裝目錄。

```
$ sudo make -j src.build CUDA_HOME=/usr/local/cuda
```

步驟 7：安裝 aws-ofi-nccl 插件

該 aws-ofi-nccl 插件將面向連接NCCL的傳輸映射APIs到 Libfabric 的無連接可靠接口。這可讓您在執行NCCL基於應用程式時使用 Libfabric 作為網路提供者。如需有關 aws-ofi-nccl 外掛程式的詳細資訊，請參閱aws-ofi-nccl 儲存庫。

若要安裝 aws-ofi-nccl 外掛程式

1. 導覽至您的主目錄。

```
$ cd $HOME
```

2. 安裝所需的公用程式。

   - Amazon Linux 2

     ```
     $ sudo yum install hwloc-devel
     ```

   - Ubuntu

     ```
     $ sudo apt-get install libhwloc-dev
     ```

3. 下載 aws-ofi-nccl 插件文件。檔案已封裝成壓縮 tarball (`.tar.gz`)。

```
$ wget https://github.com/aws/aws-ofi-nccl/releases/download/v1.11.0-aws/aws-ofi-
nccl-1.11.0-aws.tar.gz
```

4. 從壓縮的 .tar.gz 檔案中解壓縮檔案，然後導覽至解壓縮的目錄。

```
$ tar -xf aws-ofi-nccl-1.11.0-aws.tar.gz && cd aws-ofi-nccl-1.11.0-aws
```

5. 若要產生製作檔案，請執行指`configure`令碼並指定MPI、Libfabric 和CUDA安裝目錄。NCCL

```
$ ./configure --prefix=/opt/aws-ofi-nccl --with-mpi=/opt/amazon/openmpi \
--with-libfabric=/opt/amazon/efa \
--with-cuda=/usr/local/cuda \
--enable-platform-aws
```

6. 將「打開」MPI 目錄添加到PATH變量中。

```
$ export PATH=/opt/amazon/openmpi/bin/:$PATH
```

7. 安裝 aws-ofi-nccl 插件。

```
$ make && sudo make install
```

步驟 8：安裝測NCCL試

安裝NCCL測試。這些NCCL測試可讓您確認NCCL是否已正確安裝，以及是否如預期般運作。如需有關NCCL測試的詳細資訊，請參閱 nccl- test 存放庫。

安裝測NCCL試

1. 導覽至您的主目錄。

```
$ cd $HOME
```

2. 複製官方的 nccl-tests 存放庫至執行個體，並導覽至本地複製的存放庫。

```
$ git clone https://github.com/NVIDIA/nccl-tests.git && cd nccl-tests
```

3. 新增 Libfabric 目錄至 LD_LIBRARY_PATH 變數。

   - Amazon Linux 2

   ```
   $ export LD_LIBRARY_PATH=/opt/amazon/efa/lib64:$LD_LIBRARY_PATH
   ```

   - Ubuntu

   ```
   $ export LD_LIBRARY_PATH=/opt/amazon/efa/lib:$LD_LIBRARY_PATH
   ```

4. 安裝NCCL測試並指定MPINCCL、和CUDA安裝目錄。

```
$ make MPI=1 MPI_HOME=/opt/amazon/openmpi NCCL_HOME=/opt/nccl/build CUDA_HOME=/usr/
local/cuda
```

步驟 9：測試您EFA的NCCL配置

執行測試以確保您的暫存執行個體已針對EFA和設定正確NCCL。

測試您的EFA和NCCL配置

1. 建立主機檔案指定要執行測試的主機。下列命另建立名稱為 `my-hosts` 的主機檔案，包含執行個體本身的參考資料。

   IMDSv2

   ```
   [ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H
    "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
   && curl -H "X-aws-ec2-metadata-token: $TOKEN" -v http://169.254.169.254/latest/
   meta-data/local-ipv4 >> my-hosts
   ```

   IMDSv1

   ```
   [ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/local-ipv4 >> my-
   hosts
   ```

2. 運行測試並指定主機文件（`--hostfile`）和GPUs要使用的數量（`-n`）。下列命令會在執行個體本身的 8 GPUs 上執行`all_reduce_perf`測試，並指定下列環境變數。

   - `FI_EFA_USE_DEVICE_RDMA=1`—（`p4d.24xlarge`僅）使用設備的RDMA功能進行單面和雙面傳輸。
   - `NCCL_DEBUG=INFO` - 啟用詳細的除錯輸出。您也可以指`VERSION`定在測試開始時僅列印NCCL版本，或僅`WARN`接收錯誤訊息。

   有關NCCL測試參數的更多信息，請參閱官方 nccl-test 存儲庫README中的NCCL測試。

   - `p3dn.24xlarge`

   ```
   $ /opt/amazon/openmpi/bin/mpirun \
   ```

```
-x LD_LIBRARY_PATH=/opt/nccl/build/lib:/usr/local/cuda/lib64:/opt/amazon/efa/
lib:/opt/amazon/openmpi/lib:/opt/aws-ofi-nccl/lib:$LD_LIBRARY_PATH \
-x NCCL_DEBUG=INFO \
--hostfile my-hosts -n 8 -N 8 \
--mca pml ^cm --mca btl tcp,self --mca btl_tcp_if_exclude lo,docker0 --bind-to
 none \
$HOME/nccl-tests/build/all_reduce_perf -b 8 -e 1G -f 2 -g 1 -c 1 -n 100
```

- `p4d.24xlarge` 和 `p5.48xlarge`

```
$ /opt/amazon/openmpi/bin/mpirun \
-x FI_EFA_USE_DEVICE_RDMA=1 \
-x LD_LIBRARY_PATH=/opt/nccl/build/lib:/usr/local/cuda/lib64:/opt/amazon/efa/
lib:/opt/amazon/openmpi/lib:/opt/aws-ofi-nccl/lib:$LD_LIBRARY_PATH \
-x NCCL_DEBUG=INFO \
--hostfile my-hosts -n 8 -N 8 \
--mca pml ^cm --mca btl tcp,self --mca btl_tcp_if_exclude lo,docker0 --bind-to
 none \
$HOME/nccl-tests/build/all_reduce_perf -b 8 -e 1G -f 2 -g 1 -c 1 -n 100
```

3. 您可以確認EFA在列印NCCL_DEBUG記錄檔NCCL時作為基礎提供者處於作用中狀態。

```
ip-192-168-2-54:14:14 [0] NCCL INFO NET/OFI Selected Provider is efa*
```

使用 `p4d.24xlarge` 執行個體時會顯示下列其他資訊。

```
ip-192-168-2-54:14:14 [0] NCCL INFO NET/OFI Running on P4d platform, Setting
 NCCL_TOPO_FILE environment variable to /home/ec2-user/install/plugin/share/aws-
ofi-nccl/xml/p4d-24xl-topo.xml
```

步驟 10：安裝您的機器學習應用程式

在暫時執行個體上安裝機器學習應用程式。安裝程序依特定的機器學習應用程式而異。如需在 Linux 執行個體上安裝軟體的詳細資訊，請參閱管理 Amazon Linux 2 執行個體上的軟體。

> **Note**
>
> 參考機器學習應用程式的文件，以取得安裝指示。

第 11 步：創建一個EFA和NCCL啟用 AMI

安裝必要的軟體元件之後，您可以建立可重複使用AMI以啟動已啟用執行個體的EFA執行個體。

若要AMI從暫存執行個體建立

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取您建立的暫時執行個體，然後選取 Actions (動作)、Image (映像)、Create image (建立映像)。
4. 對於 Create image (建立映像)，執行下列動作：
   a. 在「影像名稱」中，輸入的描述性名稱AMI。
   b. (選擇性) 在影像說明中，輸入用途的簡短描述AMI。
   c. 選擇 Create image (建立映像)。
5. 在導覽窗格中，選擇AMIs。
6. 找到您在列表中創建的 AMI tht。等待狀態從 `pending` 變為 `available`，再繼續進行下一個步驟。

步驟 12：終止暫時執行個體

此時，您不再需要您啟動的暫時執行個體。您可以終止執行個體，以停止產生費用。

終止暫時執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取您建立的暫時執行個體，然後選取動作、執行個體狀態、終止執行個體。
4. 出現確認提示時，請選擇終止。

步驟 13：在叢集置放群組中啟動EFA並啟NCCL用執行個體

使用您之前建立的已啟用EFA和NCCL已啟用的安全性群組，將您AMI和EFA已啟用的EFA執行個體啟動到叢集置放群組。

> **Note**
>
> - 將EFA已啟用的執行個體啟動到叢集置放群組並不是絕對需求。不過，我們建議您在叢集置放群組中執行EFA已啟用的執行個體，因為它會將執行個體啟動到單一可用區域中的低延遲群組。
> - 要確保在擴展叢集執行個體時容量可用，您可以為集群放置群組建立容量保留。如需詳細資訊，請參閱在叢集置放群組中建立容量保留區。

New console

啟動暫時執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)，接著選擇 Launch Instances (啟動執行個體) 來開啟新的啟動執行個體精靈。
3. (選用) 在 Name and tags (名稱和標籤) 區段中，為執行個體命名，例如，`EFA-instance`。該名稱將指派作為執行個體的資源標籤 (Name=*`EFA-instance`*)。
4. 在 [應用程式和作業系統映像] 區段中AMIs，選擇 [我的]，然後選取AMI您在上一個步驟中建立的。
5. 在 執行個體類型 區段中，選取 `p3dn.24xlarge` 或者 `p4d.24xlarge`。
6. 在 Key pair (金鑰對) 區段中，選取要用於執行個體的金鑰對。
7. 在 Network settings (網路設定) 區段，選擇 Edit (編輯)，接著執行下列動作：
   a. 對於 Subnet (子網)，請選擇要在其中啟動執行個體的子網。如果您未選取子網路，則無法啟用的執行個體EFA。
   b. 針對 Firewall (security groups) (防火牆 (安全群組))，選取 Select existing security group (選取現有的安全群組)，接著選取您在前一個步驟中建立的安全群組。
   c. 展開 Advanced network configuration (進階網路組態) 區段，並針對 Elastic Fabric Adapter，選取 Enable (啟用)。
8. （選用）在 儲存 區段中，根據需求設定磁碟區。
9. 在 進階詳細資訊 區段中，針對 置放群組名稱，選取要在其中啟動執行個體的集群放置群組。如果您需要建立新的集群放置群組，請選取 建立新的配置群組。
10. 在右側的「摘要」面板中，針對「執行個體數目」，輸入您要啟動的EFA已啟用執行個體數目，然後選擇 Launch 執行個體。

Old console

將您EFA和NCCL已啟用的執行個體啟動到叢集置放群組

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 選擇 Launch Instance (啟動執行個體)。
3. 在 [選擇AMI頁面] 上，選擇 [我]AMIs，尋找AMI您先前建立的，然後選擇 [選取]。
4. 在 Choose an Instance Type (選取執行個體類型) 頁面上，選取 p3dn.24xlarge，然後選取 Next: Configure Instance Details (下一步：設定執行個體詳細資訊)。
5. 在 Configure Instance Details (設定執行個體詳細資訊) 頁面上，執行下列操作：
   a. 在執行個體數目中，輸入您要啟動的執行個體數目EFA和NCCL已啟用的執行個體數目。
   b. 對於網路和子網路，請選取要啟動執行個體的子網路VPC和子網路。
   c. 對於 Placement group (置放群組)，選取 Add instance to placement group (將執行個體新增至置放群組)。
   d. 對於 Placement group name (置放群組名稱)，選取 Add to a new placement group (新增至新的置放群組)，然後輸入置放群組的描述性名稱。然後為 Placement group strategy (置放群組策略) 選取 cluster (叢集)。
   e. 對於 EFA，選擇 Enable (啟用)。
   f. 在 Network Interfaces (網路介面) 區段中，針對裝置 eth0，選擇 New network interface (新網路介面)。您可以選擇性地指定主要位IPv4址和一或多個次要IPv4位址。如果您要將執行個體啟動至具有關聯IPv6CIDR區塊的子網路，您可以選擇性地指定主要位IPv6址和一或多個次要IPv6位址。
   g. 選擇 Next: Add Storage (下一步：新增儲存體)。
6. 在 [新增儲存體] 頁面上，除了指定的磁碟區 AMI (例如根裝置磁碟區) 之外，指定要連接至執行個體的磁碟區。然後選擇 Next: Add Tags (下一步：新增標籤)。
7. 在 Add Tags (新增標籤) 頁面上，為執行個體指定標籤 (例如使用者易記的名稱)，然後選擇 Next: Configure Security Group (下一步：設定安全群組)。
8. 在 Configure Security Group (設定安全群組) 頁面上，針對 Assign a security group (指派安全群組)，選取 Select an existing security group (選取現有安全群組)，然後選取您之前建立的安全群組。
9. 選擇 Review and Launch (檢閱和啟動)。
10. 在 Review Instance Launch (檢閱執行個體啟動) 頁面上，檢閱設定，然後選擇 Launch (啟動)，以選擇金鑰對並啟動執行個體。

步驟 14：啟用無密碼 SSH

若要讓應用程式能夠在叢集中的所有執行個體上執行，您必須啟用從領導節點到成員節點的無密碼SSH存取。領導節點是您在其中執行應用程式的執行個體。叢集中的其餘執行個體為成員節點。

若要在叢集中的執行個體SSH之間啟用無密碼

1. 選取叢集中的一個執行個體作為領導節點，並與它連線。
2. 在領導節點上停用 `strictHostKeyChecking` 並啟用 `ForwardAgent`。使用您偏好的文字編輯器開啟 `~/.ssh/config`，然後新增下列命令。

   ```
   Host *
       ForwardAgent yes
   Host *
       StrictHostKeyChecking no
   ```

3. 產生 RSA key pair。

   ```
   $ ssh-keygen -t rsa -N "" -f ~/.ssh/id_rsa
   ```

   在 `$HOME/.ssh/` 目錄中建立金鑰對。
4. 變更領導節點上私有金鑰的許可。

   ```
   $ chmod 600 ~/.ssh/id_rsa
   chmod 600 ~/.ssh/config
   ```

5. 使用您偏好的文字編輯器開啟 `~/.ssh/id_rsa.pub` 並複製該金鑰。
6. 對於叢集中的每個成員節點，執行以下操作：

   a. 連線到執行個體。

   b. 使用您偏好的文字編輯器開啟 `~/.ssh/authorized_keys`，然後新增您先前複製的公有金鑰。
7. 若要測試無密碼SSH是否如預期般運作，請連接至您的領導節點並執行下列命令。

   ```
   $ ssh member_node_private_ip
   ```

   您應該連線至成員節點，而不會提示您輸入金鑰或密碼。

建立彈性結構介面卡並將其連接至 Amazon EC2 執行個體

您可以創建一個EFA並將其附加到 Amazon EC2 實例，就像 Amazon 中的任何其他 elastic network interface 一樣EC2。但是，與彈性網路介面不同，EFAs無法在`running`狀態下連接至執行個體或從執行個體中分離。

考量事項

- 您可以變更與EFA. 若要啟用作業系統略過功能，EFA必須是安全性群組的成員，該群組允許進出安全性群組本身的所有輸入和輸出流量。如需詳細資訊，請參閱步驟 1：準備EFA已啟用的安全性群組。

  您可以使用變更與 elastic network interface 相關聯EFA的安全性群組相同的方式來變更與安全性群組相關聯的安全性群組。如需詳細資訊，請參閱變更安全群組。
- 您可以變更與EFA. 如果您有彈性 IP 地址，則可以將其與EFA. 如果您EFA的佈建在具有關聯IPv6CIDR區塊的子網路中，您可以將一或多個IPv6位址指派給EFA.

  您可以將彈性 IP (IPv4) 和EFA位IPv6址指派給 elastic network interface 的方式與指派 IP 位址相同。如需詳細資訊，請參閱管理 IP 地址。

任務

- 創建一個 EFA
- 附加EFA至已停止的執行個體
- 啟動執行個體EFA時附加
- 新增EFA至啟動範本

創建一個 EFA

您可以EFA在子網路中建立VPC. 您無法在建立子網路之後將其移EFA至其他子網路，而且只能將其附加到相同可用區域中已停止的執行個體。

若要使用主控台EFA建立新的

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Network Interfaces (網路介面)。
3. 選擇 Create Network Interface (建立網路介面)。

4. 在「描述」中，輸入的描述性名稱EFA。
5. 對於子網路，選取要在其中建立的子網路EFA。
6. 對於私人 IP，請輸入主要私人位IPv4址。如果您未指定IPv4位址，我們會從選取的子網路中選取可用的私人IPv4位址。
7. (IPv6僅限) 如果您選取具有關聯IPv6CIDR區塊的子網路，您可以選擇性地在 IPv6IP 欄位中指定位IPv6址。
8. 針對 Security groups (安全群組)，選取一個或多個安全群組。
9. 對於 EFA，選擇 Enable (啟用)。
10. 選擇 Yes, Create (是，建立)。

若要EFA使用建立新的 AWS CLI

使用指create-network-interface令`interface-type`，並指定`efa`，如下列範例所示。

```
aws ec2 create-network-interface --subnet-id subnet-01234567890 --
description example_efa --interface-type efa
```

附加EFA至已停止的執行個體

您可以將一個附加EFA到任何處於該`stopped`狀態的受支援執行個體。您無法附加EFA至處於該`running`狀態的執行個體。如需支援的執行個體類型的詳細資訊，請參閱支援的執行個體類型。

連接EFA至執行個體的方式與將網路介面連接至執行個體的方式相同。如需詳細資訊，請參閱連接網路介面。

啟動執行個體EFA時附加

啟動實例EFA時附加現有的步驟（AWS CLI）

使用執行個體指令 NetworkInterfaceId，並指定的識別碼EFA，如下列範例所示。

```
aws ec2 run-instances --image-id ami_id --count 1 --instance-
type c5n.18xlarge --key-name my_key_pair --network-interfaces
 DeviceIndex=0,NetworkInterfaceId=efa_id,Groups=sg_id,SubnetId=subnet_id
```

若要EFA在啟動執行個體時附加新的執行個體 (AWS CLI)

使用 run-instance 指令，並指efa定 InterfaceType，如下列範例所示。

```
aws ec2 run-instances --image-id ami_id --count 1 --instance-
type c5n.18xlarge --key-name my_key_pair --network-interfaces
 DeviceIndex=0,InterfaceType=efa,Groups=sg_id,SubnetId=subnet_id
```

新增EFA至啟動範本

您可以建立啟動範本，其中包含啟動EFA已啟動執行個體所需的組態資訊。若要建立EFA已啟用的啟動範本，請建立新的啟動範本，並指定支援的執行個體類型、EFA已AMI啟用的執行個體和EFA已啟用的安全性群組。如需詳細資訊，請參閱開始使用EFA和處理 Amazon 上MPI的HPC工作負載 EC2。

您可以利用啟動範本，透過其他 AWS 服務 (例如AWS Batch或) 啟動EFA已啟用的執行個體AWS ParallelCluster。

如需建立啟動範本的詳細資訊，請參閱創建一個 Amazon EC2 啟動模板。

從 Amazon EC2 執行個體EFA中分離並刪除

您可以EFA從 Amazon EC2 執行個體中分離，並以與 Amazon 中任何其他 elastic network interface 相同的方式刪除它EC2。

分離 EFA

若要EFA從執行個體中分離，您必須先停止執行個體。您無法EFA從處於執行中狀態的執行個體分離。

您可以使用與執行個體中斷連結 elastic network interface 的方式相同，從執行個體中分離執行個體。EFA如需詳細資訊，請參閱卸離網路介面。

刪除一個 EFA

若要刪除EFA，您必須先將其從例證中分離。貼附至例證EFA時，您無法刪除該物件。

刪EFAs除彈性網路介面的方式與刪除彈性網路介面的方式相同。如需詳細資訊，請參閱刪除網路介面。

在 Amazon 上監控彈性織物適配器 EC2

您可以使用下列功能來監控 Elastic Fabric Adapter 的效能。

Amazon VPC 流程日誌

您可以建立 Amazon VPC 流程日誌，以擷取有關進出EFA. 流程日誌資料可以發佈到 Amazon CloudWatch 日誌和 Amazon S3。建立流量日誌之後，您可以在選擇的目標中擷取及檢視其資料。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的VPC流程日誌。

您可以使用與為 elastic network interface 建立EFA流程記錄檔相同的方式來建立流程記錄。有關詳情，請參閱 Amazon VPC 使用者指南中的建立流程日誌。

在流程記錄項目中，EFA流量由`srcAddress`和識別`destAddress`，這兩者都格式化為MAC位址，如下列範例所示。

```
version accountId  eniId        srcAddress        destAddress       sourcePort destPort
 protocol packets bytes start      end        action log-status
2       3794735123 eni-10000001 01:23:45:67:89:ab 05:23:45:67:89:ab -          -
 -        9       5689  1521232534 1524512343 ACCEPT OK
```

Amazon CloudWatch

Amazon CloudWatch 提供的指標可讓您即時監控您EFAs的資訊。您可以收集和追蹤指標、建立自訂儀板表，以及設定警示，在特定指標達到您指定的閾值時通知您或採取動作。如需詳細資訊，請參閱使用監控執行個體 CloudWatch。

使用總和檢查碼驗證EFA安裝程式

您可以選擇性地使用或總和檢查碼來驗證EFA壓縮排檔 (.tar.gz 檔案)。MD5 SHA256我們建議您執行這項操作來確認軟體發行者的身分識別，並檢查應用程式自發行以來並未遭到變更或損毀。

驗證 tarball 的步驟

請使用 md5sum 公用程式作為總和檢查碼，或使用 sha256 MD5 sum 公用程式做為總和SHA256檢查碼，並指定壓縮標籤檔案名稱。您必須從儲存 tarball 檔案的目錄執行命令。

- MD5

```
$  md5sum tarball_filename.tar.gz
```

- SHA256

```
$  sha256sum tarball_filename.tar.gz
```

這些命令應該會傳回下列格式的檢查總和值。

```
checksum_value tarball_filename.tar.gz
```

將命令傳回的檢查總和值與下表所提供的檢查總和值進行比較。如果檢查總和相符，則可以安全地執行安裝指令碼。如果檢查總和不相符，請勿執行安裝指令碼，然後聯絡 AWS Support。

例如，下列指令會使用總和檢查碼來驗證 EFA 1.9.4 壓縮標籤。SHA256

```
$  sha256sum aws-efa-installer-1.9.4.tar.gz
```

輸出範例：

```
1009b5182693490d908ef0ed2c1dd4f813cc310a5d2062ce9619c4c12b5a7f14 aws-efa-
installer-1.9.4.tar.gz
```

下表列出最新版本的總和檢查碼。EFA

版本	下載 URL	檢查總和
EFA1.34.0	https://efa-installer.amazonaws.com/aws-efa-installer-1.34.0.塔爾.	MD5: `5cd4b28d27a31677c16139b54c9acb45` SHA256: `bd68839e741b0afd3ec2e37d50603803cfa7a279c120f0a736cc57c2ff2d7fdc`
EFA1.33.0	https://efa-installer.amazonaws.com/aws-efa-installer-1.33.0.tar.gz	MD5: `e2f61fccbcaa11e2ccfddd3660522276` SHA256: `0372877b87c6a7337bb7791d255e1053b907d030489fb2c3732ba70069185fce`
EFA1.32.0	https://efa-installer.amazonaws.com/aws-efa-installer-1.32.0.1 公里.	MD5: `db8d65cc028d8d08b5a9f2d88881c1b1`

版本	下載 URL	檢查總和
		SHA256: 5f7233760be57f6fee6de8c09acbfbf59238de848e06048dc54d156ef578fc66
EFA1.31.0	https://efa-installer.amazonaws.com/aws-efa-installer-1.31.	MD5: 856352f12bef2ccbadcd75e35aa52aaf SHA256: 943325bd37902a4300ac9e5715163537d56ecb4e7b87b37827c3e547aa1897bf
EFA1.30.0	https://efa-installer.amazonaws.com/aws-efa-installer-1.30.0.1.	MD5: 31f48e1a47fe93ede8ebd273fb747358 SHA256: 876ab9403e07a0c3c91a1a34685a52eced890ae052df94857f6081c5f6c78a0a
EFA1.29.1	https://efa-installer.amazonaws.com/aws-efa-installer-1.29.1.1.	MD5: e1872ca815d752c1d7c2b5c175e52a16 SHA256: 178b263b8c25845b63dc93b25bcdff5870df5204ec509af26f43e8d283488744
EFA1.29.0	https://efa-installer.amazonaws.com/aws-efa-installer-1.29.0.塔爾.	MD5: 39d06a002154d94cd982ed348133f385 SHA256: 836655f87015547e733e7d9f7c760e4e24697f8bbc261bb5f3560abd4206bc36

版本	下載 URL	檢查總和
EFA1.28.0	https://efa-installer.amazonaws.com/aws-efa-installer-1.28.0.塔爾.	MD5: 9dc13b744666582260 5e66febe074035 SHA256: 2e625d2d6d3e073b51 78e8e861891273d896 b66d03cb1a32244fd5 6789f1c435
EFA1.27.0	https://efa-installer.amazonaws.com/aws-efa-installer-1.27.0.1。塔爾。	MD5: 98bfb515ea3e8d93f5 54020f3837fa15 SHA256: 1d49a97b0bf8d964d9 1652a79ac851f2550e 33a5bf9d0cf86ec935 7ff6579aa3
EFA1.26.1	https://efa-installer.amazonaws.com/aws-efa-installer-1.26.1.TAR.GZ	MD5: 884e74671fdef47255 01f7cd2d451d0c SHA256: c616994c924f54ebfa bfab32b7fe8ac56947 fae00a0ff453d975e2 98d174fc96
EFA1.26.0	https://efa-installer.amazonaws.com/aws-efa-installer-1.26.0.g.	MD5: f8839f12ff2e3b9ba0 9ae8a82b30e663 SHA256: bc1abc1f76e97d204d 3755d2a9ca307fc423 e51c63141f798c2f15 be3715aa11

版本	下載 URL	檢查總和
EFA1.25.1	https://efa-installer.amazonaws.com/aws-efa-installer-1.25. 塔爾.	MD5: 6d876b894547847a45bb8854d4431f18 SHA256: d2abc553d22b89a4ce92882052c1fa6de450d3a801fe005da718b7d4b9602b06
EFA1.25.0	https://efa-installer.amazonaws.com/aws-efa-installer-1.25.0.1.	MD5: 1993836ca749596051da04694ea0d00c SHA256: 98b7b26ce031a2d6a93de2297cc71b03af647194866369ca53b60d82d45ad342
EFA1.24.1	https://efa-installer.amazonaws.com/aws-efa-installer-1.24.1.塔爾.	MD5: 211b249f39d53086f3cb0c07665f4e6f SHA256: 120cfeec233af0955623ac7133b674143329f9561a9a8193e473060f596aec62
EFA1.24.0	https://efa-installer.amazonaws.com/aws-efa-installer-1.24.0.塔爾.	MD5: 7afe0187951e2dd2c9cc4b572e62f924 SHA256: 878623f819a0d9099d76ecd41cf4f569d4c3aac0c9bb7ba9536347c50b6bf88e

版本	下載 URL	檢查總和
EFA1.23.1	https://efa-installer.amazonaws.com/aws-efa-installer-1.23.1.tar.gz	MD5: 22491e114b6ee7160a8290145dca0c28 SHA256: 5ca848d8e0ff4d1571cd443c36f8d27c8cdf2a0c97e9068ebf000c303fc40797
EFA1.23.0	https://efa-installer.amazonaws.com/aws-efa-installer-1.23.0.tar.gz	MD5: 38a6d7c1861f5038dba4e441ca7683ca SHA256: 555d497a60f22e3857fdeb3dfc53aa86d05926023c68c916d15d2dc3df6525bd
EFA1.22.1	https://efa-installer.amazonaws.com/aws-efa-installer-1.22.1.TAR.G.	MD5: 600c0ad7cdbc06e8e846cb763f92901b SHA256: f90f3d5f59c031b9a964466b5401e86fd0429272408f6c207c3f9048254e9665
EFA1.22.0	https://efa-installer.amazonaws.com/aws-efa-installer-1.22.0.塔爾.	MD5: 8f100c93dc8ab519c2aeb5dab89e98f8 SHA256: f329e7d54a86a03ea51da6ea9a5b68fb354fbae4a57a02f9592e21fce431dc3a

版本	下載 URL	檢查總和
EFA1.21.0	https://efa-installer.amazonaws.com/aws-efa-installer-1.21.0.塔爾.	MD5: 959ccc3a4347461909ec02ed3ba7c372 SHA256: c64e6ca34ccfc3ebe8e82d08899ae8442b3ef552541cf5429c43d11a04333050
EFA1.20.0	https://efa-installer.amazonaws.com/aws-efa-installer-1.20. 塔爾.	MD5: 7ebfbb8e85f1b94709df4ab3db47913b SHA256: aeefd2681ffd5c4c631d1502867db5b831621d6eb85b61fe3ec80df983d1dcf0
EFA1.19.0	https://efa-installer.amazonaws.com/aws-efa-installer-1.19.0.塔爾.	MD5: 2fd45324953347ec5518da7e3fefa0ec SHA256: 99b77821b9e72c8dea015cc92c96193e8db307deee05b91a58094cc331f16709
EFA1.18.0	https://efa-installer.amazonaws.com/aws-efa-installer-1.18.0.g.	MD5: fc2571a72f5d3c7b7b576ce2de38d91e SHA256: acb18a0808aedb9a5e485f1469225b9ac97f21db9af78e4cd6939700debe1cb6

版本	下載 URL	檢查總和
EFA1.17.3	https://efa-installer.amazonaws.com/aws-efa-installer-1.17.3.	MD5: `0517df4a190356ab559235147174cafd` SHA256: `5130998b0d2883bbae189b21ab215ecbc1b01ae0231659a9b4a17b0a33ebc6ca`
EFA1.17.2	https://efa-installer.amazonaws.com/aws-efa-installer-1.17.2.	MD5: `a329dedab53c4832df218a24449f4c9a` SHA256: `bca1fdde8b32b00346e175e597ffab32a09a08ee9ab136875fb38283cc4cd099`
EFA1.17.1	https://efa-installer.amazonaws.com/aws-efa-installer-1.17.1.1 塔爾.	MD5: `733ae2cfc9d14b52017eaf0a2ab6b0ff` SHA256: `f29322640a88ae9279805993cb836276ea240623820848463ca686c8ce02136f`
EFA1.17.0	https://efa-installer.amazonaws.com/aws-efa-installer-1.17.0.塔爾.	MD5: `d430fc841563c11c3805c5f82a4746b1` SHA256: `75ab0cee4fb6bd38889dce313183f5d3a83bd233e0a6ef6205d8352821ea901d`

版本	下載 URL	檢查總和
EFA1.16.0	https://efa-installer.amazonaws.com/aws-efa-installer-1.16.0.	MD5: 399548d3b0d2e812d74dd67937b696b4 SHA256: cecec36495a1bc6fdc82f97761a541e4fb6c9a3cbf3cfcb145acf25ea5dbd45b
EFA1.15.2	https://efa-installer.amazonaws.com/aws-efa-installer-1.15.2.	MD5: 955fea580d5170b05823d51acde7ca21 SHA256: 84df4fbc1b3741b6c073176287789a601a589313accc8e6653434e8d4c20bd49
EFA1.15.1	https://efa-installer.amazonaws.com/aws-efa-installer-1.15.1.tar.gz	MD5: c4610267039f72bbe4e35d7bf53519bc SHA256: be871781a1b9a15fca342a9d169219260069942a8bda7a8ad06d4baeb5e2efd7
EFA1.15.0	https://efa-installer.amazonaws.com/aws-efa-installer-1.15.0.1 公里。	MD5: 9861694e1cc00d884fadac07d22898be SHA256: b329862dd5729d2d098d0507fb486bf859d7c70ce18b61c302982234a3a5c88f

版本	下載 URL	檢查總和
EFA1.14.1	https://efa-installer.amazonaws.com/aws-efa-installer-1.14.1.1. 塔爾.	MD5: `50ba56397d359e57872fde1f74d4168a` SHA256: `c7b1b48e86fe4b3eaa4299d3600930919c4fe6d88cc6e2c7e4a408a3f16452c7`
EFA1.14.0	https://efa-installer.amazonaws.com/aws-efa-installer-1.14.0.g.	MD5: `40805e7fd842c36ececb9fd7f921b1ae` SHA256: `662d62c12de85116df33780d40e0533ef7dad92709f4f613907475a7a1b60a97`
EFA1.13.0	https://efa-installer.amazonaws.com/aws-efa-installer-1.13.0.g.	MD5: `c91d16556f4fd53becadbb345828221e` SHA256: `ad6705eb23a3fce44af3afc0f7643091595653a723ad0374084f4f2b715192e1`
EFA1.12.3	https://efa-installer.amazonaws.com/aws-efa-installer-1.12.3.	MD5: `818aee81f097918cfaebd724eddea678` SHA256: `2c225321824788b8ca3fbc118207b944cdb096b847e1e0d1d853ef2f0d727172`

版本	下載 URL	檢查總和
EFA1.12.2	https://efa-installer.amazonaws.com/aws-efa-installer-1.12.2.	MD5: `956bb1fc5ae0d6f0f87d2e481d49fccf` SHA256: `083a868a2c212a5a4fcf3e4d732b685ce39cceb3ca7e5d50d0b74e7788d06259`
EFA1.12.1	https://efa-installer.amazonaws.com/aws-efa-installer-1.12.1.	MD5: `f5bfe52779df435188b0a2874d0633ea` SHA256: `5665795c2b4f09d5f3f767506d4d4c429695b36d4a17e5758b27f033aee58900`
EFA1.12.0	https://efa-installer.amazonaws.com/aws-efa-installer-1.12.0.1 公里。	MD5: `d6c6b49fafb39b770297e1cc44fe68a6` SHA256: `28256c57e9ecc0b0778b41c1f777a9982b4e8eae782343dfe1246079933dca59`
EFA 1.11.2	https://efa-installer.amazonaws.com/aws-efa-installer-1.11.2.	MD5: `2376cf18d1353a4551e35c33d269c404` SHA256: `a25786f98a3628f7f54f7f74ee2b39bc6734ea9374720507d37d3e8bf8ee1371`

版本	下載 URL	檢查總和
EFA 1.11.1	https://efa-installer.amazonaws.com/aws-efa-installer-1.11.1.1. 塔爾.	MD5: `026b0d9a0a48780cc7406bd51997b1c0` SHA256: `6cb04baf5ffc58ddf319e956b5461289199c8dd805fe216f8f9ab8d102f6d02a`
EFA 1.11.0	https://efa-installer.amazonaws.com/aws-efa-installer-1.11.0.1 公里。	MD5: `7d9058e010ad65bf2e14259214a36949` SHA256: `7891f6d45ae33e822189511c4ea1d14c9d54d000f6696f97be54e915ce2c9dfa`
EFA 1.10.1	https://efa-installer.amazonaws.com/aws-efa-installer-1.10.1.	MD5: `78521d3d668be22976f46c6fecc7b730` SHA256: `61564582de7320b21de319f532c3a677d26cc46785378eb3b95c636506b9bcb4`
EFA 1.10.0	https://efa-installer.amazonaws.com/aws-efa-installer-1.10.0.g.	MD5: `46f73f5a7afe41b4bb918c81888fefa9` SHA256: `136612f96f2a085a7d98296da0afb6fa807b38142e2fc0c548fa986c41186282`

版本	下載 URL	檢查總和
EFA 1.9.5	https://efa-installer.amazonaws.com/aws-efa-installer-1.9.5.tar.gz	MD5: 95edb8a209c18ba8d250409846eb6ef4 SHA256: a4343308d7ea4dc943ccc21bcebed913e8868e59bfb2ac93599c61a7c87d7d25
EFA 1.9.4	https://efa-installer.amazonaws.com/aws-efa-installer-1.9.4.tar.gz	MD5: f26dd5c350422c1a985e35947fa5aa28 SHA256: 1009b5182693490d908ef0ed2c1dd4f813cc310a5d2062ce9619c4c12b5a7f14
EFA 1.9.3	https://efa-installer.amazonaws.com/aws-efa-installer-1.9.3.tar.gz	MD5: 95755765a097802d3e6d5018d1a5d3d6 SHA256: 46ce732d6f3fcc9edf6a6e9f9df0ad136054328e24675567f7029edab90c68f1
EFA 1.8.4	https://efa-installer.amazonaws.com/aws-efa-installer-1.8.4.tar.gz	MD5: 85d594c41e831afc6c9305263140457e SHA256: 0d974655a09b213d7859e658965e56dc4f23a0eee2dc44bb41b6d039cc5bab45

Amazon EC2 實例拓撲

說明您的執行個體拓撲可提供執行個體之間相對鄰近度的階層式檢視。您可以使用這些資訊來管理大規模的高效能運算 (HPC) 和機器學習 (ML) 運算基礎架構，同時最佳化工作安排。HPC和 ML 工作對延遲

和輸送量很敏感。您可以使用執行個體拓撲來偵測執行個體的位置，然後使用這項資訊在實際上彼此較近的執行個體上執行這些工作，以最佳化HPC和 ML 工作。

您可以使用執行個體拓撲來偵測現有執行個體的位置，但無法使用它來選擇實際啟動靠近現有執行個體的新執行個體。 若要影響例證放置，您可以使用在叢集置放群組中建立容量保留區。

定價

說明您的執行個體拓撲不需額外費用。

目錄

- Amazon EC2 執行個體拓撲如何運作
- Amazon EC2 執行個體拓撲的先決條
- Amazon EC2 執行個體拓撲範例

考量事項

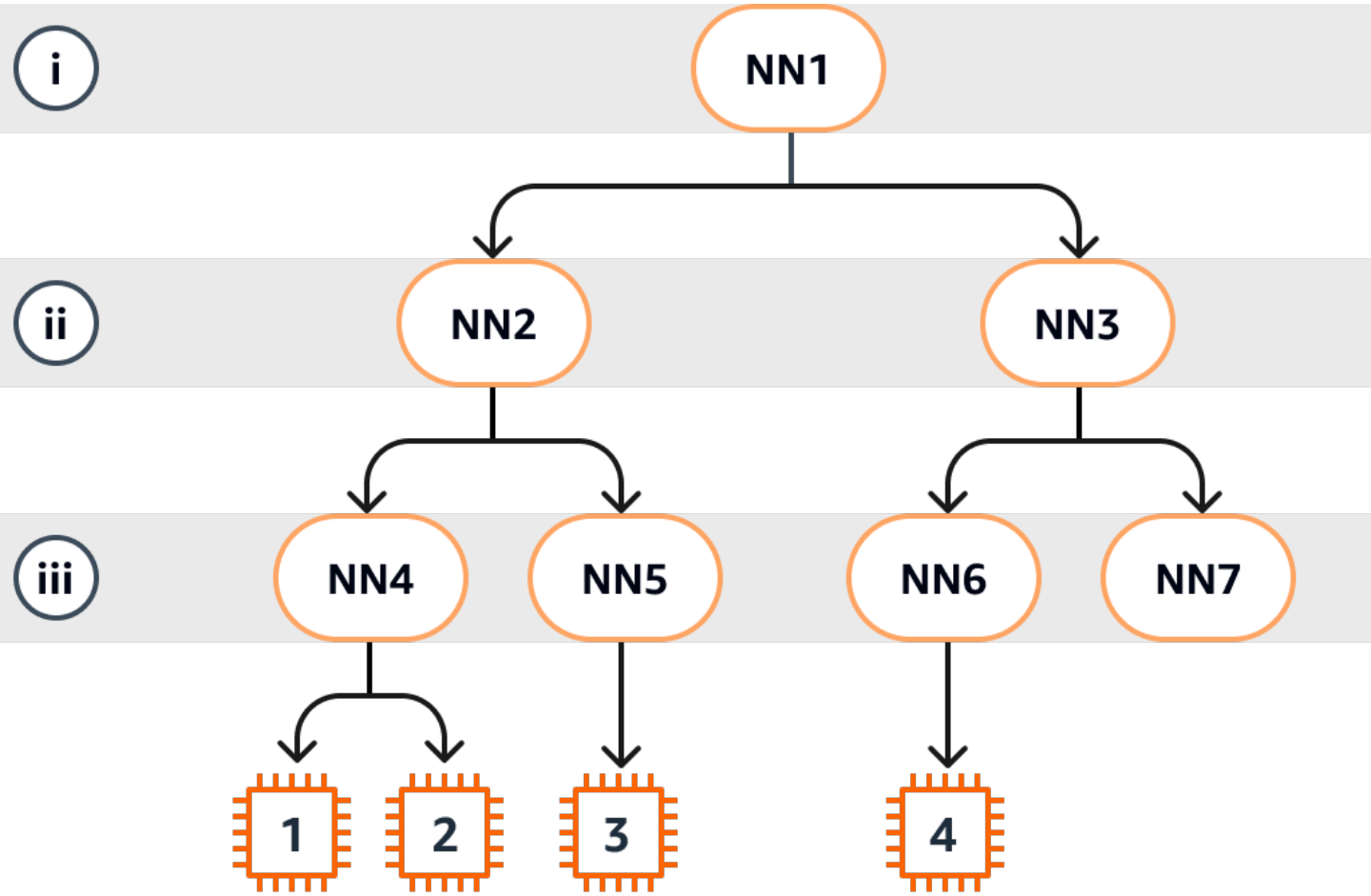
- 執行個體必須處於`running`狀態。
- 每個執行個體拓撲檢視對每個帳戶都是唯一的。
- AWS Management Console 不支援檢視例證拓樸。

Amazon EC2 執行個體拓撲如何運作

每個EC2執行個體都會連線到一個節點集。節點集由三個網路節點組成，每個節點代表 AWS 網路中的不同圖層。網路層是以 3 層或更多層的階層進行排列。節點集合提供此階層的由上而下檢視，其底層與執行個體最接近。

有關節點集的資訊稱為執行個體拓撲。

下圖提供可用來瞭解執行個體拓撲的視覺化表示法。網路節點會識別為 NN1— NN7。數字 i、ii 和 iii 可識別網路層。數字 1、2、3 和 4 可識別EC2例證。例證連接到底層中的節點，由 iii 標識。可有多個執行個體連接到同一個節點。



在此範例中：

- 例證 1 連接到第 iii 層中的網路節點 4 (NN4)。NN4連接到第 ii 層中的網絡節點 2 (NN2)，並NN2連接到第 i 層中的網絡節點 1 (NN1)，這是本例中網絡層次結構的頂部。網路節點集包括NN1、和NN2NN4，以階層方式從上方圖層到底部圖層表示。
- 執行個體 2 也會連線至網路節點 4 (NN4)。執行個體 1 和執行個體 2 共用相同的網路節點集：NN1NN2、和NN4。
- 執行個體 3 連線至網路節點 5 (NN5)。NN5連接NN2並NN2連接到NN1。針對執行個體 3 設定的網路節點為NN1NN2、和NN5。
- 執行個體 4 連線至網路節點 6 (NN6)。其網路節點集為NN1NN3、和NN6。

考慮例證 1、2 和 3 的鄰近性時，例證 1 和 2 彼此更接近，因為它們連接到相同的網路節點 (NN4)，而例證 3 則因為它連接到不同的網路節點 (NN5) 而更遠。

當考量此圖中所有執行個體的接近程度時，執行個體 1、2 和 3 彼此比執行個體更接近執行個體 4，因為它們NN2在網路節點集中共用。

一般而言，如果連接到任何兩個執行個體的網路節點相同，這兩個執行個體實際位置彼此相近，就像執行個體 1 和 2 的情況一樣。此外，網路節點之間的躍點數越少，執行個體彼此之間的距離就越接近。例如，執行個體 1 和 3 與執行個體 4 相同的網路節點 (NN2) 跳至一般網路節點 (NN1) 的躍點少於它們與執行個體 4 共同的網路節點 ()，因此彼此較接近執行個體 4。

在此範例中，沒有在網路節點 7 (NN7) 下執行的執行個體，因此API輸出不會包含在內NN7。

如何解釋輸出

您可以使用取得執行個體拓撲資訊DescribeInstanceTopologyAPI。輸出會提供執行個體之基礎網路拓撲的階層式檢視。

下列輸出範例對應上圖中四個執行個體的網路拓撲資訊。為了說明本範例，輸出範例中會包含註解。

輸出中的以下資訊非常重要：

- `NetworkNodes` 說明執行個體的網路節點集。
- 在每個網路節點集中，網路節點會以由上至下的階層順序列出。
- 連接至執行個體的網路節點是清單中的最後一個網路節點 (底層)。
- 若要找出哪些執行個體彼此靠近，請先在底層中尋找一般網路節點。如果底層沒有共同網路節點，請往上層尋找共同網路節點。

在下列範例中，輸出`i-1111111111example`和`i-2222222222example`位於與此範例中的其他執行個體相較之間最接近的位置，因為它們在底層`nn-4444444444example`中具有共同的網路節點。

```
{
    "Instances": [
        {
            "InstanceId": "i-1111111111example",  //Corresponds to instance 1
            "InstanceType": "p4d.24xlarge",
            "GroupName": "ML-group",
            "NetworkNodes": [
                "nn-1111111111example",           //Corresponds to NN1 in layer i
                "nn-2222222222example",           //Corresponds to NN2 in layer ii
                "nn-4444444444example"            //Corresponds to NN4 in layer iii -
 bottom layer, connected to the instance
            ],
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        },
```

```
        {
            "InstanceId": "i-2222222222example",  //Corresponds to instance 2
            "InstanceType": "p4d.24xlarge",
            "NetworkNodes": [
                "nn-1111111111example",           //Corresponds to NN1 - layer i
                "nn-2222222222example",           //Corresponds to NN2 - layer ii
                "nn-4444444444example"            //Corresponds to NN4 - layer iii -
 connected to instance
            ],
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        },
        {
            "InstanceId": "i-3333333333example",  //Corresponds to instance 3
            "InstanceType": "trn1.32xlarge",
            "NetworkNodes": [
                "nn-1111111111example",           //Corresponds to NN1 - layer i
                "nn-2222222222example",           //Corresponds to NN2 - layer ii
                "nn-5555555555example"            //Corresponds to NN5 - layer iii -
 connected to instance
            ],
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        },
        {
            "InstanceId": "i-444444444example",  //Corresponds to instance 4
            "InstanceType": "trn1.2xlarge",
            "NetworkNodes": [
                "nn-1111111111example",          //Corresponds to NN1 - layer i
                "nn-3333333333example",          //Corresponds to NN3 - layer ii
                "nn-6666666666example"           //Corresponds to NN6 - layer iii -
 connected to instance
            ],
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        }
    ],
    "NextToken": "SomeEncryptedToken"
}
```

Amazon EC2 執行個體拓撲的先決條

在說明執行個體的執行個體拓撲之前，請確定您的執行個體符合下列需求。

說明執行個體拓撲的需求

- AWS 區域
- 執行個體類型
- 執行個體狀態
- IAM許可

AWS 區域

支援 AWS 區域：

- 美國東部 (維吉尼亞北部)、美國東部 (俄亥俄)、美國西部 (加利佛尼亞北部)、美國西部 (奧勒岡)
- 亞太區域 (首爾)、亞太區域 (東京)
- 加拿大 (中部)
- 歐洲 (法蘭克福)、歐洲 (愛爾蘭)、歐洲 (斯德哥爾摩)

執行個體類型

支援的執行個體類型：

- `hpc6a.48xlarge` | `hpc6id.32xlarge` | `hpc7a.12xlarge` | `hpc7a.24xlarge` | `hpc7a.48xlarge` | `hpc7a.96xlarge` | `hpc7g.4xlarge` | `hpc7g.8xlarge` | `hpc7g.16xlarge`
- `p3dn.24xlarge` | `p4d.24xlarge` | `p4de.24xlarge` | `p5.48xlarge`
- `trn1.2xlarge` | `trn1.32xlarge` | `trn1n.32xlarge`

查看特定區域的可用執行個體類型

可用的執行個體類型因區域而異。若要查看「區域」中是否有執行個體類型可用，請將指describe-instance-types-offerings令與`--region`參數搭配使用。包含 `--filters` 參數可將結果範圍限定為您想使用的執行個體系列或執行個體類型，包含 `--query` 參數則可將輸出範圍限定為 `InstanceType` 的值。

```
aws ec2 describe-instance-type-offerings \
    --region us-east-2 \
    --filters 'Name=instance-type, Values=trn1*' \
    --query 'InstanceTypeOfferings[].InstanceType'
```

預期的輸出結果

```
[
    "trn1.2xlarge",
    "trn1.32xlarge",
    "trn1n.32xlarge"
]
```

執行個體狀態

執行個體必須處於 `running` 狀態。若執行個體處於其他狀態，您無法取得其執行個體拓撲資訊。

IAM許可

您的IAM身分 (使用者、使用者群組或角色) 需要下列IAM權限：

- `ec2:DescribeInstanceTopology`

Amazon EC2 執行個體拓撲範例

您可以使用指describe-instance-topologyCLI令描述EC2執行個體的執行個體拓撲。

若不搭配參數或篩選條件使用 `describe-instance-topology` 命令，回應會包含指定區域中符合此命令所支援執行個體類型的所有執行個體。您可以藉由包含 `--region` 參數或設定預設區域來指定區域。如需設定預設區域的詳細資訊，請參閱：為您的 Amazon EC2 資源選擇一個區域。

您可以加入參數以傳回符合指定例證IDs或放置群組名稱的例證。您也可以加入篩選條件，以傳回符合指定執行個體類型或執行個體系列的執行個體，或是指定可用區域或本地區域中的執行個體。您可以包含單一參數或篩選條件，或是參數和篩選條件的組合。

系統會將輸出會分頁，預設每頁最多 20 個執行個體。您可以使用 `--max-results` 參數指定每頁最多 100 個執行個體。

如需詳細資訊，請參閱 AWS CLI 命令參考中的 describe-instance-topology。

所需的許可

描述執行個體拓撲需要下列權限：

- `ec2:DescribeInstanceTopology`

範例

- 範例 1：無參數或篩選條件
- 範例 2：instance-type 篩選條件
  - 範例 2a：指定執行個體類型的完全相符篩選條件
  - 範例 2b：執行個體系列的萬用字元篩選條件
  - 範例 2c：結合執行個體系列和完全相符篩選條件
- 範例 3：zone-id 篩選條件
  - 範例 3a：可用區域篩選條件
  - 範例 3b：本地區域篩選條件
  - 範例 3c：結合可用區域與本地區域篩選條件
- 範例 4：結合 instance-type 和 zone-id 篩選條件
- 範例 5：放置群組名稱參數
- 範例 6 — 執行個體 IDs

範例 1：無參數或篩選條件

說明所有執行個體的執行個體拓撲

使用指describe-instance-topologyCLI令而不指定任何參數或篩選器。

```
aws ec2 describe-instance-topology --region us-west-2
```

回應只會傳回符合此支援執行個體類型的執行個體API。執行個體可以位於不同可用區域、本地區域 (ZoneId) 和放置群組 (GroupName)。如果執行個體不在放置群組中，GroupName 欄位就不會出現在輸出中。在下列輸出範例中，只有一個執行個體位於放置群組中。

範例輸出

```
{
    "Instances": [
        {
            "InstanceId": "i-1111111111example",
            "InstanceType": "p4d.24xlarge",
            "GroupName": "my-ml-cpg",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
```

```
                "nn-3333333333example"
            ],
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        },
        {
            "InstanceId": "i-2222222222example",
            "InstanceType": "p4d.24xlarge",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3333333333example"
            ],
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        },
        {
            "InstanceId": "i-3333333333example",
            "InstanceType": "trn1.32xlarge",
            "NetworkNodes": [
                "nn-1212121212example",
                "nn-1211122211example",
                "nn-1311133311example"
            ],
            "ZoneId": "usw2-az4",
            "AvailabilityZone": "us-west-2d"
        },
        {
            "InstanceId": "i-444444444example",
            "InstanceType": "trn1.2xlarge",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-5434334334example",
                "nn-1235301234example"
            ],
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        }
    ],
    "NextToken": "SomeEncryptedToken"
}
```

範例 2：instance-type 篩選條件

您可以依指定的執行個體類型進行篩選 (完全相符)，或依執行個體系列進行篩選 (使用萬用字元)。您也可以結合指定的執行個體類型篩選器和執行個體族群篩選。

範例 2a：指定執行個體類型的完全相符篩選條件

說明符合指定執行個體類型的所有執行個體之執行個體拓撲

將describe-instance-topologyCLI指令與`instance-type`篩選器搭配使用。在此範例中，系統會針對 `trn1n.32xlarge` 執行個體篩選輸出。回應只會傳回符合指定執行個體類型的執行個體。

```
aws ec2 describe-instance-topology \
    --region us-west-2 \
    --filters Name=instance-type,Values=trn1n.32xlarge
```

範例輸出

```
{
    "Instances": [
        {
            "InstanceId": "i-2222222222example",
            "InstanceType": "trn1n.32xlarge",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3333333333example"
            ],
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        }
    ],
    "NextToken": "SomeEncryptedToken"
}
```

範例 2b：執行個體系列的萬用字元篩選條件

說明符合特定執行個體系列的所有執行個體之執行個體拓撲

將describe-instance-topologyCLI指令與`instance-type`篩選器搭配使用。在此範例中，系統會針對 `trn1*` 執行個體篩選輸出。回應只會傳回符合指定執行個體系列的執行個體。

```
aws ec2 describe-instance-topology \
    --region us-west-2 \
    --filters Name=instance-type,Values=trn1*
```

範例輸出

```
{
    "Instances": [
        {
            "InstanceId": "i-2222222222example",
            "InstanceType": "trn1n.32xlarge",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3333333333example"
            ],
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        },
        {
            "InstanceId": "i-3333333333example",
            "InstanceType": "trn1.32xlarge",
            "NetworkNodes": [
                "nn-1212121212example",
                "nn-1211122211example",
                "nn-1311133311example"
            ],
            "ZoneId": "usw2-az4",
            "AvailabilityZone": "us-west-2d"
        },
        {
            "InstanceId": "i-444444444example",
            "InstanceType": "trn1.2xlarge",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-5434334334example",
                "nn-1235301234example"
            ],
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        }
    ],
    "NextToken": "SomeEncryptedToken"
```

```
}
```

範例 2c：結合執行個體系列和完全相符篩選條件

說明符合特定執行個體系列或指定執行個體類型的所有執行個體之執行個體拓撲

將describe-instance-topologyCLI指令與`instance-type`篩選器搭配使用。在此範例中，系統會針對 `pd4d*` 或 `trn1n.32xlarge` 執行個體篩選輸出。回應只會傳回符合任一個篩選條件的執行個體。

```
aws ec2 describe-instance-topology \
    --region us-west-2 \
    --filters "Name=instance-type,Values=p4d*,trn1n.32xlarge"
```

範例輸出

```
{
    "Instances": [
        {
            "InstanceId": "i-1111111111example",
            "InstanceType": "p4d.24xlarge",
            "GroupName": "ML-group",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3333333333example"
            ],
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        },
        {
            "InstanceId": "i-2222222222example",
            "InstanceType": "trn1n.32xlarge",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-4343434343example"
            ],
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        }
    ],
    "NextToken": "SomeEncryptedToken"
```

```
}
```

範例 3：zone-id 篩選條件

您可以使用 `zone-id` 篩選條件來依可用區域或本地區域進行篩選。您也可以結合可用區域篩選條件和本地區域篩選條件。

範例 3a：可用區域篩選條件

說明符合指定可用區域的所有執行個體之執行個體拓撲

將describe-instance-topologyCLI指令與`zone-id`篩選器搭配使用。在此範例中，輸出會使用可用區域 ID 進行篩選`use1-az1`。回應只會傳回符合指定可用區域的執行個體。

```
aws ec2 describe-instance-topology \
    --region us-east-1 \
    --filters Name=zone-id,Values=use1-az1
```

範例輸出

```
{
    "Instances": [
        {
            "InstanceId": "i-2222222222example",
            "InstanceType": "trn1n.32xlarge",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3214313214example"
            ],
            "ZoneId": "use1-az1",
            "AvailabilityZone": "us-east-1a"
        }
    ],
    "NextToken": "SomeEncryptedToken"
}
```

範例 3b：本地區域篩選條件

說明符合指定本地區域的所有執行個體之執行個體拓撲

將describe-instance-topologyCLI指令與`zone-id`篩選器搭配使用。在此範例中，會使用本機區域 ID 篩選輸出`use1-atl2-az1`。回應只會傳回符合指定本地區域的執行個體。

```
aws ec2 describe-instance-topology \
    --region us-east-1 \
    --filters Name=zone-id,Values=use1-atl2-az1
```

範例輸出

```
{
    "Instances": [
        {
            "InstanceId": "i-1111111111example",
            "InstanceType": "p4d.24xlarge",
            "GroupName": "ML-group",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3333333333example"
            ],
            "ZoneId": "use1-atl2-az1",
            "AvailabilityZone": "us-east-1-atl-2a"
        }
    ],
    "NextToken": "SomeEncryptedToken"
}
```

範例 3c：結合可用區域與本地區域篩選條件

說明符合指定可用區域或本地區域的所有執行個體之執行個體拓撲

將describe-instance-topologyCLI指令與`zone-id`篩選器搭配使用。在此範例中，輸出會使用可用區域 ID `use1-az1` 和本機區域 ID 進行篩選`use1-atl2-az1`。回應只會傳回符合任一個篩選條件的執行個體。

```
aws ec2 describe-instance-topology \
    --region us-east-1 \
    --filters Name=zone-id,Values=use1-az1,use1-atl2-az1
```

範例輸出

```
{
    "Instances": [
```

```
        {
            "InstanceId": "i-1111111111example",
            "InstanceType": "p4d.24xlarge",
            "GroupName": "ML-group",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3333333333example"
            ],
            "ZoneId": "use1-atl2-az1",
            "AvailabilityZone": "us-east-1-atl-2a"
        },
        {
            "InstanceId": "i-2222222222example",
            "InstanceType": "trn1n.32xlarge",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3214313214example"
            ],
            "ZoneId": "use1-az1",
            "AvailabilityZone": "us-east-1a"
        }
    ],
    "NextToken": "SomeEncryptedToken"
}
```

範例 4：結合 instance-type 和 zone-id 篩選條件

您可將所有篩選條件結合為單一命令。

說明符合指定執行個體類型、執行個體系列、可用區域或本地區域的所有執行個體之執行個體拓撲

將describe-instance-topologyCLI指令與`instance-type`和`zone-id`篩選器搭配使用。在此範例中，系統會針對執行個`p4d*`體系列、`trn1n.32xlarge`執行個體類型、`use1-az1`可用區域 ID 和`use1-atl2-az1`本機區域 ID 篩選輸出。回應會傳回符合 `us-east-1a` 或 `us-east-1-atl-2a` 區域中 `p4d*` 或 `trn1n.32xlarge` 執行個體的執行個體。

```
aws ec2 describe-instance-topology \
    --region us-east-1 \
    --filters "Name=instance-type,Values=p4d*,trn1n.32xlarge" "Name=zone-
id,Values=use1-az1,use1-atl2-az1"
```

範例輸出

```
{
    "Instances": [
        {
            "InstanceId": "i-1111111111example",
            "InstanceType": "p4d.24xlarge",
            "GroupName": "ML-group",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3333333333example"
            ],
            "ZoneId": "use1-atl2-az1",
            "AvailabilityZone": "us-east-1-atl-2a"
        },
        {
            "InstanceId": "i-2222222222example",
            "InstanceType": "trn1n.32xlarge",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3214313214example"
            ],
            "ZoneId": "use1-az1",
            "AvailabilityZone": "us-east-1a"
        }
    ],
    "NextToken": "SomeEncryptedToken"
}
```

範例 5：放置群組名稱參數

描述指定放置群組中所有執行個體的執行個體拓撲

將describe-instance-topologyCLI指令與group-names參數搭配使用。在下列範例中，執行個體可以位於ML-group或放HPC-group置群組中。回應會傳回位於其中一個放置群組中的執行個體。

```
aws ec2 describe-instance-topology \
    --region us-west-2 \
    --group-names ML-group HPC-group
```

範例輸出

```
{
    "Instances": [
        {
            "InstanceId": "i-1111111111example",
            "InstanceType": "p4d.24xlarge",
            "GroupName": "ML-group",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3333333333example"
            ],
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        },
        {
            "InstanceId": "i-2222222222example",
            "InstanceType": "trn1n.32xlarge",
            "GroupName": "HPC-group",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3214313214example"
            ],
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        }
    ],
    "NextToken": "SomeEncryptedToken"
}
```

範例 6 — 執行個體 IDs

說明指定執行個體的執行個體拓撲

將describe-instance-topologyCLI指令與`--instance-ids`參數搭配使用。響應將返回匹配指定實例的實例IDs。

```
aws ec2 describe-instance-topology \
    --region us-west-2 \
    --instance-ids i-1111111111example i-2222222222example
```

範例輸出

```
{
    "Instances": [
        {
            "InstanceId": "i-1111111111example",
            "InstanceType": "p4d.24xlarge",
            "GroupName": "ML-group",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3333333333example"
            ],
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        },
        {
            "InstanceId": "i-2222222222example",
            "InstanceType": "trn1n.32xlarge",
            "GroupName": "HPC-group",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3214313214example"
            ],
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        }
    ],
    "NextToken": "SomeEncryptedToken"
}
```

Amazon EC2 執行個體的放置群組

為了滿足工作負載的需求，您可以將一組相依的EC2執行個體啟動到放置群組中，以影響其位置。

視工作負載類型而定，您可以採取下列其中一個置放策略來建立置放群組：

- 叢集：會將執行個體緊密地包裝在一個可用區域內。此策略可讓工作負載達到高效能運算 () 應用程式典型之緊密結合 node-to-node 通訊所需的低延遲網路效能。HPC
- 分區：會將執行個體分散到邏輯分區，使一個分區中的執行個體群組不會與不同分區中的執行個體群組共用底層硬體。大量分散和複寫的工作負載 (例如 Hadoop、Cassandra 和 Kafka) 通常採取此策略。

- 分散：嚴格地將一小組執行個體分散到不同的底層硬體，以減少相互關聯的故障。

置放群組是選用的程序。如果您沒有將執行個體啟動到置放群組中，請EC2嘗試將執行個體放置在所有執行個體都散佈在基礎硬體上，以將相關的故障降到最低。

定價

建立置放群組無需收費。

規則與限制

在使用置放群組之前，請記住下列規則：

- 一個例證一次可以放置在一個放置群組中；您無法將例證放置在多個放置群組中。
- 置放群組無法合併。
- 隨需容量保留和區域預留執行個體可讓您為可用區域中的EC2執行個體預留容量。啟動執行個體時，如果執行個體屬性與隨需容量保留或區域預留執行個體指定的屬性相符，則該執行個體會自動使用預留容量。如果您將例證啟動到放置群組中，也是如此。
- 您無法在置放群組中啟動專用執行個體。
- 您無法啟動在置放群組中設定為中斷時停止或休眠的 Spot 執行個體。

目錄

- 放置群組的放置策略
- 為您的 Amazon EC2 執行個體建立放置群組
- 變更 Amazon EC2 執行個體的放置位置
- 刪除置放群組
- 共用置放群組
- 放置群組 AWS Outposts

放置群組的放置策略

您可以使用下列其中一種置放策略來建立置放群組。

置放策略

- 集群放置群組
- 分區置放群組

- 分散放置群組

集群放置群組

集群放置群組是單一可用區域中的執行個體邏輯分組。執行個體不會隔離到單一機架。叢集置放群組可跨越相同區域中的對等虛擬私人網路 (VPCs)。相同叢集置放群組中的執行個體可享有更高的 TCP /IP 流量每個流量輸送量限制，並位於網路的相同高二分頻寬區段中。

下列映像會顯示放入集群放置群組中的執行個體。

能夠受惠於低網路延遲、高網路傳輸量或這兩者的應用，建議採用集群放置群組。當大部分的網路流量都在群組中的執行個體之間時，也建議使用這些群組。若要為置放群組提供最低延遲和最高的 packet-per-second 網路效能，請選擇支援增強型聯網的執行個體類型。如需詳細資訊，請參閱增強型聯網。

我們建議您以下列方式啟動執行個體：

- 使用單一啟動要求，在置放群組中啟動您所需的執行個體數目。
- 對置放群組中的所有執行個體使用相同的執行個體類型。

如果之後您試著在置放群組中加入更多執行個體，或是在置放群組中啟動超過一種執行個體類型，將會提高出現容量不足錯誤的機率。

如果先停止置放群組中的執行個體，然後再啟動，此執行個體仍會在置放群組中執行。不過，如果執行個體可用的容量不足，起始作業將會失敗。

若置放群組中已經擁有正在運作的執行個體，當您在此置放群組中啟動執行個體時，如果收到容量錯誤，請停止和啟動該置放群組中的所有執行個體，然後試著再次啟動。啟動執行個體的動作，可能會讓這些執行個體轉移到容量足夠的硬體，能夠提供容量給所有請求的執行個體使用。

規則與限制

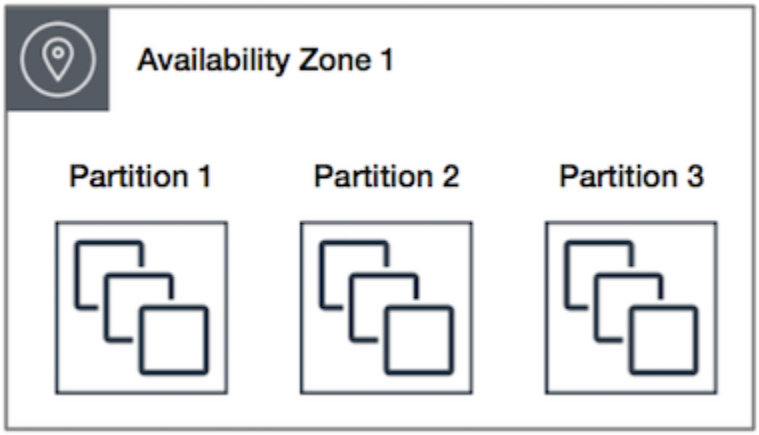
集群放置群組適用的規則如下：

- 支援下列執行個體類型：
  - 目前一代執行個體，高載效能執行個體 (例如 T2) 、Mac1 執行個體和 M7I-Flex 執行個體除外。
  - 下列上一代執行個體：A1、C3、C4、I2、M4、R3 和 R4。
- 集群放置群組不能涵跨多個可用區域。
- 在集群放置群組中，兩個執行個體之間傳送資料的最高網路傳輸速度，會受限於這兩個執行個體之中較慢者的速度。針對需要高傳輸量的應用程式，請選擇符合您需求之網路連線能力的執行個體類型。
- 下列規則適用於為增強聯網所啟用的執行個體：
  - 集群放置群組內的執行個體，單一傳輸流的速度最多可達 10 Gbps。不在集群放置群組內的執行個體，單一傳輸流的速度最高可達 5 Gbps。
  - 透過公有 IP 地址空間或VPC端點在相同區域內往返 Amazon S3 儲存貯體的流量可以使用所有可用的執行個體彙總頻寬。
- 您可以在集群放置群組中啟動多種執行個體類型。不過，這會降低啟動成功所需容量可用的機率。我們建議在集群放置群組中，針對所有執行個體使用相同的執行個體類型。
- 建議您在叢集置放群組中建立隨需容量保留，藉此在叢集置放群組中明確保留容量。請注意，您無法使用區域預留執行個體保留容量，因為它們無法在置放群組中明確保留容量。
- 叢集置放群組的網際網路和透過內部部署資源 AWS Direct Connect 連線的網路流量限制為 5 Gbps。

分區置放群組

分區置放群組有助於降低應用程式發生相關硬體故障的可能性。使用分割區放置群組時，Amazon 會將每個EC2群組分成稱為分割區的邏輯區段。Amazon EC2 確保置放群組中的每個分割區都有自己的機架組。每個機架有其自己的網路和電源。一個置放群組內不會有兩個分割區共用相同的機架，這樣可讓您隔離硬體故障在應用程式內造成的影響。

下圖顯示在單一可用區域中分區置放群組的簡單視覺化呈現。其中顯示放入分區置放群組中的執行個體，而該群組具有三個分區 - 分區 1、分區 2 和分區 3。每個分區包含多個執行個體。一個分割區中的執行個體不會與其他分割區中的執行個體共用機架，可讓您將單一硬體故障的影響僅限縮在相關聯的分割區中。



分割區放置群組可用於跨不同機架部署大型分散式和複寫的工作負載 HDFSHBase，例如、和 Cassandra。當您將執行個體啟動到分割區放置群組時，Amazon EC2 會嘗試將執行個體平均分配到您指定的分割區數目。您也可以將執行個體啟動到特定的分區，讓您能夠更有效控制放置執行個體的位置。

分區置放群組的分割區可以在同一區域中的多個可用區域。分區置放群組在每個可用區域最多可有 7 個分區。可啟動到分區置放群組中的執行個體數，僅會受到帳戶限制的限制。

此外，分區置放群組可提供對分區的可見性 - 您可以看到執行個體位於哪個分區中。您可以與拓撲感知應用程式共用此資訊，例如HDFSHBase、和 Cassandra。這些應用程式會使用此資訊來進行智能資料複寫決策，以提高資料可用性與持久性。

如果開始或啟動分區置放群組中的執行個體，但是沒有足夠的唯一硬體來滿足請求，則請求會失敗。Amazon EC2 會隨著時間推移提供更多獨特的硬體，因此您可以稍後再次嘗試請求。

規則與限制

分區置放群組適用的規則如下：

- 分區置放群組在每個可用區域最多支援 7 個分區。您可以在分區置放群組中啟動的執行個體數，僅會受到帳戶限制的限制。
- 將執行個體啟動到分割區放置群組時，Amazon EC2 會嘗試將執行個體平均分配到所有分割區。Amazon EC2 不保證在所有分區之間均勻分佈執行個體。
- 具有 專用執行個體 的分區置放群組最多可有 2 個分區。
- 容量保留不能在分區置放群組中預留容量。

分散放置群組

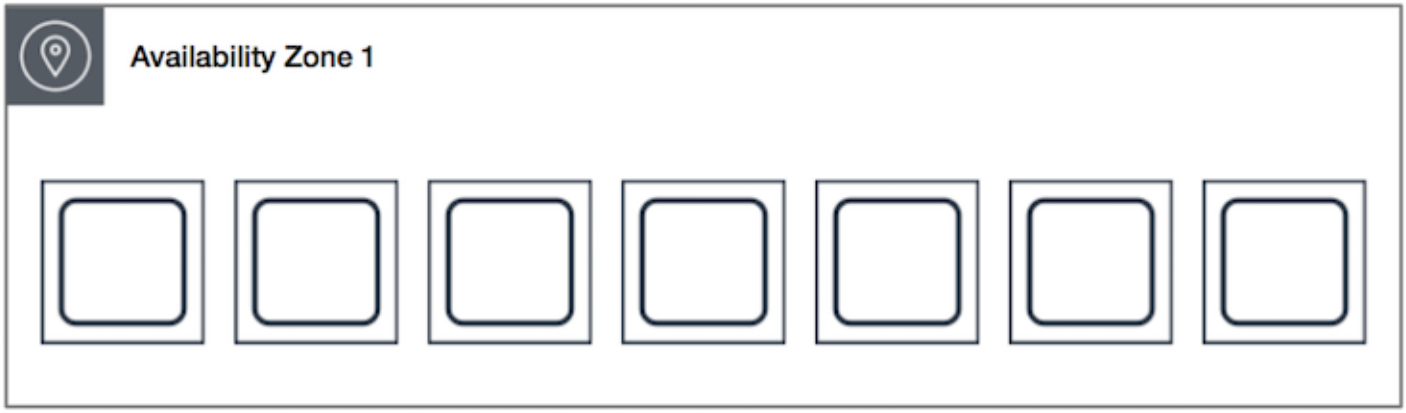
分散置放群組是一組放在不同硬體上的執行個體。

若應用程式具有應該分開保存的少量重要執行個體，建議使用分散置放群組。透過分散層級置放群組來啟動執行個體，可降低同時發生故障的風險，這種情況可能會在執行個體共用相同設備時發生。分散層級置放群組提供對不同硬體的存取，因此適合混合的執行個體類型或不同時間啟動執行個體。

如果開始或啟動分散置放群組中的執行個體，但是沒有足夠的唯一硬體來滿足請求，則請求會失敗。Amazon EC2 會隨著時間推移提供更多獨特的硬體，因此您可以稍後再次嘗試請求。放置群組可以跨機架或主機分散放置執行個體。機架層級分攤放置群組可用於 AWS 區域和開啟 AWS Outposts。主機層級分攤放置群組 AWS Outposts 只能搭配使用。

機架層級分攤放置群組

下列映像會顯示在單一可用區域中放入分散置放群組中的 7 個執行個體。七個執行個體放在七個不同機架上，而各機架有其自己的網路和電源。



機架層級分攤放置群組可跨越相同區域中的多個可用區域。在一個區域中，每個群組的每個可用區域，機架層級分攤置放群組最多可以有七個執行中執行個體。使用 Outposts，機架級分攤放置群組可以容納與 Outpost 部署中機架相同數量的執行個體。

主機層級分散置放群組

主機層級分攤放置群組僅適用於 AWS Outposts。主機分攤層級置放群組可以容納與 Outpost 部署中主機一樣多的執行個體。如需詳細資訊，請參閱the section called “放置群組 AWS Outposts”。

規則與限制

分散置放群組適用的規則如下：

- 機架分散置放群組支援每個可用區域最多可有七個執行中的執行個體。例如，在具有三個可用區域的區域中，您在該群組共可執行 21 個執行個體 (每個可用區域七個)。如果您嘗試在相同可用區域和在相同的分散置放群組中啟動第 8 個執行個體，則該執行個體不會啟動。如果您在一個可用區域中需要七個以上的執行個體，建議使用多重分散置放群組。使用多重分散置放群組並不能保證群組之間的執行個體分散，但可有助於確保每個群組的分散，從而限制某些類型失敗的影響。
- 專用執行個體 不支援分散置放群組。
- 主機層級分攤放置群組僅支援在上的放置群組 AWS Outposts。主機層級分攤置放群組可以容納與 Outpost 部署中主機一樣多的執行個體。
- 在一個區域中，每個群組的每個可用區域，機架層級分攤置放群組最多可以有七個執行中執行個體。使用 AWS Outposts機架層級分攤放置群組可容納與 Outpost 部署中機架相同數量的執行個體。
- 容量保留不能在分散置放群組中預留容量。

為您的 Amazon EC2 執行個體建立放置群組

您可以使用放置群組來控制例證相對於彼此的位置。建立放置群組後，您可以在放置群組中啟動例證。

限制

每個區域最多可以建立 500 個放置群組。

Console

建立放置群組

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Placement Groups (置放群組)。
3. 選擇建立放置群組。
4. 指定群組的名稱。
5. 選擇群組的放置策略：「叢集」、「分攤」或「分割區」。

   如果選擇「分攤」，則必須選擇分攤層級：「機架」或「主機」。

   如果選擇「分割區」，則必須輸入群組的分割區數目。
6. (選擇性) 若要新增標記，請選擇 [新增標記]，然後輸入金鑰和值。
7. 選擇 Create group (建立群組)。

AWS CLI

使用指create-placement-group令。

建立叢集置放群組

下列範例會建立使用cluster放置策略的放置群組，purpose並套用鍵值為的標籤production。

```
aws ec2 create-placement-group \
    --group-name my-cluster \
    --strategy cluster \
    --tag-specifications 'ResourceType=placement-
group,Tags={Key=purpose,Value=production}'
```

建立分割區放置群組

下列範例會建立使用放置策略的partition放置群組，並使用--partition-count參數指定五個分割區。

```
aws ec2 create-placement-group \
    --group-name HDFS-Group-A \
    --strategy partition \
    --partition-count 5
```

PowerShell

建立放置群組

下列New-EC2PlacementGroup指令會建立叢集置放群組。

```
New-EC2PlacementGroup -GroupName my-placement-group -Strategy cluster
```

變更 Amazon EC2 執行個體的放置位置

您可以使用下列任一方式變更執行個體的置放群組：

- 將例證新增至放置群組
- 在置放群組之間移動執行個體
- 從置放群組中移除執行個體

例證必須處於stopped狀態，才能變更例證的放置群組。

Console

變更例證放置

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體。
4. 選擇 Actions (動作)、Instance settings (執行個體設定)、Modify instance placement (修改執行個體的配置)。
5. 對於「放置」群組，請執行下列任一項作業：
   - 若要將例證新增至放置群組，請選擇放置群組。
   - 若要將例證從一個放置群組移至另一個放置群組，請選擇放置群組。
   - 若要從放置群組中移除例證，請選擇「無」(None)。
6. 選擇 Save (儲存)。

AWS CLI

將執行個體移到置放群組

下列指modify-instance-placement令會將指定的例證移至指定的放置群組。

```
aws ec2 modify-instance-placement \
    --instance-id i-0123a456700123456 \
    --group-name MySpreadGroup
```

從置放群組中移除執行個體

下列指modify-instance-placement令會指定放置群組名稱的空字串，這會將例證從其目前的放置群組中移除。

```
aws ec2 modify-instance-placement \
    --instance-id i-0123a456700123456 \
    --group-name ""
```

PowerShell

將執行個體移到置放群組

使用具有放置群組名稱的Edit-EC2InstancePlacement指令。

從置放群組中移除執行個體

使用具有空字串的Edit-EC2InstancePlacement指令做為放置群組名稱。

刪除置放群組

如果需要更換或不再需要某個置放群組，可以刪除該置放群組。您可以使用下列其中一種方法來刪除置放群組。

先決條件

置放群組不得包含任何執行個體，您才可予以刪除。您可以終止例證、將其移至其他放置群組，或將其從放置群組中移除。

Console

刪除置放群組

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Placement Groups (置放群組)。
3. 選取置放群組並依序選取 Actions (動作)、Delete (刪除)。
4. 出現確認提示時，請輸入 **Delete**，然後選擇 Delete (刪除)。

AWS CLI

刪除置放群組

下列指delete-placement-group令會刪除指定的放置群組。

```
aws ec2 delete-placement-group --group-name my-cluster
```

PowerShell

刪除置放群組

下列指Remove-EC2PlacementGroup令會刪除指定的放置群組。

```
Remove-EC2PlacementGroup -GroupName my-cluster
```

共用置放群組

位置群組共用可讓您影響獨立 AWS 帳戶擁有的相互從屬例證的位置。您可以在多個 AWS 帳戶 或組織內共用刊登位置群組。您可以在共用置放群組中啟動執行個體。

置放群組擁有者可與下列項目共用置放群組：

- 組織內部或外部的特定 AWS 帳戶
- 組織內的組織單位
- 整個 組織

> **Note**
>
> 您要共用放置群組的 AWS 帳戶必須具備下列IAM原則的權限。
>
> - `ec2:PutResourcePolicy`
> - `ec2:DeleteResourcePolicy`

目錄

- 規則與限制
- 跨可用區域共用
- 共用置放群組
- 識別共用置放群組
- 在共用置放群組中啟動執行個體
- 取消共用已共用置放群組

規則與限制

當您共用置放群組或與您共用置放群組時，下列規則和限制適用。

- 若要共用刊登位置群組，您必須在 AWS 帳戶中擁有該群組。您無法將已共用給您的置放群組再共用出去。
- 共用分割區或分散置放群組時，置放群組限制不會變更。共用分割區置放群組在每個可用區域最多可有七個分割區，而共用分散置放群組在每個可用區域最多可支援七個執行中的執行個體。

- 若要與您的組織或組織中的組織單位共用放置群組，您必須啟用與共用功能 AWS Organizations。如需詳細資訊，請參閱共用您的 AWS 資源。
- 您必須負責管理共用置放群組中您所擁有的執行個體。
- 您無法檢視或修改與共用置放群組相關聯但非您擁有的執行個體和容量保留。

跨可用區域共用

為確保資源分配至區域中的所有可用區域，可用區域會獨立對應至各個帳戶的名稱。這可能導致帳戶之間的可用區域命名出現差異。例如，您 AWS 帳戶的可`us-east-1a`用區域可能與其他 AWS 帳戶`us-east-1a`的位置不同。

若要基於您的帳戶來識別專用執行個體的相對位置，您必須使用可用區域 ID (AZ ID)。可用區域 ID 是可用區域在所有 AWS 帳戶之間唯一且一致的識別符。例如，`use1-az1` 是 `us-east-1` 區域的可用區域 ID，它在每一個 AWS 帳戶的位置都相同。

若要檢視您帳戶中可IDs用區域的可用區域

1. 請在 https://console.aws.amazon.com/ram 開啟 AWS RAM 主控台。
2. 目前區域的可IDs用區域會顯示在右側面板的「您的 AZ ID」下方。

共用置放群組

若要共用置放群組，您必須將它新增至資源共用。資源共用是一 AWS RAM 種可讓您跨 AWS 帳號共用資源的資源。資源共享指定要共用的資源，以及共用它們的消費者。

如果您屬於組織內 AWS Organizations 共用的組織已啟用，您組織中的用戶就會獲得共用置放群組的存取權。

如果放置群組與組織外部的 AWS 帳戶共用， AWS 帳戶擁有者將會收到加入資源共用的邀請。他們可以在接受邀請後存取共用置放群組。

您可以使用 https://console.aws.amazon.com/ram 或 AWS CLI跨 AWS 帳戶共用放置群組。

AWS RAM console

若要使用 https://console.aws.amazon.com/ram 共用您擁有的置放群組，請參閱建立資源共用。

AWS CLI

若要共用您擁有的放置群組，請使用create-resource-share指令。

識別共用置放群組

放置群組的 Amazon 資源名稱 (ARN) 包含擁有放置群組之帳戶的 12 位數帳戶 ID。您可以使用帳戶 ID 來識別與您共用的置放群組的擁有者。

您可以ARN使用下列其中一種方法尋找放置群組。

Amazon EC2 console

若要識別共用放置群組

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在功能窗格的 [網路與安全性] 底下，選擇 [置放群組]。
3. 「放置群組」(Placement group) 表格會列出您所擁有並與您共用的所有放置群組。「群組」(Group) ARN 欄會顯示放置群組ARN。

   如果看不到 [群組] ARN 欄，請選擇右上角的 [設定] (⚙)，開啟 [群組]ARN，然後選擇 [確認]。

AWS CLI

若要識別共用放置群組

使用describe-placement-groups指令列出您所擁有並與您共用的所有放置群組。在回應中，`GroupId`參數會顯示放置群組ARN的。

在共用置放群組中啟動執行個體

> **Important**
> 使用啟動共 AWS CLI 用放置群組中的例證時，您必須使用`GroupId`參數指定放置群組 ID。

只有當您是要共用之置放群組的擁有者時，才能使用放置群組名稱。我們建議您使用放置群組 ID，以避免 AWS 帳戶之間潛在的放置群組名稱衝突。

您可以在 Amazon EC2 主控台的「放置群組」畫面或使用describe-placement-groups AWS CLI 指令找到放置群組的 ID。

Console

將例證啟動到共用放置群組

1. 請按照以下程序啟動執行個體，但在完成下列步驟指定置放群組的設定之前，請勿啟動執行個體。
2. 在 Instance type (執行個體類型) 中，選取支援的執行個體類型。
3. 展開 [進階詳細資料]，然後依下列方式配置置群組設定：
   a. 在「放置」群組中，選取與您共用的放置群組。

      > **Note**
      > 如果有相同名稱的置放群組，請檢查放置群組 ID，確定您選取了正確的放置群組。

   b. 如果您選擇具有分割區策略的置放群組，請針對「目標分割區」選擇要在其中啟動執行個體的分割區。
4. 在「摘要」面板中，執行下列動作：
   a. 對於 Number of instances (執行個體的數目)，輸入您在這個置放群組中所需的執行個體總數，因為您之後可能會無法在此置放群組中加入執行個體。
   b. 檢閱您的執行個體組態，然後選擇 Launch 執行個體。

如需詳細資訊，請參閱使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體。

AWS CLI

若要在共用置放群組中啟動執行個體

使用「執行例證」指令並指定共用放置群組的放置群組 ID。

```
aws ec2 run-instances —-placement "GroupId = pg-0123456789example"
```

若要將執行個體啟動到共用分割區置放群組的特定分割區

使用 run-instance 指令，並指定共用放置群組的放置群組 ID 和磁碟分割編號。

```
aws ec2 run-instances —-placement "GroupId = pg-0123456789example, PartitionNumber
 = 3"
```

> **Tip**
> 使用VPC對等互連來連接不同 AWS 帳戶所擁有的執行個體，並取得共用叢集置放群組所提供的完整延遲優勢。如需詳細資訊，請參閱什麼是對VPC等互連？

取消共用已共用置放群組

置放群組擁有者可以隨時取消共用已共用置放群組。

當您取消共用已共用置放群組時，下列變更將會生效。

- 共用置放群組的 AWS 帳戶將無法再啟動執行個體或保留容量。
- 如果您的執行個體是在共用置放群組中執行，則它們會與置放群組中斷關聯，但仍會繼續在您的 AWS 帳戶中正常執行。
- 如果您在共用置放群組中有容量保留，這些保留項目將會與放置群組中斷關聯，但您仍可在 AWS 帳戶中繼續存取這些資料。

您可以使用下列其中一種方法來取消共用已共用置放群組。

AWS RAM console

若要使用 https://console.aws.amazon.com/ram 取消共用放置群組，請參閱刪除資源共用。

AWS CLI

若要使用取消共用放置群組的共用 AWS Command Line Interface，請使用指disassociate-resource-share令。

放置群組 AWS Outposts

AWS Outposts 是一項全受管服務，可將 AWS 基礎架構APIs、服務和工具延伸至客戶場所。透過提供 AWS 受管理基礎架構的本機存取權， AWS Outposts 讓客戶能夠使用與 Region 相同的程式設計介面在 AWS 內部部署建置和執行應用程式，同時使用本機運算和儲存資源來降低延遲和本機資料處理需求。

Outpost 是部署在客戶站點的 AWS 計算和儲存容量集區。 AWS 作為 AWS 區域的一部分來操作、監控和管理此容量。

您可以在帳戶中建立的 Outposts 中建立置放群組。這樣，您就可以在站點上的 Outpost 中跨基礎硬體分散執行個體。在 Outposts 中建立和使用置放群組的方式，與在一般可用區域中建立和使用置放群組的方式相同。在 Outpost 中建立具有分散策略的置放群組時，可以選擇讓置放群組在主機層級或是機架層級分散執行個體。在主機層級分散執行個體可讓您透過單一機架 Outpost 使用分散策略。

考量

- 機架層級分攤放置群組可容納與 Outpost 部署中機架相同數量的執行個體。
- 主機層級分攤置放群組可以容納與 Outpost 部署中主機一樣多的執行個體。

先決條件

您的站點必須安裝 Outpost。如需詳細資訊，請參閱 AWS Outposts 使用者指南中的建立 Outpost 並訂購 Outpost 容量。

在 Outpost 中使用置放群組

1. 在 Outpost 上建立子網。如需詳細資訊，請參閱 AWS Outposts 使用者指南中的建立子網。
2. 在 Outpost 的關聯區域中建立置放群組。如果您使用分攤策略建立置放群組，您可以選擇主機或機架層級分攤，以決定群組如何將執行個體分散到 Outpost 上的基礎硬體。如需詳細資訊，請參閱the section called “建立置放群組”。
3. 在置放群組中啟動執行個體。在 Subnet (子網路) 欄位中，選取您在步驟 1 建立的子網路，在 Placement group name (置放群組名稱) 欄位中，選取您在步驟 2 建立的置放群組。如需詳細資訊，請參閱 AWS Outposts 使用者指南中的在 Outpost 上啟動執行個體。

EC2執行個體的網路最大傳輸單元 (MTU)

網路連線的最大傳輸單位 (MTU) 是可透過連線傳遞的最大允許封包的大小 (以位元組為單位)。連接越MTU大，可以在單個數據包中傳遞的數據越多。乙太網幀內含封包 (或您實際傳送的資料) 和環繞的網路額外負荷資訊。

乙太網路訊框具有不同格式，最常見的格式是標準乙太網路 v2 訊框格式。它支持 1500MTU，這是大多數互聯網上支持的最大以太網數據包大小。執行個體支援MTU的最大值取決於其執行個體類型。

下列規則適用於 Wavelength 區域中的執行個體：

- VPC在相同 Wavelength 區域內從一個實例到另一個實例MTU的流量具有 1300。
- 從一個實例到另一個在 Wavelength 區域內使用運營商 IP MTU 的流量具有 1500。
- 在 Wavelength 區域與使用公用 IP 位址的區域之間，從一個執行個體傳輸到另一個執行個體MTU的流量為 1500。
- 在 Wavelength 區域和使用私有 IP 地址的區域之間從一個實例傳輸到另一個實例MTU的流量具有 1300。

下列規則適用於 Outposts 中的執行個體：

- 從 Outposts 中MTU的執行個體傳輸到區域中某個執行個體的流量為 1300。

目錄

- 巨型框MTU架
- 路徑MTU探索
- MTU為您的 Amazon EC2 實例設置
- 疑難排解

巨型框MTU架

巨型訊框可增加單一封包的承載大小，進而增加不屬於封包成本的封包比例，藉此允許超過 1500 位元組的資料。傳送相同數量的可用資料，所需的封包數即會減少。但是，在以下情況下，流量限制為最多 MTU 1500：

- 網際網路閘道的流量
- 通過區域間對VPC等連接連接的流量
- 透過VPN連線的流量
- 指定 AWS 區域以外的流量

封包若超過 1500 位元組將被切割為片段，若其 IP 標頭設有 `Don't Fragment` 旗標，則會將其捨棄。

Jumbo 框架應該小心使用受到網際網路限制的流量，或是任何離開. VPC 封包由中介系統切割，因此拖累流量速度。若要在繫結至外部的流量VPC而非緩慢流量內使用 Jumbo 框架VPC，您可以依路由來設定MTU大小，或使用具有不同大MTU小和不同路由的多個彈性網路介面。

以配置於集群放置群組內的執行個體而言，巨型訊框有助實現最大網路傳輸量，此情況建議使用巨型訊框。如需詳細資訊，請參閱Amazon EC2 執行個體的放置群組。

您可以使用 Jumbo 框架來處理您VPCs與內部部署網路之間的流量 AWS Direct Connect。如需詳細資訊，以及如何驗證超大型框架功能，請參閱《AWS Direct Connect 使用指南》MTU中的〈設定網路〉。

所有 Amazon EC2 執行個體類型都支援 1500 個MTU，目前所有的執行個體類型都支援巨型框架 下列前一代執行個體類型支援巨型框架：A1、C3、I2、M3 和 R3。

如需支援MTU大小的詳細資訊：

- 如需NAT閘道，請參閱 NAT Amazon VPC 使用者指南中的閘道基礎知識。
- 有關傳輸閘道的資訊，請參閱 Amazon VPC 交通閘道使用者指南MTU中的。
- 如需 Local Zones 相關資訊，請參閱《AWS Local Zones 使用者指南》中的考量。

路徑MTU探索

路徑MTU探索 (PMTUD) 可用來判斷兩個裝置MTU之間的路徑。路徑MTU是原始主機和接收主機之間路徑所支援的最大封包大小。當兩個主機之間的網路MTU大小有所差異時，PMTUD可讓接收主機以ICMP訊息回應來源主機。此ICMP訊息會指示原始主機沿著網路路徑使用最MTU小的大小，並重新傳送要求。如果不這樣協議，可能會因為請求太大，使得接收端主機無法接受，而發生封包捨棄。

對於IPv4，當主機傳送的封包大於接收主機MTU的封包或大於該MTU路徑上的裝置時，接收主機或裝置會捨棄封包，然後傳回下列ICMP訊息：`Destination Unreachable: Fragmentation Needed and Don't Fragment was Set`(Type 3，Code 4)。這會指示傳輸主機將承載分割成多個較小的封包，然後重新傳輸它們。

通IPv6訊協定不支援網路中的片段。當主機傳送的封包大於接收主機的MTU封包，或大於路徑上裝置MTU的封包時，接收主機或裝置會捨棄封包，然後傳回下列ICMP訊息：`ICMPv6 Packet Too Big (PTB)`(Type 2)。這會指示傳輸主機將承載分割成多個較小的封包，然後重新傳輸它們。

系統會自動追蹤透過某些元件 NAT (例如閘道和負載平衡器) 建立的連線。這意味著系統會自動為您的傳出連線嘗試啟用安全群組追蹤。如果自動追蹤連線，或者您的安全群組規則允許輸入ICMP流量，您可以收到PMTUD回應。

請注意ICMP，即使在安全性群組層級允許流量，也可以封鎖流量，例如您的網路存取控制清單項目拒絕子網路的ICMP流量。

> **Important**
>
> 路徑MTU探索不保證某些路由器不會捨棄巨型框架。您的VPC意願中的互聯網網關僅轉發最多 1500 字節的數據包。建議使用 1500 個MTU數據包用於互聯網流量。

MTU為您的 Amazon EC2 實例設置

網路連線的最大傳輸單位 (MTU) 是可透過連線傳遞的最大允許封包的大小 (以位元組為單位)。所有 Amazon EC2 執行個體都支援標準框架 (1500MTU)，目前所有的執行個體類型都支援巨型框架 (9001MTU)。

您可以檢視 Amazon EC2 執行個體的、檢視執行個體與其他主機MTU之間的路徑，以及將執行個體設定為使用標準或巨型框架。MTU

任務

- 檢查兩台主機MTU之間的路徑
- 檢查您MTU的實例
- MTU為您的執行個體設定

檢查兩台主機MTU之間的路徑

您可以檢查執行個EC2體和其他主機MTU之間的路徑。您可以指定DNS名稱或 IP 位址做為目的地。如果目的地是另一個EC2執行個體，請確認其安全性群組允許輸入UDP流量。

您使用的程序視執行個體的作業系統而定。

Linux 執行個體

在執行個體上執行tracepath命令，以檢查執行EC2個體和指定目的地MTU之間的路徑。這個命令是軟件`iputils`包的一部分，默認情況下在許多 Linux 發行版中都可用。

此範例會檢查執行個EC2體和MTU之間的路徑`amazon.com`。

```
[ec2-user ~]$ tracepath amazon.com
```

在此範例輸出中，路徑MTU為 1500。

```
 1?: [LOCALHOST]      pmtu 9001
```

```
 1:  ip-172-31-16-1.us-west-1.compute.internal (172.31.16.1)   0.187ms pmtu 1500
 1:  no reply
 2:  no reply
 3:  no reply
 4:  100.64.16.241 (100.64.16.241)                              0.574ms
 5:  72.21.222.221 (72.21.222.221)                             84.447ms asymm 21
 6:  205.251.229.97 (205.251.229.97)                           79.970ms asymm 19
 7:  72.21.222.194 (72.21.222.194)                             96.546ms asymm 16
 8:  72.21.222.239 (72.21.222.239)                             79.244ms asymm 15
 9:  205.251.225.73 (205.251.225.73)                           91.867ms asymm 16
...
31:  no reply
     Too many hops: pmtu 1500
     Resume: pmtu 1500
```

Windows 執行個體

使用曲線路線檢查路徑MTU的步驟

1. 從以下位置下載mturoute.exe至您的EC2執行個體。 http://www.elifulkerson.com/projects/mturoute.php
2. 開啟命令提示字元視窗，並變更至您下載 mturoute.exe 所在的目錄。
3. 使用以下命令檢查EC2實例和指定目的地MTU之間的路徑。此範例會檢查執行個EC2體和MTU之間的路徑`www.elifulkerson.com`。

   ```
   .\mturoute.exe www.elifulkerson.com
   ```

   在此範例輸出中，路徑MTU為 1500。

   ```
   * ICMP Fragmentation is not permitted. *
   * Speed optimization is enabled. *
   * Maximum payload is 10000 bytes. *
   + ICMP payload of 1472 bytes succeeded.
   - ICMP payload of 1473 bytes is too big.
   Path MTU: 1500 bytes.
   ```

檢查您MTU的實例

您可以檢查執行個體的MTU值。部分執行個體設定為使用巨型訊框，其他則使用標準訊框大小。

您使用的程序視執行個體的作業系統而定。

Linux 執行個體

若要檢查 Linux 執行處理上的MTU設定

在執行EC2個體上執行下列ip命令。如果主要網路介面不是`eth0`，請`eth0`以您的網路介面取代。

```
[ec2-user ~]$ ip link show eth0
```

在此示例輸出中，*`mtu 9001`* 表示執行個體使用超大型框架。

```
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc pfifo_fast state UP mode
 DEFAULT group default qlen 1000
    link/ether 02:90:c0:b7:9e:d1 brd ff:ff:ff:ff:ff:ff
```

Windows 執行個體

您使用的程序取決於執行個體上的驅動程式。

ENA driver

版本 2.1.0 及更高版本

若要取得MTU值，請在EC2執行個體上使用下列Get-NetAdapterAdvancedProperty指令。使用萬用字元 (星號) 取得所有乙太網路名稱。檢查介面名稱的輸出`*JumboPacket`。9015 值表示已啟用巨型訊框。巨型訊框預設為停用。

```
Get-NetAdapterAdvancedProperty -Name "Ethernet*"
```

版本 1.5 及更早版本

若要取得MTU值，請在EC2執行個體上使用下列Get-NetAdapterAdvancedProperty指令。檢查介面名稱的輸出`MTU`。9001 值表示已啟用巨型訊框。巨型訊框預設為停用。

```
Get-NetAdapterAdvancedProperty -Name "Ethernet"
```

Intel SRIOV 82599 driver

若要取得MTU值，請在EC2執行個體上使用下列Get-NetAdapterAdvancedProperty指令。檢查介面名稱 `*JumboPacket` 的項目。9014 值表示已啟用巨型訊框。（請注意，MTU大小包括標題和有效負載。）巨型訊框預設為停用。

```
Get-NetAdapterAdvancedProperty -Name "Ethernet"
```

AWS PV driver

若要取得MTU值，請在EC2執行個體上使用下列指令。介面名稱可能有所不同。請在輸出中尋找具備名稱 "Ethernet"、"Ethernet 2" 或 "Local Area Connection" 的項目。您將需要介面名稱來啟用或停用巨型訊框。9001 值表示已啟用巨型訊框。

```
netsh interface ipv4 show subinterface
```

MTU為您的執行個體設定

您可能想要在網際網路流量VPC和標準框架內使用巨型框架來處理網路流量。無論您的使用案例為何，我們都建議您驗證執行個體是否如預期般運作。

您使用的程序視執行個體的作業系統而定。

Linux 執行個體

若要在 Linux 執行個體上設定MTU值

1. 在執行個體上執行下列ip命令。它將所需的MTU值設置為 1500，但您可以使用 9001 來代替。

   ```
   [ec2-user ~]$ sudo ip link set dev eth0 mtu 1500
   ```

2. (選擇性) 若要在重新開機後保留您的網路MTU設定，請根據您的作業系統類型修改下列組態檔案。

   - 若為 Amazon Linux 2，請將下列一行新增至 `/etc/sysconfig/network-scripts/ifcfg-eth0` 檔案：

     ```
     MTU=1500
     ```

     將下行新增至 `/etc/dhcp/dhclient.conf` 檔案：

     ```
     request subnet-mask, broadcast-address, time-offset, routers, domain-name,
      domain-search, domain-name-servers, host-name, nis-domain, nis-servers, ntp-
     servers;
     ```

- 對於 Amazon LinuxAMI，請將以下幾行添加到您的`/etc/dhcp/dhclient-eth0.conf`文件中。

```
interface "eth0" {
supersede interface-mtu 1500;
}
```

- 若是其他 Linux 發行版本，請參閱其特定文件。

3. (選擇性) 重新啟動執行個體，並確認MTU設定正確無誤。

Windows 執行個體

您使用的程序取決於執行個體上的驅動程式。

ENA driver

您可以MTU使用「裝置管理員」或執行個體上的Set-NetAdapterAdvancedProperty指令來變更。

版本 2.1.0 及更高版本

使用下列命令來啟用巨型框架。

```
Set-NetAdapterAdvancedProperty -Name "Ethernet" -RegistryKeyword "*JumboPacket" -
RegistryValue 9015
```

使用下列命令來停用巨型框架。

```
Set-NetAdapterAdvancedProperty -Name "Ethernet" -RegistryKeyword "*JumboPacket" -
RegistryValue 1514
```

版本 1.5 及更早版本

使用下列命令來啟用巨型框架。

```
Set-NetAdapterAdvancedProperty -Name "Ethernet" -RegistryKeyword "MTU" -
RegistryValue 9001
```

使用下列命令來停用巨型框架。

```
Set-NetAdapterAdvancedProperty -Name "Ethernet" -RegistryKeyword "MTU" -
RegistryValue 1500
```

Intel SRIOV 82599 driver

您可以MTU使用「裝置管理員」或執行個體上的Set-NetAdapterAdvancedProperty指令來變更。

使用下列命令來啟用巨型框架。

```
Set-NetAdapterAdvancedProperty -Name "Ethernet" -RegistryKeyword "*JumboPacket" -
RegistryValue 9014
```

使用下列命令來停用巨型框架。

```
Set-NetAdapterAdvancedProperty -Name "Ethernet" -RegistryKeyword "*JumboPacket" -
RegistryValue 1514
```

AWS PV driver

您可以在執行個體上MTU使用netsh指令來變更。您無法更改MTU使用「設備管理器」。

使用下列命令來啟用巨型框架。

```
netsh interface ipv4 set subinterface "Ethernet" mtu=9001
```

使用下列命令來停用巨型框架。

```
netsh interface ipv4 set subinterface "Ethernet" mtu=1500
```

疑難排解

如果您在使用巨型框架時遇到EC2執行個體和 Amazon Redshift 叢集之間的連線問題，請參閱 Amazon Redshift 管理指南中的查詢似乎停止回應。

適用於EC2執行個體的虛擬私有雲

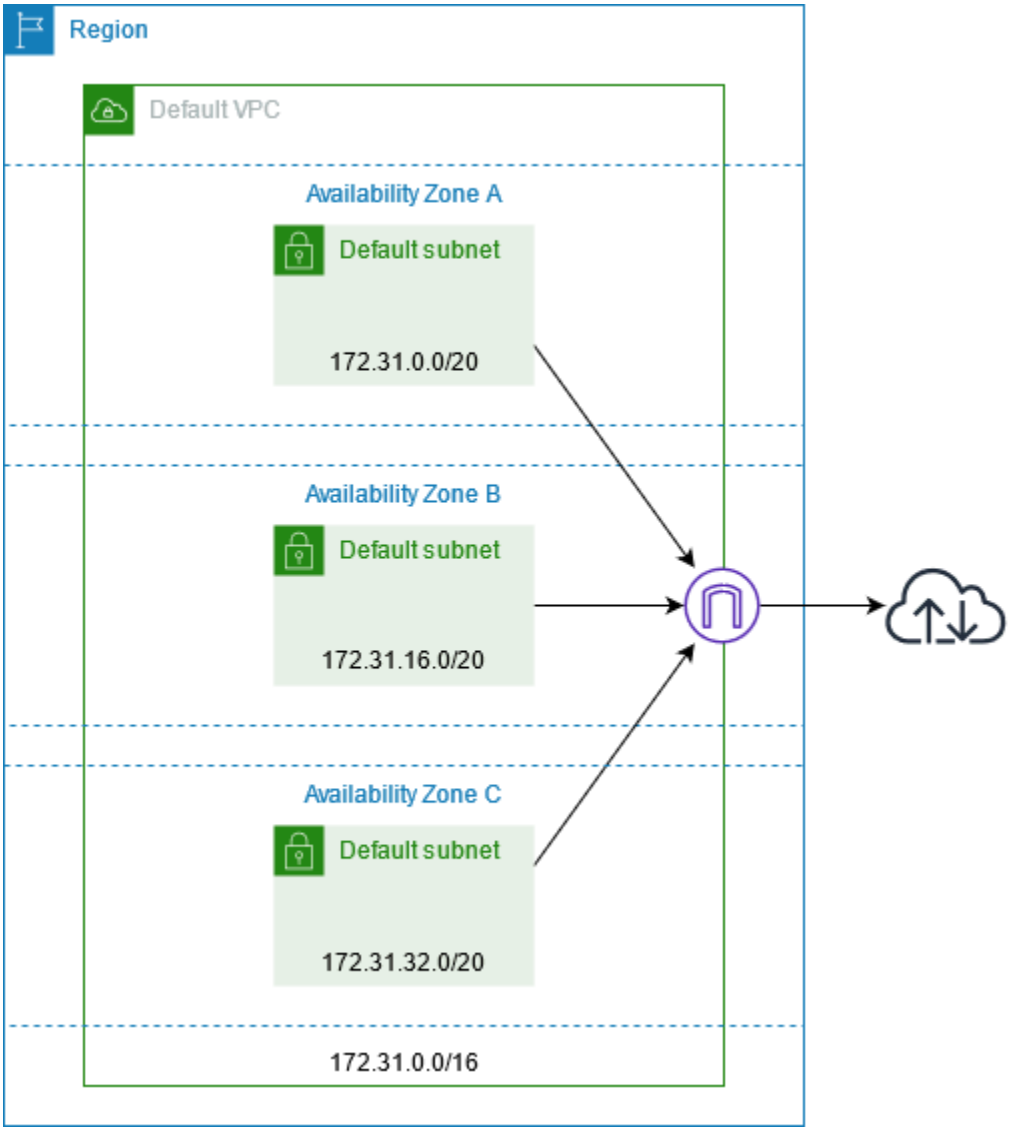
Amazon Virtual Private Cloud (AmazonVPC) 可讓您在雲端內的邏輯隔離區域中定義虛擬網路，稱為虛擬私有雲或VPC。 AWS 您可以在. 的子網路中建立 AWS 資源，EC2例如 Amazon 執行個體。VPC您的網路非常類VPC似於您可能在自己的資料中心中操作的傳統網路，並具有使用可擴充基礎架構的優點。 AWS您可以設定您的VPC；您可以選取其 IP 位址範圍、建立子網路，以及設定路由表、網路閘道和安全性設定。您可以將您中的執行個體連線VPC到網際網路或您自己的資料中心。

目錄

- 您的預設 VPCs
- 非預設 VPCs
- 網際網路存取
- 共用子網路
- IPv6只有子網

您的預設 VPCs

當您建立 AWS 帳戶時，我們會VPC在每個地區建立一個預設值。預設值VPC是VPC已設定且可供您使用的預設值。例如，每個預設值中每個可用區域都有一個預設子網路VPC，連接到網際網路閘道VPC，主路由表中有一個路由，可將所有流量 (0.0.0.0/0) 傳送至網際網路閘道。您可以視需要修改預設VPCs的組態。例如，您可以新增子網路和路由表。



非預設 VPCs

您可以建立自己的資源，而不是使用預設VPC資源VPC，如 Amazon 使用VPC者指南VPC中的建立 a 中所述。

以下是為執行個體建立EC2執行個體時需要考量VPC的一些事項。

- 您可以使用該IPv4CIDR塊的默認建議或輸入應用程序或網絡所需的CIDR塊。
- 若要確保高可用性，請在多個可用區域中建立子網路。
- 如果必須可從網際網路存取執行個體，請執行以下其中一項操作：
  - 如果您的執行個體可以位於公有子網路中，請新增公用子網路。保持啟用這兩個DNS選項。您可以選擇現在或稍後新增私有子網路。
  - 如果您的執行個體必須位於私有子網路中，請僅新增私有子網路。您可以新增NAT閘道，為私有子網路中的執行個體提供網際網路存取權。如果您的執行個體跨可用區域傳送或接收大量流量，請在每個可用區域中建立NAT閘道。否則，您可以僅在其中一個可用區域中建立NAT閘道，並啟動執行個體，以便在與NAT閘道相同的可用區域內傳送或接收跨區域流量。

網際網路存取

在預設值中啟動至預設子網路的執行個體可以存取網際網路，因VPCs為預設值會設定為指派公用 IP 位址和DNS主機名稱，而主路由表則設定為連接至網際網路閘道的VPC路由。VPC

對於在非預設子網路中啟動的執行個體VPCs，您可以使用下列其中一個選項來確保您在這些子網路中啟動的執行個體可存取網際網路：

- 設定網際網路閘道。如需詳細資訊，請參閱 Amazon 使用VPC者指南中的使用網際網路閘道 Connect 到網際網路。
- 設定公用NAT閘道。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的從私有子網路存取網際網路。

共用子網路

將EC2執行個體啟動到共用VPC子網路時，請注意下列事項：

- 參與者可以指定共用子網路的 ID，在共用子網路中執行執行個體。參與者必須擁有他們指定的任何安全群組或網路介面。

- 參與者可以啟動、停止、終止和描述他們在共用子網路中建立的執行個體。參與者無法啟動、停止、終止或描述VPC擁有者在共用子網路中建立的執行個體。
- VPC擁有者無法啟動、停止、終止或描述共用子網路中參與者建立的執行個體。
- 參與者可以使用執行個體 Connect 端點連線到共用子網路中的EC2執行個體。參與者必須在共用子網路中建立EC2執行個體 Connect 端點。參與者無法使用VPC擁有者在共用子網路中建立的EC2執行個體 Connect 端點。

如需詳細資訊，請參閱 Amazon VPC 使用者指南中的「VPC與其他帳戶共用您的帳戶」。

IPv6只有子網

在IPv6唯一子網路中啟動的EC2執行個體會接收IPv6位址，但不接收IPv4位址。您啟動到IPv6唯一子網路的任何執行個體都必須是 AWS Nitro System 上建置的執行個體。

Amazon 的安全性 EC2

雲安全 AWS 是最高的優先級。身為 AWS 客戶，您可以從資料中心和網路架構中獲益，這些架構是專為滿足對安全性最敏感的組織的需求而打造的。

安全是 AWS 與您之間共同承擔的責任。共同責任模型將其描述為雲端的安全性和雲端中的安全性：

- 雲端的安全性 — AWS 負責保護在 AWS 雲端中執行 AWS 服務的基礎架構。 AWS 還為您提供可以安全使用的服務。若要了解適用於 Amazon 的合規計劃EC2，請參閱AWS 合規計劃的合規計劃AWS服務範的服務。
- 雲端內部的安全 – 您的責任包含下列領域：
  - 控制執行個體的網路存取，例如透過設定VPC和安全群組。如需詳細資訊，請參閱控制網路流量。
  - 管理用於連線到執行個體的登入資料。
  - 管理部署至訪客作業系統的訪客作業系統和軟體，包括更新和安全性修補程式。如需詳細資訊，請參閱Amazon EC2 視窗執行個體更新管理。
  - 設定附加至執行個體的IAM角色，以及與這些角色相關聯的權限。如需詳細資訊，請參閱IAMAmazon 的角色 EC2。

本文件可協助您了解如何在使用 Amazon 時應用共同的責任模型EC2。它說明如何設定 Amazon EC2 以符合您的安全和合規目標。您也會學到如何使用其他可 AWS 協助您監控和保護 Amazon EC2 資源的服務。

目錄

- Amazon 的數據保護 EC2
- Amazon 基礎設施安全 EC2
- Amazon 的韌性 EC2
- Amazon 的合規驗證 EC2
- Amazon 的身份和訪問管理 EC2
- Amazon EC2 視窗執行個體更新管理
- Windows 執行個體的安全性最佳做法
- Amazon EC2 密鑰對和 Amazon EC2 實例
- 適用於EC2執行個體的 Amazon EC2 安全群組

- Amazon TPM EC2 實例的硝基
- Windows 執行個體的認證保護
- EC2使用界面VPC端點訪問 Amazon

Amazon 的數據保護 EC2

AWS 共同責任模型適用於 Amazon 彈性運算雲端中的資料保護。如此模型中所述， AWS 負責保護執行所有 AWS 雲端. 您負責維護在此基礎設施上託管內容的控制權。您也同時負責所使用 AWS 服務 的安全組態和管理任務。如需有關資料隱私權的詳細資訊，請參閱資料隱私權FAQ。 如需歐洲資料保護的相關資訊，請參閱AWS 安全性GDPR部落格上的AWS 共同責任模型和部落格文章。

基於資料保護目的，我們建議您使用 AWS IAM Identity Center 或 AWS Identity and Access Management (IAM) 保護 AWS 帳戶 認證並設定個別使用者。如此一來，每個使用者都只會獲得授與完成其任務所必須的許可。我們也建議您採用下列方式保護資料：

- 對每個帳戶使用多重要素驗證 (MFA)。
- 使用SSL/TLS與 AWS 資源溝通。我們需要 TLS 1.2 並推薦 TLS 1.3。
- 使用設定API和使用者活動記錄 AWS CloudTrail。
- 使用 AWS 加密解決方案以及其中的所有默認安全控制 AWS 服務。
- 使用進階的受管安全服務 (例如 Amazon Macie)，協助探索和保護儲存在 Amazon S3 的敏感資料。
- 如果 AWS 透過命令列介面或存取時需要 FIPS 140-3 驗證的密碼編譯模組API，請使用端點。FIPS 如需有關可用FIPS端點的詳細資訊，請參閱聯邦資訊處理標準 (FIPS) 140-3。

我們強烈建議您絕對不要將客戶的電子郵件地址等機密或敏感資訊，放在標籤或自由格式的文字欄位中，例如名稱欄位。這包括當您 AWS 服務 使用控制台、、EC2或與 Amazon 或其他API人 AWS CLI 合作時 AWS SDKs。您在標籤或自由格式文字欄位中輸入的任何資料都可能用於計費或診斷日誌。如果您提供URL給外部伺服器，我們強烈建議您不要在中包含認證資訊，URL以驗證您對該伺服器的要求。

目錄

- Amazon EBS 數據安全
- 靜態加密
- 傳輸中加密

Amazon EBS 數據安全

Amazon EBS 磁碟區會以原始、未格式化的區塊裝置呈現給您。這些裝置是在EBS基礎設施上建立的邏輯裝置，Amazon EBS 服務可確保在客戶使用或重複使用之前，裝置邏輯上是空的 (也就是說，原始區塊歸零或包含加密虛擬隨機資料)。

如果您有需要在使用之後或之前（或兩者）使用特定方法（或兩者）擦除所有數據的程序，例如 DoD 5220.22-M（國家工業安全計劃操作手冊）或 NIST800-88（媒體清理指南）中詳述的程序，則可以在 Amazon 上執行此操作。EBS該區塊層級活動將反映到 Amazon EBS 服務中的基礎儲存媒體。

靜態加密

EBS磁碟區

Amazon EBS 加密是適用於EBS磁碟區和快照的加密解決方案。它使用 AWS KMS keys。如需詳細資訊，請參閱 Amazon EBS使用者指南中的 Amazon EBS 加密。

[Windows 執行個體] 您也可以使用 Microsoft EFS 和NTFS權限進行資料夾層級和檔案層級加密。

執行個體儲存體磁碟區

NVMe執行個體儲存磁碟區上的資料會使用 XTS AES -256 密碼加密，並在執行個體的硬體模組上實作。用於加密寫入本機連接NVMe儲存裝置的資料金鑰是依客戶和每個磁碟區而定。這些金鑰由硬體模組產生且僅駐留在其中， AWS 人員無法存取此模組。這些加密金鑰會在執行個體停止或終止時銷毀，且無法復原。您無法停用此加密，也無法提供您自己的加密金鑰。

H1、D3 和 D3en 執行個體上執行個體儲存磁碟區上的資料會使用 XTS AES -256 和一次性金鑰加密。HDD

當您讓執行個體停止、休眠或終止時，執行個體存放區磁碟區中的所有儲存區塊都會重設。因此，資料無法透過另一個執行個體的執行個體存放區存取。

記憶體

下列執行個體會啟用記憶體加密：

- 具有 AWS 重力子處理器的執行個體。 AWS 引力 2、重力 on AWS 3 和 AWS 重力 on3E 支援永遠在線的記憶體加密。加密金鑰會在主機系統內安全地產生，不能離開主機系統，並在主機重新啟動或關閉電源時銷毀。如需詳細資訊，請參閱 AWS Graviton 處理器。
- 搭載第三代 Intel Xeon 可擴充處理器 (Ice Lake) 的執行個體 (例如 M6i 執行個體) 以及搭載第四代 Intel Xeon 可擴充處理器 (Sapphire Rapids) 的執行特體 (例如 M7i 執行個體)。這些處理器使用 Intel 總記憶體加密 () TME 支援永遠開啟的記憶體加密。

- 具有第三代AMDEPYC處理器 (米蘭) 的執行個體，例如 M6a 執行個體，以及第 4 代AMDEPYC處理器 (Genoa)，例如 M7a 執行個體。這些處理器使用AMD安全記憶體加密 () SME 支援永遠在線的記憶體加密。具有第三代AMDEPYC處理器 (Milan) 的執行個體也支援AMD安全加密虛擬化安全巢狀分頁 (-)。SEV SNP

傳輸中加密

在實體層加密

在離開 AWS 安全設施之前，透過 AWS 全球網路跨 AWS 區域流動的所有資料都會在實體層自動加密。之間AZs的所有流量都經過加密。額外的加密層 (包括本節所列的加密層) 可能會提供額外的保護。

Amazon VPC 對等互連和 Transit Gateway 跨區域對等互連提供的加密

所有使用 Amazon 對等互連和 Transit Gateway VPC 對等的跨區域流量在退出區域時都會自動大量加密。如本節先前所述，在所有流量離開 AWS 安全設施之前，會在實體層自動為所有流量提供額外的加密層。

執行個體間的加密

AWS 在所有類型的EC2執行個體之間提供安全和私有的連線。此外，某些執行個體類型使用基礎 Nitro System 硬體的卸載功能，以自動加密執行個體之間的傳輸中流量。此加密使用具有 256 位元加密與關聯資料 (AEAD) 演算法的驗證加密。這對網路效能沒有影響。若要支援執行個體之間額外的傳輸中流量加密，必須符合下列要求：

- 執行個體使用下列執行個體類型：
  - 一般用途：M5 DN，M5 N，兆兆安，M6i，M6i，M6 ID，M6 英寸，M7 安培，M7 克，M7gD，M7 億，M7i，M7-柔性
  - 計算優化：C5 安培，C5 公升，C5n，C6 安，C6 一，C6 標識，C6 英寸，C7 克，C7 克，C7 吉 D，C7 吉 D，C7 指數，C7 I，C7-柔性
  - 記憶體最佳化：R5 DN、R5 N、六安、R6 I、R6 英寸、R6 英寸、R7 A、R7 G、R7GD、R7 安培、R7 伊茲、R8 克、U-3 TB1、U-6TB1、U -12 TB 1、U 在 16 TB, 乌 7 英寸-24 TB, 乌 7 英寸 32 TB, X2 IDN, X2 一个
  - 儲存最佳化：D3、D3en、I3en、I4G、I4i、I4GN、i4 世代
  - 加速運算：DL1,, G4adDL2q, G4dN, G5, G6, G6e, GR6, 匯率 1, 二次, P4dN, P4d, P4de, P5, 中一, TRN1n, VT1
  - 高效能運算：Hpc6a、Hpc6id、Hpc7a、Hpc7g
- 這些執行個體位於相同的區域中。

- 執行個體位於相同VPC或對等VPCs，而且流量不會透過虛擬網路裝置或服務 (例如負載平衡器或傳輸閘道) 傳輸。

如本節先前所述，在所有流量離開 AWS 安全設施之前，會在實體層自動為所有流量提供額外的加密層。

使用 AWS CLI檢視加密執行個體之間傳輸中流量的執行個體類型

使用下列 describe-instance-types 命令。

```
aws ec2 describe-instance-types \
    --filters Name=network-info.encryption-in-transit-supported,Values=true \
    --query "InstanceTypes[*].[InstanceType]" \
    --output text | sort
```

來回加密 AWS Outposts

Outpost 會建立特殊的網路連線，稱為服務連結至其本 AWS 位目錄區域，並選擇性地建立您指定之子網VPC路的私人連線。透過這些連線的所有流量都會完整加密。如需詳細資訊，請參閱 AWS Outposts 使用者指南中的透過服務連結連線和傳輸中加密。

遠端存取加密

SSH和通訊RDP協定提供安全的通訊通道，讓您可以直接或透過EC2執行個體 Connect 進行遠端存取。使用 AWS Systems Manager 工作階段管理員或執行命令對執行個體的遠端存取是使用 TLS 1.2 進行加密的，建立連線的請求會使用 Sigv4 簽署，並經過AWS Identity and Access Management驗證和授權。

您有責任使用加密協定 (例如傳輸層安全性 (TLS) 來加密用戶端和 Amazon EC2 執行個體之間傳輸中的敏感資料。

(Windows 執行個體) 請務必只允許EC2執行個體與 AWS API端點或其他敏感遠端網路服務之間的加密連線。您可以透過傳出安全群組或 Windows 防火牆規則來強制執行此操作。

Amazon 基礎設施安全 EC2

作為受管服務，Amazon 彈性運算雲端受到 AWS 全球網路安全的保護。有關 AWS 安全服務以及如何 AWS 保護基礎架構的詳細資訊，請參閱AWS 雲端安全 若要使用基礎架構安全性的最佳做法來設計您的 AWS 環境，請參閱安全性支柱架構良 AWS 好的架構中的基礎結構保護。

您可以使用 AWS 已發佈的API呼叫EC2透過網路存取 Amazon。使用者端必須支援下列專案：

- 傳輸層安全性 (TLS)。我們需要 TLS 1.2 並推薦 TLS 1.3。
- 具有完美前向保密（ ）的密碼套件，例如（短暫的迪菲-赫爾曼PFS）或DHE（橢圓曲線短暫迪菲-赫爾曼）。ECDHE現代系統(如 Java 7 和更新版本)大多會支援這些模式。

此外，請求必須使用存取金鑰 ID 和與IAM主體相關聯的秘密存取金鑰來簽署。或者，您可以透過 AWS Security Token Service (AWS STS) 來產生暫時安全憑證來簽署請求。

如需詳細資訊，請參閱安全性支柱 — AWS Well-Architected 的架構中的基礎結構保護。

網路隔離

虛擬私有雲（VPC）是您自己在 AWS 雲中邏輯隔離區域中的虛擬網絡。使用個別VPCs來依工作負載或組織實體隔離基礎結構。

子網路是一個中的 IP 位址範圍VPC。當您啟動執行個體時，您可以將其啟動到VPC. 使用子網路將應用程式的層 (例如 Web、應用程式和資料庫) 隔離在單VPC一層。如果不應該從網際網路直接存取，則針對您的執行個體使用私有子網。

要EC2API從您VPC使用私有 IP 地址調用 Amazon，請使用 AWS PrivateLink. 如需詳細資訊，請參閱EC2使用界面VPC端點訪問 Amazon。

實體主機上的隔離

相同實體主機上的不同EC2執行個體會彼此隔離，就像它們位於不同的實體主機上一樣。Hypervisor 會隔離CPU和記憶體，並提供虛擬化磁碟，而非原始磁碟裝置的存取權。

當您停止或終止執行個體時，Hypervisor 會先清除配置到該執行個體的記憶體 (設定為零)，然後再將之配置到新執行個體，而且會重設儲存體的每個區塊。這可確保不會蓄意地向其他執行個體公開您的資料。

網路MAC位址是由網 AWS 路基礎結構動態指派給執行個體。IP 位址是由 AWS 網路基礎結構動態指派給執行個體，或是由EC2系統管理員透過驗證的API要求指派。 AWS 網路允許執行個體僅從指派給執行個體的MAC和 IP 位址傳送流量。否則，流量會遭到捨棄。

依預設，執行個體無法接收不是特別定址給它的流量。如果您需要在執行個體上執行網路位址轉譯 (NAT)、路由或防火牆服務，您可以停用網路介面的來源/目的地檢查。

控制網路流量

請考慮下列選項來控制EC2執行個體的網路流量：

- 使用安全性群組限制您執行個體的存取權限。設定允許最低所需網路流量的規則。例如，您可以僅允許來自公司網路位址範圍的流量，或僅允許特定通訊協定的流量，例如HTTPS。對於 Windows 執行個體，請允許 Windows 管理流量和最小輸出連線。
- 利用安全群組做為控制 Amazon EC2 執行個體網路存取的主要機制。必要時，請ACLs謹慎使用網路來提供無狀態的粗粒網路控制。ACLs由於安全性群組能夠執行可設定狀態封包篩選，並建立參照其他安全性群組的規則，因此安全性群組比網路更具用途。但是，網路ACLs可以作為拒絕特定流量子集或提供高階子網路防護軌的次要控制項有效。此外，由於網路ACLs適用於整個子網路，因此可以 defense-in-depth 在沒有正確安全性群組的情況下無意中啟動執行個體時使用它們。
- [Windows 執行個體] 使用群組原則物件 (GPO) 集中管理 Windows 防火牆設定，以進一步強化網路控制。客戶通常會使用 Windows 防火牆進一步了解網路流量，以及補充安全群組篩選條件，以建立進階規則以封鎖特定應用程式存取網路，或篩選來自子集 IP 地址的流量。例如，Windows 防火牆可以限制特定使用者或應用程式對EC2中繼資料服務 IP 位址的存取。或者，面向公眾的服務可能會使用安全群組來限制特定連接埠的流量，並使用 Windows 防火牆來維護明確封鎖之 IP 地址的清單。
- 如果不應該從網際網路直接存取，則針對您的執行個體使用私有子網。使用防禦主機或NAT閘道，從私有子網路中的執行個體存取網際網路。
- [Windows 執行個體] 使用安全的管理通訊協定，例如RDP封裝在SSL/TLS上。遠端桌面閘道快速入門提供部署遠端桌面閘道的最佳做法，包括設定RDP為使用SSL/TLS。
- [Windows 執行個體] 使用「作用中目錄」，或 AWS Directory Service 嚴格集中控制和監控互動式使用者和群組對 Windows 執行個體的存取，並避免本機使用者權限。亦請避免使用網域管理員，而是建立更精細、以應用程式特定角色為基礎的帳戶。只有足夠的管理 (JEA) 允許在沒有互動式或管理員存取權的情況下管理 Windows 執行個體的變更。此外，JEA可讓組織鎖定執行個體管理所需之 Windows PowerShell 命令子集的管理存取權。如需其他資訊，請參閱AWS 安全性最佳實務白皮書中關於管理作業系統層EC2級 Amazon 的存取權限一節。
- [Windows 執行個體] 系統管理員應使用存取權限有限的 Windows 帳戶來執行日常活動，並且僅在必要時提升存取權限以執行特定組態變更。此外，只有在絕對必要時才能直接存取 Windows 執行個體。而是利用中央組態管理系統，例如EC2執行命令、系統中心組態管理員 (SCCM) PowerShell DSC、Windows 或 Amazon EC2 Systems Manager (SSM)，將變更推送到 Windows 伺服器。
- 使用所需的最低網VPC路路由設定 Amazon 子網路路由表。例如，只將重新報告直接網際網路存取的 Amazon EC2 執行個體放入具有網際網路閘道路徑的子網路中，並且只將需要直接存取內部網路的 Amazon EC2 執行個體放入具有路由到虛擬私有閘道的子網路中。
- 請考慮使用其他安全群組或網路界面來控制和稽核 Amazon EC2 執行個體管理流量，與一般應用程式流量分開。此方法可讓客戶實作變更控制的特殊IAM原則，讓稽核安全性群組規則或自動化規則驗證指令碼的變更更加容易。使用多個網路介面也會提供其他控制網路流量的選項，包括建立以主機為基礎的路由原則，或根據網路介面的指派VPC子網路運用不同的子網路路由規則。

- 使 AWS Direct Connect 用 AWS Virtual Private Network 或建立從遠端網路到您的VPCs. 如需詳細資訊，請參閱網路到 Amazon VPC 連線選項。
- 使用VPC流量記錄來監控到達執行個體的流量。
- 使用GuardDuty 惡意程式碼防護來識別執行個體上指示惡意軟體的可疑行為，這些行為可能會損害您的工作負載、重新利用資源以供惡意使用，以及取得未經授權的資料存取權。
- 使用執GuardDuty 行階段監控來識別並回應執行個體的潛在威脅。如需詳細資訊，請參閱執行階段監控如何搭配 Amazon EC2 執行個體運作
- 使用AWS Security Hub、可 Reachability Analyzer 或網路存取分析器來檢查執行個體是否有意外的網路可存取性。
- 使用EC2執行個體 Connec t 線，透過 Secure Shell (SSH) 連線至您的執行個體，而不需要共用和管理SSH金鑰。
- 使用AWS Systems Manager 工作階段管理員遠端存取執行個體，而不是開啟輸入SSH或RDP連接埠，以及管理金鑰配對。
- 使用AWS Systems Manager 執行命令自動執行一般管理工作，而不是連線至執行個體。
- [Windows 執行個體] 許多 Windows 作業系統角色和 Microsoft 商務應用程式也提供增強的功能，例如 IP 位址範圍內的限制IIS、Microsoft SQL 伺服器中的 TCP /IP 篩選原則，以及 Microsoft Exchange 中的連線篩選原則。應用程式層內的網路限制功能可為重要的商務應用程式伺服器提供額外的防禦層。

Amazon VPC 支援其他網路安全控制，例如閘道、代理伺服器和網路監控選項。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的控制網路流量。

Amazon 的韌性 EC2

AWS 全球基礎架構是圍繞區 AWS 域和可用區域建立的。區域提供多個分開且隔離的實際可用區域，並以低延遲、高輸送量和高度備援網路連線相互連結。透過可用區域，您可以設計與操作的應用程式和資料庫，在可用區域之間自動容錯移轉而不會發生中斷。可用區域的可用性、容錯能力和擴充能力，均較單一或多個資料中心的傳統基礎設施還高。

如果您需要在更大的地理距離內複製資料或應用程式，請使用 AWS Local Zones。「 AWS 本地區域」是您使用者所在地理位置附近的「 AWS 區域」的延伸。Local Zones 有自己的網際網路連線，並支援 AWS Direct Connect。與所有區 AWS 域一樣，L AWS ocal Zones 與其他 AWS 區域完全隔離。

如果您需要在 AWS 本機區域中複寫資料或應用程式， AWS 建議您使用下列其中一個區域做為容錯移轉區域：

- 另一個 Local Zone
- 區域中不是父區域的可用區域。您可以使用describe-availability-zones指令檢視父系區域。

如需區域和可用區域的相關 AWS 資訊，請參閱AWS 全域基礎結構。

除了 AWS 全球基礎設施之外，Amazon 還EC2提供下列功能來支援您的資料恢復能力：

- AMIs跨區域複製
- 跨區域複製EBS快照
- AMIs使用 Amazon Data EBS Lifecycle Manager 自動化支援
- 使用 Amazon Data Lifecycle Manager 自動化EBS快照
- 使用 Amazon EC2 Auto Scaling 維護叢集的運作狀態和可用性
- 使用 Elastic Load Balancing 將輸入流量分配到單一可用區域或多個可用區域中的多個執行個體。

Amazon 的合規驗證 EC2

若要瞭解 AWS 服務 是否屬於特定規範遵循方案的範圍內，請參閱AWS 服務 遵循規範計劃方案中的，並選擇您感興趣的合規方案。如需一般資訊，請參閱AWS 規範計劃AWS。

您可以使用下載第三方稽核報告 AWS Artifact。如需詳細資訊，請參閱下載中的報告中的 AWS Artifact。

您在使用時的合規責任取決 AWS 服務 於您資料的敏感性、公司的合規目標以及適用的法律和法規。AWS 提供下列資源以協助遵循法規：

- 安全性與合規性快速入門指南 — 這些部署指南討論架構考量，並提供部署以安全性和合規性 AWS 為重點的基準環境的步驟。
- 在 Amazon Web Services 上進行HIPAA安全與合規架構 — 本白皮書說明公司如何使用建立符合資格的應 AWS 用程HIPAA式。

> **Note**
>
> 並非所有 AWS 服務 人都HIPAA符合資格。如需詳細資訊，請參閱合HIPAA格服務參考資料。

- AWS 合規資源AWS — 此工作簿和指南集合可能適用於您的產業和所在地。

- AWS 客戶合規指南 — 透過合規的角度瞭解共同的責任模式。這份指南總結了在多個架構 (包括美國國家標準 AWS 服務 與技術研究所 (NIST)、支付卡產業安全標準委員會 () 和國際標準化組織 ()) 中保護安全控制指引的最佳實務作法，並將其對應至安全性控制。PCI ISO
- 使用AWS Config 開發人員指南中的規則評估資源 — 此 AWS Config 服務會評估您的資源組態符合內部實務、產業準則和法規的程度。
- AWS Security Hub— 這 AWS 服務 提供了內部安全狀態的全面視圖 AWS。Security Hub 使用安全控制，可評估您的 AWS 資源並檢查您的法規遵循是否符合安全業界標準和最佳實務。如需支援的服務和控制清單，請參閱 Security Hub controls reference。
- Amazon GuardDuty — 透過監控環境中的 AWS 帳戶可疑和惡意活動，藉此 AWS 服務 偵測您的工作負載、容器和資料的潛在威脅。 GuardDuty 可協助您因應各種合規性需求 PCIDSS，例如符合特定合規性架構所要求的入侵偵測需求。
- AWS Audit Manager— 這 AWS 服務 有助於您持續稽核您的 AWS 使用情況，以簡化您管理風險的方式，以及遵守法規和業界標準的方式。

Amazon 的身份和訪問管理 EC2

AWS Identity and Access Management (IAM) 可協助系統管理員安全地控制 AWS 資源存取權。 AWS 服務 IAM管理員控制誰可以驗證 (登入) 和授權 (具有權限) 使用 Amazon EC2 資源。IAM是一種您 AWS 服務 可以使用，無需額外費用。

您的安全登入資料可識別您的服務， AWS 並授予您 AWS 資源的存取權，例如 Amazon EC2 資源。您可以使用 Amazon 的功能，EC2並IAM允許其他使用者、服務和應用程式使用您的 Amazon EC2 資源，而無需共用您的安全登入資料。您可以用IAM來控制其他使用者如何使用您的資源 AWS 帳戶，也可以使用安全群組控制對 Amazon EC2 執行個體的存取。您可以選擇允許完全或有限使用 Amazon EC2 資源。

如果您是開發人員，則可以使用IAM角色來管理在執行個體上執行的應用程式所需的安全性認EC2證。將IAM角色附加到執行個體後，在執行個體上執行的應用程式就可以從執行個體中繼資料服務 (IMDS) 擷取認證。

如需使用保護資 AWS 源安全的最佳做法IAM，請參閱《使用IAM者指南》IAM中的「安全性最佳做法」。

目錄

- Amazon 的基於身份的政策 EC2
- 控制 Amazon 訪問的示例政策 EC2 API

- 控制 Amazon EC2 主控台存取權的範例政策
- AWS Amazon 的受管政策 EC2
- IAMAmazon 的角色 EC2

Amazon 的基於身份的政策 EC2

預設情況下，使用者沒有使用 Amazon、Amazon EC2 主控台或建立或修改 Amazon EC2 EC2 API 資源或執行任務的權限CLI。若要允許使用者建立或修改資源以及執行工作，您必須建立IAM政策，以授與使用者使用所需特定資源和API動作的權限，然後將這些原則附加至需要這些權限的使用者、群組或IAM角色。

將政策連接到使用者、使用者群組或角色時，政策會允許或拒絕使用者在特定資源上執行特定任務的許可。如需有關IAM策略的更多一般資訊，請參閱《IAM使用指南》IAM中的〈原則和權限〉。如需有關管理和建立自訂IAM原則的詳細資訊，請參閱管理IAM原則。

IAM政策必須授予或拒絕許可才能使用一個或多個 Amazon EC2 動作。政策中還必須指定可用於動作的資源，可為所有資源，或在某些案例中為特定的資源。政策也可以包含您套用到資源的條件。

若要開始使用，您可以檢查 Amazon 的 AWS 受管政策是否EC2符合您的需求。否則，您可以建立自己的自訂原則。如需詳細資訊，請參閱the section called “AWS 受管政策”。

目錄

- 政策語法
- Amazon 的行動 EC2
- 支援 Amazon 動作的資源層級許可 EC2 API
- Amazon 資源名稱（ARNs）Amazon EC2
- Amazon 的條件密鑰 EC2
- 使用以屬性為基礎的存取控制存取
- 向使用者、群組和角色授予許可
- 檢查使用者是否擁有必要的許可

政策語法

IAM策略是由一個或多個陳述式組成的JSON文件。每個陳述式的結構如下所示。

```
{
  "Statement":[{
    "Effect":"effect",
    "Action":"action",
    "Resource":"arn",
    "Condition":{
      "condition":{
        "key":"value"
        }
      }
    }
  ]
}
```

陳述式由各種元素組成：

- Effect (效果)：效果 可以是 `Allow` 或 `Deny`。根據預設，使用者沒有使用資源和API動作的權限，因此拒絕所有要求。明確允許覆寫預設值。明確拒絕覆寫任何允許。
- 動作：動作是您授與或拒絕權限的特定API動作。若要了解指定動作，請參閱Amazon 的行動 EC2。
- Resource (資源)：受動作影響的資源。某些 Amazon EC2 API 動作可讓您在政策中包含可透過動作建立或修改的特定資源。您可以使用 Amazon 資源名稱 (ARN) 或使用萬用字元 (*) 來指定資源，表示該陳述式適用於所有資源。如需詳細資訊，請參閱支援 Amazon 動作的資源層級許可 EC2 API。
- Condition (條件)：條件為選擇性。您可以使用它們來控制何時政策開始生效。如需有關指定 Amazon 條件的詳細資訊EC2，請參閱Amazon 的條件密鑰 EC2。

如需有關策略需求的詳細資訊，請參閱IAM使用者指南中的IAMJSON政策參考資料。如需 Amazon 的IAM政策聲明範例EC2，請參閱控制 Amazon 訪問的示例政策 EC2 API。

Amazon 的行動 EC2

在IAM政策聲明中，您可以從支持的任何服務中指定任何API操作IAM。對於 AmazonEC2，請使用以下前綴和API動作名稱：`ec2:`。例如：`ec2:RunInstances` 和 `ec2:CreateImage`。

若要在單一陳述式中指定多個動作，請用逗號分隔，如下所示：

```
"Action": ["ec2:action1", "ec2:action2"]
```

您也可以使用萬用字元指定多個動作。例如，您可以指定名稱開頭有「Describe」文字的所有動作，如下所示：

```
"Action": "ec2:Describe*"
```

> **Note**
>
> 目前，Amazon EC2 說明 * API 動作不支援資源層級許可。如需 Amazon 資源層級許可的詳細資訊EC2，請參閱。Amazon 的基於身份的政策 EC2

若要指定所有 Amazon EC2 API 動作，請使用 * 萬用字元，如下所示：

```
"Action": "ec2:*"
```

如需 Amazon EC2 動作的清單，請參閱服務授權參考EC2中 Amazon 定義的動作。

支援 Amazon 動作的資源層級許可 EC2 API

資源層級許可能夠讓您指定使用者可執行動作的資源。Amazon EC2 對資源級許可提供部分支持。這表示對於某些 Amazon EC2 動作，您可以根據必須滿足的條件或允許使用者使用的特定資源，控制何時允許使用者使用這些動作。例如，您可以授與使用者啟動執行個體的權限，但只能授與特定類型的執行個體，而且只能使用特定的執行個體AMI。

若要在IAM政策陳述式中指定資源，請使用其 Amazon 資源名稱 (ARN)。若要取得有關指定ARN值的更多資訊，請參閱〈〉Amazon 資源名稱（ARNs）Amazon EC2。如果某個API動作不支援個別ARNs，您必須使用萬用字元 (*) 來指定所有資源都可以受到該動作的影響。

若要查看識別哪些 Amazon EC2 API 動作支援資源層級許可的表格，以ARNs及可在政策中使用的和條件金鑰，請參閱 Amazon 的動作、資源和條件金鑰。EC2

請記住，您可以在用於 Amazon 動作的IAM政策中套用以標籤為基礎的資源層級許可。EC2 API這可讓您更有效地控制使用者可以建立、修改或使用的資源。如需詳細資訊，請參閱授予在建立期間標記 Amazon EC2 資源的權限。

Amazon 資源名稱（ARNs）Amazon EC2

每個IAM策略聲明都適用於您使用它們指定的資源ARNs。

ARN具有以下一般語法：

```
arn:aws:[service]:[region]:[account-id]:resourceType/resourcePath
```

服務

服務 (例如，ec2)。

region

資源的區域 (例如，us-east-1)。

account-id

AWS 帳號 ID，不含連字號 (例如，123456789012)。

resourceType

資源類型 (例如，instance)。

resourcePath

識別資源的路徑。您可以在路徑中使用 * 萬用字元。

例如，您可以使用以下方式在陳述式中指示特定的執行個體 (i-1234567890abcdef0)。ARN

```
"Resource": "arn:aws:ec2:us-east-1:123456789012:instance/i-1234567890abcdef0"
```

您也可以使用 * 萬用字元指定所有屬於特定帳戶的執行個體，如下所示。

```
"Resource": "arn:aws:ec2:us-east-1:123456789012:instance/*"
```

您也可以使用 * 萬用字元來指定屬於特定帳戶的所有 Amazon EC2 資源，如下所示。

```
"Resource": "arn:aws:ec2:us-east-1:123456789012:*"
```

若要指定所有資源或特定API動作不支援ARNs，請在元素中使用 * 萬用字Resource元，如下所示。

```
"Resource": "*"
```

許多 Amazon EC2 API 操作涉及多種資源。例如，AttachVolume將 Amazon EBS 磁碟區附加到執行個體，因此使用者必須擁有使用磁碟區和執行個體的許可。若要在單一陳述式中指定多個資源，請以逗號ARNs分隔它們，如下所示。

```
"Resource": ["arn1", "arn2"]
```

如需 Amazon EC2 資源ARNs的清單，請參閱 Amazon 定義的資源類型EC2。

Amazon 的條件密鑰 EC2

在政策陳述式中，您可以選擇性的指定生效時發揮控制效果的條件。每個條件都包含一或多個索引鍵/值對。條件鍵不區分大小寫。我們已定義 AWS 全域條件金鑰，以及其他服務特定條件金鑰。

如需 Amazon 服務特定條件金鑰的清單EC2，請參閱 Amazon 的條件金鑰。EC2Amazon EC2 還實現了 AWS 全局條件密鑰。如需詳細資訊，請參閱《IAM使用指南》中所有請求中的可用資訊。

所有 Amazon EC2 操作都支持`aws:RequestedRegion`和`ec2:Region`條件密鑰。如需詳細資訊，請參閱範例：限制特定區域的存取。

若要在IAM原則中使用條件索引鍵，請使用`Condition`陳述式。例如，下列政策會授與使用者新增及移除任何安全群組之傳入和傳出規則的許可。它會使用`ec2:Vpc`條件索引鍵來指定這些動作只能在特定的安全性群組上執行VPC。

```
{
"Version": "2012-10-17",
  "Statement":[{
    "Effect":"Allow",
    "Action": [
       "ec2:AuthorizeSecurityGroupIngress",
       "ec2:AuthorizeSecurityGroupEgress",
       "ec2:RevokeSecurityGroupIngress",
       "ec2:RevokeSecurityGroupEgress"],
     "Resource": "arn:aws:ec2:region:account:security-group/*",
      "Condition": {
        "StringEquals": {
          "ec2:Vpc": "arn:aws:ec2:region:account:vpc/vpc-11223344556677889"
        }
      }
    }
  ]
}
```

如果您在單一條件中指定多個條件或多個鍵，我們會使用邏輯AND運算來評估它們。若您針對單一索引鍵使用多個值指定單一條件，我們會使用邏輯 OR 操作評估條件。若要授予許可，必須符合所有的條件。

您也可以在指定條件時使用預留位置。如需詳細資訊，請參閱IAM使用指南中的IAM政策元素：變數和標籤。

> **Important**
>
> 許多條件索引鍵特定於資源，而某些API動作則使用多個資源。若您撰寫的政策具備條件鍵，請使用陳述式的 `Resource` 元素來指定套用該條件鍵的資源。若未這麼做，該政策可能會導致使用者完全無法執行該動作，因為未套用條件金鑰之資源的條件檢查會失敗。如果您不想指定資源，或者您已將原則的`Action`元素撰寫為包含多個API動作，則必須使用`...IfExists`條件類型，以確保不使用該資源的資源忽略條件索引鍵。如需詳細資訊，請參閱... IfExists 《IAM使用者指南》中的條件。

條件索引鍵

- ec2:Attribute 條件金鑰
- ec2:ResourceID 條件索引鍵
- ec2:SourceInstanceARN 條件金鑰

ec2:Attribute 條件金鑰

`ec2:Attribute` 條件金鑰可用於按資源屬性篩選存取的條件。

此條AttributeValue件索引鍵僅支援原始資料類型的屬性 (例如字串或整數)，或僅包含 Value 屬性 (例如ModifyImageAttributeAPI動作的「描述」或「ImdsSupport物件」) 的複雜物件。條件索引鍵不能用於包含多個屬性的複雜物件，例如的LaunchPermission物件ModifyImageAttribute。

例如，下列原則會使用`ec2:Attribute/Description`條件索引鍵，依動作的複雜「描述」物件篩選存ModifyImageAttributeAPI取。條件金鑰僅允許將映像描述修改為 `Production` 或 `Development` 的請求。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:ModifyImageAttribute",
      "Resource": "arn:aws:ec2:us-east-1::image/ami-*",
      "Condition": {
        "StringEquals": {
          "ec2:Attribute/Description": [
            "Production",
            "Development"
```

```
          ]
        }
      }
    }
  ]
}
```

下列範例原則會使用`ec2:Attribute`條件索引鍵，依動作的原始屬性內容篩選存ModifyImageAttributeAPI取。條件金鑰會拒絕嘗試修改映像描述的所有請求。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": "ec2:ModifyImageAttribute",
      "Resource": "arn:aws:ec2:us-east-1::image/ami-*",
      "Condition": {
        "StringEquals": {
          "ec2:Attribute": "Description"
        }
      }
    }
  ]
}
```

ec2:ResourceID 條件索引鍵

將下列`ec2:`*`Resource`*`ID`條件索引鍵與指定的動API作搭配使用時，會使用條件索引鍵值來指定API動作所建立的產生資源。 `ec2:`*`Resource`*`ID`條件鍵不能用於指定API請求中指定的源資源。如果您使用下列其中一個`ec2:`*`Resource`*`ID`條件索引鍵搭配指定API，則必須一律指定萬用字元 (*)。如果指定不同的值，則該條件在執行期期間一律會解析為 *。例如，若要搭配使用`ec2:ImageId`條件索引鍵CopyImageAPI，您必須指定條件索引鍵，如下所示：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:CopyImage",
      "Resource": "arn:aws:ec2:us-east-1::image/ami-*",
      "Condition": {
```

```
        "StringEquals": {
          ec2:ImageID": "*"
        }
      }
    }
  ]
}
```

建議您避免在下列動作中使用這些條件索引API鍵：

- ec2:DhcpOptionsID – CreateDhcpOptions
- ec2:ImageID— CopyImage CreateImage、ImportImage、和 RegisterImage
- ec2:InstanceID— RunInstances 和 ImportInstance
- ec2:InternetGatewayID – CreateInternetGateway
- ec2:NetworkAclID – CreateNetworkAcl
- ec2:NetworkInterfaceID – CreateNetworkInterface
- ec2:PlacementGroupName – CreatePlacementGroup
- ec2:RouteTableID – CreateRouteTable
- ec2:SecurityGroupID – CreateSecurityGroup
- ec2:SnapshotID— CopySnapshot CreateSnapshot、CreateSnapshots、和 ImportSnapshots
- ec2:SubnetID – CreateSubnet
- ec2:VolumeID— CreateVolume 和 ImportVolume
- ec2:VpcID – CreateVpc
- ec2:VpcPeeringConnectionID – CreateVpcPeeringConnection

若要根據特定資源篩選存取IDs，建議您使用Resource原則元素，如下所示。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:CopyImage",
      "Resource": "arn:aws:ec2:us-east-1::image/ami-01234567890abcdef"
    }
```

```
  ]
}
```

ec2:SourceInstanceARN 條件金鑰

用`ec2:SourceInstanceARN`於指定ARN發出要求的執行個體。這是一個AWS 全局條件密鑰，這意味著您可以將其與 Amazon 以外的服務一起使用EC2。如需政策範例，請參閱「範例：允許特定執行個體檢視其他 AWS 服務中的資源」。

使用以屬性為基礎的存取控制存取

當您建立授與使用者使用EC2資源之權限的IAM策略時，您可以在原則的`Condition`元素中包含標籤資訊，以根據標籤控制存取。這就是所謂的基於屬性的訪問控制（ ）ABAC。ABAC可以更好地控制使用者可以修改、使用或刪除哪些資源。如需詳細資訊，請參閱有什麼ABAC用途 AWS？

例如：您可以建立一個政策，允許使用者終止執行個體，但如果執行個體具有標籤 `environment=production`，則拒絕該動作。若要這樣做，您可以使用 `aws:ResourceTag` 條件金鑰，根據連接至資源的標籤允許或拒絕存取資源。

```
"StringEquals": { "aws:ResourceTag/environment": "production" }
```

若要了解 Amazon EC2 API 動作是否支援使用`aws:ResourceTag`條件金鑰控制存取，請參閱 Amazon 的動作、資源和條件金鑰EC2。由於 `Describe` 動作不支援資源層級許可，您必須在不同的陳述式中指定它們，無需條件。

如需IAM原則範例，請參閱控制 Amazon 訪問的示例政策 EC2 API。

如果您允許或拒絕使用者根據標籤存取資源，請務必考慮明確拒絕使用者將這些標籤新增至相同資源或從中移除的能力。否則，使用者可能透過修改標籤來避開您的限制，並取得資源的存取。

向使用者、群組和角色授予許可

若要提供存取權，請新增權限至您的使用者、群組或角色：

- 使用者和群組位於 AWS IAM Identity Center：

  建立權限合集。請按照 AWS IAM Identity Center 使用者指南 中的 建立權限合集 說明進行操作。
- IAM透過身分識別提供者管理的使用者：

  建立聯合身分的角色。請遵循《使用指南》中的〈為第三方身分識別提供IAM者 (同盟) 建立角色〉中的指示進行。

- IAM使用者：
  - 建立您的使用者可擔任的角色。請按照《用戶指南》中的「為IAM用戶創建角色」中的IAM說明進行操作。
  - (不建議) 將政策直接附加至使用者，或將使用者新增至使用者群組。遵循《使用者指南》中的「向使用者 (主控台) 新增權限」IAM 中的指示進行。

檢查使用者是否擁有必要的許可

建立IAM原則之後，建議您先檢查原則是否授與使用者使用所需特定API動作和資源的權限，然後再將原則投入生產環境。

首先，建立用於測試目的的的使用者，然後將您建立的IAM策略附加到測試使用者。接著，以測試使用者的身分提出請求。

如果您正在測試的 Amazon EC2 動作建立或修改資源，則應使用`DryRun`參數提出請求 (或使用`--dry-run`選項執行 AWS CLI 命令)。在此案例中，呼叫會完成授權檢查，但不會完成操作。例如，您可以檢查使用者是否可以終止特定執行個體，而無須實際終止它。若測試使用者具有必要的許可，請求會傳回 `DryRunOperation`。否則，會傳回 `UnauthorizedOperation`。

如果政策未授予使用者預期的許可，或授予過多許可，您可以視需要調整政策並重新測試，直到您取得所要的結果。

> **Important**
>
> 政策變更的散佈可能需要幾分鐘時間才能生效。因此，我們建議您等待五分鐘，然後再測試您的政策更新。

如果授權檢查失敗，請求將傳回包含診斷資訊的編碼訊息。您可使用 `DecodeAuthorizationMessage` 動作將訊息解碼。若要取得更多資訊，請參閱〈DecodeAuthorizationMessage AWS Security Token ServiceAPI參考〉和〈AWS CLI 指令參考〉decode-authorization-message中的〈〉。

控制 Amazon 訪問的示例政策 EC2 API

您可以使用IAM政策授予使用者使用 Amazon 所需的許可EC2。如需 step-by-step 指示，請參閱IAM使用指南中的建立IAM策略。

下列範例顯示可用來授與使用者使用 Amazon 權限的政策陳述式EC2。這些原則是針對使用 AWS CLI 或發出的請求而設計的 AWS SDK。在下面的例子中，替換每個 *user input placeholder* 使用您自己的信息。

範例

- 範例：唯讀存取
- 範例：限制特定區域的存取
- 使用執行個體
- 啟動執行個體 (RunInstances)
- 使用 競價型執行個體
- 範例：使用 預留執行個體
- 範例：標籤資源
- 範例：使用 IAM 角色
- 範例：使用路由表
- 範例：允許特定執行個體檢視其他 AWS 服務中的資源
- 範例：使用啟動範本
- 使用執行個體中繼資料
- 使用 Amazon EBS 磁碟區和快照

如需在 Amazon EC2 主控台中工作的政策範例，請參閱控制 Amazon EC2 主控台存取權的範例政策。

範例：唯讀存取

下列政策授予使用者使用名稱開頭為之所有 Amazon EC2 API 動作的許可`Describe`。`Resource`元素使用萬用字元來指示使用者可以透過這些API動作指定所有資源。在API動作不支援資源層級權限的情況下，* 萬用字元也是必要的。如需有關ARNs可用於哪些 Amazon EC2 API 動作的詳細資訊，請參閱Amazon 的動作、資源和條件金鑰EC2。

用戶沒有對資源執行任何操作的權限（除非另一個語句授予他們這樣做的權限），因為他們默認情況下被拒絕使用API操作的權限。

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

```
      "Action": "ec2:Describe*",
      "Resource": "*"
    }
   ]
}
```

範例：限制特定區域的存取

除非區域為歐洲 (法蘭克福)，否則下列政策拒絕使用者使用所有 Amazon EC2 API 動作的權限。它使用全域條件金鑰`aws:RequestedRegion`，所有 Amazon EC2 API 動作都支援此金鑰。

```
{
  "Version":"2012-10-17",
  "Statement": [
       {
      "Effect": "Deny",
      "Action": "ec2:*",
      "Resource": "*",
      "Condition": {
        "StringNotEquals": {
          "aws:RequestedRegion": "eu-central-1"
        }
      }
    }
  ]
}
```

或者，您可以使用條件金鑰`ec2:Region`，該金鑰專屬於 Amazon，EC2且所有 Amazon EC2 API 動作都支援此金鑰。

```
{
  "Version":"2012-10-17",
  "Statement": [
       {
      "Effect": "Deny",
      "Action": "ec2:*",
      "Resource": "*",
      "Condition": {
        "StringNotEquals": {
          "ec2:Region": "eu-central-1"
        }
      }
```

```
    }
  ]
}
```

使用執行個體

範例

- 範例：描述、啟動、停止、開始及終止所有執行個體
- 範例：描述所有執行個體，並只停止、啟動及終止特定執行個體

範例：描述、啟動、停止、開始及終止所有執行個體

下列策略授與使用者使用`Action`元素中指定API動作的權限。`Resource`元素使用 * 萬用字元表示使用者可以透過這些API動作指定所有資源。在API動作不支援資源層級權限的情況下，* 萬用字元也是必要的。如需有關ARNs可用於哪些 Amazon EC2 API 動作的詳細資訊，請參閱 Amazon 的動作、資源和條件金鑰EC2。

用戶沒有使用任何其他API操作的權限（除非另一個語句授予他們這樣做的權限），因為默認情況下，用戶被拒絕使用API操作的權限。

```
{
   "Version": "2012-10-17",
   "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeInstances",
        "ec2:DescribeImages",
        "ec2:DescribeKeyPairs",
        "ec2:DescribeSecurityGroups",
        "ec2:DescribeAvailabilityZones",
        "ec2:RunInstances",
        "ec2:TerminateInstances",
        "ec2:StopInstances",
        "ec2:StartInstances"
      ],
      "Resource": "*"
    }
   ]
}
```

範例：描述所有執行個體，並只停止、啟動及終止特定執行個體

下列政策允許使用者描述所有執行個體、只開始及停止執行個體 i-1234567890abcdef0 和 i-0598c7d356eba48d7，並只終止美國東部 (維吉尼亞北部) 區域 (us-east-1) 中資源標籤為 "purpose=test" 的執行個體。

第一個陳述式的 Resource 元素使用 * 萬用字元，表示使用者可以為該動作指定所有資源；在本例中，他們可以列出所有執行個體。在API動作不支援資源層級權限的情況下，ec2:DescribeInstances* 萬用字元也是必要的 (在本例中為)。如需有關ARNs可用於哪些 Amazon EC2 API 動作的詳細資訊，請參閱 Amazon 的動作、資源和條件金鑰EC2。

第二個陳述式的 StopInstances 和 StartInstances 動作使用資源層級許可。特定例證在Resource元素ARNs中以其指示。

第三個陳述式可讓使用者終止美國東部 (維吉尼亞北部) 區域 (us-east-1) 中屬於指定 AWS 帳戶的所有執行個體，但僅限執行個體具有標籤的執行個體"purpose=test"。Condition 元素可限定政策陳述式生效的條件。

```
{
   "Version": "2012-10-17",
   "Statement": [
   {
   "Effect": "Allow",
      "Action": "ec2:DescribeInstances",
      "Resource": "*"
   },
   {
      "Effect": "Allow",
      "Action": [
        "ec2:StopInstances",
        "ec2:StartInstances"
      ],
      "Resource": [
        "arn:aws:ec2:us-east-1:account-id:instance/i-1234567890abcdef0",
        "arn:aws:ec2:us-east-1:account-id:instance/i-0598c7d356eba48d7"
      ]
    },
    {
      "Effect": "Allow",
      "Action": "ec2:TerminateInstances",
      "Resource": "arn:aws:ec2:us-east-1:account-id:instance/*",
      "Condition": {
         "StringEquals": {
```

```
            "aws:ResourceTag/purpose": "test"
         }
      }
   }

   ]
}
```

啟動執行個體 (RunInstances)

此RunInstancesAPI動作會啟動一或多個隨需執行個體或一或多個 Spot 執行個體。 `RunInstances`需要AMI並建立執行個體。使用者可以在請求中指定金鑰對和安全性群組。啟動到VPC需要子網路，並建立網路介面。從 Amazon EBS 支持啟動AMI創建一個卷。因此，使用者必須具有使用這些 Amazon EC2 資源的許可。您可以建立需要使用者在 `RunInstances` 上指定選用參數，或限制使用者使用特定參數值的政策陳述式。

如需啟動執行個體所需資源層級許可的詳細資訊，請參閱 Amazon 的動作、資源和條件金鑰。EC2

根據預設，使用者不具描述、開始、停止或終止所產生執行個體的許可。將管理所產生執行個體之許可授予使用者的一個方式，就是為每個執行個體建立特定標籤，再建立讓他們管理具有該標籤之執行個體的陳述式。如需詳細資訊，請參閱 使用執行個體。

資源

- AMIs
- 執行個體類型
- 子網
- EBS 磁碟區
- 標籤
- 啟動範本中的標籤
- 彈性 GPUs
- 啟動範本

AMIs

下列原則允許使用者僅使用指定AMIs的`ami-9e1670f7`和來啟動執行個體`ami-45cf5c3c`。用戶無法使用 other 啟動實例AMIs（除非另一個語句授予用戶這樣做的權限）。

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
    {
       "Effect": "Allow",
       "Action": "ec2:RunInstances",
       "Resource": [
         "arn:aws:ec2:region::image/ami-9e1670f7",
         "arn:aws:ec2:region::image/ami-45cf5c3c",
         "arn:aws:ec2:region:account-id:instance/*",
         "arn:aws:ec2:region:account-id:volume/*",
         "arn:aws:ec2:region:account-id:key-pair/*",
         "arn:aws:ec2:region:account-id:security-group/*",
         "arn:aws:ec2:region:account-id:subnet/*",
         "arn:aws:ec2:region:account-id:network-interface/*"
       ]
     }
    ]
}
```

或者，下列政策允許使用者從 Amazon AMIs 擁有的所有執行個體或特定受信任且經過驗證的合作夥伴啟動執行個體。第一個陳述式的 `Condition` 元素會測試 `ec2:Owner` 是否為 `amazon`。用戶無法使用 other 啟動實例AMIs（除非另一個語句授予用戶這樣做的權限）。

```
{
    "Version": "2012-10-17",
    "Statement": [
          {
       "Effect": "Allow",
       "Action": "ec2:RunInstances",
       "Resource": [
          "arn:aws:ec2:region::image/ami-*"
       ],
       "Condition": {
          "StringEquals": {
             "ec2:Owner": "amazon"
          }
       }
    },
    {
       "Effect": "Allow",
       "Action": "ec2:RunInstances",
       "Resource": [
          "arn:aws:ec2:region:account-id:instance/*",
```

```
            "arn:aws:ec2:region:account-id:subnet/*",
            "arn:aws:ec2:region:account-id:volume/*",
            "arn:aws:ec2:region:account-id:network-interface/*",
            "arn:aws:ec2:region:account-id:key-pair/*",
            "arn:aws:ec2:region:account-id:security-group/*"
            ]
        }
    ]
}
```

執行個體類型

下列政策只允許使用者使用 `t2.micro` 或 `t2.small` 執行個體類型啟動執行個體，您可能會為了控制成本而這樣做。因為第一個陳述式的 `Condition` 元素會測試 `ec2:InstanceType` 是否為 `t2.micro` 或 `t2.small`，所以使用者無法啟動更大的執行個體。

```
{
    "Version": "2012-10-17",
    "Statement": [
         {
        "Effect": "Allow",
        "Action": "ec2:RunInstances",
        "Resource": [
            "arn:aws:ec2:region:account-id:instance/*"
        ],
        "Condition": {
            "StringEquals": {
                "ec2:InstanceType": ["t2.micro", "t2.small"]
            }
        }
    },
    {
        "Effect": "Allow",
        "Action": "ec2:RunInstances",
        "Resource": [
            "arn:aws:ec2:region::image/ami-*",
            "arn:aws:ec2:region:account-id:subnet/*",
            "arn:aws:ec2:region:account-id:network-interface/*",
            "arn:aws:ec2:region:account-id:volume/*",
            "arn:aws:ec2:region:account-id:key-pair/*",
            "arn:aws:ec2:region:account-id:security-group/*"
            ]
        }
```

```
   ]
}
```

或者，您可以建立政策來拒絕使用者啟動 t2.micro 和 t2.small 執行個體類型以外之任何執行個體的許可。

```
{
   "Version": "2012-10-17",
   "Statement": [
        {
      "Effect": "Deny",
      "Action": "ec2:RunInstances",
      "Resource": [
         "arn:aws:ec2:region:account-id:instance/*"
      ],
      "Condition": {
         "StringNotEquals": {
            "ec2:InstanceType": ["t2.micro", "t2.small"]
         }
      }
   },
   {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": [
         "arn:aws:ec2:region::image/ami-*",
         "arn:aws:ec2:region:account-id:network-interface/*",
         "arn:aws:ec2:region:account-id:instance/*",
         "arn:aws:ec2:region:account-id:subnet/*",
         "arn:aws:ec2:region:account-id:volume/*",
         "arn:aws:ec2:region:account-id:key-pair/*",
         "arn:aws:ec2:region:account-id:security-group/*"
         ]
      }
   ]
}
```

子網

下列政策只允許使用者使用指定的子網 subnet-*12345678* 啟動執行個體。群組無法將執行個體啟動至任何其他子網 (除非其他陳述式授予使用者該許可)。

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
     {
       "Effect": "Allow",
       "Action": "ec2:RunInstances",
       "Resource": [
         "arn:aws:ec2:region:account-id:subnet/subnet-12345678",
         "arn:aws:ec2:region:account-id:network-interface/*",
         "arn:aws:ec2:region:account-id:instance/*",
         "arn:aws:ec2:region:account-id:volume/*",
         "arn:aws:ec2:region::image/ami-*",
         "arn:aws:ec2:region:account-id:key-pair/*",
         "arn:aws:ec2:region:account-id:security-group/*"
       ]
     }
    ]
}
```

或者，您可以建立拒絕使用者將執行個體啟動至任何其他子網之許可的政策。陳述式執行此作業的方式是拒絕建立網路介面的許可，但指定子網 subnet-*12345678* 的介面除外。此拒絕會覆寫為允許將執行個體啟動至其他子網所建立的任何其他政策。

```
{
    "Version": "2012-10-17",
    "Statement": [
          {
       "Effect": "Deny",
       "Action": "ec2:RunInstances",
       "Resource": [
          "arn:aws:ec2:region:account-id:network-interface/*"
       ],
       "Condition": {
          "ArnNotEquals": {
             "ec2:Subnet": "arn:aws:ec2:region:account-id:subnet/subnet-12345678"
          }
       }
    },
    {
       "Effect": "Allow",
       "Action": "ec2:RunInstances",
       "Resource": [
          "arn:aws:ec2:region::image/ami-*",
          "arn:aws:ec2:region:account-id:network-interface/*",
```

```
            "arn:aws:ec2:region:account-id:instance/*",
            "arn:aws:ec2:region:account-id:subnet/*",
            "arn:aws:ec2:region:account-id:volume/*",
            "arn:aws:ec2:region:account-id:key-pair/*",
            "arn:aws:ec2:region:account-id:security-group/*"
            ]
        }
    ]
}
```

EBS 磁碟區

只有在執行個體的EBS磁碟區已加密時，下列原則才允許使用者啟動執行個體。使用者必須從使用加密快照建立AMI的執行個體啟動執行個體，以確保根磁碟區已加密。使用者在啟動期間連接至執行個體的任何其他磁碟區也必須經過加密。

```
{
    "Version": "2012-10-17",
    "Statement": [
                {
            "Effect": "Allow",
            "Action": "ec2:RunInstances",
            "Resource": [
                "arn:aws:ec2:*:*:volume/*"
            ],
            "Condition": {
                "Bool": {
                    "ec2:Encrypted": "true"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": "ec2:RunInstances",
            "Resource": [
                "arn:aws:ec2:*::image/ami-*",
                "arn:aws:ec2:*:*:network-interface/*",
                "arn:aws:ec2:*:*:instance/*",
                "arn:aws:ec2:*:*:subnet/*",
                "arn:aws:ec2:*:*:key-pair/*",
                "arn:aws:ec2:*:*:security-group/*"
            ]
        }
```

```
    ]
}
```

標籤

建立時為執行個體加上標籤

下列政策允許使用者在建立期間啟動執行個體並為執行個體套用標籤。針對套用標籤的資源建立動作，使用者必須具有使用 `CreateTags` 動作的許可。第二個陳述式使用 `ec2:CreateAction` 條件鍵限制使用者在 `RunInstances` 的條件下才可建立標籤，且僅限為執行個體建立。使用者無法為現有資源套用標籤，也無法使用 `RunInstances` 請求為磁碟區套用標籤。

如需詳細資訊，請參閱 授予在建立期間標記 Amazon EC2 資源的權限。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
         "ec2:RunInstances"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
         "ec2:CreateTags"
      ],
      "Resource": "arn:aws:ec2:us-east-1:account-id:instance/*",
      "Condition": {
         "StringEquals": {
             "ec2:CreateAction" : "RunInstances"
          }
       }
    }
  ]
}
```

使用特定標籤建立時為執行個體和磁碟區加上標籤

下列政策包含 aws:RequestTag 條件鍵，需要使用者使用標籤 RunInstances 和 environment=production 為 purpose=webserver 建立的任何執行個體和磁碟區套用標籤。如果使用者未傳遞這些特定標籤，或完全未指定標籤，請求會失敗。

```
{
  "Version": "2012-10-17",
  "Statement": [
   {
      "Effect": "Allow",
      "Action": [
         "ec2:RunInstances"
      ],
      "Resource": [
         "arn:aws:ec2:region::image/*",
         "arn:aws:ec2:region:account-id:subnet/*",
         "arn:aws:ec2:region:account-id:network-interface/*",
         "arn:aws:ec2:region:account-id:security-group/*",
         "arn:aws:ec2:region:account-id:key-pair/*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
         "ec2:RunInstances"
      ],
      "Resource": [
          "arn:aws:ec2:region:account-id:volume/*",
          "arn:aws:ec2:region:account-id:instance/*"
      ],
      "Condition": {
         "StringEquals": {
             "aws:RequestTag/environment": "production" ,
             "aws:RequestTag/purpose": "webserver"
          }
       }
    },
    {
      "Effect": "Allow",
      "Action": [
         "ec2:CreateTags"
      ],
      "Resource": "arn:aws:ec2:region:account-id:*/*",
      "Condition": {
```

```
          "StringEquals": {
              "ec2:CreateAction" : "RunInstances"
           }
        }
    }
  ]
}
```

使用至少一個特定標籤建立時為執行個體和磁碟區加上標籤

下列政策在 ForAnyValue 條件的部分使用 aws:TagKeys 修飾詞，表示至少必須在請求中指定一個標籤，而且它必須包含 environment 鍵或 webserver 鍵。此標籤必須同時套用至執行個體和磁碟區。使用者可以在請求中指定任何標籤值。

```
{
  "Version": "2012-10-17",
  "Statement": [
   {
      "Effect": "Allow",
      "Action": [
         "ec2:RunInstances"
      ],
      "Resource": [
         "arn:aws:ec2:region::image/*",
         "arn:aws:ec2:region:account-id:subnet/*",
         "arn:aws:ec2:region:account-id:network-interface/*",
         "arn:aws:ec2:region:account-id:security-group/*",
         "arn:aws:ec2:region:account-id:key-pair/*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
          "ec2:RunInstances"
      ],
      "Resource": [
          "arn:aws:ec2:region:account-id:volume/*",
          "arn:aws:ec2:region:account-id:instance/*"
      ],
      "Condition": {
          "ForAnyValue:StringEquals": {
              "aws:TagKeys": ["environment","webserver"]
          }
```

```
        }
    },
    {
      "Effect": "Allow",
      "Action": [
          "ec2:CreateTags"
      ],
      "Resource": "arn:aws:ec2:region:account-id:*/*",
      "Condition": {
          "StringEquals": {
              "ec2:CreateAction" : "RunInstances"
          }
        }
    }
  ]
}
```

如果執行個體在建立時被標籤，則必須使用特定標籤來加以標籤

在下列政策中，使用者不需要在請求中指定標籤，但若需指定，標籤必須是 purpose=test。不允許其他標籤。使用者可以套用標籤至 RunInstances 請求中任何可套用標籤的資源。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
         "ec2:RunInstances"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
         "ec2:CreateTags"
      ],
      "Resource": "arn:aws:ec2:region:account-id:*/*",
      "Condition": {
         "StringEquals": {
             "aws:RequestTag/purpose": "test",
             "ec2:CreateAction" : "RunInstances"
          },
```

```
          "ForAllValues:StringEquals": {
              "aws:TagKeys": "purpose"
          }
       }
    }
  ]
}
```

禁止任何人在創建時調用標籤 RunInstances

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowRun",
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-1::image/*",
                "arn:aws:ec2:us-east-1:*:subnet/*",
                "arn:aws:ec2:us-east-1:*:network-interface/*",
                "arn:aws:ec2:us-east-1:*:security-group/*",
                "arn:aws:ec2:us-east-1:*:key-pair/*",
                "arn:aws:ec2:us-east-1:*:volume/*",
                "arn:aws:ec2:us-east-1:*:instance/*",
                "arn:aws:ec2:us-east-1:*:spot-instances-request/*"
            ]
        },
        {
            "Sid": "VisualEditor0",
            "Effect": "Deny",
            "Action": "ec2:CreateTags",
            "Resource": "*"
        }
    ]
}
```

僅允許的特定標籤 spot-instances-request。不一致的數字 2 會意外在此發揮作用。在一般情況下，不指定任何標籤會導致「未驗證」狀態。在的情況下 spot-instances-request，如果沒有標籤，則不會評估此原則，因此非標 spot-instances-request 籤 Spot 執行時要求將會成功。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowRun",
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-1::image/*",
                "arn:aws:ec2:us-east-1:*:subnet/*",
                "arn:aws:ec2:us-east-1:*:network-interface/*",
                "arn:aws:ec2:us-east-1:*:security-group/*",
                "arn:aws:ec2:us-east-1:*:key-pair/*",
                "arn:aws:ec2:us-east-1:*:volume/*",
                "arn:aws:ec2:us-east-1:*:instance/*",
            ]
        },
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": "ec2:RunInstances",
            "Resource": "arn:aws:ec2:us-east-1:*:spot-instances-request/*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/environment": "production"
                }
            }
        }
    ]
}
```

啟動範本中的標籤

在下列範例中，使用者可以啟動執行個體，但前提是他們要使用特定的啟動範本 (`lt-09477bcd97b0d310e`)。`ec2:IsLaunchTemplateResource` 條件鍵可防止使用者覆寫在啟動

範本中指定的任何資源。陳述式的第二部分允許使者在建立時為執行個體套用標籤 – 如果在啟動範本中指定執行個體的標籤，則需要陳述式的這個部分。

```
{
  "Version": "2012-10-17",
  "Statement": [
   {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": "*",
      "Condition": {
         "ArnLike": {
             "ec2:LaunchTemplate": "arn:aws:ec2:region:account-id:launch-template/
lt-09477bcd97b0d310e"
          },
          "Bool": {
             "ec2:IsLaunchTemplateResource": "true"
          }
       }
    },
    {
      "Effect": "Allow",
      "Action": [
         "ec2:CreateTags"
      ],
      "Resource": "arn:aws:ec2:region:account-id:instance/*",
      "Condition": {
         "StringEquals": {
             "ec2:CreateAction" : "RunInstances"
          }
       }
    }
  ]
}
```

彈性 GPUs

在下列原則中，使用者可以啟動執行個體，並指定GPU要連接至執行個體的彈性設定。使用者可以在任何區域啟動執行個體，但只能在區`us-east-2`域啟動GPU期間附加彈性執行個體。

`ec2:ElasticGpuType`條件索引鍵可確保執行個體使用`eg1.medium`或`eg1.large`彈性GPU類型。

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
             {
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances"
            ],
            "Resource": [
                "arn:aws:ec2:*:account-id:elastic-gpu/*"
            ],
            "Condition": {
                "StringEquals": {
                    "ec2:Region": "us-east-2",
                    "ec2:ElasticGpuType": [
                        "eg1.medium",
                        "eg1.large"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": "ec2:RunInstances",
            "Resource": [
                "arn:aws:ec2:*::image/ami-*",
                "arn:aws:ec2:*:account-id:network-interface/*",
                "arn:aws:ec2:*:account-id:instance/*",
                "arn:aws:ec2:*:account-id:subnet/*",
                "arn:aws:ec2:*:account-id:volume/*",
                "arn:aws:ec2:*:account-id:key-pair/*",
                "arn:aws:ec2:*:account-id:security-group/*"
            ]
        }
    ]
}
```

啟動範本

在下列範例中，使用者可以啟動執行個體，但前提是他們要使用特定的啟動範本 (`lt-09477bcd97b0d310e`)。使用者可以透過在 `RunInstances` 動作中指定參數，來覆寫啟動範本中的任何參數。

```
{
```

```
  "Version": "2012-10-17",
  "Statement": [
         {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": "*",
      "Condition": {
         "ArnLike": {
             "ec2:LaunchTemplate": "arn:aws:ec2:region:account-id:launch-template/
lt-09477bcd97b0d310e"
          }
       }
    }
  ]
}
```

在此範例中，使用者只有在使用啟動範本時才能啟動執行個體。原則會使用`ec2:IsLaunchTemplateResource`條件索引鍵來防止使用者覆寫啟動範本中任何預先存ARNs在的項目。

```
{
  "Version": "2012-10-17",
  "Statement": [
         {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": "*",
      "Condition": {
         "ArnLike": {
             "ec2:LaunchTemplate": "arn:aws:ec2:region:account-id:launch-template/*"
          },
          "Bool": {
             "ec2:IsLaunchTemplateResource": "true"
          }
       }
    }
  ]
}
```

下列政策範例允許使用者啟動執行個體，但前提是他們要使用啟動範本。使用者無法覆寫請求中的子網和網路介面參數；這些參數只能在啟動範本中予以指定。陳述式的第一部分會使用NotResource元素來

允許除子網路和網路介面以外的所有其他資源。陳述式的第二部分允許子網和網路介面資源，但前提是它們來自啟動範本。

```
{
  "Version": "2012-10-17",
  "Statement": [
        {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "NotResource": ["arn:aws:ec2:region:account-id:subnet/*",
                      "arn:aws:ec2:region:account-id:network-interface/*" ],
      "Condition": {
         "ArnLike": {
             "ec2:LaunchTemplate": "arn:aws:ec2:region:account-id:launch-template/*"
          }
       }
    },
   {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": ["arn:aws:ec2:region:account-id:subnet/*",
                   "arn:aws:ec2:region:account-id:network-interface/*" ],
      "Condition": {
         "ArnLike": {
             "ec2:LaunchTemplate": "arn:aws:ec2:region:account-id:launch-template/*"
          },
          "Bool": {
             "ec2:IsLaunchTemplateResource": "true"
          }
       }
    }
  ]
}
```

下列範例只允許使用者在使用啟動範本且該啟動範本具有標籤 Purpose=Webservers 時，啟動執行個體。使用者無法覆寫 RunInstances 動作中的任何啟動範本參數。

```
{
  "Version": "2012-10-17",
  "Statement": [
        {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
```

```
      "NotResource": "arn:aws:ec2:region:account-id:launch-template/*",
      "Condition": {
         "ArnLike": {
             "ec2:LaunchTemplate": "arn:aws:ec2:region:account-id:launch-template/*"
          },
         "Bool": {
             "ec2:IsLaunchTemplateResource": "true"
          }
       }
    },
    {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": "arn:aws:ec2:region:account-id:launch-template/*",
      "Condition": {
       "StringEquals": {
           "aws:ResourceTag/Purpose": "Webservers"
        }
       }
     }
  ]
}
```

使用 競價型執行個體

您可以使用此 RunInstances 動作建立競價型執行個體請求，並在建立時標記競價型執行個體請求。要為其指定的資源 RunInstances 為`spot-instances-request`。

在IAM策略中評估`spot-instances-request`資源，如下所示：

- 如果您在建立時未標記競價型執行個體請求，Amazon EC2 不會評估 RunInstances陳述式中的`spot-instances-request`資源。
- 如果您在建立時標記競價型執行個體請求，Amazon EC2 會評估 RunInstances陳述式中的`spot-instances-request`資源。

因此，對於`spot-instances-request`資源，下列規則會套用至IAM策略：

- 如果您使 RunInstances 用建立競價型執行個體請求，且不打算在建立時標記競價型執行個體請求，則不需要明確允許`spot-instances-request`資源；呼叫將會成功。
- 如果您使 RunInstances 用建立競價型執行個體請求並打算在建立時標記競價型執行個體請求，則必須在 RunInstances allow 陳述式中包含`spot-instances-request`資源，否則呼叫將會失敗。

- 如果您使用 RunInstances 建立競價型執行個體請求並打算在建立時標記競價型執行個體請求，則必須在 CreateTags allow 陳述式中指定`spot-instances-request`資源或`*`萬用字元，否則呼叫將會失敗。

您可以使用 RunInstances 或請求 Spot 執行個體 RequestSpotInstances。下列範例IAM政策僅適用於請求 Spot 執行個體使用 RunInstances。

範例：請求 Spot 執行個體 RunInstances

下列原則允許使用者使用動作來請求 Spot 執 RunInstances 行個體。由 RunInstances建立的`spot-instances-request`資源請求 Spot 執行個體。

> **Note**
>
> 若要用 RunInstances 來建立競價型執行個體請求，如果您不打算在建立時標記 Spot 執行個體請求，則可以從`Resource`清單`spot-instances-request`中省略。這是因為如果競價型執行個體請求在建立時未標記，Amazon 不EC2會評估 RunInstances陳述式中的`spot-instances-request`資源。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowRun",
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-1::image/*",
                "arn:aws:ec2:us-east-1:*:subnet/*",
                "arn:aws:ec2:us-east-1:*:network-interface/*",
                "arn:aws:ec2:us-east-1:*:security-group/*",
                "arn:aws:ec2:us-east-1:*:key-pair/*",
                "arn:aws:ec2:us-east-1:*:volume/*",
                "arn:aws:ec2:us-east-1:*:instance/*",
                "arn:aws:ec2:us-east-1:*:spot-instances-request/*"
            ]
        }
```

```
    ]
}
```

> **Warning**
>
> NOTSUPPORTED— 範例：拒絕使用者請求 Spot 執行個體使用的權限 RunInstances `spot-instances-request` 資源不支援下列政策。
> 下列政策旨在提供使用者啟動 隨需執行個體 的許可，但拒絕使用者請求 競價型執行個體 的許可。由 RunInstances建立的`spot-instances-request`資源是請求 Spot 執行個體的資源。第二個陳述式是為了拒絕`spot-instances-request`資源的 RunInstances 動作。但是，不支援此條件，因為如果競價型執行個體請求在建立時未標記，Amazon 不EC2會評估 RunInstances 陳述式中的`spot-instances-request`資源。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowRun",
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-1::image/*",
                "arn:aws:ec2:us-east-1:*:subnet/*",
                "arn:aws:ec2:us-east-1:*:network-interface/*",
                "arn:aws:ec2:us-east-1:*:security-group/*",
                "arn:aws:ec2:us-east-1:*:key-pair/*",
                "arn:aws:ec2:us-east-1:*:volume/*",
                "arn:aws:ec2:us-east-1:*:instance/*"
            ]
        },
        {
            "Sid": "DenySpotInstancesRequests - NOT SUPPORTED - DO NOT USE!",
            "Effect": "Deny",
            "Action": "ec2:RunInstances",
            "Resource": "arn:aws:ec2:us-east-1:*:spot-instances-request/*"
        }
    ]
}
```

範例：在建立時標記 Spot 執行個體請求

下列政策可讓使用者標記執行個體啟動期間建立的所有資源。第一個語句允 RunInstances 許創建列出的資源。由 RunInstances建立的`spot-instances-request`資源是請求 Spot 執行個體的資源。第二個陳述式提供 * 萬用字元，允許在執行個體啟動時建立所有資源時加上標記。

> **Note**
>
> 如果您在建立時標記競價型執行個體請求，Amazon EC2 會評估 RunInstances 陳述式中的`spot-instances-request`資源。因此，您必須明確允許 RunInstances 動作的`spot-instances-request`資源，否則呼叫將失敗。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowRun",
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-1::image/*",
                "arn:aws:ec2:us-east-1:*:subnet/*",
                "arn:aws:ec2:us-east-1:*:network-interface/*",
                "arn:aws:ec2:us-east-1:*:security-group/*",
                "arn:aws:ec2:us-east-1:*:key-pair/*",
                "arn:aws:ec2:us-east-1:*:volume/*",
                "arn:aws:ec2:us-east-1:*:instance/*",
                "arn:aws:ec2:us-east-1:*:spot-instances-request/*"
            ]
        },
        {
            "Sid": "TagResources",
            "Effect": "Allow",
            "Action": "ec2:CreateTags",
            "Resource": "*"
        }
    ]
}
```

範例：拒絕在建立時標記 Spot 執行個體請求

下列政策拒絕使用者標記執行個體啟動期間建立的資源的許可。

第一個語句允 RunInstances 許創建列出的資源。由 RunInstances建立的`spot-instances-request`資源是請求 Spot 執行個體的資源。第二個陳述式會提供 * 萬用字元，以拒絕在執行個體啟動時建立所有標記的資源。如果`spot-instances-request`或任何其他資源在創建時被標記，則 RunInstances 調用將失敗。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowRun",
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-1::image/*",
                "arn:aws:ec2:us-east-1:*:subnet/*",
                "arn:aws:ec2:us-east-1:*:network-interface/*",
                "arn:aws:ec2:us-east-1:*:security-group/*",
                "arn:aws:ec2:us-east-1:*:key-pair/*",
                "arn:aws:ec2:us-east-1:*:volume/*",
                "arn:aws:ec2:us-east-1:*:instance/*",
                "arn:aws:ec2:us-east-1:*:spot-instances-request/*"
            ]
        },
        {
            "Sid": "DenyTagResources",
            "Effect": "Deny",
            "Action": "ec2:CreateTags",
            "Resource": "*"
        }
    ]
}
```

> **Warning**
>
> NOTSUPPORTED— 範例：只有在已指派特定標籤時，才允許建立 Spot 執行個體請求 `spot-instances-request` 資源不支援下列政策。

以下政策旨在僅在請求使 RunInstances 用特定標籤標記時，才授予建立競價型執行個體請求的權限。
第一個語句允 RunInstances 許創建列出的資源。
第二個陳述式向使用者授予許可，以便僅在請求具有 `environment=production` 標籤時才建立 Spot 執行個體請求。如果此條件套用至由建立的其他資源 RunInstances，則不指定任何標籤會導致錯`Unauthenticated`誤。但是，如果沒有為競價型執行個體請求指定標籤，Amazon EC2 不會評估 RunInstances 陳述式中的`spot-instances-request`資源，這會導致由 RunInstances建立未標記的 Spot 執行個體請求。
請注意，指定其他標籤會`environment=production`導致`Unauthenticated`錯誤，因為如果使用者標記競價型執行個體請求，Amazon EC2 會評估 RunInstances 陳述式中的`spot-instances-request`資源。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowRun",
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-1::image/*",
                "arn:aws:ec2:us-east-1:*:subnet/*",
                "arn:aws:ec2:us-east-1:*:network-interface/*",
                "arn:aws:ec2:us-east-1:*:security-group/*",
                "arn:aws:ec2:us-east-1:*:key-pair/*",
                "arn:aws:ec2:us-east-1:*:volume/*",
                "arn:aws:ec2:us-east-1:*:instance/*"
            ]
        },
        {
            "Sid": "RequestSpotInstancesOnlyIfTagIs_environment=production - NOT
 SUPPORTED - DO NOT USE!",
            "Effect": "Allow",
            "Action": "ec2:RunInstances",
            "Resource": "arn:aws:ec2:us-east-1:*:spot-instances-request/*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/environment": "production"
                }
```

```
            }
        },
        {
            "Sid": "TagResources",
            "Effect": "Allow",
            "Action": "ec2:CreateTags",
            "Resource": "*"
        }

    ]
}
```

範例：如果系統指派特定標籤，則拒絕建立 Spot 執行個體請求

如果請求標記為，下列政策會拒絕 RunInstances 建立競價型執行個體請求的權限。`environment=production`

第一個語句允 RunInstances 許創建列出的資源。

如果請求具有 `environment=production` 標籤，第二個陳述式會拒絕使用者建立 Spot 執行個體請求的許可。指定 `environment=production` 為標籤會導致 `Unauthenticated` 錯誤。指定其他標籤或不指定標籤將導致建立 Spot 執行個體請求。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowRun",
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-1::image/*",
                "arn:aws:ec2:us-east-1:*:subnet/*",
                "arn:aws:ec2:us-east-1:*:network-interface/*",
                "arn:aws:ec2:us-east-1:*:security-group/*",
                "arn:aws:ec2:us-east-1:*:key-pair/*",
                "arn:aws:ec2:us-east-1:*:volume/*",
                "arn:aws:ec2:us-east-1:*:instance/*",
                "arn:aws:ec2:us-east-1:*:spot-instances-request/*"
```

```
            ]
        },
        {
            "Sid": "DenySpotInstancesRequests",
            "Effect": "Deny",
            "Action": "ec2:RunInstances",
            "Resource": "arn:aws:ec2:us-east-1:*:spot-instances-request/*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/environment": "production"
                }
            }
        },
        {
            "Sid": "TagResources",
            "Effect": "Allow",
            "Action": "ec2:CreateTags",
            "Resource": "*"
        }
    ]
}
```

範例：使用 預留執行個體

下列政策可提供使用者檢視、修改及購買您帳戶中 預留執行個體 的許可。

您無法針對個別的 預留執行個體 設定資源層級的權限。此政策表示使用者可存取帳戶中的所有 預留執行個體。

`Resource` 元素使用 * 萬用字元，表示使用者可以為此動作指定所有資源；在本例中，他們可以列出及修改帳戶中的所有預留執行個體。他們也可以使用帳戶登入資料購買 預留執行個體。在API動作不支援資源層級權限的情況下，* 萬用字元也是必要的。

```
{
   "Version": "2012-10-17",
   "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeReservedInstances",
        "ec2:ModifyReservedInstances",
        "ec2:PurchaseReservedInstancesOffering",
        "ec2:DescribeAvailabilityZones",
```

```
        "ec2:DescribeReservedInstancesOfferings"
      ],
      "Resource": "*"
    }
   ]
}
```

用以允許使用者檢視及修改您帳戶中的預留執行個體，但不允許購買新的預留執行個體。

```
{
  "Version": "2012-10-17",
   "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeReservedInstances",
        "ec2:ModifyReservedInstances",
        "ec2:DescribeAvailabilityZones"
      ],
      "Resource": "*"
    }
  ]
}
```

範例：標籤資源

下列政策只允許使用者在標籤包含 `CreateTags` 鍵和 `environment` 值時，使用 `production` 動作將標籤套用至執行個體。不允許其他標籤，並且使用者無法將標籤用於任何其他資源類型。

```
{
    "Version": "2012-10-17",
    "Statement": [
              {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:region:account-id:instance/*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/environment": "production"
```

```
                }
            }
        }
    ]
}
```

下列政策允許使用者套用標籤至任何已有 owner 金鑰和使用者名稱值之標籤的可套用標籤資源。此外，使用者也必須在請求中使用 anycompany:environment-type 鍵和 test 值或 prod 值指定標籤。使用者可以在請求中指定其他標籤。

```
{
    "Version": "2012-10-17",
    "Statement": [
                {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:region:account-id:*/*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/anycompany:environment-type": ["test","prod"],
                    "aws:ResourceTag/owner": "${aws:username}"
                }
            }
        }
    ]
}
```

您可以建立允許使用者刪除資源的特定標籤的IAM策略。例如，下列政策允許使用者在請求中指定的標籤鍵為 environment 或 cost-center 時，刪除磁碟區的標籤。您可以為標籤指定任何值，但標籤鍵必須符合指定的任一鍵。

> **Note**
>
> 如果您刪除資源，也會刪除與該資源相關聯的所有標籤。使用者不需要使用 ec2:DeleteTags 動作的許可，也能刪除具有標籤的資源；他們只需要執行刪除動作的許可。

```
{
```

```
  "Version": "2012-10-17",
  "Statement": [
       {
      "Effect": "Allow",
      "Action": "ec2:DeleteTags",
      "Resource": "arn:aws:ec2:us-east-1:account-id:volume/*",
      "Condition": {
        "ForAllValues:StringEquals": {
          "aws:TagKeys": ["environment","cost-center"]
        }
      }
    }
  ]
}
```

此政策只允許使用者刪除任何資源上的 environment=prod 標籤，而且僅限資源已套用 owner 金鑰和使用者名稱值之標籤的情況。使用者無法刪除資源的任何其他標籤。

```
{
  "Version": "2012-10-17",
  "Statement": [
      {
      "Effect": "Allow",
      "Action": [
        "ec2:DeleteTags"
      ],
      "Resource": "arn:aws:ec2:region:account-id:*/*",
      "Condition": {
        "StringEquals": {
          "aws:RequestTag/environment": "prod",
          "aws:ResourceTag/owner": "${aws:username}"
        },
        "ForAllValues:StringEquals": {
          "aws:TagKeys": ["environment"]
        }
      }
    }
  ]
}
```

範例：使用 IAM 角色

下列原則可讓使用者將IAM角色附加、取代和卸離至具有標記的執行個體`department=test`。取代或卸離IAM角色需要關聯 ID，因此政策也會授與使用者使用該`ec2:DescribeIamInstanceProfileAssociations`動作的權限。

使用者必須具有使用 `iam:PassRole` 動作的許可，才能將角色傳遞給執行個體。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:AssociateIamInstanceProfile",
        "ec2:ReplaceIamInstanceProfileAssociation",
        "ec2:DisassociateIamInstanceProfile"
      ],
      "Resource": "arn:aws:ec2:us-east-1:account-id:instance/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/department":"test"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": "ec2:DescribeIamInstanceProfileAssociations",
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": "iam:PassRole",
      "Resource": "arn:aws:iam::account-id:role/DevTeam*"
    }
  ]
}
```

下列原則可讓使用者附加或取代任何執行個體的IAM角色。使用者只能附加或取代名稱開頭為的IAM角色`TestRole-`。針對`iam:PassRole`動作，請確定您指定IAM角色的名稱，而非執行個體設定檔 (如果名稱不同)。如需詳細資訊，請參閱 執行個體描述檔。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AssociateIamInstanceProfile",
                "ec2:ReplaceIamInstanceProfileAssociation"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "ec2:DescribeIamInstanceProfileAssociations",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::account-id:role/TestRole-*"
        }
    ]
}
```

範例：使用路由表

下列原則可讓使用者針對VPCvpc-ec43eb89僅與相關聯的路由表新增、移除及取代路由。若要VPC為ec2:Vpc條件索引鍵指定一個，您必須指定完整ARN的VPC。

```
{
    "Version": "2012-10-17",
    "Statement": [
              {
            "Effect": "Allow",
            "Action": [
                "ec2:DeleteRoute",
                "ec2:CreateRoute",
                "ec2:ReplaceRoute"
            ],
            "Resource": [
                "arn:aws:ec2:region:account-id:route-table/*"
            ],
            "Condition": {
```

```
                "StringEquals": {
                    "ec2:Vpc": "arn:aws:ec2:region:account-id:vpc/vpc-ec43eb89"
                }
            }
        }
    ]
}
```

範例：允許特定執行個體檢視其他 AWS 服務中的資源

以下是您可能附加至IAM角色的原則範例。此原則可讓執行個體檢視各種 AWS 服務中的資源。它會使用`ec2:SourceInstanceARN`全域條件索引鍵來指定從中發出要求的執行個體必須是執行個體`i-093452212644b0dd6`。如果相同IAM角色與另一個實例相關聯，則另一個實例將無法執行這些動作中的任何一個。

```
{
    "Version": "2012-10-17",
    "Statement": [
              {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeVolumes",
                "s3:ListAllMyBuckets",
                "dynamodb:ListTables",
                "rds:DescribeDBInstances"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "ArnEquals": {
                    "ec2:SourceInstanceARN": "arn:aws:ec2:region:account-id:instance/
i-093452212644b0dd6"
                }
            }
        }
    ]
}
```

範例：使用啟動範本

下列政策允許使用者建立啟動範本版本並修改啟動範本，但僅限特定啟動範本 (`lt-09477bcd97b0d3abc`)。使用者無法使用其他啟動範本。

```
{
  "Version": "2012-10-17",
  "Statement": [
   {
      "Action": [
        "ec2:CreateLaunchTemplateVersion",
        "ec2:ModifyLaunchTemplate"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:ec2:region:account-id:launch-template/lt-09477bcd97b0d3abc"
    }
  ]
}
```

下列政策允許使用者刪除任何啟動範本和啟動範本版本，但前提是啟動範本具有標籤 `Purpose=Testing`。

```
{
  "Version": "2012-10-17",
  "Statement": [
       {
      "Action": [
        "ec2:DeleteLaunchTemplate",
        "ec2:DeleteLaunchTemplateVersions"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:ec2:region:account-id:launch-template/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/Purpose": "Testing"
        }
      }
    }
  ]
}
```

使用執行個體中繼資料

下列原則可確保使用者只能使用執行個體中繼資料服務版本 2 (IMDSv2) 擷取執行個體中繼資料。您可以將以下四個政策合併為一個包含四個陳述式的政策。當組合為一個策略時，您可以使用該策略作為服務控制策略 (SCP)。它同樣適用於您套用至現有原IAM則 (取消並限制現有權限) 的拒絕原則，或做為跨帳戶、組織單位 (OU) 或整個組織全域套用的拒絕原則。SCP

> **Note**
>
> 下列中 RunInstances 繼資料選項原則必須與授予主體權限的原則搭配使用，以啟動執行個體 RunInstances。如果主參與者沒有 RunInstances 權限，就無法啟動執行個體。如需詳細資訊，請參閱使用執行個體和啟動執行個體 (RunInstances)中的政策。

> **Important**
>
> 如果您使用 Auto Scaling 群組，且需要IMDSv2在所有新執行個體上使用，則 Auto Scaling 群組必須使用啟動範本。
> 當 Auto Scaling 群組使用啟動範本時，會在建立新的 Auto Scaling 群組時檢查IAM主參與者的`ec2:RunInstances`權限。當現有的 Auto Scaling 群組更新為使用新的啟動範本或新版本的啟動範本時，也會檢查這些許可。
> 只`RunInstances`有在建立或更新使用啟動範本的 Auto Scaling 群組時，才會檢查IMDSv1在IAM主參與者上使用的限制。對於設為使用 `Latest` or `Default` 啟動範本的 Auto Scaling 群組，建立新版本的啟動範本時，不會檢查許可。對於要檢查的許可，您必須將 Auto Scaling 群組設為使用特定版本的啟動範本。
>
> 若要在 Auto Scaling 群組啟動的執行個體IMDSv2上強制使用，則需要執行下列其他步驟：
>
> 1. 針對所建立的新主參與者使用服務控制原則 (SCPs) 或IAM權限界限，針對組織中的所有帳戶停用啟動組態。對於具有 Auto Scaling 群組權限的現有IAM主參與者，請使用此條件金鑰更新其相關聯的原則。若要停用啟動設定的使用，請使用指定的值為的`"autoscaling:LaunchConfigurationName"`條件索引鍵來建立或修改相關SCP、權限界限或IAM原則`null`。
> 2. 對於新的啟動範本，請在啟動範本中設定執行個體中繼資料選項。對於現有的啟動範本，請建立新版本的啟動範本，並在新版本中設定執行個體中繼資料選項。
> 3. 在授予任何委託人許可使用啟動範本的政策中，藉由指定 `"autoscaling:LaunchTemplateVersionSpecified": "true"` 來限制 `$latest` 和 `$default` 的關聯。藉由限制使用特定版本的啟動範本，可以確保使

用已設定執行個體中繼資料選項的版本來啟動新執行個體。如需詳細資訊，請參閱 LaunchTemplateSpecificationAmazon EC2 Auto Scaling 參API考中的特別`Version`參數。

4. 對於使用啟動組態的 Auto Scaling 群組，將啟動組態取代為啟動範本。如需詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南中的使用啟動範本取代啟動組態。
5. 對於使用啟動範本的 Auto Scaling 群組，請確保其將新啟動範本與設定的執行個體中繼資料選項搭配使用，或將新版本的目前啟動範本與設定的執行個體中繼資料選項搭配使用。若要取得更多資訊，請參閱《指AWS CLI 令參考》update-auto-scaling-group 中的。

範例

- 需要使用 IMDSv2
- 拒絕選擇退出 IMDSv2
- 指定最大跳轉限制
- 限制可修改執行個體中繼資料選項的人員
- 需要從中擷取角色認證 IMDSv2

需要使用 IMDSv2

下列原則指定您無法呼叫， RunInstances API除非執行個體也選擇加入要求使用 IMDSv2 (以表示`"ec2:MetadataHttpTokens": "required"`)。如果您未指定執行個`UnauthorizedOperation`體需要IMDSv2，則在呼叫 RunInstances API.

```
{
    "Version": "2012-10-17",
    "Statement": [
                {
            "Sid": "RequireImdsV2",
            "Effect": "Deny",
            "Action": "ec2:RunInstances",
            "Resource": "arn:aws:ec2:*:*:instance/*",
            "Condition": {
                "StringNotEquals": {
                    "ec2:MetadataHttpTokens": "required"
                }
            }
        }
    ]
}
```

拒絕選擇退出 IMDSv2

下列原則指定您無法呼叫`ModifyInstanceMetadataOptionsAPI`並允許IMDSv1或的選項IMDSv2。如果您呼叫 `ModifyInstanceMetadataOptionsAPI`，則必須將`HttpTokens`屬性設定為`required`。

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Sid": "DenyIMDSv1HttpTokensModification",
        "Effect": "Deny",
        "Action": "ec2:ModifyInstanceMetadataOptions",
        "Resource": "arn:aws:ec2:*:*:instance/*",
        "Condition": {
            "StringNotEquals": {
                "ec2:Attribute/HttpTokens": "required"
            },
            "Null": {
                "ec2:Attribute/HttpTokens": false
            }
        }
    }]
}
```

指定最大跳轉限制

下列原則指定 RunInstances API除非您同時指定躍點限制，否則您無法呼叫，且躍點限制不能超過3。如果你沒有這樣做，你會得到一個`UnauthorizedOperation`錯誤，當你調用 RunInstances API.

```
{
    "Version": "2012-10-17",
    "Statement": [
               {
            "Sid": "MaxImdsHopLimit",
            "Effect": "Deny",
            "Action": "ec2:RunInstances",
            "Resource": "arn:aws:ec2:*:*:instance/*",
            "Condition": {
                "NumericGreaterThan": {
                    "ec2:MetadataHttpPutResponseHopLimit": "3"
                }
            }
        }
```

```
    ]
}
```

限制可修改執行個體中繼資料選項的人員

下列政策只允許具有角色 `ec2-imds-admins` 的使用者對執行個體中繼資料選項進行變更。如果`ec2-imds-admins`角色以外的任何主體嘗試呼叫 ModifyInstanceMetadataOptions API，就會收到錯`UnauthorizedOperation`誤。這個陳述式可用來控制使用 ModifyInstanceMetadataOptions API；目前沒有針對 ModifyInstanceMetadataOptions API.

```
{
    "Version": "2012-10-17",
    "Statement": [
               {
            "Sid": "AllowOnlyImdsAdminsToModifySettings",
            "Effect": "Deny",
            "Action": "ec2:ModifyInstanceMetadataOptions",
            "Resource": "*",
            "Condition": {
                "StringNotLike": {
                    "aws:PrincipalARN": "arn:aws:iam::*:role/ec2-imds-admins"
                }
            }
        }
    ]
}
```

需要從中擷取角色認證 IMDSv2

下列原則指定如果將此原則套用至角色，且該角色由EC2服務承擔，且產生的認證是用來簽署要求，則必須由從中擷取的EC2角色認證來簽署要求IMDSv2。否則，其所有API呼叫都會`UnauthorizedOperation`出現錯誤。通常可以套用此陳述式/原則，因為如果要求未由EC2角色認證簽署，則沒有任何作用。

```
{
    "Version": "2012-10-17",
    "Statement": [
               {
            "Sid": "RequireAllEc2RolesToUseV2",
            "Effect": "Deny",
            "Action": "*",
            "Resource": "*",
```

```
            "Condition": {
                "NumericLessThan": {
                    "ec2:RoleDelivery": "2.0"
                }
            }
        }
    ]
}
```

使用 Amazon EBS 磁碟區和快照

如需使用 Amazon EBS 磁碟區和快照的政策範例，請參閱 Amazon 以身分識別為基礎的政策範例。EBS

控制 Amazon EC2 主控台存取權的範例政策

您可以使用IAM政策授予使用者使用 Amazon 所需的許可EC2。如需 step-by-step 指示，請參閱IAM使用指南中的建立IAM策略。

主控台會針對其功能使用其他API動作，因此這些原則可能無法如預期般運作。例如，在嘗試在主控台中檢視磁碟區時，具有僅使用`DescribeVolumes`API動作權限的使用者將會遇到錯誤。本節中所示範的政策，可讓使用者使用主控台的特定部分。如需為 Amazon EC2 主控台建立政策的其他資訊，請參閱下列 AWS 安全部落格文章：授與使用者在 Amazon EC2 主控台中工作的權限。

下列範例顯示可用來授與使用者使用 Amazon 權限的政策陳述式EC2。替換每個 *`user input placeholder`* 使用您自己的信息。這些原則是針對使用 AWS Management Console. Amazon 主EC2控台可能會呼叫多個API動作來顯示單一資源，而且在使用者嘗試執行任務且主控台顯示錯誤之前可能並不明顯。如需詳細資訊，請參閱下列 AWS 安全部落格文章：授與使用者在 Amazon EC2 主控台中工作的權限。

範例

- 範例：唯讀存取
- 範例：使用EC2啟動執行個體精靈
- 範例：使用安全群組
- 範例：使用彈性 IP 地址
- 範例：使用 預留執行個體

為了協助您確定在主控台中執行工作所需的API動作，您可以使用記錄呼叫的服務，例如 AWS CloudTrail。如果政策並未授予權限來建立或修改特定的資源，主控台會顯示編碼的訊息，其中包含診

斷資訊。您可以使用的DecodeAuthorizationMessageAPI動作或中的 AWS STSdecode-authorization-message指令來解碼郵件 AWS CLI。

範例：唯讀存取

若要允許使用者檢視 Amazon EC2 主控台中的所有資源，您可以使用與下列範例相同的政策：範例：唯讀存取除非有另一個陳述式授予使用者此種權限，否則使用者無法對其他資源執行任何動作，也無法建立新資源。

檢視執行個體AMIs、和快照

或者，您可以提供對資源子集的唯讀存取權限。若要這麼做，請將`ec2:Describe`API動作中的 * 萬用字元取代為每個資源的特定`ec2:Describe`動作。以下政策允許使用者在 Amazon EC2 主控台中AMIs檢視所有執行個體和快照。該`ec2:DescribeTags`操作允許用戶查看公共AMIs。主控台需要標記資訊才能顯示為公開AMIs；不過，您可以移除此動作以允許使用者只檢視私人資訊AMIs。

```
{
   "Version": "2012-10-17",
   "Statement": [{
      "Effect": "Allow",
      "Action": [
         "ec2:DescribeInstances",
         "ec2:DescribeImages",
         "ec2:DescribeTags",
         "ec2:DescribeSnapshots"
      ],
      "Resource": "*"
   }
   ]
}
```

> **Note**
>
> Amazon 動EC2`ec2:Describe*`API作不支援資源層級許可，因此您無法控制使用者可以在主控台中檢視哪些個別資源。因此，在上述的陳述式中，`Resource` 元素必須包含 * 萬用字元。如需有關ARNs可用於哪些 Amazon EC2 API 動作的詳細資訊，請參閱 Amazon 的動作、資源和條件金鑰EC2。

檢視執行個體和 CloudWatch 指標

下列政策允許使用者在 Amazon EC2 主控台中檢視執行個體，以及在「執行個體」頁面的「監控」索引標籤中檢視 CloudWatch警示和指標。Amazon 主 EC2控台會使 CloudWatch API用顯示警示和指標，因此您必須授與使用者使用`cloudwatch:DescribeAlarms`、`cloudwatch:DescribeAlarmsForMetric``cloudwatch:ListMet`和`cloudwatch:GetMetricData`動作的權限。

```
{
   "Version": "2012-10-17",
   "Statement": [{
      "Effect": "Allow",
      "Action": [
         "ec2:DescribeInstances",
         "ec2:DescribeInstanceTypes",
         "cloudwatch:DescribeAlarms",
         "cloudwatch:DescribeAlarmsForMetric",
         "cloudwatch:ListMetrics",
         "cloudwatch:GetMetricStatistics",
         "cloudwatch:GetMetricData"
      ],
      "Resource": "*"
   }
   ]
}
```

範例：使用EC2啟動執行個體精靈

Amazon EC2 啟動執行個體精靈是一個畫面，其中包含設定和啟動執行個體的選項。您的原則必須包含使用允許使用者使用精靈選項之API動作的權限。如果政策未包含使用這些動作的權限，可能無法適當地載入精靈中的某些項目，使用者就無法完成啟動。

基本啟動執行個體精靈存取權

若要成功完成啟動，使用者必須獲得使用`ec2:RunInstances`API動作的權限，以及至少下列API動作：

- `ec2:DescribeImages`：檢視並選取一個AMI。
- `ec2:DescribeInstanceTypes`：檢視和選取執行個體類型。
- `ec2:DescribeVpcs`：檢視可用的網路選項。
- `ec2:DescribeSubnets`：檢視所選VPC項目的所有可用子網路。

- `ec2:DescribeSecurityGroups` 或 `ec2:CreateSecurityGroup`：檢視和選取現有的安全群組，或建立新的安全群組。
- `ec2:DescribeKeyPairs` 或 `ec2:CreateKeyPair`：選取現有的金鑰對，或建立新的金鑰對。
- `ec2:AuthorizeSecurityGroupIngress`：新增傳入規則。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeInstances",
                "ec2:DescribeImages",
                "ec2:DescribeInstanceTypes",
                "ec2:DescribeKeyPairs",
                "ec2:DescribeVpcs",
                "ec2:DescribeSubnets",
                "ec2:DescribeSecurityGroups",
                "ec2:CreateSecurityGroup",
                "ec2:AuthorizeSecurityGroupIngress",
                "ec2:CreateKeyPair"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "ec2:RunInstances",
            "Resource": "*"
        }
    ]
}
```

您可以將API動作新增至原則，為使用者提供更多選項，例如：

- `ec2:DescribeAvailabilityZones`：檢視並選取特定的可用區域。
- `ec2:DescribeNetworkInterfaces`：針對選取的子網，檢視並選取現有的網路介面。
- 若要將輸出規則新增至VPC安全性群組，使用者必須獲得使用此`ec2:AuthorizeSecurityGroupEgressAPI`動作的權限。若要修改或刪除現有規則，使用者必須獲得使用相關`ec2:RevokeSecurityGroup*`API動作的權限。

- `ec2:CreateTags`：標記由 `RunInstances` 所建立的資源。如需詳細資訊，請參閱 授予在建立期間標記 Amazon EC2 資源的權限。如果使用者不具有使用此動作的許可，而且試著在 啟動執行個體精靈的標記頁面上使用標籤，則啟動會失敗。

> **Important**
>
> 啟動執行個體時指定 Name (名稱) 可建立標籤，並且需要 `ec2:CreateTags` 動作。請小心授與使用者使用 `ec2:CreateTags` 動作的許可，因為這樣做會限制您使用 `aws:ResourceTag` 條件金鑰來限制使用者使用其他資源的能力。如果您授與使用者使用 `ec2:CreateTags` 動作的許可，他們可以變更資源的標籤，以略過這些限制。如需詳細資訊，請參閱 使用以屬性為基礎的存取控制存取。

- 若要在選取時使用「Systems Manager」參數AMI，您必須將`ssm:DescribeParameters`和新增`ssm:GetParameters`至原則。 `ssm:DescribeParameters`授予您的使用者檢視和選取 Systems Manager 參數的權限。 `ssm:GetParameters`授予您的使用者取得「Systems Manager」參數值的權限。您也可以限制對特定 Systems Manager 參數的存取。如需詳細資訊，請參閱本節稍後的 Restrict access to specific Systems Manager parameters (限制特定 Systems Manager 參數的存取)。

目前 Amazon EC2 `Describe*` API 動作不支援資源層級許可，因此您無法限制使用者可以在啟動執行個體精靈中檢視的個別資源。不過，您可以對動作套用資源層級權限，以限制使用者可以使用哪些資源來啟`ec2:RunInstances`API動執行個體。如果使用者選取了自己無使用授權的選項，啟動會失敗。

Restrict access to a specific instance type, subnet, and Region (限制對特定執行個體類型、子網和區域的存取)

以下政策允許使用者使用 Amazon AMIs 擁有的`t2.micro`執行個體啟動執行個體，並且只能啟動到特定子網路 (`subnet-1a2b3c4d`)。使用者只能在指定的地區啟動。如果使用者選取不同的區域，或在啟動執行個體精靈中選取不同的執行個體類型或子網路，則啟動會失敗。AMI

第一個陳述式授與使用者許可來檢視啟動執行個體精靈中的選項，或建立新的選項。如上列的範例所解釋。第二個陳述式授與使用者使用網路介面、磁碟區、key pair、安全性群組和子網路資源的權限，這些`ec2:RunInstances`動作是將執行個體啟動到VPC. 如需關於使用 `ec2:RunInstances` 動作的詳細資訊，請參閱 啟動執行個體 (RunInstances)。第三個和第四個陳述式分別授予使用者使用執行個體和AMI資源的權限，但前提是執行個體是執`t2.micro`行個體，且僅當執行個體屬AMI於 Amazon 或特定受信任且經過驗證的合作夥伴擁有時。

```
{
   "Version": "2012-10-17",
   "Statement": [{
      "Effect": "Allow",
      "Action": [
         "ec2:DescribeInstances",
         "ec2:DescribeImages",
         "ec2:DescribeInstanceTypes",
         "ec2:DescribeKeyPairs",
         "ec2:CreateKeyPair",
         "ec2:DescribeVpcs",
         "ec2:DescribeSubnets", "ec2:DescribeSecurityGroups",
         "ec2:CreateSecurityGroup",
         "ec2:AuthorizeSecurityGroupIngress"
   ],
   "Resource": "*"
   },
   {
      "Effect": "Allow",
      "Action":"ec2:RunInstances",
      "Resource": [
         "arn:aws:ec2:region:111122223333:network-interface/*",
         "arn:aws:ec2:region:111122223333:volume/*",
         "arn:aws:ec2:region:111122223333:key-pair/*",
         "arn:aws:ec2:region:111122223333:security-group/*",
         "arn:aws:ec2:region:111122223333:subnet/subnet-1a2b3c4d"
      ]
   },
   {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": [
         "arn:aws:ec2:region:111122223333:instance/*"
      ],
      "Condition": {
         "StringEquals": {
            "ec2:InstanceType": "t2.micro"
         }
      }
   },
   {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
```

```
      "Resource": [
            "arn:aws:ec2:region::image/ami-*"
      ],
      "Condition": {
         "StringEquals": {
            "ec2:Owner": "amazon"
         }
      }
   }
   ]
}
```

Restrict access to specific Systems Manager parameters (限制對特定 Systems Manager 參數的存取)

下列政策授與使用具有特定名稱之 Systems Manager 參數的存取權。

AMI在啟動執行個體精靈中選取時，第一個陳述式會授與使用者檢視 Systems Manager 參數的權限。第二個陳述式會授與使用者只使用命名為 `prod-*` 參數的許可。

```
{
   "Version": "2012-10-17",
   "Statement": [{
      "Effect": "Allow",
      "Action": [
         "ssm:DescribeParameters"
      ],
      "Resource": "*"
   },
   {
      "Effect": "Allow",
      "Action": [
         "ssm:GetParameters"
      ],
     "Resource": "arn:aws:ssm:region:123456123456:parameter/prod-*"
   }
   ]
}
```

範例：使用安全群組

檢視安全群組，以及新增和移除規則

下列政策授予使用者在 Amazon EC2 主控台中檢視安全群組、新增和移除輸入和輸出規則，以及列出和修改具有標籤之現有安全群組的規則說明的權限`Department=Test`。

在第一個陳述式中，`ec2:DescribeTags` 動作可讓使用者在主控台中檢視標籤，如此就能更容易找出使用者獲得允許修改的安全群組。

```
{
   "Version": "2012-10-17",
   "Statement": [{
      "Effect": "Allow",
      "Action": [
         "ec2:DescribeSecurityGroups",
         "ec2:DescribeSecurityGroupRules",
         "ec2:DescribeTags"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
         "ec2:AuthorizeSecurityGroupIngress",
         "ec2:RevokeSecurityGroupIngress",
         "ec2:AuthorizeSecurityGroupEgress",
         "ec2:RevokeSecurityGroupEgress",
         "ec2:ModifySecurityGroupRules",
         "ec2:UpdateSecurityGroupRuleDescriptionsIngress",
         "ec2:UpdateSecurityGroupRuleDescriptionsEgress"
      ],
      "Resource": [
         "arn:aws:ec2:region:111122223333:security-group/*"
      ],
      "Condition": {
         "StringEquals": {
            "aws:ResourceTag/Department": "Test"
         }
      }
   },
   {
      "Effect": "Allow",
      "Action": [
         "ec2:ModifySecurityGroupRules"
      ],
      "Resource": [
```

```
        "arn:aws:ec2:region:111122223333:security-group-rule/*"
      ]
    }
]}
```

使用 Create Security Group (建立安全群組) 對話方塊

您可以建立政策，讓使用者在 Amazon EC2 主控台中使用「建立安全群組」對話方塊。若要使用此對話方塊，使用者必須獲得至少使用下列API動作的權限：

- `ec2:CreateSecurityGroup`：建立新的安全群組。
- `ec2:DescribeVpcs`：檢視清單VPCs中現有的VPC清單。

使用者可利用這些權限來順利建立新的安全群組，但無法新增任何規則到安全群組。若要使用 [建立安全性群組] 對話方塊中的規則，您可以將下列API動作新增至您的原則：

- `ec2:AuthorizeSecurityGroupIngress`：新增傳入規則。
- `ec2:AuthorizeSecurityGroupEgress`：將輸出規則新增至VPC安全性群組。
- `ec2:RevokeSecurityGroupIngress`：修改或刪除現有的傳入規則。這是一項實用的動作，可讓使用者在主控台中使用 Copy to new (複製到新的) 功能。這項功能會開啟 Create Security Group (建立安全群組) 對話方塊，並使用與所選取安全群組相同的規則，來填入規則。
- `ec2:RevokeSecurityGroupEgress`：修改或刪除VPC安全群組的輸出規則。這是一項實用的動作，可讓使用者修改或刪除預設的外傳規則 (此規則允許所有外傳流量)。
- `ec2:DeleteSecurityGroup`：適用於無法儲存個別規則的情況。主控台會先建立安全群組，然後再加入指定的規則。如果規則無效，動作會失敗，而主控台會試著刪除安全群組。使用者仍在使用 Create Security Group (建立安全群組) 對話方塊，因此可以修正無效的規則，然後再次試著建立安全群組。不需要API執行此動作，但是如果未授與使用者使用該動作的權限，並嘗試建立含有無效規則的安全性群組，則會建立安全性群組，而且使用者之後必須新增它們。
- `ec2:UpdateSecurityGroupRuleDescriptionsIngress`：新增或更新傳入 (輸入) 安全群組規則的描述。
- `ec2:UpdateSecurityGroupRuleDescriptionsEgress`：新增或更新傳出 (輸出) 安全群組規則的描述。
- `ec2:ModifySecurityGroupRules`：修改安全群組規則。
- `ec2:DescribeSecurityGroupRules`：列出安全群組規則。

下列原則授與使用者使用 [建立安全性群組] 對話方塊的權限，以及為與特定 VPC (vpc-1a2b3c4d) 相關聯的安全性群組建立輸入和輸出規則。使用者可以為 a 建立安全性群組VPC，但無法將任何規則新增至其中。同樣地，使用者無法將任何規則新增至任何未與關聯的現有安全性群組VPCvpc-1a2b3c4d。使用者也會獲得權限，來檢視主控台中的所有安全群組。如此，使用者就能更輕鬆地找出可以加入傳入規則的安全群組。此原則也會授與使用者刪除與相關聯之安全性群組的權限VPCvpc-1a2b3c4d。

```
{
   "Version": "2012-10-17",
   "Statement": [{
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeSecurityGroups",
        "ec2:CreateSecurityGroup",
        "ec2:DescribeVpcs"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:DeleteSecurityGroup",
        "ec2:AuthorizeSecurityGroupIngress",
        "ec2:AuthorizeSecurityGroupEgress"
      ],
      "Resource": "arn:aws:ec2:region:111122223333:security-group/*",
      "Condition":{
         "ArnEquals": {
            "ec2:Vpc": "arn:aws:ec2:region:111122223333:vpc/vpc-1a2b3c4d"
         }
      }
    }
   ]
}
```

範例：使用彈性 IP 地址

若要允許使用者在 Amazon EC2 主控台中檢視彈性 IP 地址，您必須授與使用者使用該ec2:DescribeAddresses動作的權限。

若要讓使用者使用彈性 IP 地址，您可以將下列的動作加入政策。

- `ec2:AllocateAddress`：配置彈性 IP 地址。
- `ec2:ReleaseAddress`：釋出彈性 IP 地址。
- `ec2:AssociateAddress`：建立彈性 IP 地址與執行個體或網路介面的關聯。
- `ec2:DescribeNetworkInterfaces` 和 `ec2:DescribeInstances`：使用 Associate address (關聯地址) 畫面。此畫面會顯示可以和彈性 IP 地址建立關聯的執行個體或網路介面。
- `ec2:DisassociateAddress`：取消彈性 IP 地址與執行個體或網路介面的關聯。

下列政策可讓使用者檢視和配置彈性 IP 地址，並建立這些地址與執行個體的關聯。使用者不能建立彈性 IP 地址與網路介面的關聯、取消彈性 IP 地址的關聯，也不能釋出這些地址。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeAddresses",
                "ec2:AllocateAddress",
                "ec2:DescribeInstances",
                "ec2:AssociateAddress"
            ],
            "Resource": "*"
        }
    ]
}
```

範例：使用 預留執行個體

以下政策允許使用者檢視和修改帳戶中的預留執行個體，以及在 AWS Management Console中購買新的預留執行個體。

此政策允許使用者檢視帳戶中的所有預留執行個體，以及帳戶中的所有隨需執行個體。您無法針對個別的預留執行個體設定資源層級的許可。

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

```
            "ec2:DescribeReservedInstances",
            "ec2:ModifyReservedInstances",
            "ec2:PurchaseReservedInstancesOffering",
            "ec2:DescribeInstances",
            "ec2:DescribeInstanceTypes",
            "ec2:DescribeAvailabilityZones",
            "ec2:DescribeReservedInstancesOfferings"
        ],
        "Resource": "*"
    }
    ]
}
```

為了確保 Amazon EC2 主控台可以顯示您可以在其中購買預留執`ec2:DescribeAvailabilityZones`行個體之可用區域的相關資訊，必須執行此動作。`ec2:DescribeInstances` 動作則非必要，但可確保使用者能夠檢視帳戶中的執行個體，並購買保留，以配合目前的規格。

您可以調整動API作以限制使用者存取權，例如移除`ec2:DescribeInstances`並`ec2:DescribeAvailabilityZones`表示使用者具有唯讀存取權。

AWS Amazon 的受管政策 EC2

若要新增使用者、群組和角色的權限，使用 AWS 受管理的原則比自己撰寫原則更容易。建立IAM客戶管理的政策需要時間和專業知識，以便為您的團隊提供他們所需的權限。若要快速開始使用，您可以使用我們的 AWS 受管政策。這些政策涵蓋常見使用案例，並可在您的 AWS 帳戶中使用。如需有關 AWS 受管理策略的詳細資訊，請參閱IAM使用指南中的AWS 受管理策略。

AWS 服務會維護和更新 AWS 受管理的策略。您無法變更 AWS 受管理原則中的權限。服務偶爾會在 AWS 受管政策中新增其他許可以支援新功能。此類型的更新會影響已連接政策的所有身分識別 (使用者、群組和角色)。當新功能啟動或新操作可用時，服務很可能會更新 AWS 受管政策。服務不會從 AWS 受管理的政策移除權限，因此政策更新不會破壞您現有的權限。

此外，還 AWS 支援跨多個服務之工作職能的受管理原則。例如，ReadOnlyAccess AWS 受管理的策略提供對所有 AWS 服務和資源的唯讀存取。當服務啟動新功能時，會為新作業和資源新 AWS 增唯讀權限。如需工作職能原則的清單與說明，請參閱《IAM使用指南》中的工作職能AWS 受管理原則。

AWS 受管理策略：AmazonEC2FullAccess

您可以將`AmazonEC2FullAccess`原則附加至您的IAM身分識別。此政策授予允許完整存取 Amazon 的許可EC2。

若要檢視此原則的權限，請參閱AWS 受管理AmazonEC2FullAccess的策略參考中的。

AWS 受管理策略：AmazonEC2ReadOnlyAccess

您可以將`AmazonEC2ReadOnlyAccess`原則附加至您的IAM身分識別。此政策授予允許 Amazon 唯讀存取權限的許可EC2。

若要檢視此原則的權限，請參閱AWS 受管理AmazonEC2ReadOnlyAccess的策略參考中的。

AWS 受管理策略：AWSEC2CapacityReservationFleetRolePolicy

此政策連接至名為 AWSServiceRoleForEC2CapacityReservationFleet 的服務連結角色，可讓容量保留功能代表您建立、修改和取消容量保留。如需詳細資訊，請參閱容量保留機群的服務連結角色。

若要檢視此原則的權限，請參閱AWS 受管理AWSEC2CapacityReservationFleetRolePolicy的策略參考中的。

AWS 受管理策略：AWSEC2FleetServiceRolePolicy

此原則會附加至名為的服務連結角色，AWSServiceRoleForEC2Fleet以允許EC2叢集代表您要求、啟動、終止和標記執行個體。如需詳細資訊，請參閱 叢集的EC2服務連結角色。

若要檢視此原則的權限，請參閱AWS 受管理AWSEC2FleetServiceRolePolicy的策略參考中的。

AWS 受管理策略：AWSEC2SpotFleetServiceRolePolicy

此政策連接到名為 AWSServiceRoleForEC2SpotFleet 的服務連結角色，允許 Spot Fleet 代表您啟動和管理執行個體。如需詳細資訊，請參閱 Spot Fleet 的服務連結角色。

若要檢視此原則的權限，請參閱AWS 受管理AWSEC2SpotFleetServiceRolePolicy的策略參考中的。

AWS 受管理策略：AWSEC2SpotServiceRolePolicy

此政策附加至名為的服務連結角色，AWSServiceRoleForEC2Spot以允許 Amazon EC2 代表您啟動和管理 Spot 執行個體。如需詳細資訊，請參閱 Spot 執行個體請求的服務連結角色。

若要檢視此原則的權限，請參閱AWS 受管理AWSEC2SpotServiceRolePolicy的策略參考中的。

AWS 受管理策略：AWSEC2VssSnapshotPolicy

您可以將此受管政策附加到IAM用於 Amazon EC2 Windows 執行個體的執行個體設定檔角色。該政策授予許可，以允許 Amazon EC2 代表您建立和管理VSS快照。

若要檢視此原則的權限，請參閱AWS 受管理AWSEC2VssSnapshotPolicy的策略參考中的。

AWS 受管理策略：EC2FastLaunchFullAccess

您可以將`EC2FastLaunchFullAccess`政策附加到執行個體設定檔或其他IAM角色。此原則會授與EC2快速啟動動作的完整存取權，以及鎖定目標權限，如下所示。

許可詳細資訊

- EC2快速啟動 — 授與管理存取權，以便角色可以啟用或停用EC2快速啟動，並說明EC2快速啟動映像檔。
- Amazon EC2 — Amazon 授與存取權限 EC2 RunInstances，CreateTags 並描述驗證資源許可所需的動作。
- IAM— 授與存取權以取得和使用名稱包含的執行個體設定檔，`ec2fastlaunch`以建立EC2FastLaunchServiceRolePolicy服務連結角色。

若要檢視此原則的權限，請參閱AWS 受管理EC2FastLaunchFullAccess的策略參考中的。

AWS 受管理策略：EC2FastLaunchServiceRolePolicy

此政策附加到名為的服務連結角色上，可AWSServiceRoleForEC2FastLaunch讓 Amazon EC2 建立和管理一組預先佈建的快照，從而減少啟用EC2快速啟動功能啟動執行個體所需的時間。AMI如需詳細資訊，請參閱 the section called “服務連結角色”。

若要檢視此原則的權限，請參閱AWS 受管理EC2FastLaunchServiceRolePolicy的策略參考中的。

AWS 受管理策略：Ec2InstanceConnectEndpoint

此原則會附加至名為的服務連結角色，AWSServiceRoleForEC2InstanceConnect以允許EC2執行個體Connect 端點代表您執行動作。如需詳細資訊，請參閱 EC2執行個體連線端點的服務連結角色。

若要檢視此原則的權限，請參閱AWS 受管理Ec2InstanceConnectEndpoint的策略參考中的。

Amazon EC2 更新受 AWS 管政策

檢視有關 Amazon AWS 受管政策更新的詳細資訊，EC2因為此服務開始追蹤這些變更。

變更	描述	日期
EC2FastLaunchFullAccess – 新政策	Amazon EC2 新增此政策以從執API行個體執行與EC2快速啟動功能相關的動作。針對從啟	2024 年 5 月 14 日

變更	描述	日期
	用EC2快速啟動啟動的執行個體啟動的執行個體，政策可以附加至執行個體設定檔AMI。	
AWSEC2VssSnapshotPolicy – 新政策	Amazon EC2 添加了`AWSEC2VssSnapshotPolicy` 政策，其中包含創建和添加標籤到 Amazon 機器映像（AMIs）和EBS快照的許可。	2024年3月28日
EC2FastLaunchServiceRolePolicy – 新政策	Amazon EC2 新增了EC2快速啟動功能，透過建立一組預先佈建的快照，讓 Windows AMIs 能夠更快地啟動執行個體。	2021 年 11 月 26 日
Amazon EC2 開始跟踪變化	Amazon EC2 開始追蹤其 AWS 受管政策的變更	2021 年 3 月 1 日

IAMAmazon 的角色 EC2

應用程式必須使用 AWS 認證簽署其API要求。因此，如果您是應用程式開發人員，則需要針對在執行個EC2體上執行的應用程式管理認證的策略。例如，您可安全將 AWS 登入資料分配給執行個體，讓在這些執行個體上運作的應用程式能夠使用此登入資料來簽署請求，同時避免其他使用者取得您的登入資料。但是，要將憑證安全地分配到每個執行個體，尤其是代表您 AWS 建立的執行個體 (例如 Spot 執行個體或 Auto Scaling 群組中的執行個體) 是一件困難的事。輪換認證時，您也必須能夠更新每個執行個體上的 AWS 認證。

我們設計了IAM角色，讓您的應用程式可以安全地向執行個體發出要API求，而不需要您管理應用程式使用的安全登入資料。您可以使用IAM角色委派API要求的權限，而不是建立和散佈 AWS 認證，如下所示：

1. 建立 IAM 角色。
2. 定義哪些帳戶或 AWS 服務可以擔任該角色。
3. 定義應用程式在擔任該角色之後可以使用的API動作和資源。

4. 啟動執行個體時指定該角色，或將該角色連接至現有執行個體。
5. 讓應用程式擷取一組暫時登入資料並加以使用。

例如，您可以使用IAM角色向需要在 Amazon S3 中使用儲存貯體的執行個體上執行的應用程式授予許可。您可以透過以JSON格式建立策略來指定IAM角色的權限。這些政策類似於您為使用者建立的政策。若您變更角色，此變更會傳播至所有執行個體。

> **Note**
>
> Amazon EC2 IAM 角色登入資料不受該角色中設定的最長工作階段持續時間限制。如需詳細資訊，請參閱《使用指南》中的IAM〈使用IAM角色〉。

建立IAM角色時，請關聯限制存取應用程式所需之特定API呼叫的最低權限IAM原則。對於 Windows 至 Windows 通訊，請使用妥善定義且妥善記載的 Windows 群組和角色，授與 Windows 執行個體之間的應用程式層級存取權。群組和角色可讓客戶定義最低權限的應用程式和NTFS資料夾層級權限，以限制對應用程式特定需求的存取。

您只能將一個IAM角色附加至執行個體，但您可以將相同的角色附加至多個執行個體。如需有關建立和使用IAM角色的詳細資訊，請參閱《使IAM用指南》中的〈角色〉。

您可以將資源層級權限套用至IAM政策，以控制使用者為執行個體附加、取代或卸離IAM角色的能力。如需詳細資訊，請參閱 支援 Amazon 動作的資源層級許可 EC2 API及下列範例：範例：使用 IAM 角色。

內容

- 執行個體描述檔
- 您使用案例的權限
- 從執行個體中繼資料中擷取安全登入資料
- 授與將IAM角色附加至執行個體的權限
- 將IAM角色附加至執行個體
- Amazon EC2 執行個體的執行個體身分角

執行個體描述檔

Amazon EC2 使用執行個體設定檔做為IAM角色的容器。使用主控台建立IAM角色時，IAM主控台會自動建立執行個體設定檔，並提供與其對應角色相同的名稱。如果您使用 Amazon EC2 主控台啟動具有角色的執行個體或將IAM角IAM色附加到執行個體，則可以根據執行個體設定檔名稱清單選擇角色。

如果您使用 AWS CLI API、或建立角色，您可以 AWS SDK將角色和執行個體設定檔建立為個別動作，名稱可能不同。如果您接著使用 AWS CLI API、或 AWS SDK來啟動具有角色的執行個體，或將IAM角色附加至執行個IAM體，請指定執行個體設定檔名稱。

執行個體設定檔只能包含一個IAM角色。此限制無法提高。

如需詳細資訊，請參閱IAM使用指南中的執行個體設定檔。

您使用案例的權限

當您第一次為應用程式建立IAM角色時，有時可能會授與超出所需權限的權限。在生產環境中啟動應用程式之前，您可以根據IAM角色的存取活動產生IAM原則。IAMAccess Analyzer 會檢閱您的 AWS CloudTrail 記錄檔，並產生原則範本，其中包含指定日期範圍內角色所使用的權限。您可以使用範本建立具有精細權限的受管理原則，然後將其附加至IAM角色。如此一來，您只會針對特定使用案例授與角色與 AWS 資源互動所需的權限。這有助於您符合授予最低權限的最佳實務。如需詳細資訊，請參閱IAM使IAM用者指南中的存取分析器原則產生。

從執行個體中繼資料中擷取安全登入資料

執行個體上的應用程式會從執行個體中繼資料項目 `iam/security-credentials/role-name`，擷取角色提供的安全登入資料。如此即授予該應用程式許可，能夠執行您在建立角色與安全登入資料的關聯時所定義的角色動作與資源。這些安全登入資料為暫時的，我們會定期更換。舊的登入資料過期前至少五分鐘，我們就會提供新的登入資料。

如需執行個體中繼資料的詳細資訊，請參閱使用執行個體中繼資料管理EC2執行個。

> **Warning**
>
> 如果您將執行個體中繼資料與IAM角色搭配使用的服務，請確定在服務代表您撥HTTP打呼叫時，不會公開認證。可能會暴露您的憑據的服務類型包括HTTP代理，HTML/CSS驗證器服務以及支持XML包含的XML處理器。

對於 Amazon EC2 工作負載，建議您使用下述方法擷取工作階段登入資料。這些認證應該可讓您的工作負載發出要 AWS API求，而不需`sts:AssumeRole`要使用來承擔與執行個體已關聯的相同角色。除

非您需要為屬性型存取控制 (ABAC) 傳遞工作階段標籤，或是傳遞工作階段原則來進一步限制角色的權限，否則這類角色假設呼叫是不必要的，因為它們會建立一組新的相同暫時角色工作階段認證。

如果您的工作負載使用角色來擔任它自己，則必須建立信任政策，明確允許該角色擔任它自己。如果您不建立信任原則，就會收到AccessDenied錯誤訊息。如需詳細資訊，請參閱《IAM使用指南》中的〈修改角色信任原則〉。

下列命令會擷取名為之IAM角色的安全性認證s3access。

cURL

IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-
aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-
data/iam/security-credentials/s3access
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/iam/security-
credentials/s3access
```

PowerShell

IMDSv2

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-
seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
 GET -Uri http://169.254.169.254/latest/meta-data/iam/security-credentials/s3access
```

IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/iam/security-
credentials/s3access
```

下列為範例輸出。如果您無法擷取安全登入資料，請參閱《IAM使用者指南》中的「我無法存取EC2執行個體上的臨時安全登入資料」。

```
{
  "Code" : "Success",
  "LastUpdated" : "2012-04-26T16:39:16Z",
  "Type" : "AWS-HMAC",
  "AccessKeyId" : "ASIAIOSFODNN7EXAMPLE",
  "SecretAccessKey" : "wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY",
  "Token" : "token",
  "Expiration" : "2017-05-17T15:09:54Z"
}
```

對於在執行個體上執行的應用程式 AWS CLI、和 Windows 專用工具 PowerShell 命令，您不需要明確取得暫時的安全性登入資料 — AWS SDKs AWS CLI、和 Windows Tools PowerShell 會自動從執行EC2個體中繼資料服務取得認證並加以使用。若要使用臨時安全性認證 (例如，測試IAM原則) 在執行個體外部進行呼叫，您必須提供存取金鑰、秘密金鑰和工作階段 Token。如需詳細資訊，請參閱使用指南中的使用臨時安全登入資料要求 AWS 資源存取IAM權。

授與將IAM角色附加至執行個體的權限

您 AWS 帳戶的身分 (例如IAM使用者) 必須具有特定的許可，才能啟動具有IAM角色的 Amazon EC2 執行個體、將IAM角色附加至執行個體、取代執行個體的IAM角色，或從執行個體中分離IAM角色。您必須授與權限才能視需要使用下列API動作：

- `iam:PassRole`
- `ec2:AssociateIamInstanceProfile`
- `ec2:DisassociateIamInstanceProfile`
- `ec2:ReplaceIamInstanceProfileAssociation`

> **Note**
>
> 如果您將資源指定`iam:PassRole`為`*`，這將授予將任何IAM角色傳遞給執行個體的存取權。若要遵循最小權限的最佳作法，請指定具有ARNs的特定IAM角色`iam:PassRole`，如以下範例政策所示。

程式化存取的範例原則

下列IAM政策授予使用或 Amazon 啟動具有IAM角色的執行個體、將IAM角色附加至執行個體或取代執行個體IAM角色的 AWS CLI 許可EC2API。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
         "ec2:RunInstances",
         "ec2:AssociateIamInstanceProfile",
         "ec2:DisassociateIamInstanceProfile",
         "ec2:ReplaceIamInstanceProfileAssociation"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": "iam:PassRole",
      "Resource": "arn:aws:iam::123456789012:role/DevTeam*"
    }
  ]
}
```

主控台存取的其他需求

若要授與使用 Amazon EC2 主控台完成相同任務的許可，您還必須包括`iam:ListInstanceProfilesAPI`動作。

將IAM角色附加至執行個體

您可以建立IAM角色，並在啟動期間或之後將其附加至執行個體。您也可以取代或卸離IAM角色。

若要在啟動時使用 Amazon EC2 主控台將IAM角色附加到執行個體，請展開進階詳細資料。針對IAM執行個體設定檔，選取IAM角色。

> **Note**
>
> 如果您是使用IAM主控台建立IAM角色，則系統會為您建立執行個體設定檔，並提供與該角色相同的名稱。如果您使用、或建立IAM角色 AWS CLI API AWS SDK，可能已為執行個體設定檔指定與角色不同的名稱。

您可以將IAM角色附加至執行中或已停止的執行個體。如果執行個體已附加IAM角色，您必須將其取代為新IAM角色。

Console

若要將IAM角色附加至執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體。
4. 選擇動作、安全性、修改IAM角色。
5. 針對 IAMrole，請選取IAM執行個體設定檔。
6. 選擇 [更新IAM角色]。

AWS CLI

若要將IAM角色附加至執行個體

使用指associate-iam-instance-profile令將IAM角色附加至執行個體。指定執行個體設定檔時，可以使用執行個體設定檔的 Amazon 資源名稱 (ARN)，也可以使用其名稱。

```
aws ec2 associate-iam-instance-profile \
    --instance-id i-1234567890abcdef0 \
    --iam-instance-profile Name="TestRole-1"
```

下列為範例輸出。

```
{
    "IamInstanceProfileAssociation": {
        "InstanceId": "i-1234567890abcdef0",
        "State": "associating",
        "AssociationId": "iip-assoc-0dbd8529a48294120",
        "IamInstanceProfile": {
            "Id": "AIPAJLNLDX3AMYZNWYYAY",
            "Arn": "arn:aws:iam::123456789012:instance-profile/TestRole-1"
        }
    }
}
```

PowerShell

若要將IAM角色附加至執行個體

- Get-EC2Instance
- Register-EC2IamInstanceProfile

若要取代已有附加IAM角色的執行個體上的IAM角色，執行個體必須處於該`running`狀態。如果您想要變更執行個體的IAM角色而不先分離現有的執行個體角色，可以這麼做。例如，您可以這樣做，以確保執行個體上API執行的應用程式執行的動作不會中斷。

Console

若要取代執行個體的IAM角色

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體。
4. 選擇動作、安全性、修改IAM角色。
5. 針對 IAMrole，請選取IAM執行個體設定檔。
6. 選擇 [更新IAM角色]。

AWS CLI

若要取代執行個體的IAM角色

1. 如有必要，請說明您的IAM執行個體設定檔關聯，以取得要取代之IAM執行個體設定檔的關聯ID。

```
aws ec2 describe-iam-instance-profile-associations
```

2. 透過指定現有IAM執行個體設定檔的replace-iam-instance-profile關聯 ID，以及應取代它的執行個體設定檔ARN或名稱，以取代執行個體設定檔，以取代執行個體設定檔。

```
aws ec2 replace-iam-instance-profile-association \
    --association-id iip-assoc-0044d817db6c0a4ba \
    --iam-instance-profile Name="TestRole-2"
```

下列為範例輸出。

```
{
    "IamInstanceProfileAssociation": {
        "InstanceId": "i-087711ddaf98f9489",
        "State": "associating",
        "AssociationId": "iip-assoc-09654be48e33b91e0",
        "IamInstanceProfile": {
            "Id": "AIPAJCJEDKX7QYHWYK7GS",
            "Arn": "arn:aws:iam::123456789012:instance-profile/TestRole-2"
        }
    }
}
```

PowerShell

若要取代執行個體的IAM角色

- Get-EC2IamInstanceProfileAssociation
- Set-EC2IamInstanceProfileAssociation

您可以從執行中或已停止的執行個體中斷連結IAM角色。

Console

若要從執行個體卸離IAM角色

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體。
4. 選擇動作、安全性、修改IAM角色。
5. 對於IAM角色，請選擇無IAM角色。
6. 選擇 [更新IAM角色]。
7. 升級以進行確認時，請輸入「分離」，然後選擇「分離」。

AWS CLI

若要從執行個體卸離IAM角色

1. 如有必要，請使用 describe-iam-instance-profile- sociation 描述您的IAM執行個體設定檔關聯，並取得要分離之IAM執行個體設定檔的關聯 ID。

```
aws ec2 describe-iam-instance-profile-associations
```

下列為範例輸出。

```
{
    "IamInstanceProfileAssociations": [
        {
            "InstanceId": "i-088ce778fbfeb4361",
            "State": "associated",
            "AssociationId": "iip-assoc-0044d817db6c0a4ba",
            "IamInstanceProfile": {
                "Id": "AIPAJEDNCAA64SSD265D6",
                "Arn": "arn:aws:iam::123456789012:instance-profile/TestRole-2"
            }
        }
    ]
}
```

2. 使用指disassociate-iam-instance-profile令，使用其關聯 ID 分離IAM執行個體設定檔。

```
aws ec2 disassociate-iam-instance-profile --association-id iip-
assoc-0044d817db6c0a4ba
```

下列為範例輸出。

```
{
    "IamInstanceProfileAssociation": {
        "InstanceId": "i-087711ddaf98f9489",
        "State": "disassociating",
        "AssociationId": "iip-assoc-0044d817db6c0a4ba",
        "IamInstanceProfile": {
            "Id": "AIPAJEDNCAA64SSD265D6",
            "Arn": "arn:aws:iam::123456789012:instance-profile/TestRole-2"
        }
    }
```

```
}
```

PowerShell

若要從執行個體卸離IAM角色

- Get-EC2IamInstanceProfileAssociation
- Unregister-EC2IamInstanceProfile

Amazon EC2 執行個體的執行個體身分角

您啟動的每個 Amazon EC2 執行個體都有一個代表其身分識別的執行個體身分角色。執行個體識別角色是一種IAM角色類型。 AWS 整合以使用執行個體身分識別角色的服務和功能可以使用它來識別服務的執行個體。

執行個體識別角色認證可從執行個體中繼資料服務 (IMDS) 存取，位於`/identity-credentials/ec2/security-credentials/ec2-instance`。憑據由 AWS 臨時訪問 key pair 和會話令牌組成。它們是用來簽 AWS 署 Sigv4 要求，以使用執行個體身分識別角色的 AWS 服務。無論執行個體上是否已啟用使用執行個體身分角色的服務或功能，憑證都會顯示在執行個體中繼資料中。

執行個體身分角色在執行個體啟動時自動建立，沒有角色信任政策文件，且不受任何身分或資源政策的約束。

支援的服務

下列 AWS 服務使用執行個體識別角色：

- Amazon EC2 — EC2執行個體 Connect 使用執行個體身分角色更新 Linux 執行個體的主機金鑰。
- Amazon GuardDuty — 執行階段監控使用執行個體身分角色，允許執行階段代理程式將安全性遙測傳送到 GuardDuty VPC端點。
- AWS Security Token Service (AWS STS) — 執行個體身分識別角色認證可與 AWS STS `GetCallerIdentity`動作搭配使用。
- AWS Systems Manager— 使用預設主機管理組態時， AWS Systems Manager 會使用執行個體識別角色提供的識別來註冊EC2執行個體。在識別執行個體之後，Systems Manager 可以將您的`AWSSystemsManagerDefaultEC2InstanceManagementRole`IAM角色傳遞給執行個體。

執行個體身分識別角色無法與其他 AWS 服務或功能搭配使用，因為這些角色未與執行個體身分識別角色整合。

執行個體識別角 ARN

執行個體識別角色ARN的格式如下：

```
arn:aws-partition:iam::account-number:assumed-role/aws:ec2-instance/instance-id
```

例如：

```
arn:aws:iam::0123456789012:assumed-role/aws:ec2-instance/i-0123456789example
```

如需有關的詳細資訊ARNs，請參閱IAM使用者指南中的 Amazon 資源名稱 (ARNs)。

Amazon EC2 視窗執行個體更新管理

我們建議您定期修補、更新並保護EC2執行個體上的作業系統和應用程式。您可以使用 AWS Systems Manager Patch Manager 會針對作業系統及應用程式自動化安裝安全性相關更新的程序。

對於 Auto Scaling 群組中的EC2執行個體，您可以使用 AWS-PatchAsgInstancerunbook 協助避免正在進行修補的執行個體遭到取代。或者，您也可以使用任何自動更新服務或建議的程序，以安裝應用程式廠商提供的更新。

資源

- AL2023 — 在 Amazon 用戶指南中更新 AL2 0 23。
- AL2— 在 Amazon Linux 2 用戶指南中管理您的 Amazon Linux 2 實例上的軟件。
- 視窗實例 —the section called “更新管理”.

Windows 執行個體的安全性最佳做法

我們建議您遵循 Windows 執行個體的這些安全性最佳做法。

目錄

- 高階安全性最佳做法
- 更新管理
- 組態管理
- 變更管理

- Amazon EC2 視窗執行個體的稽核與責任

高階安全性最佳做法

您應該遵守下列適用於 Windows 執行個體的高階安全性最佳做法：

- 最少存取權 — 僅授予對受信任且預期的系統和位置的存取權。這適用於所有 Microsoft 產品，例如 Active Directory、Microsoft 企業生產力伺服器，以及基礎結構服務，例如遠端桌面服務、反向代理伺服器、IIS網頁伺服器等。使用 Amazon EC2 執行個體安全群組、網路存取控制清單 (ACLs) 和 Amazon VPC 公有/私有子網路等 AWS 功能，將架構中多個位置的安全性分層。在 Windows 執行個體中，客戶可以使用 Windows 防火牆在部署中進一步分層 defense-in-depth 策略。僅安裝系統依設計運作所需的作業系統元件和應用程式。設定基礎結構服務，例如IIS在服務帳戶下執行，或使用應用程式集區識別等功能，在您的基礎結構本機和遠端存取資源。
- 最小權限 — 決定執行個體和帳戶執行其功能所需的最低權限集。限制這些伺服器和使用者只允許這些已定義的許可。使用角色型存取控制這類技術來減少管理帳戶的表面區域，並建立最受限的角色來完成任務。使用作業系統功能，例如加密檔案系統 (EFS) NTFS 來加密靜態的敏感資料，並控制應用程式和使用者存取。
- 組態管理 — 建立基準伺服器組態，其中包含 up-to-date 安全性修補程式和主機型保護套件，其中包括防毒、反惡意程式碼、入侵偵測/預防，以及檔案完整性監控。根據目前記錄的基準來評定每部伺服器，以識別並標示任何偏差。確保每部伺服器都已設定為產生並安全地存放適當的日誌和稽核資料。
- 變更管理 — 建立程序以控制伺服器組態基準的變更，並朝著完全自動化的變更流程進行。此外，利用 Windows 的「剛夠管理」(JEA)， PowerShellDSC將系統管理存取限制為最低限度的必要功能。
- 修補程式管理 — 實作可定期修補、更新及保護EC2執行個體上作業系統和應用程式的程序。
- 稽核日誌 — 稽核 Amazon EC2 執行個體的存取和所有變更，以驗證伺服器完整性並確保僅進行授權的變更。利用增強型記錄等功能IIS來增強預設記錄功能。 AWS 諸如VPC流程記錄等功能， AWS CloudTrail 也可用於稽核網路存取，包括分別允許/拒絕的要求和API呼叫。

更新管理

為了確保在 Amazon 上執行 Windows 伺服器時獲得最佳結果EC2，我們建議您實作下列最佳實務：

- Configure Windows Update
- Update drivers
- Use the latest Windows AMIs

- Test performance before migration
- Update launch agents
- 安裝更新之後，請重新啟動 Windows 執行個體。如需詳細資訊，請參閱重新啟動您的執行個體。

如需如何將 Windows 執行個體升級或遷移至較新版 Windows Server 的資訊，請參閱將EC2視窗執行個體升級至較新版本的視窗伺服器。

設定視窗更新

根據預設，從 AWS Windows 伺服器啟動的執行個體AMIs不會透過 Windows 更新接收更新。

更新 Windows 驅動程式

維護所有 Windows EC2 執行個體上的最新驅動程式，以確保在您的叢集中套用最新的問題修正和效能增強功能。根據您的執行個體類型，您應該更新 AWS PV ENA、Amazon 和 AWS NVMe驅動程式。

- 使用SNS主題接收新驅動程式版本的更新。
- 使用 AWS Systems Manager 自動化手冊 AWSSupport- 輕鬆地UpgradeWindowsAWSDrivers將更新套用到您的執行個體。

使用最新的視窗啟動執行個體 AMIs

AWS AMIs每個月都會發佈新的 Windows，其中包含最新的作業系統修補程式、驅動程式和啟動代理程式。AMI當您啟動新執行個體或建立自己的自訂映像檔時，您應該利用最新版本。

- 若要檢視視 AWS 窗每個發行版本的更新AMIs，請參閱 AWS Windows AMI 版本歷程記錄。
- 若要使用可用的最新版本進行建置AMIs，請參閱AMI使用 Systems Manager 參數存放區查詢最新Windows。
- 如需有關可用來啟動資料庫執行個體的專用 Windows AMIs 的詳細資訊，以及相容性強化使用案例，請參閱 Windows AMI 參考AMIs中的特定AWS Windows。

在遷移前測試系統/應用程式效能

將企業應用程式移轉至 AWS 可能涉及許多變數和組態。始終測試EC2解決方案的性能以確保：

- 已正確配置執行個體類型，包括執行個體大小、增強型聯網和租用 (共用或專用)。
- 執行個體拓撲適用於工作負載，並在必要時運用高效能功能，例如專用租用、置放群組、執行個體存放區磁碟區、裸機。

更新啟動代理程式

更新至最新的 EC2Launch v2 代理程式，以確保在您的叢集中套用最新的增強功能。如需詳細資訊，請參閱the section called “遷移到第 EC2Launch 2 版”。

如果您有混合的叢集，或者想要繼續使用 EC2Launch (Windows Server 2016 和 2019) 或 EC2 Config (僅限舊版作業系統) 代理程式，請更新為相應代理程式的最新版本。

下列 Windows Server 版本和啟動代理程式組合支援自動更新。您可以在 Amazon EC2 Launch 代理程式下的SSM快速設定主機管理主控台中選擇加入自動更新。

Windows 版本	EC2Launch v1	EC2Launch v2
2016	✓	✓
2019	✓	✓
2022		✓

- 如需更新為 EC2Launch v2 的詳細資訊，請參閱the section called “安裝第 EC2Launch 2 版”。
- 如需手動更新的資訊EC2Config，請參閱the section called “安裝 EC2Config”。
- 如需手動更新的資訊EC2Launch，請參閱the section called “安裝 EC2Launch”。

組態管理

Amazon Machine Image (AMIs) 提供 Amazon EC2 執行個體的初始組態，其中包括 Windows 作業系統和選用的客戶特定自訂項目，例如應用程式和安全控制。建立包含自訂安全性組態基準的AMI目錄，以確保所有 Windows 執行個體都以標準安全性控制項啟動。安全性基準可在EC2執行個AMI體啟動時進行動態啟動載入，或封裝為產品，以便透過 AWS Service Catalog 產品組合進行統一發佈。如需有關保護安全的詳細資訊AMI，請參閱建置AMI.

每個 Amazon EC2 執行個體都應遵守組織安全標準。不要勿安裝任何不需要的 Windows 角色和功能，並安裝軟體以防範惡意程式碼 (防毒、反惡意軟體、漏洞防護)、監控主機完整性，以及執行入侵偵測。設定安全軟體以監控和維護作業系統安全設定、保護重要作業系統檔案的完整性，以及在偏離安全基準時發出提醒。請考慮實作 Microsoft、網際網路安全中心 () 或美國國家標準與技術研究所 (CIS) 所發行的建議安全性設定效能標竿。NIST請考慮針對特定的應用程式伺服器使用其他 Microsoft 工具，例如伺服SQL器的最佳作法分析程式。

AWS 客戶也可以執行 Amazon Inspector 評估，以改善 Amazon 執行EC2個體上部署之應用程式的安全性和合規性。Amazon Inspector 會自動評估應用程式的弱點或偏離最佳實務，並包含數百個與常見安全合規標準 (例如 PCIDSS) 和弱點定義對應的規則的知識庫。內建規則的範例包括檢查是否已啟用遠端根登入，或是否已安裝易受攻擊的軟體版本。這些規則由 AWS 安全研究人員定期更新。

保護 Windows 執行個體時，我們建議您實作 Active Directory Domain Services，為分散式位置啟用可擴展、安全且可管理的基礎設施。此外，從 Amazon EC2 主控台啟動執行個體或使用 Amazon EC2 佈建工具 (例如) 後 AWS CloudFormation，最好在發生組態偏移時利用原生作業系統功能 (例如 Microsoft Windows) PowerShell DSC 來維護組態狀態。

變更管理

在啟動時將初始安全基準套用至 Amazon EC2 執行個體後，請控制進行中的 Amazon EC2 變更以維護虛擬機器的安全性。建立變更管理程序，以授權並納入 AWS 資源 (例如安全性群組、路由表和網路 ACLs) 的變更，以及作業系統和應用程式組態 (例如 Windows 或應用程式修補、軟體升級或組態檔案更新)。

AWS 提供數種工具來協助管理 AWS 資源的變更 AWS CloudTrail AWS Config AWS CloudFormation AWS Elastic Beanstalk，包括系統中心作業管理員和系統中心虛擬機器管理員的管理組件 AWS OpsWorks、、和管理組件。請注意，Microsoft 在每個月的第二個星期二（或根據需要）發布 Windows 修補程序，並在 Microsoft 發布修補程序後的五天 AWS 內 AWS 更新所有AMIs管理的 Windows。因此，請務必使用最新版本持續修補所有基準AMIs、更新 AWS CloudFormation 範本和 Auto Scaling 群組組態 AMIIDs，並實作工具以自動執行執行的執行個體修補程式管理。

Microsoft 提供了數個選項，用於管理 Windows 作業系統和應用程式變更。SCCM例如，提供環境修改的完整生命週期涵蓋範圍。選擇可滿足業務需求並控制變更對應用程式SLAs、容量、安全性和災難復原程序的影響方式的工具。避免手動變更，而是利用自動化組態管理軟體或命令列工具 (例如EC2執行命令或 Windows) PowerShell 來實作可重複指令碼的變更程序。為了協助滿足此需求，請將具有增強記錄功能的堡壘主機用於與 Windows 執行個體的所有互動，以確保自動記錄所有事件和任務。

Amazon EC2 視窗執行個體的稽核與責任

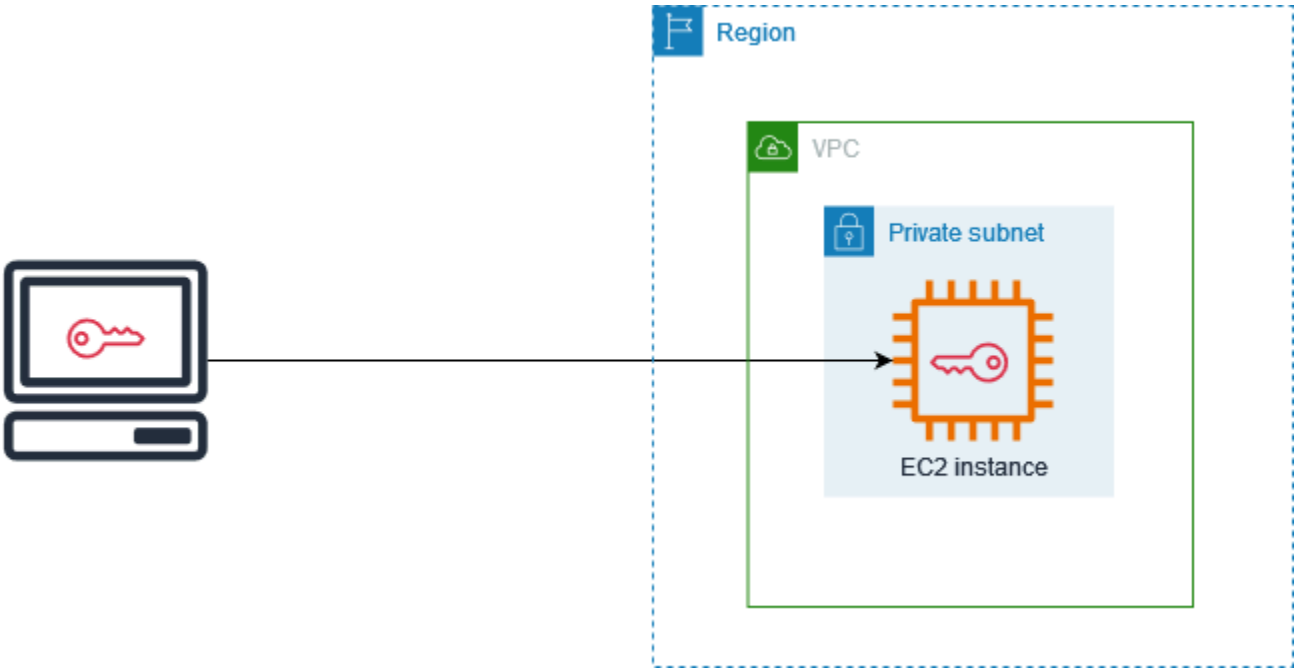
AWS CloudTrail、 AWS Config，並 AWS Config 規則 提供稽核與變更追蹤功能，以便稽核 AWS 資源變更。將 Windows 事件日誌設定為將本機日誌檔傳送至集中式日誌管理系統，來保留日誌資料以進行安全性和操作行為分析。Microsoft 系統中心作業管理員 (SCOM) 彙總部署到 Windows 執行個體的 Microsoft 應用程式的相關資訊，並根據應用程式角色和服務套用預先設定和自訂規則集。以系統中心管理組件為基礎，SCOM提供應用程式特定的監視和組態指引。這些管理組件支援視窗伺服器作用中目錄、 SharePoint 伺服器 2013 年、交換伺服器 2013 年、Lync 伺服器 2013 年、SQL伺服器 2014 年，以及更多的伺服器和技術。

除了 Microsoft 系統管理工具之外，客戶還可以使用 Amazon CloudWatch 監控執行個體使用CPU率、磁碟效能、網路 I/O，以及執行主機和執行個體狀態檢查。EC2ConfigEC2Launch、和 EC2Launch v2 啟動代理程式可讓您存取 Windows 執行個體的其他進階功能。例如，他們可以將 Windows 系統、安全性、應用程式和網際網路資訊服務 (IIS) 記錄匯出到 CloudWatch 日誌，然後可與 Amazon CloudWatch 指標和警示整合。客戶也可以建立將 Windows 效能計數器匯出至 Amazon CloudWatch 自訂指標的指令碼。

Amazon EC2 密鑰對和 Amazon EC2 實例

key pair (包含公開金鑰和私密金鑰) 是一組安全登入資料，可在連線至 Amazon EC2 執行個體時用來證明您的身分。對於 Linux 執行個體，私密金鑰可讓您安全地SSH進入執行個體。對於 Windows 執行個體，需要私密金鑰才能解密管理員密碼，然後您可以使用該密碼連線至執行個體。

Amazon 會將公開金鑰EC2存放在您的執行個體上，並儲存私密金鑰，如下圖所示。請務必將私密金鑰儲存在安全的地方，因為擁有您私密金鑰的任何人都可以連線到使用該 key pair 的執行個體。



啟動執行個體時，您可以指定 key pair，以便使用需要金 key pair 的方法連線至執行個體。根據您管理安全性的方式，您可以為所有執行個體指定相同的金鑰對，也可以指定不同的金鑰對。

對於 Linux 執行個體，當您的執行個體首次啟動時，您在啟動時指定的公開金鑰會放置在 Linux 執行個體中的項目中`~/.ssh/authorized_keys`。當您使用連線到 Linux 執行個體時SSH，若要登入，您必須指定與公開金鑰對應的私密金鑰。

如需連線至EC2執行個體的詳細資訊，請參閱Connect 至您的EC2執行個體。

> **Important**
>
> 由於 Amazon EC2 不會保留私鑰的副本，因此如果您遺失私密金鑰，將無法復原私密金鑰。但是，仍然有一種方法可以連線到遺失私有金鑰的執行個體。如需詳細資訊，請參閱 我的私有金鑰遺失。如何連線到我的執行個體？

作為金鑰配對的替代方案，您可AWS Systems Manager Session Manager以使用互動式按一下瀏覽器殼層或 AWS Command Line Interface (AWS CLI) 連線至執行個體。

目錄

- 為您的 Amazon EC2 執行個體建立 key pair
- 標記金鑰對
- 描述您的金鑰配對
- 刪除您的金鑰對
- 在 Linux 執行個體上新增或取代公開金鑰
- 驗證金鑰對的指紋

為您的 Amazon EC2 執行個體建立 key pair

您可以使EC2用 Amazon 建立金鑰對，也可以使用第三方工具建立金鑰配對，然後將其匯入 Amazon EC2。

Amazon 針對 Linux 和視窗執行個EC2體支援 2048 位元 SSH -2 RSA 金鑰。Amazon EC2 還支持 Linux 實例的ED25519密鑰。

如需在建立 key pair SSH 之後使用連線至 Linux 執行個體的步驟，請參閱the section called “使用 Connect 到您的 Linux 執行個體 SSH”。

如需在建立 key pair RDP 之後使用連線至 Windows 執行個體的步驟，請參閱the section called “使用 Connect 至您的 Windows 執行個體 RDP”。

目錄

- 使用 Amazon 創建一個 key pair EC2
- 使用建立 key pair AWS CloudFormation
- 使用第三方工具建立 key pair，並將公開金鑰匯入 Amazon EC2

使用 Amazon 創建一個 key pair EC2

當您使用 Amazon 創建 key pair 時EC2，公鑰將存儲在 Amazon 中EC2，並存儲私鑰。

每個區域最多可以建立 5,000 個金鑰配對。若要請求增加，請建立支援案例。如需詳細資訊，請參閱 AWS Support 使用者指南中的建立支援案例。

Console

若要使用 Amazon 建立 key pair EC2

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 請在導覽窗格的 Network & Security (網路與安全性) 下方，選擇 Key Pairs (金鑰對)。
3. 選擇 Create key pair (建立金鑰對)。
4. 對於 Name (名稱)，輸入金鑰對的描述性名稱。Amazon EC2 會將公開金鑰與您指定為金鑰名稱的名稱相關聯。金鑰名稱最多可包含 255 個ASCII字元。它不能包括前置或結尾空格。
5. 選取適合您作業系統的 key pair 類型：

   (Linux 執行個體) 針對金鑰配對類型，請選擇RSA或ED25519。

   (Windows 執行個體) 針對金鑰配對類型，選擇RSA。ED25519視窗執行個體不支援金鑰。
6. 針對 Private key file format (私有金鑰檔案格式)，選擇私有金鑰的儲存格式。若要以可與「開啟」搭配使用的格式儲存私密金鑰SSH，請選擇 pem。若要以可與 Pu 搭配使用的格式儲存私密金鑰TTY，請選擇 ppk。
7. 若要向公有金鑰新增標籤，請選擇 Add tag (新增標籤)，然後輸入標籤的金鑰和值。針對每個標籤，重複執行上述動作。
8. 選擇 Create key pair (建立金鑰對)。
9. 您的瀏覽器會自動下載私有金鑰檔案。基礎檔案名稱是您所指定的金鑰對名稱，而副檔名是根據您選擇的檔案格式所決定。將私有金鑰檔案存放在安全的地方。

   > **Important**
   > 這是您儲存私有金鑰檔案的唯一機會。

10. 如果您打算使用 macOS 或 Linux 電腦上的用SSH戶端連線至 Linux 執行個體，請使用下列指令來設定私密金鑰檔案的權限，以便只有您才能讀取該檔案。

```
chmod 400 key-pair-name.pem
```

如果您未設定這些許可，則無法使用此金鑰對來連接至您的執行個體。如需詳細資訊，請參閱錯誤：未受保護的私有金鑰檔案。

AWS CLI

若要使用 Amazon 建立 key pair EC2

1. 按如下說明，使用 create-key-pair 命令，產生金鑰對並將私有金鑰儲存到 `.pem` 檔案中。

   對於 `--key-name`，指定公有金鑰的名稱。名稱最多可ASCII包含 255 個字元。

   對於 `--key-type`，指定 `rsa` 或 `ed25519`. 如果您未包含 `--key-type` 參數，預設會建立 `rsa` 金鑰。請注意，Windows 執行個體不支援ED25519金鑰。

   對於 `--key-format`，指定 `pem` 或 `ppk`。如果您未包含 `--key-format` 參數，預設會建立 `pem` 檔案。

   `--query "KeyMaterial"` 將私有金鑰材料列印到輸出。

   `--output text > my-key-pair.pem` 會將私有金鑰資料儲存在有特定副檔名的檔案中。副檔名可為 `.pem` 或 `.ppk`。私有金鑰的名稱可以與公有金鑰名稱不同，但為了方便使用，請使用相同的名稱。

   ```
   aws ec2 create-key-pair \
       --key-name my-key-pair \
       --key-type rsa \
       --key-format pem \
       --query "KeyMaterial" \
       --output text > my-key-pair.pem
   ```

2. 如果您打算使用 macOS 或 Linux 電腦上的用SSH戶端連線至 Linux 執行個體，請使用下列指令來設定私密金鑰檔案的權限，以便只有您才能讀取該檔案。

   ```
   chmod 400 key-pair-name.pem
   ```

   如果您未設定這些許可，則無法使用此金鑰對來連接至您的執行個體。如需詳細資訊，請參閱錯誤：未受保護的私有金鑰檔案。

PowerShell

若要使用 Amazon 建立 key pair EC2

使用如下New-EC2KeyPair AWS Tools for Windows PowerShell 命令來生成密鑰並將其保存到.pem或.ppk文件中。

對於 -KeyName，指定公有金鑰的名稱。名稱最多可ASCII包含 255 個字元。

對於 -KeyType，指定 rsa 或 ed25519. 如果您未包含 -KeyType 參數，預設會建立 rsa 金鑰。請注意，Windows 執行個體不支援ED25519金鑰。

對於 -KeyFormat，指定 pem 或 ppk。如果您未包含 -KeyFormat 參數，預設會建立 pem 檔案。

KeyMaterial 將私有金鑰材料列印到輸出。

Out-File -Encoding ascii -FilePath *C:\path\my-key-pair*.pem 會將私有金鑰資料儲存在有特定副檔名的檔案中。副檔名可為 .pem 或 .ppk。私有金鑰的名稱可以與公有金鑰名稱不同，但為了方便使用，請使用相同的名稱。

```
PS C:\> (New-EC2KeyPair -KeyName "my-key-pair" -KeyType "rsa" -KeyFormat
 "pem").KeyMaterial | Out-File -Encoding ascii -FilePath C:\path\my-key-pair.pem
```

使用建立 key pair AWS CloudFormation

使用建立新 key pair 時 AWS CloudFormation，私密金鑰會儲存至 AWS Systems Manager 參數存放區。參數名稱具有以下格式：

```
/ec2/keypair/key_pair_id
```

如需詳細資訊，請參閱《AWS Systems Manager 使用者指南》中的 AWS Systems Manager 參數存放區。

若要使用建立 key pair AWS CloudFormation

1. 在範本中指定AWSEC2::: KeyPair 資源。

```
Resources:
```

```
  NewKeyPair:
    Type: 'AWS::EC2::KeyPair'
    Properties:
      KeyName: new-key-pair
```

2. 按如下說明，使用 describe-key-pairs 命令以取得金鑰對 ID。

```
aws ec2 describe-key-pairs --filters Name=key-name,Values=new-key-pair --query
 KeyPairs[*].KeyPairId --output text
```

下列為範例輸出。

```
key-05abb699beEXAMPLE
```

3. 按如下說明，使用 get-parameter 命令以取得金鑰參數，並將金鑰材料儲存在 `.pem` 檔案中。

```
aws ssm get-parameter --name /ec2/keypair/key-05abb699beEXAMPLE --with-decryption
 --query Parameter.Value --output text > new-key-pair.pem
```

必要的IAM權限

若 AWS CloudFormation 要啟用以代表您管理參數存放區參數， AWS CloudFormation 或您的使用者所擔任的IAM角色必須具有下列權限：

- `ssm:PutParameter` – 授予許可，以建立私有金鑰材料的參數。
- `ssm:DeleteParameter` - 准許刪除用來存放私有金鑰材料的參數。無論金鑰對是否是由 AWS CloudFormation匯入或建立，都需要此許可。

AWS CloudFormation 刪除由堆疊建立或匯入的 key pair 時， AWS CloudFormation 會執行權限檢查以判斷您是否具有刪除參數的權限，即使只在建立 key pair 時建立參數，而不是在匯入 key pair 時建立參數。 AWS CloudFormation 使用與您帳戶中任何參數不相符的製造參數名稱來測試所需權限。因此，您可能會在 `AccessDeniedException` 錯誤訊息中看到虛構參數名稱。

使用第三方工具建立 key pair，並將公開金鑰匯入 Amazon EC2

Linux 執行個體

您可以使用第三方工具EC2來創建RSA或ED25519密 key pair，然後將公鑰導入 Amazon，而不是使用 Amazon 創建密鑰對EC2。

金鑰對的需求

- 支援的類型：RSA和ED25519。Amazon EC2 不接受DSA鑰匙。
- 支援的格式：
  - 開啟SSH公開金鑰格式 (格式為`~/.ssh/authorized_keys`)。如果您在使用「EC2執行個體連線」SSH 時使用連線API，也支援此SSH2格式。
  - SSH私密金鑰檔案格式必須是PEM或 PPK
  - (RSA僅) 基 64 編碼DER格式
  - （RSA僅）RFC47 16 中指定的SSH公鑰文件格式
- 支援的長度：1024、2048 和 4096。如果您在使用EC2執行個體連線SSH時使用連線API，支援的長度為 2048 和 4096。

使用第三方工具來產生金鑰對

1. 使用您自選的第三方工具來產生金鑰對 例如，您可以使用 ssh-keygen (隨標準 Open SSH 安裝提供的工具)。或者，Java，Ruby，Python 和許多其他編程語言提供標準庫，您可以用來創建RSA 或 ED25519 key pair。

   > **Important**
   > 私密金鑰必須為PEM或PPK格式。例如，使用`ssh-keygen -m PEM`以產生格PEM式的 Open SSH 金鑰。

2. 將公有金鑰儲存到本機檔案。例如，`~/.ssh/my-key-pair.pub`。此檔案的副檔名不重要。
3. 將私有金鑰儲存到具有 `.pem` 或 `.ppk` 副檔名的本機檔案。例如 `~/.ssh/my-key-pair.pem` 或 `~/.ssh/my-key-pair.ppk`。

   > **Important**
   > 將私有金鑰檔案存放在安全的地方。您在啟動執行個體時需要提供公有金鑰的名稱，且每次連線至執行個體時也需要提供對應的私有金鑰。

Windows 執行個體

您可以使用第三方工具EC2來創建 RSA key pair，然後將公鑰導入 Amazon，而不是使用 Amazon 創建密鑰對EC2。

金鑰對的需求

- 支持的類型：RSA。Amazon EC2 不接受DSA鑰匙。

> **Note**
> ED25519視窗執行個體不支援金鑰。

- 支援的格式：
  - SSH公開金鑰格式
  - SSH私密金鑰檔案格式必須是PEM或 PPK
  - (RSA僅) 基 64 編碼DER格式
  - （RSA僅）RFC47 16 中指定的SSH公鑰文件格式
- 支援的長度：1024、2048 和 4096。

使用第三方工具來產生金鑰對

1. 使用您自選的第三方工具來產生金鑰對 例如，您可以使用 ssh-keygen (隨標準 Open SSH 安裝提供的工具)。或者，Java，Ruby，Python 和許多其他編程語言提供標準庫，您可以用來創建 RSA key pair。

   > **Important**
   > 私密金鑰必須為PEM或PPK格式。例如，使用`ssh-keygen -m PEM`以產生格PEM式的 Open SSH 金鑰。

2. 將公有金鑰儲存到本機檔案。例如，`C:\keys\my-key-pair.pub`。此檔案的副檔名不重要。
3. 將私有金鑰儲存到具有 `.pem` 或 `.ppk` 副檔名的本機檔案。例如 `C:\keys\my-key-pair.pem` 或 `C:\keys\my-key-pair.ppk`。此檔案的副檔名很重要，因為從EC2主控台連線到 Windows 執行個體時，只`.pem`能選取檔案。

   > **Important**
   > 將私有金鑰檔案存放在安全的地方。您在啟動執行個體時需要提供公有金鑰的名稱，且每次連線至執行個體時也需要提供對應的私有金鑰。

建立 key pair 之後，請使用下列其中一種方法將公開金鑰匯入 Amazon EC2。

Console

將公鑰導入 Amazon EC2

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Key Pairs (金鑰對)。
3. 選擇 Import key pair (匯入金鑰對)。
4. 在 Name (名稱) 處，輸入公有金鑰的描述性名稱。名稱最多可包含 255 個ASCII字元。它不能包括前置或結尾空格。

> **Note**
>
> 當您從主控台連線至執行個體時，EC2主控台會建議您的私密金鑰檔案名稱使用此名稱。

5. 選取 Browse (瀏覽) 以導覽並選取您的公有金鑰，或將公有金鑰的內容貼到 Public key contents (公有金鑰內容) 欄位。
6. 選擇 Import key pair (匯入金鑰對)。
7. 確認您匯入的公開金鑰出現在金鑰對清單中。

AWS CLI

將公鑰導入 Amazon EC2

使用 import-key-pair AWS CLI 命令。

確認是否已成功匯入金鑰對

使用 describe-key-pairs AWS CLI 命令。

PowerShell

將公鑰導入 Amazon EC2

使用 Import-EC2KeyPair AWS Tools for Windows PowerShell 命令。

確認是否已成功匯入金鑰對

使用 Get-EC2KeyPair AWS Tools for Windows PowerShell 命令。

標記金鑰對

為了協助分類和管理您使用 Amazon 建立EC2或匯入到 Amazon 的金鑰配對EC2，您可以使用自訂中繼資料標記它們。如需有關標籤運作方式的詳細資訊，請參閱標記您的 Amazon EC2 資源。

Console

檢視、新增或刪除 key pair 的標籤

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Key Pairs (金鑰對)。
3. 選取公有金鑰，然後選取 Actions (動作)、Manage tags (管理標籤)。
4. Manage tags (管理標籤) 頁面會顯示指派給公有金鑰的任何標籤。
   - 若要加入標籤，請選擇 Add tag (新增標籤)，然後輸入標籤索引鍵和值。每個金鑰最多可新增 50 個標籤。如需詳細資訊，請參閱 標籤限制。
   - 若要移除標籤，請選擇標籤旁的 Remove (移除) 以移除。
5. 選擇 Save (儲存)。

AWS CLI

檢視金鑰配對的標籤

使用 describe-tags AWS CLI 命令。下列範例描述所有公有金鑰的標籤。

```
aws ec2 describe-tags --filters "Name=resource-type,Values=key-pair"
```

```
{
    "Tags": [
    {
        "Key": "Environment",
        "ResourceId": "key-0123456789EXAMPLE",
        "ResourceType": "key-pair",
        "Value": "Production"
    },
    {
        "Key": "Environment",
```

```
        "ResourceId": "key-9876543210EXAMPLE",
        "ResourceType": "key-pair",
        "Value": "Production"
    }]
}
```

描述 key pair 的標籤

使用 describe-key-pairs AWS CLI 命令。

```
aws ec2 describe-key-pairs --key-pair-ids key-0123456789EXAMPLE
```

```
{
    "KeyPairs": [
    {
        "KeyName": "MyKeyPair",
        "KeyFingerprint":
 "1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f",
        "KeyPairId": "key-0123456789EXAMPLE",
        "Tags": [
        {
            "Key": "Environment",
            "Value": "Production"
        }]
    }]
}
```

標記 key pair

使用 create-tags AWS CLI 命令。在下列範例中，會使用 `Key=Cost-Center` 和 `Value=CC-123` 來標記公有金鑰。

```
aws ec2 create-tags --resources key-0123456789EXAMPLE --tags Key=Cost-
Center,Value=CC-123
```

從金鑰對刪除標籤

使用 delete-tags AWS CLI 命令。如需範例，請參閱 AWS CLI 命令參考中的範例。

PowerShell

若要檢視金鑰配對的標籤

使用 Get-EC2Tag 命令。

描述 key pair 的標籤

使用 Get-EC2KeyPair 命令。

標記 key pair

使用 New-EC2Tag 命令。

從金鑰對刪除標籤

使用 Remove-EC2Tag 命令。

描述您的金鑰配對

您可以描述存放在 Amazon 中的金鑰對EC2。您也可以擷取公有金鑰材料，並辨識啟動時指定的公有金鑰。

主題

- 描述您的金鑰配對
- 擷取公有金鑰資料
- 識別啟動時指定的公有金鑰

描述您的金鑰配對

您可以查看有關 Amazon 中存儲的公鑰的以下信息EC2：公鑰名稱，ID，密鑰類型，指紋，公鑰材料，日期和時間（在UTC時區中）由 Amazon 創建的密鑰EC2（如果密鑰是由第三方工具創建的，則是將密鑰導入到 Amazon 的日期和時間EC2）以及與公鑰關聯的任何標籤。

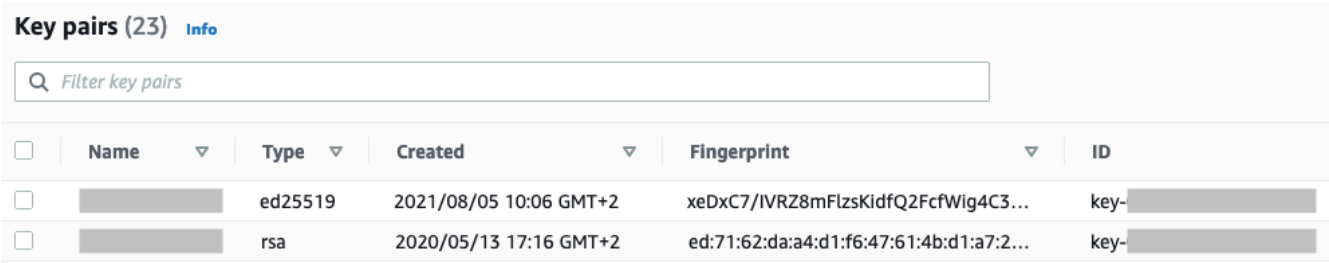
您可以使用 Amazon EC2 主控台或 AWS CLI 檢視有關公開金鑰的資訊。

Console

檢視您的公有金鑰相關資訊

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側導覽器中，選擇 Key Pairs (金鑰對)。

3. 您可以在金鑰對表中檢視每個公有金鑰的相關資訊。



4. 若要檢視公有金鑰標籤，請勾選金鑰旁的核取方塊，然後選取 Actions (動作)、Manage tags (管理標籤)。

AWS CLI

描述公有金鑰

使用 describe-key-pairs 命令，並指定 `--key-names` 參數。

```
aws ec2 describe-key-pairs --key-names key-pair-name
```

範例輸出

```
{
    "KeyPairs": [
        {
            "KeyPairId": "key-0123456789example",
            "KeyFingerprint":
 "1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f",
            "KeyName": "key-pair-name",
            "KeyType": "rsa",
            "Tags": [],
            "CreateTime": "2022-04-28T11:37:26.000Z"
        }
    ]
}
```

您也可以不要 `--key-names`，而是指定 `--key-pair-ids` 參數來識別公有金鑰。

```
aws ec2 describe-key-pairs --key-pair-ids key-0123456789example
```

若要查看輸出中的公有金鑰資料，您必須指定 `--include-public-key` 參數。

```
aws ec2 describe-key-pairs --key-names key-pair-name --include-public-key
```

範例輸出：在輸出中，PublicKey 欄位會包含公有金鑰資料。

```
{
    "KeyPairs": [
        {
            "KeyPairId": "key-0123456789example",
            "KeyFingerprint":
 "1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f",
            "KeyName": "key-pair-name",
            "KeyType": "rsa",
            "Tags": [],
            "PublicKey": "ssh-ed25519
 AAAAC3NzaC1lZDI1NTE5AAAAIIj7azlDjVHAsSxgcpCRZ3oWnTm0nAFM64y9jd22ioI/ my-key-pair",
            "CreateTime": "2022-04-28T11:37:26.000Z"
        }
    ]
}
```

擷取公有金鑰資料

您可以使用多種不同的方法來存取公有金鑰資料。您可以從本機電腦上相符的私密金鑰、使用公開金鑰啟動之執行個體上的執行個體中繼資料，或使用describe-key-pairs AWS CLI 指令擷取公開金鑰資料。對於 Linux 執行個體，也可以從執行個體上的authorized_keys檔案擷取公開金鑰材料。

使用下列其中一種方法擷取公有金鑰材料。

Linux 執行個體

From the private key

從私有金鑰中擷取公有金鑰材料

在本機的 Linux 或 macOS 電腦上，您可以使用 ssh-keygen 命令來擷取金鑰對的公有金鑰。指定已將私有金鑰 (.pem 檔案) 下載至其中的路徑。

```
ssh-keygen -y -f /path_to_key_pair/my-key-pair.pem
```

此命令會傳回公有金鑰，如下列範例所示。

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQClKsfkNkuSevGj3eYhCe53pcjqP3maAhDFcvBS7O6V
hz2ItxCih+PnDSUaw+WNQn/mZphTk/a/gU8jEzoOWbkM4yxyb/wB96xbiFveSFJuOp/d6RJhJOI0iBXr
lsLnBItntckiJ7FbtxJMXLvvwJryDUilBMTjYtwB+QhYXUMOzce5Pjz5/i8SeJtjnV3iAoG/cQk+0FzZ
qaeJAAHco+CY/5WrUBkrHmFJr6HcXkvJdWPkYQS3xqC0+FmUZofz221CBt5IMucxXPkX4rWi+z7wB3Rb
BQoQzd8v7yeb7OzlPnWOyN0qFU0XA246RA8QFYiCNYwI3f05p6KLxEXAMPLE
```

如果此命令失敗，請執行下列命令，來確保您已在私有金鑰對檔案中變更許可，以便只有您能夠檢視它。

```
chmod 400 key-pair-name.pem
```

From the instance metadata

您可以使用執行個體中繼資料服務第 2 版或執行個體中繼資料服務第 1 版，從執行個體中繼資料擷取公有金鑰。

> **Note**
>
> 如果您變更用於連線至執行個體的 key pair，Amazon EC2 不會更新執行個體中繼資料以顯示新的公開金鑰。執行個體中繼資料會繼續顯示啟動執行個體時指定之金鑰對的公有金鑰。

從執行個體中繼資料擷取公有金鑰

從執行個體使用下列其中一個命令。

IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-
aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-
data/public-keys/0/openssh-key
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key
```

範例輸出

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQClKsfkNkuSevGj3eYhCe53pcjqP3maAhDFcvBS7O6V
```

```
hz2ItxCih+PnDSUaw+WNQn/mZphTk/a/gU8jEzoOWbkM4yxyb/wB96xbiFveSFJuOp/d6RJhJOI0iBXr
lsLnBItntckiJ7FbtxJMXLvvwJryDUilBMTjYtwB+QhYXUMOzce5Pjz5/i8SeJtjnV3iAoG/cQk+0FzZ
qaeJAAHco+CY/5WrUBkrHmFJr6HcXkvJdWPkYQS3xqC0+FmUZofz221CBt5IMucxXPkX4rWi+z7wB3Rb
BQoQzd8v7yeb7OzlPnWOyN0qFU0XA246RA8QFYiCNYwI3f05p6KLxEXAMPLE key-pair-name
```

如需執行個體中繼資料的詳細資訊，請參閱訪問實例的實EC2例元數據。

From the instance

若您在啟動 Linux 執行個體時指定金鑰對，當執行個體第一次啟動時，公有金鑰內容會放置在 `~/.ssh/authorized_keys` 內項目的執行個體上。

從執行個體擷取公有金鑰材料

1. 連線到您的執行個體。
2. 在終端機視窗中，使用您最愛的文字編輯器 (例如 `authorized_keys` 或 vim) 開啟 nano 檔案。

   ```
   [ec2-user ~]$ nano ~/.ssh/authorized_keys
   ```

   `authorized_keys` 檔案開啟，會顯示公有金鑰，後面為金鑰對的名稱。以下是名為的 key pair 的範例項目 *`key-pair-name`*.

   ```
   ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQClKsfkNkuSevGj3eYhCe53pcjqP3maAhDFcvBS7O6V
   hz2ItxCih+PnDSUaw+WNQn/mZphTk/a/gU8jEzoOWbkM4yxyb/wB96xbiFveSFJuOp/d6RJhJOI0iBXr
   lsLnBItntckiJ7FbtxJMXLvvwJryDUilBMTjYtwB+QhYXUMOzce5Pjz5/i8SeJtjnV3iAoG/cQk+0FzZ
   qaeJAAHco+CY/5WrUBkrHmFJr6HcXkvJdWPkYQS3xqC0+FmUZofz221CBt5IMucxXPkX4rWi+z7wB3Rb
   BQoQzd8v7yeb7OzlPnWOyN0qFU0XA246RA8QFYiCNYwI3f05p6KLxEXAMPLE key-pair-name
   ```

From describe-key-pairs

使用 **`describe-key-pairs`**AWS CLI 命令擷取公有金鑰

使用 describe-key-pairs 命令，然後指定 `--key-names` 參數來識別公有金鑰。若要在輸出中包含公有金鑰資料，請指定 `--include-public-key` 參數。

```
aws ec2 describe-key-pairs --key-names key-pair-name --include-public-key
```

範例輸出：在輸出中，`PublicKey` 欄位會包含公有金鑰資料。

```
{
    "KeyPairs": [
        {
            "KeyPairId": "key-0123456789example",
            "KeyFingerprint":
 "1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f",
            "KeyName": "key-pair-name",
            "KeyType": "rsa",
            "Tags": [],
            "PublicKey": "ssh-ed25519
 AAAAC3NzaC1lZDI1NTE5AAAAIIj7azlDjVHAsSxgcpCRZ3oWnTm0nAFM64y9jd22ioI/ my-key-pair",
            "CreateTime": "2022-04-28T11:37:26.000Z"
        }
    ]
}
```

您也可以不要 `--key-names`，而是指定 `--key-pair-ids` 參數來識別公有金鑰。

```
aws ec2 describe-key-pairs --key-pair-ids key-0123456789example --include-public-key
```

Windows 執行個體

From the private key

從私有金鑰中擷取公有金鑰材料

在本機 Windows 電腦上，您可以使uTTYgen 用 P 取得金鑰組的公開 key pair。

啟動 P uTTYgen 並選擇載入。選取 `.ppk` 或 `.pem` 私有金鑰檔案。P uTTYgen 顯示公共密鑰下粘貼到打開SSH授權的 _keys 文件中的公鑰。您也可以選擇 Save public key (儲存公有金鑰)、指定檔案名稱、儲存檔案，然後開啟檔案，以檢視公有金鑰。

From the instance metadata

您可以使用執行個體中繼資料服務第 2 版或執行個體中繼資料服務第 1 版，從執行個體中繼資料擷取公有金鑰。

> **Note**
>
> 如果您變更用於連線至執行個體的 key pair，Amazon EC2 不會更新執行個體中繼資料以顯示新的公開金鑰。執行個體中繼資料會繼續顯示啟動執行個體時指定之金鑰對的公有金鑰。

從執行個體中繼資料擷取公有金鑰

從執行個體使用下列其中一個命令。

IMDSv2

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-
seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
 GET -Uri http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key
```

IMDSv1

```
PS C:\> Invoke-RestMethod -uri  http://169.254.169.254/latest/meta-data/public-
keys/0/openssh-key
```

範例輸出

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQClKsfkNkuSevGj3eYhCe53pcjqP3maAhDFcvBS7O6V
hz2ItxCih+PnDSUaw+WNQn/mZphTk/a/gU8jEzoOWbkM4yxyb/wB96xbiFveSFJuOp/d6RJhJOI0iBXr
lsLnBItntckiJ7FbtxJMXLvvwJryDUilBMTjYtwB+QhYXUMOzce5Pjz5/i8SeJtjnV3iAoG/cQk+0FzZ
qaeJAAHco+CY/5WrUBkrHmFJr6HcXkvJdWPkYQS3xqC0+FmUZofz221CBt5IMucxXPkX4rWi+z7wB3Rb
BQoQzd8v7yeb7OzlPnWOyN0qFU0XA246RA8QFYiCNYwI3f05p6KLxEXAMPLE key-pair-name
```

如需執行個體中繼資料的詳細資訊，請參閱訪問實例的實EC2例元數據。

From describe-key-pairs

使用 **describe-key-pairs**AWS CLI 命令擷取公有金鑰

使用 describe-key-pairs 命令，然後指定 `--key-names` 參數來識別公有金鑰。若要在輸出中包含公有金鑰資料，請指定 `--include-public-key` 參數。

```
aws ec2 describe-key-pairs --key-names key-pair-name --include-public-key
```

範例輸出：在輸出中，`PublicKey` 欄位會包含公有金鑰資料。

```
{
    "KeyPairs": [
        {
```

```
            "KeyPairId": "key-0123456789example",
            "KeyFingerprint":
 "1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f",
            "KeyName": "key-pair-name",
            "KeyType": "rsa",
            "Tags": [],
            "PublicKey": "ssh-ed25519
 AAAAC3NzaC1lZDI1NTE5AAAAIIj7azlDjVHAsSxgcpCRZ3oWnTm0nAFM64y9jd22ioI/ my-key-pair",
            "CreateTime": "2022-04-28T11:37:26.000Z"
        }
    ]
}
```

您也可以不要 `--key-names`，而是指定 `--key-pair-ids` 參數來識別公有金鑰。

```
aws ec2 describe-key-pairs --key-pair-ids key-0123456789example --include-public-key
```

識別啟動時指定的公有金鑰

如果您在啟動執行個體時指定了公有金鑰，執行個體會記錄公有金鑰名稱。

識別啟動時指定的公有金鑰

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選取 Instances (執行個體)，然後選取您的執行個體。
3. 在詳細資訊標籤中，執行個體詳細資訊下的啟動時指派的金鑰對欄位會顯示您在啟動執行個體時指定的公有金鑰名稱。

> **Note**
>
> 即使您變更執行個體上的公有金鑰，或新增公有金鑰，啟動時指派的金鑰對欄位的值也不會變更。

刪除您的金鑰對

您可以刪除 key pair，這將刪除存儲在 Amazon 中的公鑰EC2。刪除 key pair 並不會刪除相符的私密金鑰。

當您使用下列方法刪除公開金鑰時，您只會在建立或匯入金鑰組EC2時刪除儲存在 Amazon 中的公開 key pair。無論在啟動執行個體時或啟動之後，移除公有金鑰都不會從您新增金鑰的任何執行個體中移除公有金鑰。它也不會刪除您本機電腦上的私有金鑰。只要您仍然擁有私鑰 (`.pem`) 檔案，就可以繼續連線到使用已從 Amazon EC2 刪除的公開金鑰啟動的執行個體。

> **Important**
>
> 如果使用 Auto Scaling 群組 (例如，在 Elastic Beanstalk 環境中)，請確保在相關聯的啟動範本或啟動組態中並未指定您要刪除的公有金鑰。如果 Amazon EC2 Auto Scaling 偵測到運作狀態不佳的執行個體，就會啟動替換執行個體 不過，如果找不到公有金鑰，執行個體啟動就會失敗。如需詳細資訊，請參閱 Amazon EC2 自動擴展使用者指南中的啟動範本。

Console

在 Amazon 上刪除您的公鑰 EC2

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Key Pairs (金鑰對)。
3. 選取要刪除的金鑰對，然後選取 Actions (動作)、Delete (刪除)。
4. 在確認欄位中，輸入 `Delete`，然後選擇 Delete (刪除)。

AWS CLI

在 Amazon 上刪除您的公鑰 EC2

使用 delete-key-pair AWS CLI 命令。

PowerShell

在 Amazon 上刪除您的公鑰 EC2

使用 Remove-EC2KeyPair AWS Tools for Windows PowerShell 命令。

在 Linux 執行個體上新增或取代公開金鑰

如果遺失私密金鑰，您將無法存取任何使用 key pair 的執行個體。如需使用與您在啟動時指定之金鑰組不同的 key pair 連線至執行個體的詳細資訊，請參閱我遺失私密金鑰。

當您啟動執行個體時，您可指定金鑰對。若您在啟動時指定金鑰對，當執行個體第一次啟動時，公有金鑰材料會放置在 `~/.ssh/authorized_keys` 內項目的 Linux 執行個體上。

您可以變更用來存取執行個體預設系統帳戶的金鑰對，方法是在執行個體上新增公有金鑰，或是取代執行個體上的公有金鑰 (刪除現有公有金鑰並在執行個體上新增一個)。您也可以從執行個體中移除所有公有金鑰。若要新增或取代金鑰對，您必須能夠連線至您的執行個體。

您可以基於下列原因新增或取代 key pair：

- 如果組織中的使用者需要使用不同的金鑰對來存取系統使用者，您可以將該共有金鑰新增到執行個體。
- 如果某些人擁有私有金鑰的複本 (`.pem` 檔案)，而您希望阻止他們連線到您的執行個體 (例如，如果他們已離開您的組織)，則您可刪除執行個體上的公有金鑰，並以新的金鑰取代。
- 如果您是AMI從執行個體建立 Linux，公開金鑰材料會從執行個體複製到AMI. 如果您從啟動執行個體 AMI，則新執行個體會包含來自原始執行個體的公開金鑰。若要防止擁有私密金鑰的使用者連線至新的執行個體，您可以在建立之前從原始執行個體移除公開金鑰AMI。

使用下列程序修改預設使用者的 key pair，例如`ec2-user`。如需將使用者新增至執行個體的相關資訊，請參閱執行個體上作業系統的說明文件。

新增或替換金鑰對

1. 使用 Amazon EC2 主控台或第三方工具建立新的 key pair。
2. 從新的金鑰對擷取公有金鑰。如需詳細資訊，請參閱擷取公有金鑰資料。
3. 使用現有的私有金鑰，連線到執行個體。
4. 使用您自選的文字編輯器，在執行個體上開啟 `.ssh/authorized_keys` 檔案。貼上新金鑰對的公有金鑰資訊 (在現有公有金鑰資訊的下方)。儲存檔案。
5. 中斷與執行個體的連線，然後測試是否能使用新的私有金鑰檔案來連線到執行個體。
6. (選用) 如果要替換現有的金鑰對，請連線到執行個體，然後從 `.ssh/authorized_keys` 檔案，刪除原始金鑰對的公有金鑰資訊。

> **Important**
>
> 如果使用 Auto Scaling 群組，請確保在啟動範本或啟動組態中並未指定您要取代的金鑰對。如果 Amazon EC2 Auto Scaling 偵測到運作狀態不佳的執行個體，就會啟動替換執行個體 不

過，如果找不到金鑰對，執行個體啟動就會失敗。如需詳細資訊，請參閱 Amazon EC2 自動擴展使用者指南中的啟動範本。

若要從執行個體中移除公有金鑰

1. 連線到您的執行個體。
2. 使用您自選的文字編輯器，在執行個體上開啟 `.ssh/authorized_keys` 檔案。刪除公有金鑰資訊，然後儲存檔案。

> **Warning**
> 從執行個體移除所有公開金鑰並中斷與執行個體的連線後，除非AMI提供另一種登入方式，否則您無法再次連線至該執行個體。

驗證金鑰對的指紋

若要驗證 key pair 的指紋，請將 Amazon EC2 主控台「金鑰配對」頁面上顯示的指紋或describe-key-pairs命令傳回的指紋與您使用本機電腦上的私密金鑰產生的指紋進行比較。這些指紋應該相符。

當 Amazon EC2 計算指紋時，Amazon EC2 可能會在帶有=字符的指紋上附加填充。其他工具，例如 ssh-keygen，可能會忽略此填補。

如果您嘗試驗證 Linux EC2 執行個體的指紋，而不是 key pair 的指紋，請參閱取得執行個體指紋。

如何計算指紋

Amazon EC2 使用不同的哈希函數來計算RSA和ED25519密鑰對的指紋。此外，對於RSA密鑰對，Amazon EC2 根據密 key pair 是由 Amazon 創建還是導入 AmazonEC2，使用不同的哈希函數計算指紋不同EC2。

下表列出用於計算 Amazon 建立並匯入至 Amazon EC2 的指紋RSA和ED25519金鑰配對的雜湊函數 EC2。

(Linux 實例) 用於計算指紋的雜湊函數

金鑰對來源	RSA金鑰配對 (視窗及 Linux)	ED25519金鑰配對
由 Amazon 創建 EC2	SHA-1	SHA-256

金鑰對來源	RSA金鑰配對 (視窗及 Linux)	ED25519金鑰配對
進口到 Amazon EC2	MD5[1]	SHA-256

[1] 如果您將公開RSA金鑰匯入 AmazonEC2，則會使用MD5雜湊函數計算指紋。無論您是如何建立 key pair，例如使用第三方工具或從使用 Amazon 建立的現有私密金鑰產生新的公開金鑰，都是如此EC2。

在不同區域中使用相同的金鑰對

如果您打算使用相同的 key pair 連線至不同的執行個體 AWS 區域，則必須將公開金鑰匯入您要使用該金鑰的所有區域。如果您使EC2用 Amazon 建立 key pair，您可擷取公有金鑰資料以將公開金鑰匯入其他區域。

> **Note**
>
> - 如果您使用 Amazon 建立金 RSA key pairEC2，然後從 Amazon EC2 私密金鑰產生公開金鑰，則匯入的公開金鑰將具有與原始公開金鑰不同的指紋。這是因為使用 Amazon 建立的原始金RSA鑰指紋EC2是使用 SHA -1 雜湊函數計算的，而匯入RSA金鑰的指紋則是使用MD5雜湊函數計算的。
> - 對於ED25519密鑰對，無論指紋是由 Amazon 創建還是導入 AmazonEC2，都將是相同的EC2，因為使用相同的 SHA -256 哈希函數來計算指紋。

從私有金鑰產生指紋

使用下列命令之一從本機電腦上的私有金鑰產生指紋。

如果您使用的是 Windows 本機電腦，您可以使用適用於 Linux 的 Windows 子系統來執行下列命令 (WSL)。使用視窗 10 安裝指南中的指示來安裝WSL和 Linux 發行版本。指示中的範例是安裝 Linux 的 Ubuntu 發行版本，但您可以安裝任何發行版本。您將會收到提示，要求您重新啟動電腦以讓變更生效。

- 如果您使用 Amazon 創建了 key pair EC2

  使用「開啟」SSL 工具產生指紋，如下列範例所示。

  對於RSA金鑰配對：

```
openssl pkcs8 -in path_to_private_key -inform PEM -outform DER -topk8 -nocrypt |
 openssl sha1 -c
```

(Linux 執行個體) 對於ED25519金鑰配對：

```
ssh-keygen -l -f path_to_private_key
```

- （僅限RSA密鑰對）如果您將公鑰導入 Amazon EC2

  無論 key pair 的建立方式為何，您都可以遵循此程序，例如使用第三方工具或從使用 Amazon 建立的現有私密金鑰產生新的公開金鑰 EC2

  使用「開啟」SSL 工具產生指紋，如下列範例所示。

```
openssl rsa -in path_to_private_key -pubout -outform DER | openssl md5 -c
```

- 如果您使用 Open SSH 7.8 或更高版本創建了一個打開密 SSH key pair，並將公鑰導入 Amazon EC2

  如下列範例所示，使用 ssh-keygen 來產生指紋。

  對於RSA金鑰配對：

```
ssh-keygen -ef path_to_private_key -m PEM | openssl rsa -RSAPublicKey_in -outform DER
 | openssl md5 -c
```

  (Linux 執行個體) 對於ED25519金鑰配對：

```
ssh-keygen -l -f path_to_private_key
```

適用於EC2執行個體的 Amazon EC2 安全群組

安全性群組可做為EC2執行個體的虛擬防火牆，以控制傳入和傳出流量。傳入規則會控制傳入至您的執行個體的流量，以及傳出規則會控制從您的執行個體傳出的流量。啟動執行個體時，您會為其指派一個或多個安全群組。如果您未指定安全群組，Amazon EC2 會使用VPC. 在您啟動執行個體之後，您可變更其安全群組。

安全是 AWS 與您之間共同承擔的責任。如需詳細資訊，請參閱Amazon 的安全性 EC2。 AWS 提供安全群組做為保護執行個體安全性的工具之一，因此您需要設定這些群組以符合安全性需求。若您有安全群組無法完全滿足的需求，您可以在使用安全群組的同時在您的任何執行個體上維持您自己的防火牆。

定價

使用安全群組無需額外收費。

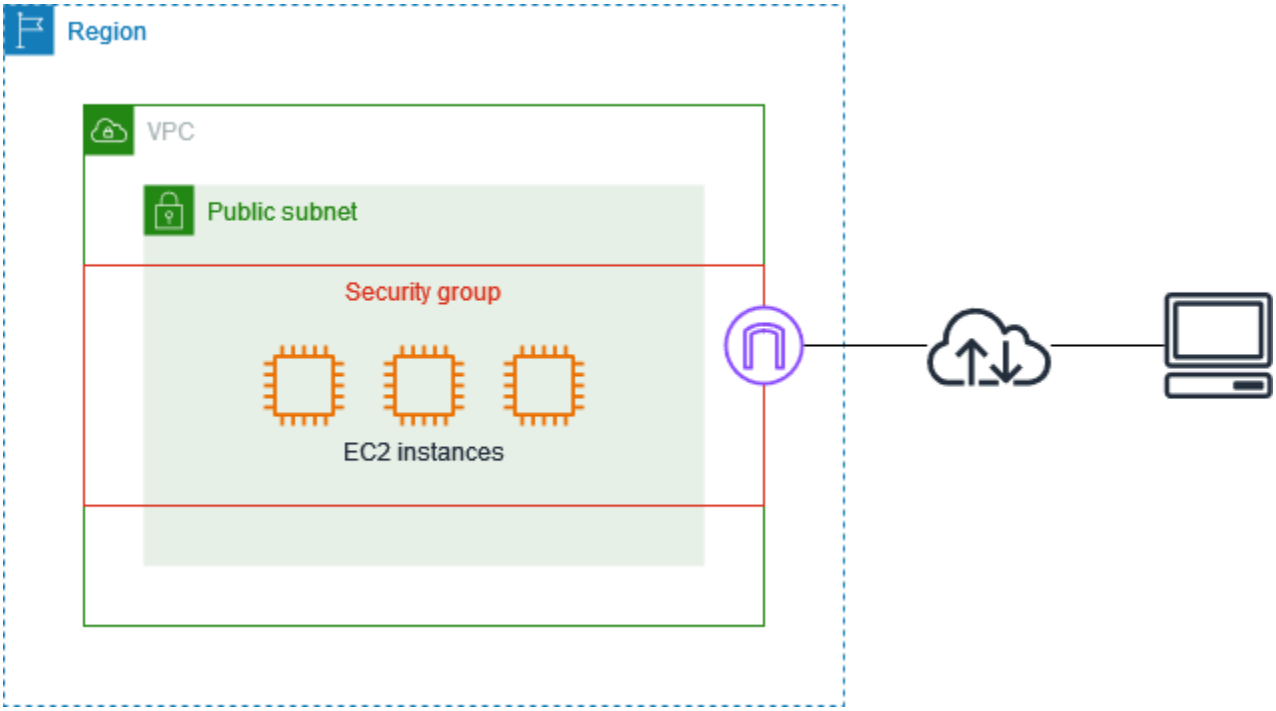
目錄

- 概要
- 為您的 Amazon EC2 執行個體建立安全群組
- 變更 Amazon EC2 執行個體的安全群組
- 刪除 Amazon 安EC2全組
- Amazon EC2 安全組連接跟踪
- 不同使用案例的安全群組規則

概要

安全性群組只能在建立安全VPC性群組的中使用。您可以將每個執行個體與多個安全群組建立關聯，也可以將每個安全群組與多個執行個體相關聯。您在各個安全群組新增規則，允許流量往返於建立關聯的執行個體。您可隨時修改安全群組規則。新的、修改過的規則會自動套用至與安全群組建立關聯的所有執行個體。當 Amazon EC2 決定是否允許流量到達執行個體時，它會評估與執行個體相關聯的所有安全群組中的所有規則。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的安全群組規則。

下圖顯示VPC具有子網路、網際網路閘道和安全群組的。子網路包含EC2執行個體。安全群組與執行個體相關聯。僅有安全群組規則允許的流量才能到達執行個體。例如，如果安全性群組包含允許來自網路SSH流量的規則，則您可以使用從電腦連線到執行個體SSH。如果安全性群組包含允許來自與其相關聯資源的所有流量的規則，則每個執行個體都可以接收從其他執行個體傳送的任何流量。



安全群組是具有狀態的群組—若您從執行個體傳送請求，該請求的回應流量將允許流入，與對內安全群組規則無關。此外，不論輸出規則為何，允許對允許的入站流量的回應流出。如需詳細資訊，請參閱連線追蹤。

為您的 Amazon EC2 執行個體建立安全群組

安全群組就像是防火牆，用於關聯的執行個體，可在執行個體層級控制傳入及傳出流量。您可以將規則新增至安全群組，以便使用 SSH (Linux 執行個體) 或 RDP (Windows 執行個體) 連線至執行個體。您也可以新增允許用戶端流量的規則，例如，以HTTP及目的地到 Web 伺服器的HTTPS流量。

啟動執行個體時，您可以將安全群組與執行個體建立關聯。當您從相關聯的安全性群組新增或移除規則時，這些變更會自動套用至您與安全性群組相關聯的所有執行個體。

啟動執行個體後，您可以建立其他安全性群組的關聯。如需詳細資訊，請參閱變更 Amazon EC2 執行個體的安全群組。

您可以在建立安全性群組時新增輸入和輸出安全性群組規則，也可以稍後新增這些規則。如需詳細資訊，請參閱設定安全群組規則。如需可新增至安全性群組的規則範例，請參閱不同使用案例的安全群組規則。

考量事項

- 根據預設，新的安全群組一開始只有允許流量離開資源的傳出規則。您必須新增規則啟用任何傳入流量，或是限制傳出流量。

- 為允許SSH或RDP存取執行個體的規則設定來源時，請勿允許從任何地方存取，因為這樣可以從網際網路上的所有 IP 位址存取您的執行個體。通常在測試環境中短暫進行此操作是沒有問題的，但用在生產環境則不安全。
- 如果特定連接埠有多個規則，Amazon 會EC2套用最寬鬆的規則。例如，如果您有一個規則允許從 IP 位址 203.0.113.1 存取TCP連接埠 22 (SSH)，以及另一個允許從任何地方存取TCP連接埠 22 的規則，則每個人都可以存取連接埠 22。TCP
- 您可以將多個安全群組與執行個體建立關聯。因此，執行個體可以有數百個適用的規則。這可能會在您存取執行個體時產生問題。但建議您盡可能緊縮您的規則。
- 當您將安全群組指定為規則的來源或目標時，規則會影響所有與安全群組相關聯的執行個體。傳入流量會根據與來源安全群組相關聯之執行個體的私有 IP 地址允許 (而非公有 IP 或彈性 IP 地址)。如需有關 IP 地址的詳細資訊，請參閱 Amazon EC2 實例 IP 定址。
- Amazon 默認情況下EC2阻止端口 25 上的流量。如需詳細資訊，請參閱使用通訊埠 25 傳送的電子郵件限制。

使用主控台建立安全群組

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Security Groups (安全群組)。
3. 選擇 Create Security Group (建立安全群組)。
4. 輸入安全性群組的描述性名稱和簡短描述。您無法在建立安全性群組之後變更其名稱和說明。
5. 對於 VPC，請選擇您要VPC在其中執行 Amazon 執行個體的EC2執行個體。
6. (選擇性) 若要新增輸入規則，請選擇輸入規則。針對每個規則，選擇 [新增規則]，然後指定通訊協定、連接埠和來源。例如，若要允許SSH流量，請選擇 [類型]，並SSH為 [來源] 指定電腦或網路的公用IPv4位址。
7. (選擇性) 若要新增輸出規則，請選擇輸出規則。針對每個規則，選擇 [新增規則]，然後指定通訊協定、連接埠和目的地。否則，您可以保留允許所有輸出流量的預設規則。
8. (選用) 若要新增標籤，請選擇 Add new tag (新增標籤)，然後輸入標籤金鑰和值。
9. 選擇 Create Security Group (建立安全群組)。

使用命令列建立安全群組

- create-security-group (AWS CLI)
- New-EC2SecurityGroup (AWS Tools for Windows PowerShell)

變更 Amazon EC2 執行個體的安全群組

您可以在啟動 Amazon EC2 執行個體時為它們指定安全群組。啟動執行個體後，您可以新增或移除安全性群組。您也可以隨時新增、移除或編輯關聯安全性群組的安全性群組規則。

安全群組與網路介面關聯。新增或移除安全性群組會變更與主要網路介面相關聯的安全性群組。您也可以變更與任何次要網路介面關聯的安全性群組。如需詳細資訊，請參閱 修改網路介面屬性。

任務

- 新增或移除安全性群組
- 設定安全群組規則

新增或移除安全性群組

啟動執行個體之後，您可以從相關聯的安全群組清單中新增或移除安全性群組。當您將多個安全群組與執行個體建立關聯時，每個安全群組的規則都會有效彙總並建立一組規則。Amazon EC2 使用這組規則來確定是否允許流量。

要求

- 執行個體必須處於 `running` 或 `stopped` 狀態。
- 安全性群組是特定於VPC. 您可以將安全群組與一或多個執行個體建立關聯。

使用主控台變更執行個體的安全群組

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後選取 動作、安全性、變更安全群組。
4. 對於關聯的安全群組，請從清單中選取安全群組，然後選取新增安全群組。

   若要移除關聯的安全群組，請為該安全群組選擇移除。
5. 選擇 Save (儲存)。

使用命令列變更執行個體的安全群組

- modify-instance-attribute (AWS CLI)
- Edit-EC2InstanceAttribute (AWS Tools for Windows PowerShell)

設定安全群組規則

建立安全性群組之後，您可以新增、更新及刪除其安全性群組規則。當您新增、更新或刪除規則時，變更會自動套用至與安全性群組相關聯的資源。

如需可新增至安全性群組的規則範例，請參閱不同使用案例的安全群組規則。

來源和目的地

您可以將下列項目指定為輸入規則的來源或輸出規則的目的地。

- 自訂 — IPv4 CIDR 區IPv6CIDR塊和封鎖、其他安全性群組或首碼清單。
- 任何地方-IPv4 — 在 0.0.0.0/0 塊 IPv4CIDR.
- 任何地方 IPv6- 的:: /0 塊 IPv6CIDR.
- 我的 IP — 本地計算機的公共地IPv4址。

> **Warning**
>
> 如果您為連接埠 22 (SSH) 或 3389 (RDP) 新增輸入規則，強烈建議您僅授權需要存取執行個體的特定 IP 位址或位址範圍。如果您選擇 Anywhere-IPv4，則允許來自所有IPv4地址的流量使用指定的通訊協定存取您的執行個體。如果您選擇 Anywhere-IPv6，則允許來自所有IPv6地址的流量使用指定的通訊協定存取您的執行個體。

使用主控台設定安全群組規則

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Security Groups (安全群組)。
3. 選取安全群組。
4. 若要編輯入站規則，請從 [動作] 或 [輸入規則] 索引標籤選擇 [編輯輸入規則]。
   a. 若要新增規則，請選擇 [新增規則]，然後輸入規則的類型、通訊協定、連接埠和來源。

      如果類型為TCP或UDP，則必須輸入要允許的通訊埠範圍。對於自定義ICMP，您必須從協議中選擇ICMP類型名稱，如果適用，則從端口範圍中選擇代碼名稱。如果是任何其他類型，則系統會自動為您設定通訊協定和連接埠範圍。
   b. 若要更新規則，請視需要變更其通訊協定、說明和來源。不過，您無法變更來源類型。例如，如果來源是IPv4CIDR區塊，則無法指定IPv6CIDR封鎖、首碼清單或安全性群組。

c. 若要刪除規則，請選擇其「刪除」按鈕。

5. 若要編輯輸出規則，請從 [動作] 或 [輸出規則] 索引標籤選擇 [編輯輸出規則]。

   a. 若要新增規則，請選擇 [新增規則]，然後輸入規則的類型、通訊協定、連接埠和目的地。您也可以輸入選擇性描述。

      如果類型為TCP或UDP，則必須輸入要允許的通訊埠範圍。對於自定義ICMP，您必須從協議中選擇ICMP類型名稱，如果適用，則從端口範圍中選擇代碼名稱。如果是任何其他類型，則系統會自動為您設定通訊協定和連接埠範圍。

   b. 若要更新規則，請視需要變更其通訊協定、說明和來源。不過，您無法變更來源類型。例如，如果來源是IPv4CIDR區塊，則無法指定IPv6CIDR封鎖、首碼清單或安全性群組。

   c. 若要刪除規則，請選擇其「刪除」按鈕。

6. 選擇儲存規則。

使用設定安全群組規則 AWS CLI

- 新增 — 使用authorize-security-group-ingress和authorize-security-group-egress指令。
- 移除 — 使用revoke-security-group-ingress和revoke-security-group-egress指令。
- 修改 — 使用modify-security-group-rules、說明輸入和update-security-group-rule說明輸出命令。update-security-group-rule

使用 Windows 的工具來設定安全性群組規則 PowerShell

- 添加-使用Grant-EC2SecurityGroupIngress和Grant-EC2SecurityGroupEgress。
- 移除 — 使用Revoke-EC2SecurityGroupIngress和Revoke-EC2SecurityGroupEgress。
- 修改 — 使用Edit-EC2SecurityGroupRuleUpdate-EC2SecurityGroupRuleIngressDescription、和Update-EC2SecurityGroupRuleEgressDescription。

刪除 Amazon 安EC2全組

完成建立用於 Amazon EC2 執行個體使用的安全群組後，您可以將其刪除。

要求

- 安全性群組無法與執行個體或網路介面建立關聯。
- 另一個安全性群組中的規則無法參考安全性群組。

使用主控台刪除安全群組

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. (選擇性) 若要確認您的安全群組與執行個體沒有關聯，請執行下列動作：
   a. 在導覽窗格中，選擇 Security Groups (安全群組)。
   b. 複製要刪除的安全性群組 ID。
   c. 在導覽窗格中，選擇 Instances (執行個體)。
   d. 在搜索欄中，添加安全組IDs等於過濾器並粘貼安全組的 ID。如果沒有結果，則表示安全性群組不會與執行個體相關聯。否則，您必須先取消安全性群組的關聯，然後才能刪除它。
3. 在導覽窗格中，選擇 Security Groups (安全群組)。
4. 選取安全群組，然後選擇動作、刪除安全群組。
5. 如果您選取多個安全性群組，系統會提示您進行確認。如果某些安全性群組無法刪除，我們會顯示每個安全群組的狀態，指出是否將其刪除。若要確認刪除，請輸入刪除。
6. 選擇 刪除 。

使用命令列刪除安全群組

- delete-security-group (AWS CLI)
- Remove-EC2SecurityGroup (AWS Tools for Windows PowerShell)

Amazon EC2 安全組連接跟踪

您的安全群組使用連線追蹤來追蹤流入和流出執行個體流量的資訊。規則會根據流量的連線狀態套用，以判斷該允許或拒絕該流量。使用這種方法，安全群組是可設定狀態的。這表示傳入流量的回應可從執行個體傳出，與對外安全群組規則無關，反之亦然。

舉例來說，假設您從家用電腦啟動 netcat 或類似執行個體之類的命令，而輸入安全性群組規則允許ICMP流量。會追蹤連線的相關資訊 (包括連接埠資訊)。來自命令執行個體的回應流量不會被追蹤為新的要求，而是以建立的連線追蹤，而是被允許流出執行個體，即使您的輸出安全性群組規則限制輸出ICMP流量也是如此。

對於TCP、UDP或以外的通訊協定ICMP，只會追蹤 IP 位址和通訊協定編號。若您的執行個體傳送流量到另一個主機，且該主機也在 600 秒內將相同類型的流量傳送至您的執行個體，無論傳入安全群組規則為何，執行個體的安全群組都會接受該流量。安全群組會接受該流量，因為將其視為原始流量的回應流量。

變更安全群組規則時，其跟蹤的連線不會立即中斷。安全群組將繼續允許封包，直到現有連線逾時為止。為了確保流量立即中斷，或者所有流量都受到防火牆規則的約束，無論追蹤狀態為何，您可以將網路用ACL於子網路。網絡ACLs是無狀態的，因此不會自動允許響應流量。新增封鎖任一方向流量的網路ACL會中斷現有的連線。如需詳細資訊，請參閱 Amazon VPC 使用者指南ACLs中的網路。

> **Note**
>
> 安全群組對進出 Route 53 解析器的DNS流量沒有影響，有時也稱為「VPC+2 IP 位址」(請參閱什麼是 Amazon Route 53 Resolver 程式？ 在 Amazon Route 53 開發人員指南中）或 'AmazonProvidedDNS'（請參閱 Amazon Virtual Private Cloud 用戶指南中的使用DHCP選項集）。如果您希望透過 Route 53 解析器篩選DNS請求，可以啟用 Route 53 解析器DNS防火牆 (請參閱 Amazon Route 53 開發人員指南中的 Route 53 解析器DNS防火牆)。

未追蹤的連線

並非所有的流量流程都會追蹤。如果安全群組規則允許TCP所有UDP流量 (0.0.0.0/0 或:: /0)，並且在另一個方向上有一個允許所有回應流量的對應規則 (0.0.0.0/0 或:: /0) 任何連接埠 (0-65535)，則不會追蹤該流量流量，除非它是自動追蹤連線的一部分。接著會依據允許回應流量的傳入或傳出規則允許未追蹤流的回應流量，而非依據追蹤資訊。

若啟用未受追蹤之流量流程的規則遭到移除或修改，則該流程會立即中斷。例如，如果您有一個開放式 (0.0.0.0/0) 輸出規則，並移除允許執行個體所有 (0.0.0.0/0) 輸入 (連接TCP埠 22) 流量的規則 SSH (或修改該規則，使其不再允許連線)，您現有SSH的執行個體連線會立即中斷。該連線之前未被追蹤，因此該變更會中斷連線。另一方面，如果您有一個較窄的輸入規則，最初允許SSH連接（表示已跟踪連接），但將該規則更改為不再允許來自當前SSH客戶端的地址的新連接，則現有SSH連接不會中斷，因為它已被跟踪。

自動追蹤連線

即使安全性群組組態不需要追蹤，透過下列項目建立的連線也會自動追蹤：

- 輸出限定網際網路閘道
- Global Accelerator 加速器
- NAT閘道
- Network Firewall 防火墻端點
- Network Load Balancer
- AWS PrivateLink (介面VPC端點)

- AWS Lambda (超平面彈性網路介面)

連線追蹤限額

Amazon EC2 定義了每個執行個體可追蹤的最大連線數。達到最大值之後，任何傳送或接收的封包都會捨棄，因為無法建立新的連線。發生這種情況時，傳送及接收封包的應用程式無法正確通訊。使用 `conntrack_allowance_available` 網路效能指標來判斷該執行個體類型仍可使用的追蹤連線數目。

若要判斷是否因為執行個體的網路流量超過可追蹤的連線數目上限而捨棄封包，請使用 `conntrack_allowance_exceeded` 網路效能指標。如需詳細資訊，請參閱監控EC2執行個體ENA設定的網路效能。

使用 Elastic Load Balancing 時，如果超過每個執行個體可追蹤的連線數目上限，建議您使用負載平衡器調整執行個體數目，或使用負載平衡器調整執行個體大小。

連線追蹤效能考量

非對稱路由是流量透過一個網路介面進入執行個體，並透過不同的網路介面離開，可降低執行個體在追蹤流程時所能達到的尖峰效能。

若要在為您的安全性群組啟用連線追蹤時維持最佳效能，我們建議您進行下列組態：

- 如果可能，請避免使用非對稱的路由拓撲。
- 而不是使用安全群組進行篩選，而是使用網路ACLs。
- 如果您必須將安全群組與連線追蹤搭配使用，請盡可能設定最短的連線逾時。

如需 Nitro 系統效能調整的詳細資訊，請參閱對於性能調整的硝基系統考量。

閒置連線追蹤逾時

安全群組會追蹤每個建立的連線，以確保傳回封包如預期交付。每個執行個體可追蹤的連線數目上限。處於閒置狀態的連線可能會導致連線追蹤耗盡，並造成不追蹤連線及捨棄封包。您可以在彈性網路介面上設定閒置連線追蹤的逾時。

> **Note**
>
> 此功能僅適用於在 AWS Nitro 系統上構建的實例。

可設定的逾時有三種：

- TCP已建立逾時：處於已建立狀態的閒置TCP連線逾時 (以秒為單位)。最小值：60 秒。最大值：432000 秒 (5 天)。預設：432000 秒。建議使用：432000 秒以下。
- UDPtimeout：閒置UDP流量僅在單一方向或單一要求-回應交易中看到流量的逾時 (以秒為單位)。最小值：30 秒。最大值：60 秒。預設：30 秒。
- UDP串流逾時：分類為已看到多個要求-回應交易之串UDP流的閒置流程逾時 (以秒為單位)。最小值：60 秒。最大值：180 秒 (3 分鐘) 預設值：180 秒。

您可修改下列任何情況的預設逾時：

- 如果您使用 Amazon EC2 網路效能指標監控追蹤的連線，conntrack_allowance_EXPUT 和 conntrack_allowance_可用指標可讓您監控丟棄的封包和追蹤連線使用率，以透過向上或向外擴充動作來主動管理EC2執行個體容量，以協助在丟棄封包之前滿足網路連線需求。如果您在EC2執行個體上觀察 conntrack_allowance_超過下降，您可能會受益於設定較低的TCP建立逾時，以解決由不正確的用戶端或網路中間方塊造成的過時TCP/UDP工作階段。
- 一般而言，負載平衡器或防火牆會在 60 到 90 分鐘的範圍內TCP建立閒置逾時。如果您執行的工作負載預期會處理來自網路防火牆等應用裝置的大量連線 (大於 100k)，建議您在網EC2路介面上設定類似逾時。
- 如果您執行的工作負載使用非對稱路由拓撲，建議您將 [已建TCP立的閒置逾時] 設定為 60 秒。
- 如果您執行的工作負載具有大量連線 (例如DNS、、SIP、Syslog SNMP、Radius 和其他主要用於服務請求的服務)，則UDP將 'UDP-stream' 逾時設定為 60s 可為現有容量提供更高的規模/效能，並防止灰色故障。
- 對於TCP/UDP透過網路負載平衡器 (NLBs) 和彈性負載平衡器 (ELB) 連線，會追蹤所有連線。TCP流程的閒置逾時值為 350 秒，UDP流量為 120 秒，且因介面層級逾時值而異。您可能想要在網路介面層級設定逾時，讓逾時的彈性比ELB/NLB的預設值更大。

執行下列操作時，您可以選擇設定連線追蹤逾時：

- 建立網路介面
- 修改網路介面屬性
- 啟動EC2執行個體
- 建立EC2執行個體啟動範本

範例

在下列範例中，安全性群組具有允許TCP和ICMP流量的輸入規則，以及允許所有輸出流量的輸出規則。

傳入

通訊協定類型	連接埠號碼	來源
TCP	SSH	203.0.113.1/32
TCP	八HTTP十	0.0.0.0/0
TCP	八HTTP十	::/0
ICMP	全部	0.0.0.0/0

傳出

通訊協定類型	連接埠號碼	目的地
全部	全部	0.0.0.0/0
全部	全部	::/0

對於執行個體或網路介面的直接網路連線，追蹤行為如下：

- 會追蹤連接埠 22 (SSH) 上的輸入和輸出TCP流量，因為輸入規則僅允許來自 203.0.113.1/32 的流量，而不是所有 IP 位址 (0.0.0/0) 的流量。
- 不會追蹤TCP通訊埠 80 (HTTP) 上的輸入和輸出流量，因為輸入和輸出規則允許來自所有 IP 位址的流量。
- ICMP流量始終被跟踪。

如果您移除IPv4流量的輸出規則，則會追蹤所有入站和輸出IPv4流量，包括通訊埠 80 (HTTP) 上的流量。如果您移除IPv6流量的輸出規則，這同樣適用於IPv6流量。

不同使用案例的安全群組規則

您可以建立安全群組並新增可反映與該安全群組相關聯之執行個體角色的規則。例如，設定為 Web 伺服器的執行個體需要允許輸入HTTP和HTTPS存取的安全群組規則。同樣地，資料庫執行個體需要允許存取資料庫類型的規則，例如透過連接埠 3306 存取 MySQL.

以下為您可以針對特定類型存取，為安全群組新增之規則類型的範例。

範例

- Web 伺服器規則
- 資料庫伺服器規則
- 從您電腦連線到執行個體的規則
- 從執行個體使用相同安全群組連線到執行個體的規則
- 平/規則 ICMP
- DNS伺服器規則
- Amazon EFS 規則
- Elastic Load Balancing 規則

Web 伺服器規則

以下輸入規則允許HTTP和從任何 IP 地址HTTPS訪問。如果您VPC已啟用IPv6，您可以新增規則來控制來自IPv6地址的入站HTTP和HTTPS流量。

通訊協定類型	通訊協定號碼	連接埠	來源 IP	備註
TCP	6	八HTTP十	0.0.0.0/0	允許從任何IPv4位址進入HTTP存取
TCP	6	HTTPS	0.0.0.0/0	允許從任何IPv4位址進入HTTPS存取
TCP	6	八HTTP十	::/0	允許從任何IPv6位址進入HTTP存取
TCP	6	HTTPS	::/0	允許從任何IPv6位址進入HTTPS存取

資料庫伺服器規則

以下對內規則是您可以為資料庫存取新增之規則的範例，取決於您在執行個體上執行的資料庫類型為何。如需 Amazon RDS 執行個體的詳細資訊，請參閱 Amazon RDS 使用者指南。

針對來源 IP，請指定下列其中一項：

- 區域網路中的特定 IP 位址或 IP 位址範圍 (以CIDR區塊標示法表示)
- 存取資料庫之執行個體群組的安全群組 ID

通訊協定類型	通訊協定號碼	連接埠	備註
TCP	6	女士 SQL	用於訪問 Microsoft SQL 服務器數據庫的默認端口，例如在 Amazon RDS 實例上
TCP	6	Aur MYSQL ora	用於存取 My SQL 或 Aurora 資料庫的預設連接埠，例如在 Amazon RDS 執行個體上
TCP	6	5439 (Redshift)	存取 Amazon Redshift 叢集資料庫的預設連接埠。
TCP	6	5432 (波斯特格雷SQL)	用於訪問 Postgre SQL 數據庫的默認端口，例如在 Amazon RDS 實例上
TCP	6	1521 (Oracle)	用於存取 Oracle 資料庫的預設連接埠，例如在 Amazon RDS 執行個體上

您可以選擇性限制來自資料庫伺服器的輸出流量。例如，您可能想允許存取網際網路以取得軟體更新，但限制所有其他類型的流量。您必須先移除允許所有對外流量的預設對外規則。

通訊協定類型	通訊協定號碼	連接埠	目標 IP	備註
TCP	6	八HTTP十	0.0.0.0/0	允許對任何IPv4位址的對外HTTP存取

通訊協定類型	通訊協定號碼	連接埠	目標 IP	備註
TCP	6	HTTPS	0.0.0.0/0	允許對任何IPv4位址的對外HTTPS存取
TCP	6	八HTTP十	::/0	(VPC僅限IPv6已啟用) 允許對外HTTP存取任何IPv6位址
TCP	6	HTTPS	::/0	(VPC僅限IPv6已啟用) 允許對外HTTPS存取任何IPv6位址

從您電腦連線到執行個體的規則

若要連線至您的執行個體，您的安全群組必須具有允許SSH存取 (針對 Linux 執行個體) 或RDP存取權 (適用於 Windows 執行個體) 的輸入規則。

通訊協定類型	通訊協定號碼	連接埠	來源 IP
TCP	6	SSH	您電腦的公用位IPv4址，或區域網路中的 IP 位址範圍。如果您的執行VPC個體已啟用，IPv6且執行個體有IPv6位址，您可以輸入IPv6地址或範圍。
TCP	6	3389 () RDP	您電腦的公用位IPv4址，或區域網路中的 IP 位址範圍。如果您的執行VPC個體已啟用，IPv6且執行個體有IPv6位址，您可以輸入IPv6地址或範圍。

從執行個體使用相同安全群組連線到執行個體的規則

若要允許與相同安全群組相關聯的執行個體彼此之間互相通訊，您必須為此明確新增規則。

> **Note**
>
> 如果您將路由設定為透過中間設備來轉遞不同子網中兩個執行個體之間的流量，則您必須確保兩個執行個體的安全群組均允許流量在執行個體之間流動。每個執行個體的安全性群組都必須

參照另一個執行個體的私有 IP 位址，或包含其他執行個體的子網路CIDR範圍做為來源。如果您參考另一個執行個體的安全群組作為來源，這不會允許流量在執行個體之間流動。

下表說明可讓相關聯的執行個體互相通訊的安全群組傳入規則。規則允許所有類型的流量。

通訊協定類型	通訊協定號碼	連接埠	來源 IP
-1 (全部)	-1 (全部)	-1 (全部)	安全性群組的識別碼，或包含其他執行個體的子網路CIDR範圍 (請參閱附註)。

平/規則 ICMP

該ping命令是一種ICMP流量類型。若要 ping 執行個體，您必須新增下列其中一個輸入ICMP規則。

Type	通訊協定	來源		
自訂 ICMP-IPv4	回聲請求	您電腦的公共IPv4位址、特定IPv4位址或來自任何地方的IPv4或IPv6位址。		
全部 ICMP-IPv4	IPv4ICMP(一)	您電腦的公共IPv4位址、特定IPv4位址或來自任何地方的IPv4或IPv6位址。		

若要使用ping6命令來 ping 執行個體的IPv6位址，您必須新增下列輸入ICMPv6規則。

Type	通訊協定	來源		
全部 ICMP-IPv6	IPv6ICMP	您電腦的IPv6位址、特定IPv4位		

Type	通訊協定	來源		
		址或來自任何地方的IPv6地址IPv4或地址。		

DNS伺服器規則

如果您已將EC2執行個體設定為DNS伺服器，則必須確保TCP且UDP流量可透過連接埠 53 到達您的DNS伺服器。

針對來源 IP，請指定下列其中一項：

- 網路中的 IP 位址或 IP 位址範圍 (以區CIDR塊標記法表示)
- 網路中需要存取DNS伺服器之執行個體集的安全群組識別碼

通訊協定類型	通訊協定號碼	連線埠
TCP	6	53
UDP	17	53

Amazon EFS 規則

如果您將 Amazon EFS 檔案系統與 Amazon EC2 執行個體搭配使用，則與 Amazon EFS 掛載目標相關聯的安全群組必須允許通過該NFS協定的流量。

通訊協定類型	通訊協定號碼	連接埠	來源 IP	備註
TCP	6	NFS	安全群組的 ID	允許NFS從與此安全群組關聯的資源 (包括掛載目標) 輸入存取

若要在 Amazon EC2 執行個體上掛載 Amazon EFS 檔案系統，您必須連線到執行個體。因此，與執行個體相關聯的安全性群組必須具有允許SSH從本機電腦或區域網路輸入的規則。

通訊協定類型	通訊協定號碼	連接埠	來源 IP	備註
TCP	6	SSH	本機電腦的 IP 位址範圍，或網路的 IP 位址範圍 (以區CIDR塊標示法表示)。	允許SSH從本機電腦輸入存取。

Elastic Load Balancing 規則

如果您向負載平衡器註冊EC2執行個體，與負載平衡器相關聯的安全性群組必須允許與執行個體進行通訊。如需詳細資訊，請參閱 Elastic Load Balancing 說明文件中的下列內容。

- 應用程式負載平衡器的安全性群組
- Network Load Balancer 的安全群組
- 設定 Classic Load Balancer 的安全群組

Amazon TPM EC2 實例的硝基

硝基可信平台模塊（NitroTPM）是由 N AWS itro 系統提供並符合 2.0 規範的虛擬設備。TPM它可安全存放用於對執行個體執行身分驗證的成品 (如密碼、憑證或加密金鑰)。Nitro TPM 可以生成密鑰並將其用於加密功能（例如散列，簽名，加密和解密）。

Nitro TPM 提供了測量的引導，引導加載程序和操作系統創建每個引導二進製文件的加密哈希，並將它們與 Nitro TPM 內部平台配置寄存器（ ）中的先前值結合起來的過程。PCRs透過測量的啟動，您可以從 Nitro 取得已簽署的PCR值，TPM並使用它們向遠端實體證明執行個體開機軟體的完整性。這稱為遠端證明。

使用 NitroTPM，可以使用特定PCR值標記密鑰和密鑰，以便在實例完整性發生變化時永遠無法訪問它們。PCR這種特殊形式的條件式存取被稱為密封和開封。例如 BitLocker，作業系統技術可以使用 Nitro TPM 密封磁碟機解密金鑰，以便只有在作業系統已正確啟動且處於已知良好狀態時，才能解密磁碟機。

若要使用 NitroTPM，您必須選取已設定為 Nitro TPM 支援的 Amazon 機器映像 (AMI)，然後使用啟動在 Nitro 系統上建置的執行個體 AWS。AMI您可以選擇亞馬遜的預構建之一，也可AMIs以自己創建一個。

定價

使用硝基TPM無需額外費用。只需為您使用的基礎資源支付費用。

目錄

- 將 Nitro TPM 與 Amazon 執EC2行個體搭配使用的要求
- AMI為硝基啟用 Linux 系統 TPM
- 驗證AMI已啟用硝基 TPM
- TPM在 Amazon EC2 實例上啟用或停止使用 Nitro
- 確認已為 Nitro 啟用 Amazon EC2 執行個體 TPM
- 擷取 Amazon EC2 執行個體的公開背書金鑰

將 Nitro TPM 與 Amazon 執EC2行個體搭配使用的要求

若要在啟用 Nitro 的情況下TPM啟動執行個體，您必須符合下列需求。

主題

- AMIs
- 執行個體類型
- 考量事項

AMIs

必AMI須TPM啟用硝基。

Linux AMIs

沒有預先設定AMIs。您必須配置自己的AMI。如需詳細資訊，請參閱AMI為硝基啟用 Linux 系統 TPM。

視窗 AMIs

下面的 Windows AMIs 預配置為啟用硝基TPM和UEFI安全啟動與 Microsoft 密鑰：

- TPM-視窗伺服器 2022-英文核心基礎
- TPM-視窗 _ 伺服器 2022-英文全基礎
- TPM-視窗 _ 服務器 2022-英語-全版-企業版 SQL
- TPM-視窗 _ 伺服器 2022-英文-完整版 SQL

- TPM-視窗伺服器 2019-英文核心基礎
- TPM-視窗服務器 2019-英語全基
- TPM-視窗 _ 伺服器 -2019-英文版-完整版 _ 企業版 SQL
- TPM-視窗伺服器 2019-英文版-完整版 SQL
- TPM-視窗伺服器 2016-英文核心基礎
- TPM-視窗伺服器 2016-英文全基礎

> **Note**
>
> 作業系統 — AMI 必須包含具有 TPM 2.0 指令回應緩衝區 (CRB) 驅動程式的作業系統。目前大多數操作系統都包含 TPM 2.0 CRB 驅動程序。
> UEFI啟動模式 — AMI 必須配置為UEFI啟動模式。如需詳細資訊，請參閱UEFIAmazon EC2實例的安全啟動。

執行個體類型

您必須使用下列其中一種虛擬化執行個體類型：

- 一般用途:M5, M5a, M5ad, M5dN, M5dn, M5 N, 兆兆焦, M6 克, M6gd, M6i, M6id, M6 ID, M6-柔性, M6 英寸, M7-柔性, T3,
- 計算優化：C5，C5a，C5 阿德，C5d，C5 安，C6 A，C6 克，六吉 D，六指數，C6 I，C6 身份，C6 英寸，C7 安培，C7 克，C7gD，C7gD，C7 指數，C7 I-柔性
- 記憶體最佳化:R5, R5 安培, R5 廣告, R5 乙, R5 DN, R5 n, R6 安, R6 克, R6 公升, R6 I, R6 英寸, R6 英寸, R6id, TB, 乌 7 英寸-24 TB, 乌 7 英寸 32 TB, X2IDN, X2iedn, 十二月, Z1D
- 儲存最佳化：D3、D3 安裝、i3en、i4i
- 加速運算：G4dN、G5、G6、G6e、GR6、第一次、二次
- 高效能運算：HPC6a、高效能

考量事項

使用 Nitro TPM 時需要注意下列事項：

- 在啟用 Nitro 的情況下TPM啟動執行個AMI體之後，如果您想要變更執行個體類型，您選擇的新執行個體類型也必須支援 Nit TPM ro。

- BitLocker 使用 Nitro TPM 型金鑰加密的磁碟區只能在原始執行個體上使用。
- Amazon EC2 控制台中不會顯示硝基TPM狀態。
- 硝基TPM狀態不包含在 Amazon EBS 快照中。
- Nitro TPM 狀態不包含在虛擬機器匯入/匯出映像中。
- 在 AWS Outposts.、局部區域或 Wavelength 區域上不支援硝基TPM。

AMI為硝基啟用 Linux 系統 TPM

若要TPM為執行個體啟用 Nitro，您必須使用已啟用 Nitro TPM 的功能來啟動執行個AMI體。註冊時，您必須AMI使用硝基TPM支持配置 Linux。您以後無法配置硝基TPM支持。

如需預先設定AMIs為硝基TPM支援的 Windows 清單，請參閱。將 Nitro TPM 與 Amazon 執EC2行個體搭配使用的要求

您必須使用AMI來建立TPM配置的RegisterImageAPI硝基。您不能使用 Amazon EC2 控制台或虛擬機導入/導出。

AMI為硝基啟用 Linux 的步驟 TPM

1. 使用您所需的 Linux 啟動臨時執行個體AMI。請記下其根磁碟區的識別碼，您可以在執行個體的 [儲存] 索引標籤上的主控台中找到該 ID。
2. 執行個體到達狀`running`態後，建立執行個體根磁碟區的快照。您可以使用控制台或下面的創建快照命令。

```
aws ec2 create-snapshot  \
    --volume-id vol-1234567890EXAMPLE \
    --description "Snapshot of the root volume"
```

3. 將您建立的快照註冊為AMI. 您必須使用註冊影像指令。在 `--tpm-support`，請指定 `v2.0`。在 `--boot-mode`，請指定 `uefi`。在區塊裝置對應中，指定您為根磁碟區建立的快照。

```
aws ec2 register-image \
    --name my-image \
    --boot-mode uefi \
    --architecture x86_64 \
    --root-device-name /dev/xvda \
    --block-device-mappings DeviceName=/dev/xvda,Ebs={SnapshotId=snapshot_id} \
    --tpm-support v2.0
```

下列為範例輸出。

```
{
    "ImageId": "ami-0123456789example"
}
```

4. 終止您在步驟 1 中啟動的暫存執行個體。

驗證AMI已啟用硝基 TPM

若要TPM為執行個體啟用 Nitro，您必須使用已啟用 Nitro TPM 的功能來啟動執行個AMI體。您可以使用`describe-images`或`describe-image-attributes`來驗證AMI是否已啟用 Nitro TPM。如果TPM已啟用硝基AMI，則的值`TpmSupport`為`"v2.0"`。

描述圖像

您可以使用描述影像指令，如下所示。

```
aws ec2 describe-images --image-ids ami-0123456789example --query Images[*].TpmSupport
```

如果TPM為啟用硝基AMI，則輸出如下。

```
[
    "v2.0"
]
```

如果TPM未啟用，則輸出為空。

```
[
]
```

描述圖像屬性

或者，如果您是AMI擁有者，則可以使用指describe-image-attribute令，如下所示，指定`tpmSupport`為`attribute`。

```
aws ec2 describe-image-attribute \
    --region us-east-1 \
    --image-id ami-0123456789example \
```

```
    --attribute tpmSupport
```

下列為範例輸出。

```
{
    "ImageId": "ami-0123456789example",
    "TpmSupport": {
        "Value": "v2.0"
    }
}
```

TPM在 Amazon EC2 實例上啟用或停止使用 Nitro

您TPM只能在啟動時為 Nitro 啟用 Amazon EC2 執行個體。一旦啟用 Nitro 的執行個體TPM，您就無法停用它。如果您不再需要使用 NitroTPM，則必須配置操作系統以停止使用它。

主題

- 啟用 Nitro TPM 的情況下啟動執行個體
- 停止TPM在實例上使用 Nitro

啟用 Nitro TPM 的情況下啟動執行個體

當您啟動具有先決條件的執行個體時，Nitro 會自動TPM在執行個體上啟用。您只能在啟動時在執行個體TPM上啟用 Nitro。如需有關啟動執行個體的詳細資訊，請參閱啟動 Amazon EC2 實例。

停止TPM在實例上使用 Nitro

在啟用 Nitro 的情況下TPM啟動執行個體後，您無法停用該執行個體TPM的 Nitro。但是，您可以使用下列工具停用執行個體上的 TPM 2.0 裝置驅動程式，將作業系統設定為停止使用 NitroTPM：

- 對於 Linux 執行個體，請使用 tPM-工具。
- 如果是 Windows 執行個體，請使用TPM管理主控台 (tpm.msc)。

如需有關停用裝置驅動程式的詳細資訊，請參閱適用於您作業系統的相關文件。

確認已為 Nitro 啟用 Amazon EC2 執行個體 TPM

確認執行個體是否已啟用 Nitro TPM

使用 describe-instances AWS CLI 命令並指定執行個體 ID。Amazon EC2 控制台不顯示該`TpmSupport`字段。

```
aws ec2 describe-instances --instance-ids i-0123456789example
```

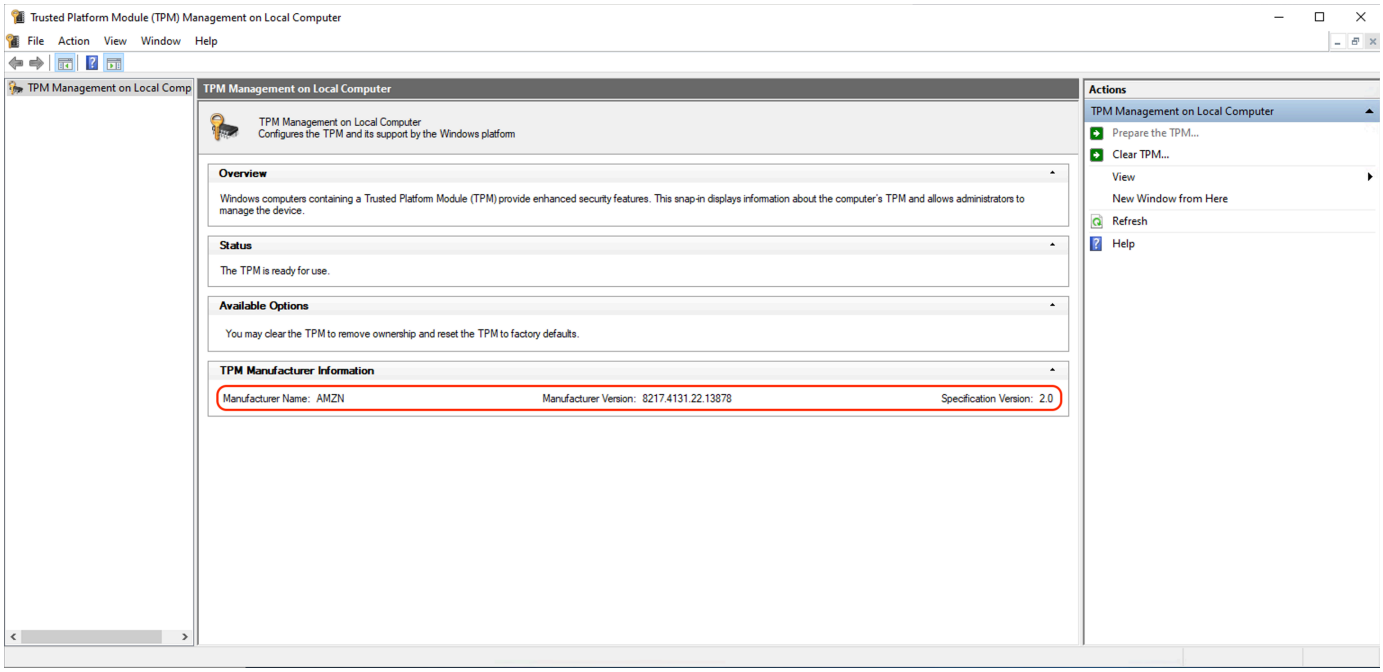
如果執行個體已啟用 Nitro TPM 支援，則會出`"TpmSupport": "v2.0"`現在輸出中。例如：

```
"Instances": {
  "InstanceId":"0123456789example",
  "InstanceType":"c5.large",
  ...
  "BootMode": "uefi",
  "TpmSupport": "v2.0"
  ...
}
```

(僅適用於 Windows 執行個體) 若要驗證硝基TPM是否可供視窗存取

1. Connect 至您的EC2視窗執行個體。
2. 在執行個體上，執行 tpm.msc 程式。

   「在本機電腦上TPM管理」視窗隨即開啟。
3. 請檢查「TPM制造商信息」字段。它包含製造商的名稱和實例上的 Nitro TPM 的版本。



擷取 Amazon EC2 執行個體的公開背書金鑰

您可以隨時使用安全地擷取執行個體的公開背書金鑰。 AWS CLI

擷取執行個體的公開背書金鑰

使用 get-instance-tpm-ek-pub AWS CLI 指令。

範例 1

下列範例命令會取得指定執行個體tpmt格式的rsa-2048公開背書金鑰。

```
aws ec2 get-instance-tpm-ek-pub \--instance-id i-01234567890abcdef \
--key-format tpmt \
--key-type rsa-2048
```

以下是範例輸出。

```
{
    "InstanceId": "i-01234567890abcdef",
    "KeyFormat": "tpmt",
    "KeyType": "rsa-2048",
    "KeyValue": "AAEACwADALIAIINxl2dEhLEXAMPLEUal1yT9UtduBlILZPKh2hszFGmqAAYAgABDA
    EXAMPLEAAABAOiRd7WmgtdGNoV1h/AxmW+CXExblG8pEUfNm0LOLiYnEXAMPLERqApiFa/UhvEYqN4
    Z7jKMD/usbhsQaAB1gKA5RmzuhSazHQkax7EXAMPLEzDthlS7HNGuYn5eG7qnJndRcakS+iNxT8Hvf
    0S1ZtNuItMs+Yp4SO6aU28MT/JZkOKsXIdMerY3GdWbNQz9AvYbMEXAMPLEPyHfzgVO0QTTJVGdDxh
    vxtXCOu9GYf0crbjEXAMPLEd4YTbWdDdgOKWF9fjzDytJSDhrLAOUctNzHPCd/92l5zEXAMPLEOIFA
    Ss50C0/802c17W2pMSVHvCCa9lYCiAfxH/vYKovAAE="
}
```

範例 2

下列範例命令會取得指定執行個體der格式的rsa-2048公開背書金鑰。

```
aws ec2 get-instance-tpm-ek-pub \
--instance-id i-01234567890abcdef \
--key-format der \
--key-type rsa-2048
```

以下是範例輸出。

```
{
    "InstanceId": "i-01234567890abcdef",
```

```
    "KeyFormat": "der",
    "KeyType": "rsa-2048",
    "KeyValue": "MIIBIjANBgEXAMPLEw0BAQEFAAOCAQ8AMIIBCgKCAQEA6JF3taEXAMPLEXWH8DGZb4
    JcTFuUbykRR82bQs4uJifaKSOv5NGoEXAMPLEG8Rio3hnuMowP+6xuGxBoAHWAoDlGbO6FJrMdEXAMP
    LEnYUHvMO2GVLsc0a5ifl4buqcmd1FxqRL6I3FPwe9/REXAMPLE0yz5inhI7ppTbwxP8lmQ4qxch0x6
    tjcZ1Zs1DP0EXAMPLERUYLQ/Id/OBU7RBNMlUZ0PGG/G1cI670Zh/RytuOdx9iEXAMPLEtZ0N2A4pYX
    1+PMPK0lIOGssA5Ry03Mc8J3/3aXnOD2/ASRQ4gUBKznQLT/zTZEXAMPLEJUe8IJr2VgKIB/Ef+9gqi
    8AAQIDAQAB"
}
```


Windows 執行個體的認證保護

AWS 硝基系統支持 Amazon 彈性計算雲（AmazonEC2）Windows 實例的憑據保護。認證保全是以 Windows 虛擬化為基礎的安全性 (VBS) 功能，可讓您建立隔離的環境，以保護 Windows 使用者認證和程式碼完整性強制執行之外的安全性資產，而不是 Windows 核心保護。當您執行 EC2 Windows 執行個體時，認證保全會使用 AWS Nitro 系統來保護 Windows 登入認證，避免從作業系統的記憶體中擷取。

目錄

- 必要條件
- 啟動支援的執行個體
- 禁用內存完整性
- 開啟認證保全
- 確認認證保全正在執行

必要條件

您的 Windows 執行個體必須符合下列必要條件，才能使用認證保全。

Amazon 機器圖像（AMIs）

AMI必須預先配置才能啟用 Nitro TPM 和UEFI安全啟動。如需支援的詳細資訊AMIs，請參閱the section called “要求”。

記憶體完整性

不支援記憶體完整性 (也稱為 Hypervisor 保護的程式碼完整性 (HVCI) 或 Hypervisor 強制執行程式碼完整性。開啟 Credential Guard 之前，您必須確定此功能已停用。如需詳細資訊，請參閱禁用內存完整性。

執行個體類型

除非另有說明，否則下列執行個體類型支援所有大小的認證保護：C5C5dC5nC6iC6idC6inC7iC7i-flex,M5,M5d,M5dn,,M5n,M5zn,M6i,,M6id,M6idn,M6in,,M7i,M7i-flex,R5,,R5b,R5d,R5dn,,R5n,R6i,R6id,,R6idn, R6inR7i,R7iz,, T3。

> **Note**
> - 雖然 Nitro TPM 具有一些共同的必要實例類型，但實例類型必須是上述實例類型之一才能支持憑據保護。
> - 憑證保護不支援：
> - 裸機執行個體。
> - 下列執行個體類型：C7i.48xlargeM7i.48xlarge、和R7i.48xlarge。

如需執行個體類型的詳細資訊，請參閱 Amazon EC2 執行個體類型指南。

啟動支援的執行個體

您可以使用 Amazon EC2 主控台或 AWS Command Line Interface (AWS CLI) 啟動可支援登入資料保護的執行個體。您將需要一個兼容的 AMI ID 來啟動實例，該 ID 對於每個實例都是唯一的 AWS 區域。

> **Tip**
> 您可以使用下列連結來探索並啟動 Amazon EC2 主控台提供AMIs的相容 Amazon 執行個體：https://console.aws.amazon.com/ec2/v2/home?#Images:visibility=public-images;v=3;search=:TPM-Windows_Server;ownerAlias=amazon

Amazon EC2 console

使用 Amazon EC2 主控台啟動執行個體

請依照下列步驟啟動執行個體，指定支援的執行個體類型和預先設定的 Windows AMI。

AWS CLI

使用啟動執行個體 AWS CLI

使用指run-instances令，使用支援的執行個體類型和預先設定的 Windows AMI 來啟動執行個體。

```
aws ec2 run-instances \
    --image-id resolve:ssm:/aws/service/ami-windows-latest/TPM-Windows_Server-2022-English-Full-Base \
    --instance-type c6i.large \
    --region us-east-1 \
    --subnet-id subnet-id
    --key-name key-name
```

PowerShell

使用啟動執行個體 AWS Tools for PowerShell

使用指New-EC2Instance令，使用支援的執行個體類型和預先設定的 Windows AMI 來啟動執行個體。

```
New-EC2Instance `
    -ImageId resolve:ssm:/aws/service/ami-windows-latest/TPM-Windows_Server-2022-English-Full-Base `
    -InstanceType c6i.large `
    -Region us-east-1 `
    -SubnetId subnet-id `
    -KeyName key-name
```

禁用內存完整性

在支援的情況下，您可以使用本機群組政策編輯器停用記憶體完整性。下列指引適用於虛擬化型程式碼完整性保護下的每個組態設定：

- 已啟用但未鎖定：將設定修改為已停用以停用記憶體完整性。
- 以UEFI鎖定啟用 — 已透過UEFI鎖定啟用記憶體完整性。一旦使用UEFI鎖定啟用記憶體完整性，就無法停用記憶體完整性。建議您建立停用記憶體完整性的新執行個體，並終止未使用且不受支援的執行個體。

使用本機群組政策編輯器停用記憶體完整性

1. 使用遠端桌面通訊協定 (RDP) 以具有管理員權限的使用者帳戶 Connect 線至執行個體。如需詳細資訊，請參閱the section called “使 Connect 用RDP戶端連線”。

2. 開啟「開始」選單並搜尋 **cmd** 以啟動命令提示。
3. 執行以下命令以開啟本機群組政策編輯器：`gpedit.msc`
4. 在本機群組政策編輯器中，選擇電腦設定、管理範本、系統、Device Guard。
5. 選取開啟基於虛擬化的安全性，然後選取編輯政策設定。
6. 開啟虛擬化型程式碼完整性保護的設定下拉式清單，選擇已停用，然後選擇套用。
7. 重新啟動執行個體以套用變更。

開啟認證保全

啟動具有支援執行個體類型且相容的 Windows 執行個體AMI，並確認記憶體完整性已停用之後，您可以開啟認證保全。

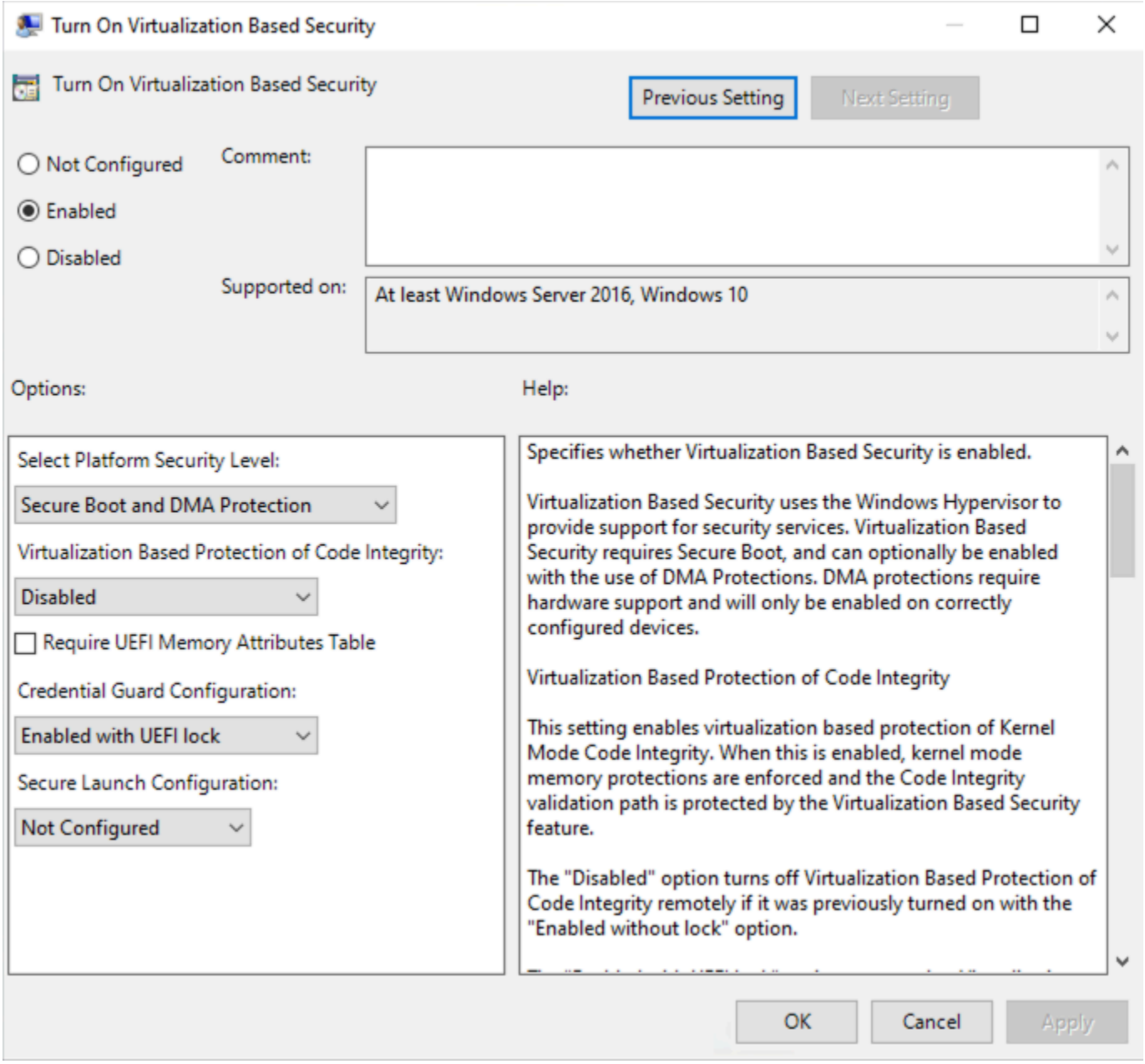
> **Important**
>
> 需要管理員權限執行以下步驟來開啟 Credential Guard。

開啟 Credential Guard

1. 使用遠端桌面通訊協定 (RDP) 以具有管理員權限的使用者帳戶 Connect 線至執行個體。如需詳細資訊，請參閱the section called “使 Connect 用RDP戶端連線”。
2. 開啟「開始」選單並搜尋 **cmd** 以啟動命令提示。
3. 執行以下命令以開啟本機群組政策編輯器：`gpedit.msc`
4. 在本機群組政策編輯器中，選擇電腦設定、管理範本、系統、Device Guard。
5. 選取開啟基於虛擬化的安全性，然後選取編輯政策設定。
6. 在開啟基於虛擬化的安全性選單中選擇已啟用。
7. 針對 [選取平台安全層級]，選擇 [安全開機和DMA保護]。
8. 在「認證保全組態」中，選擇以UEFI鎖定啟用。

   > **Note**
   >
   > 其餘的政策設定不需要來啟用 Credential Guard，而且可以保留為未設定。

   下列影像顯示如前所述所VBS設定的設定：



9. 重新啟動執行個體以套用設定。

確認認證保全正在執行

可以使用 Microsoft System Information (`Msinfo32.exe`) 工具來確認 Credential Guard 是否正在執行。

> **Important**
>
> 您必須先重新啟動執行個體，才能完成套用啟用 Credential Guard 所需的政策設定。

驗證 Credential Guard 正在執行

1. 使用遠端桌面通訊協定 (RDP) Connect 線至執行個體。如需詳細資訊，請參閱the section called "使 Connect 用RDP戶端連線"。
2. 在執行個體的RDP工作階段中，開啟 [開始] 功能表並搜尋**cmd**以啟動命令提示字元。
3. 透過執行以下命令，開啟 System Information：`msinfo32.exe`
4. Microsoft 系統資訊工具會列出VBS組態的詳細資料。在基於虛擬化的安全性服務旁邊，確認 Credential Guard 顯示為正在執行。

   以下影像顯示VBS正在執行，如前所述：

Virtualization-based security	Running
Virtualization-based security Required Security Properties	Base Virtualization Support, Secure Boot, DMA Protection
Virtualization-based security Available Security Properties	Base Virtualization Support, Secure Boot, DMA Protection, UEFI Code Readonly, Mode Based Execution Control
Virtualization-based security Services Configured	Credential Guard
Virtualization-based security Services Running	Credential Guard

EC2使用界面VPC端點訪問 Amazon

您可以通過在您VPC和 Amazon 之間創VPC建私有連接來改善您的安全狀態EC2。您可以像在您的 Amazon 一EC2樣訪問亞馬遜VPC，而無需使用互聯網網關，NAT設備，VPN連接或 AWS Direct Connect 連接。您中的執行個體VPC不需要公有 IP 地址即可存取 Amazon EC2。

如需詳細資訊，請參閱AWS PrivateLink 指南 AWS PrivateLink中的 AWS 服務 透過存取。

目錄

- 建立介面VPC端點
- 建立端點政策

建立介面VPC端點

使用下列服務名稱為 Amazon EC2 建立介面端點：

- COM. 亞馬遜。***region***.ec2 — 為 Amazon 行EC2API動創建端點。

如需詳細資訊，請參閱AWS PrivateLink 指南中的 AWS 服務 使用介面VPC端點存取。

建立端點政策

端點策略是您可以附加到接口端點的IAM資源。預設端點政策允許EC2API透過界面端點完全存取Amazon。若要控制允許EC2API從您的 Amazon 存取VPC，請將自訂端點政策附加到介面端點。

端點政策會指定以下資訊：

- 可執行動作的主體。
- 可執行的動作。
- 可供執行動作的資源。

> **Important**
>
> 將非預設政策套用至 Amazon 的介面VPC端點時EC2，某些失敗的API請求 (例如失敗的請求) 可能不會記錄到 AWS CloudTrail 或 Amazon CloudWatch。`RequestLimitExceeded`

如需詳細資訊，請參閱《AWS PrivateLink 指南》中的使用端點政策控制對服務的存取。

下列範例顯示的VPC端點政策會拒絕建立未加密磁碟區或啟動具有未加密磁碟區的執行個體的權限。範例政策也授予執行所有其他 Amazon EC2 動作的權限。

```
{
    "Version": "2012-10-17",
    "Statement": [
    {
        "Action": "ec2:*",
        "Effect": "Allow",
        "Resource": "*",
        "Principal": "*"
    },
    {
        "Action": [
            "ec2:CreateVolume"
        ],
        "Effect": "Deny",
        "Resource": "*",
        "Principal": "*",
```

```
        "Condition": {
            "Bool": {
                "ec2:Encrypted": "false"
            }
        }
    },
    {
        "Action": [
            "ec2:RunInstances"
        ],
        "Effect": "Deny",
        "Resource": "*",
        "Principal": "*",
        "Condition": {
            "Bool": {
                "ec2:Encrypted": "false"
            }
        }
    }]
}
```

適用於 Amazon EC2 執行個體的儲存選項

Amazon 為您的執行個體EC2提供彈性、符合成本效益的 easy-to-use 資料儲存選項。每種選項都有獨特的效能與耐用性組合。這些儲存選項可獨立或組合使用，因應您的要求。

區塊儲存

- Amazon EBS — Amazon EBS 提供耐用的區塊級儲存磁碟區，您可以從執行個體連接和分離這些儲存磁碟區。您可以將多個EBS磁碟區連接至執行個體。EBS磁碟區會與其關聯執行個體的生命週期獨立存在。您可以加密EBS磁碟區。若要保留資料的備份副本，您可以從EBS磁碟區建立快照。快照會存放在 Amazon S3。您可以從快照建立EBS磁碟區。
- Amazon EC2 執行個體的執行個體存放臨時區塊儲— 執行個體存放區為執行個體提供暫時的區塊層級儲存。執行個體儲存體磁碟區的數量、大小和類型取決於執行個體類型和執行個體大小。執行個體存放區磁碟區上的資料只會在相關聯執行個體的壽命期間持續保留；如果您讓執行個體停止、休眠或終止，則會遺失執行個體存放區磁碟區上的任何資料。

物件儲存

- Amazon S3— Amazon S3 可讓您存取可靠且經濟實惠的資料儲存基礎設施。它旨在讓您隨時從 Amazon 內部EC2或網路上的任何地方存放和擷取任意數量的資料，讓網路規模運算變得更容易。例如，您可以使用 Amazon S3 來存放資料和應用程式的備份複本。亞馬遜EC2使用 Amazon S3 存放EBS快照和執行個體商店支援AMIs。

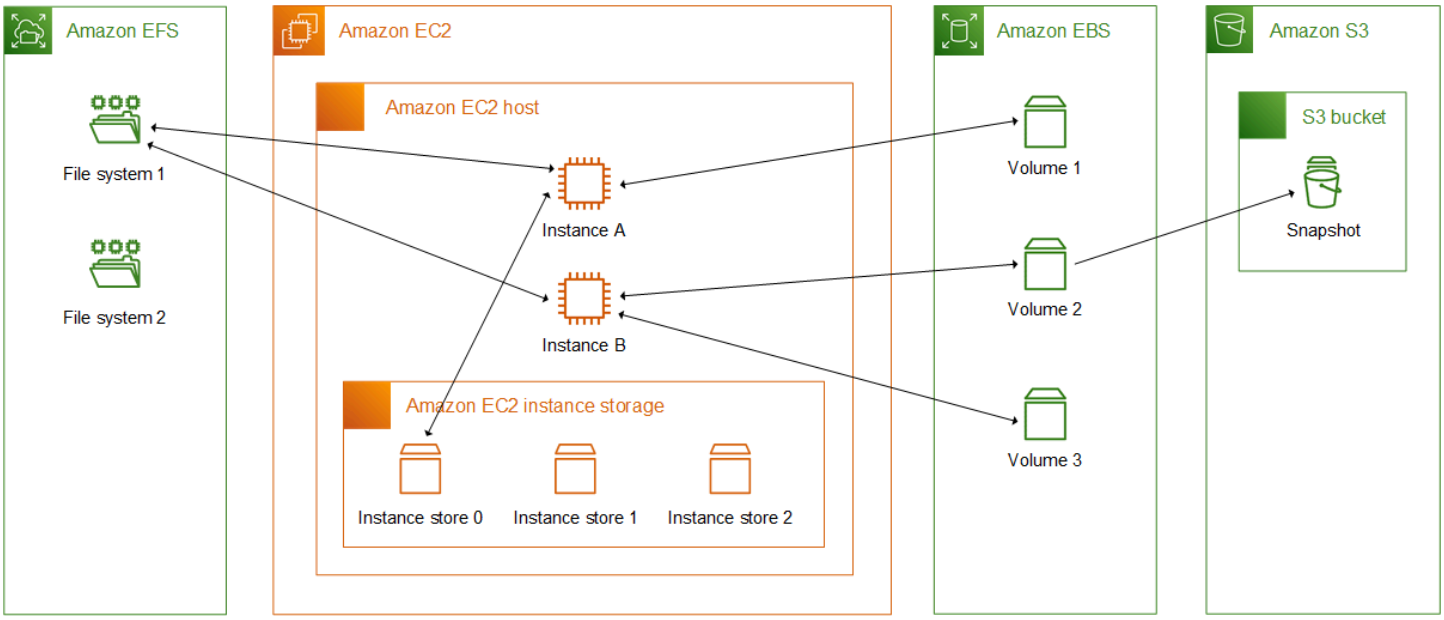
檔案儲存

- Amazon EFS(僅限 Linux 執行個體) — Amazon EFS 提供可擴展的檔案儲存，以便與 Amazon 搭配使用EC2。您可以建立EFS檔案系統，並將執行個體設定為掛載檔案系統。您可以使用EFS檔案系統做為在多個執行個體上執行的工作負載和應用程式的通用資料來源。
- Amazon FSx— 使用 AmazonFSx，您可以在雲端啟動、執行和擴展功能豐富的高效能檔案系統。Amazon FSx 是一項全受管服務，可支援各種工作負載。您可以在這些廣泛使用的檔案系統之間進行選擇：「光碟」、 NetApp ONTAP「開啟ZFS」和「Windows 檔案伺服器」。

文件緩存

- 搭配 Amazon EC2 執行個體使用 Amazon 檔案快取— Amazon 檔案快取提供臨時、高效能的快取 AWS ，以處理檔案資料。快取提供對 Amazon EC2 上運算工作負載的讀取和寫入資料存取，延遲時間低於一毫秒、高達數百 GB 的輸送量以及高達數百萬 GB 的輸送量。IOPS

下圖顯示這些儲存體選項和執行個體之間的關係。



AWS 儲存定價

開啟 [AWS 定價]，捲動至 [AWS 產品的定價]，然後選取 [儲存 選擇儲存產品，開啟其定價頁面。

Amazon EC2 執行個體的EBS持久性區塊儲存

Amazon 彈性區塊存放區 (AmazonEBS) 提供可擴展的高效能區塊儲存資源，可與 Amazon EC2 執行個體搭配使用。使用 AmazonEBS，您可以建立和管理下列區塊儲存資源：

- Amazon EBS 卷 — 這些是您連接到 Amazon EC2 實例的存儲卷。將磁碟區連接至執行個體後，您可以使用與使用區塊儲存相同的方式來使用磁碟區。執行個體可與磁碟區互動，就像與本機磁碟機一樣。
- Amazon EBS 快照 — 這些是 Amazon EBS 磁碟區的 point-in-time 備份，獨立於磁碟區本身。您可以建立快照以備份 Amazon EBS 磁碟區上的資料。然後，您可以隨時從這些快照還原新磁碟區。

您可以在啟動期間建立EBS磁碟區並將其附加到執行個體，也可以在啟動後隨時建立EBS磁碟區並將其附加到執行個體。您也可以增加磁碟區的大小或效能，而無需分離EBS磁碟區或重新啟動執行個體。

您可以在建立後隨時從EBSEBS磁碟區建立快照。您可以使用EBS快照備份儲存在磁碟區中的資料。然後，您可以使用這些快照立即還原磁碟區，或跨 AWS 帳戶區域或可用區域遷移資料。 AWS 您可以使用 Amazon Data Lifecycle Manager 或 AWS Backup 自動化EBS快照的建立、保留和刪除。

如需使用磁碟區和快照的詳細資訊，請參閱 Amazon EBS 使用者指南。

Amazon EC2 實例的EBS數量限制

您可以連接到執行個體的最大 Amazon EBS 磁碟區數量取決於執行個體類型和執行個體大小。考慮要連接至執行個體的磁碟區數目時，您應考量是否需要提高 I/O 頻寬或提高儲存體容量。

頻寬和容量

如需一致且可預測的頻寬使用案例，請使用 Amazon EBS 最佳化執行個體搭配一般用途SSD磁碟區或佈建IOPSSSD磁碟區。為了獲得最大效能，請將IOPS您為磁碟區佈建的範圍與執行個體類型的可用頻寬相符。

對於RAID配置，您可能會發現大於 8 個磁碟區的陣列因 I/O 額外負荷增加而降低了效能回報。測試個別應用程式效能，並視需要調整。

內容

- 建置於 Nitro System 上的執行個體磁碟區限制
  - 專用EBS音量限制
  - 共用EBS磁碟區限制
- Xen 型執行個體的磁碟區限制
  - Linux 執行個體
  - Windows 執行個體

建置於 Nitro System 上的執行個體磁碟區限制

Nitro System 上建置的執行個體的磁碟區限制取決於執行個體類型。某些 Nitro 執行個體類型具有專用的EBS磁碟區限制，而大多數執行個體具有共用磁碟區限制

專用EBS音量限制

下列 Nitro 執行個體類型具有專用EBS磁碟區限制，視執行個體大小而有所不同。此限制不會與其他裝置連接共用。換句話說，您可以將任意數量的EBS磁碟區附加至磁碟區附加限制，不論連接的裝置數量如何，例如NVMe執行個體儲存磁碟區和網路介面。

- 一般用途：M7a、M7i、M7-柔性
- 運算最佳化：C7a、C7i、C7i-彈性
- 最佳化記憶體：7 安培、7 i、R7 伊茲、R8 克、U7i
- 加速運算：G6

對於這些支援專用磁碟區限制的執行個體類型，磁碟區限制取決於執行個體大小。下表顯示每個執行個體大小的限制。

執行個體大小	磁碟區限制
`medium` | `large` | `xlarge` | `2xlarge` | `4xlarge` | `8xlarge` | `12xlarge`	32
`16xlarge`	48
`24xlarge`	64
`32xlarge`	88
`48xlarge`	128
`metal-16xl` | `metal-24xl`	39
`metal-32xl` | `metal-48xl`	79

共用EBS磁碟區限制

所有其他 Nitro 執行個體類型 (未列於中專用EBS音量限制) 都有磁碟區附件限制，該限制可在 Amazon EBS 磁碟區、網路界面和NVMe執行個體存放區磁碟區之間共用。您可以附加任意數量的 Amazon EBS 磁碟區，最多只能連接網路界面和NVMe執行個體存放磁碟區的數量。請記住，每個執行個體必須至少有一個網路介面，而且該NVMe執行個體儲存磁碟區會在啟動時自動連接。

大多數 Nitro 實例最多支持 28 個附件。以下範例示範如何計算您可以連接的EBS體積塊數。

範例

- 如果`m5.xlarge`執行個體只有主要網路介面，您可以連接 27 個EBS磁碟區。

  28 個磁碟區-1 個網路介面 = 27
- 如果`m5.xlarge`執行個體具有兩個額外的網路介面，您可以連接 25 個EBS磁碟區。

  28 個磁碟區-3 個網路介面 = 25 個

- 如果m5d.xlarge執行個體具有兩個額外的網路介面，您可以連接 24 個EBS磁碟區。

  28 個磁碟區-3 個網路介面-1 NVMe 個執行個體儲存磁碟區 = 24

以下是具有共用磁碟區限制的執行個體類型的例外情況。

例外狀況

- d3.8xlarge和d3en.12xlarge執行個體最多支援 3 個EBS磁碟區。
- DL2q執行個體最多支援 19 個EBS磁碟區。
- g5.48xlarge執行個體最多支援 9 個EBS磁碟區。
- inf1.xlarge和inf1.2xlarge執行個體最多支援 26 個EBS磁碟區。
- inf1.6xlarge執行個體最多支援 23 個EBS磁碟區。
- inf1.24xlarge執行個體最多支援 11 個EBS磁碟區。
- Mac2、Mac2-m2Mac2-m2pro、和Mac2-m1ultra執行個體最多支援 10 個EBS磁碟區。
- U-*tb1虛擬化執行個體最多可支援 27 個EBS磁碟區。
- 對於vt1.3xlarge和vt1.6xlarge例證而言，每個加速器都算作兩個附件。
- 對於vt1.24xlarge執行個體，加速器不會計入共用磁碟區限制。
- 對於例證以外的加速計算VT1例證，每個加速器都算作一個附件。例如，p4d.24xlarge執行個體的共用磁碟區限制為 28 GPUs、8 和 8 個NVMe執行個體存放磁碟區。這表示您最多可以連接 11 個EBS磁碟區 (28 個磁碟區-1 個網路介面-8 GPUs-8 個NVMe執行個體儲存磁碟區)。
- 大多數裸機執行個體最多支援 31 個EBS磁碟區。以下是例外情況：
  - mac1.metal執行個體最多支援 16 個EBS磁碟區。
  - U-*tb1裸機執行個體最多可支援 19 個EBS磁碟區。

Xen 型執行個體的磁碟區限制

Xen 型執行個體的磁碟區限制取決於作業系統。

Linux 執行個體

將 40 個以上的磁碟區連接到 Xen 型 Linux 執行個體，可能會導致開機失敗。此數字包括根磁碟區，以及任何連接的執行個體存放區磁碟區和 Amazon EBS 磁碟區。

如果具有大量磁碟區的執行個體發生開機問題，請停止執行個體、分離啟動程序非必要的所有磁碟區、啟動執行個體，然後在執行個體執行之後重新連接磁碟區。

> **Important**
>
> Xen 型 Linux 執行個體最佳情況可支援連接超過 40 個磁碟區，但不予保證。

Windows 執行個體

下表根據使用的驅動程式，顯示 Xen 型 Windows 執行個體的磁碟區限制。這些數字包括根磁碟區，以及任何連接的執行個體存放區磁碟區和 Amazon EBS 磁碟區。

驅動程式	磁碟區限制
AWS 光伏	26
Citrix PV	26
Red Hat PV	17

建議您不要將超過 26 個磁碟區連接至具有 AWS PV 或 Citrix PV 驅動程式的 Xen 型 Windows 執行個體，因為這可能會造成效能問題。若要判定執行個體使用哪些 PV 驅動程式，或將 Windows 執行個體從 Red Hat 升級至 Citrix PV 驅動程式，請參閱the section called “升級 PV 驅動程式”。

> **Important**
>
> Xen 型 Windows 執行個體最佳情況可支援連接超過以下數量的磁碟區，但不予保證。

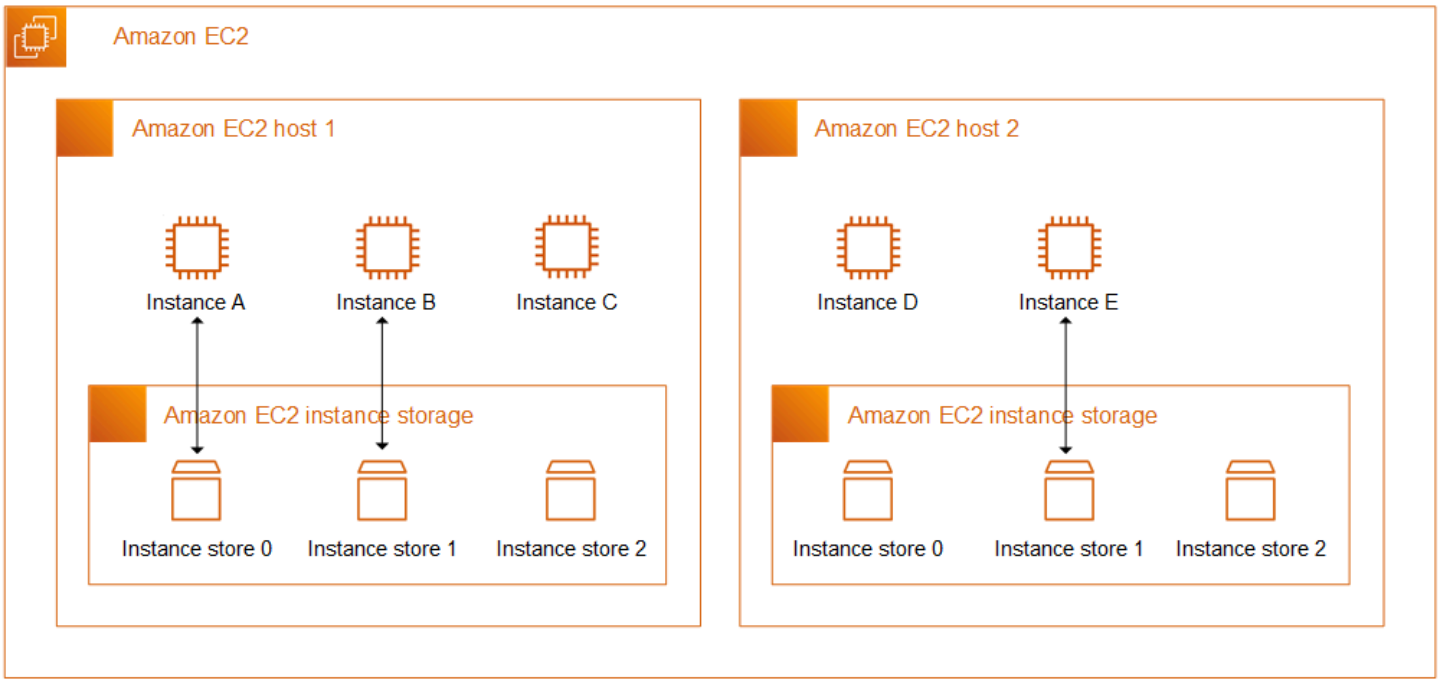
如需裝置名稱如何與磁碟區相關的詳細資訊，請參閱將磁碟映射到連接到 Amazon EC2 Windows 執行個體的磁碟區。

Amazon EC2 執行個體的執行個體存放臨時區塊儲

執行個體存放區為您EC2的執行個體提供暫時的區塊層級儲存體。此儲存體是由實體連接至主機電腦的磁碟所提供。執行個體儲存體非常適合暫時儲存不斷變更的資訊，例如緩衝區、快取、臨時資料及其他暫存的內容。也可用來存放複製到整個執行個體機群的暫時性資料，例如網頁伺服器的負載平衡集區。

執行個體存放區包含顯示為區塊型儲存設備的一個或多個執行個體存放磁碟區。執行個體儲存體的大小和可用裝置的數量，會隨執行個體類型和執行個體大小而有不同。例如，並非每個執行個體類型都提供執行個體儲存磁碟區 如需詳細資訊，請參閱Amazon EC2 執行個體的執行個體存放量限制。

執行個體儲存磁碟區的虛擬裝置會ephemeral0以從到的順序指定虛擬裝置名稱ephemeral23。例如，如果執行個體類型支援一個執行個體儲存磁碟區，則該磁碟區的虛擬裝置名稱為ephemeral0。如果執行個體類型支援四個執行個體儲存磁碟區，四個磁碟區的虛擬裝置名稱如下：ephemeral0ephemeral1、ephemeral2和ephemeral3。



執行個體儲存體定價

使用為執行個體提供的執行個體存放磁碟區無須額外付費。執行個體存放區磁碟區包含在執行個體使用費用中。

目錄

- Amazon EC2 執行個體存放磁碟區的資料持續
- Amazon EC2 執行個體的執行個體存放量限制
- SSDAmazon EC2 執行個體的執行個體存放量
- 將執行個體存放磁碟區新增至 Amazon EC2 執行個
- 為 Amazon EC2 M1 和 C1 執行個體啟用執行個體存放區交換磁碟區
- 初始化 Amazon EC2 實例上的實例存儲卷

Amazon EC2 執行個體存放磁碟區的資料持續

執行個體儲存體磁碟區只會在執行個體啟動時連接。啟動後即無法連接執行個體儲存體磁碟區。您不能將執行個體儲存體磁碟區與某個執行個體中斷連接，然後再連接到不同的執行個體。

執行個體儲存體磁碟區只存在於所連接執行個體的存留期。您無法將執行個體儲存體磁碟區設定為保留超過其關聯執行個體的存留期。

即使將執行個體重新開機，執行個體儲存體磁碟區上的資料仍會存在。但如果停止、休眠或終止執行個體，則資料不會存在。執行個體停止、休眠或終止時，會以密碼編譯方式清除執行個體儲存體磁碟區的每個區塊。

因此，如果是重要的長期資料，請不要只使用執行個體儲存體磁碟區。如果您需要將存放在執行個體存放磁碟區上的資料保留超過執行個體的生命週期，則需要手動將該資料複製到更持久的儲存，例如 Amazon EBS 磁碟區、Amazon S3 儲存貯體或 Amazon EFS 檔案系統。

部分事件可能會導致您的資料在執行個體的整個存留期中未持續存在。下表指出在特定事件期間，虛擬化和裸機執行個體的執行個體儲存體磁碟區上的資料是否持續存在。

事件	您的資料會發生什麼事？
使用者啟動的執行個體生命週期事件	
執行個體已重新啟動	數據仍然存在
執行個體已停止	數據不會持續存在
執行個體進入休眠狀態	數據不會持續存在
執行個體已終止	數據不會持續存在
執行個體類型已變更	資料不會持續存在 *
從實例創建AMI一個EBS支持	數據不會持續存在於創建的 AMI **
從執行個體 (Linux 執行AMI個體) 建立存放區支援的執行個體	數據仍然存在於上傳到 Amazon S3 的AMI服務包中 ***
使用者啟動的作業系統事件	
啟動關閉	數據不會持續 †

事件	您的資料會發生什麼事？
啟動重新啟動	數據仍然存在
AWS 排程事件	
實例停止	數據不會持續存在
實例重新啟	數據仍然存在
系統重啟	數據仍然存在
執行個體淘	數據不會持續存在
意外事件	
簡化的自動恢復	數據不會持續存在
CloudWatch 基於動作的復原	數據不會持續存在
基礎磁碟失敗	故障磁碟上的資料不會持續存在
電源故障	重新啟動後數據仍然存在

* 如果新執行個體類型支援執行個體儲存體，執行個體會具有新執行個體類型支援的執行個體儲存體磁碟區數量，但資料不會傳輸至新執行個體。如果新執行個體類型不支援執行個體儲存體，則執行個體不會取得執行個體儲存體磁碟區。

** 資料不包含在 EBS-backed 中AMI，且不包含在連接至從該AMI執行個體啟動之執行個體的執行個體儲存磁碟區中。

*** 資料包含在上傳至 Amazon S3 的AMI服務包中。當您從該執行個體啟動執行個體時AMI，執行個體會取得執行個體儲存磁碟區，以及建立執行個體時所包含的AMI資料。AMI

† 終止保護和停止保護不會防止執行個體因透過執行個體上的作業系統關機，而停止或終止。儲存在執行個體儲存體磁碟區上的資料不會保留在執行個體停止和終止事件中。

Amazon EC2 執行個體的執行個體存放量限制

執行個體儲存磁碟區的數量、大小和類型取決於執行個體類型。部分執行個體類型 (如 M6、C6 和 R6) 不支援執行個體儲存體磁碟區，其他執行個體類型 (如 M5d、C6gd 和 R6gd) 則支援執行個體儲存體磁

碟區。您無法將超過執行個體類型所支援的執行個體儲存體磁碟區數量，連接至執行個體。對於支援執行個體儲存體磁碟區的執行個體類型，執行個體儲存體磁碟區的數量和大小會依執行個體大小而異。例如，`m5d.large` 支援 1 個 75 GB 的執行個體儲存體磁碟區，而 `m5d.24xlarge` 支援 4 個 900 GB 的執行個體儲存體磁碟區。

對於具有執行個NVMe體存放磁碟區的執行個體類型，所有支援的執行個體存放磁碟區都會在啟動時自動附加至執行個 對於具有非執NVMe行個體儲存磁碟區的執行個體類型 (例如 C1、C3、M1、M2、R3、D2、H1、I2、X1 和 X1e)，您必須手動為要在啟動時連接的執行個體儲存磁碟區指定區塊裝置對映。執行個體啟動後，您必須先格式化並掛載連接的執行個體儲存體磁碟區，才能使用它們。啟動執行個體之後，即無法連接執行個體儲存體磁碟區。

某些執行個體類型使用NVMe或SATA固態硬碟 (SSD)，而其他執行個體類型則使用SATA基於硬碟 (HDD)。SSDs以極低的延遲提供高隨機 I/O 效能，但您不需要在執行個體終止時保留資料，或者您可以利用容錯架構的優勢。如需詳細資訊，請參閱SSDAmazon EC2 執行個體的執行個體存放量。

NVMe執行個體儲存磁碟區和某些HDD執行個體儲存磁碟區上的資料會在靜態時加密。如需詳細資訊，請參閱Amazon 的數據保護 EC2。

可用執行個體儲存體磁碟區

Amazon EC2 執行個體類型指南針對每個受支援的執行個體類型提供執行個體存放區磁碟區的數量、大小、類型和效能優化。如需詳細資訊，請參閱下列內容：

- 執行個體存放區規格 — 一般用
- 執行個體存放區規格 — 最佳化
- 執行個體存放區規格 — 記憶體
- 執行個體存放區規格 — 儲存優
- 執行個體商店規格 — 加速運算
- 執行個體存放區規格 — 高效能
- 執行個體存放區規格 — 上一代

若要使用擷取執行個體儲存磁碟區資訊 AWS CLI

您可以使用describe-instance-types AWS CLI 指令來顯示執行個體類型的相關資訊，例如其執行個體儲存磁碟區。下列範例顯示具有執行個體存放區磁碟區之所有 R5 執行個體的執行個體儲存體總大小。

```
aws ec2 describe-instance-types \
```

```
    --filters "Name=instance-type,Values=r5*" "Name=instance-storage-
supported,Values=true" \
    --query "InstanceTypes[].[InstanceType, InstanceStorageInfo.TotalSizeInGB]" \
    --output table
```

範例輸出

```
----------------------------
|  DescribeInstanceTypes  |
+----------------+--------+
|  r5ad.24xlarge |  3600  |
|  r5ad.12xlarge |  1800  |
|  r5dn.8xlarge  |  1200  |
|  r5ad.8xlarge  |  1200  |
|  r5ad.large    |  75    |
|  r5d.4xlarge   |  600   |
   . . .
|  r5dn.2xlarge  |  300   |
|  r5d.12xlarge  |  1800  |
+----------------+--------+
```

下列範例顯示指定執行個體類型的完整執行個體儲存體詳細資訊。

```
aws ec2 describe-instance-types \
    --filters "Name=instance-type,Values=r5d.4xlarge" \
    --query "InstanceTypes[].InstanceStorageInfo"
```

範例輸出顯示此執行個體類型有兩個 300 GB NVMe SSD 磁碟區，總計 600 GB 的執行個體儲存體。

```
[
    {
        "TotalSizeInGB": 600,
        "Disks": [
            {
                "SizeInGB": 300,
                "Count": 2,
                "Type": "ssd"
            }
        ],
        "NvmeSupport": "required"
    }
]
```

SSDAmazon EC2 執行個體的執行個體存放量

與其他執行個體存放磁碟區一樣，您必須在啟動SSD執行個體時對應執行個體的執行個體儲存磁碟區。SSD執行個體磁碟區上的資料只會在其關聯執行個體的生命週期內保留。如需詳細資訊，請參閱將執行個體存放磁碟區新增至 Amazon EC2 執行個。

NVMeSSD卷

某些執行個體提供非揮發性記憶體 express (NVMe) 固態硬碟 (SSD) 執行個體儲存磁碟區。關於每種執行個體類型所支援的執行個體存放磁碟區類型，詳細資訊請參閱Amazon EC2 執行個體的執行個體存放量限制。

執行個體儲存NVMe體上的資料會使用在執XTS行個體上的硬體模組中實作 AES -256 區塊密碼加密。加密金鑰是使用硬體模組產生的，每個NVMe執行個體儲存裝置都是唯一的。所有加密金鑰會在執行個體停止或終止時銷毀，且無法復原。您無法停用此加密，也無法提供您自己的加密金鑰。

Linux 執行個體

若要存取NVMe磁碟區，必須安裝NVMe驅動程式。以下內容AMIs符合此要求：

- AL2023
- Amazon Linux 2
- Amazon Linux AMI 及更高版本
- Ubuntu 14.04 或更新版本 (帶 `linux-aws` 核心)

> **Note**
>
> AWS 以重力為基礎的執行個體類型需要 Ubuntu 18.04 或更新版本搭配核心 `linux-aws`

- Red Hat Enterprise Linux 7.4 或更新版本
- SUSELinux 企業伺服器 12 SP2 或更新版本
- CentOS 7.4.1708 或更新版本
- 免費 BSD 11.1 或更高版本
- GNUDebian 9 或更新版本

- Bottlerocket

連線到執行個體之後，您可以使用lspci指令列出NVMe裝置。以下是支援四個NVMe裝置的`i3.8xlarge`執行個體輸出範例。

```
[ec2-user ~]$ lspci
00:00.0 Host bridge: Intel Corporation 440FX - 82441FX PMC [Natoma] (rev 02)
00:01.0 ISA bridge: Intel Corporation 82371SB PIIX3 ISA [Natoma/Triton II]
00:01.1 IDE interface: Intel Corporation 82371SB PIIX3 IDE [Natoma/Triton II]
00:01.3 Bridge: Intel Corporation 82371AB/EB/MB PIIX4 ACPI (rev 01)
00:02.0 VGA compatible controller: Cirrus Logic GD 5446
00:03.0 Ethernet controller: Device 1d0f:ec20
00:17.0 Non-Volatile memory controller: Device 1d0f:cd01
00:18.0 Non-Volatile memory controller: Device 1d0f:cd01
00:19.0 Non-Volatile memory controller: Device 1d0f:cd01
00:1a.0 Non-Volatile memory controller: Device 1d0f:cd01
00:1f.0 Unassigned class [ff80]: XenSource, Inc. Xen Platform Device (rev 01)
```

如果您使用的是受支援的作業系統，但看不到NVMe裝置，請使用下列指令確認NVMe模組是否已載入。

- Amazon Linux, Amazon Linux 2, Ubuntu 14/16, 紅帽企業 Linux, Linux 企業伺服器, CentOS 7 SUSE

  ```
  $ lsmod | grep nvme
  nvme           48813  0
  ```

- Ubuntu 18

  ```
  $ cat /lib/modules/$(uname -r)/modules.builtin | grep nvme
  s/nvme/host/nvme-core.ko
  kernel/drivers/nvme/host/nvme.ko
  kernel/drivers/nvmem/nvmem_core.ko
  ```

磁NVMe碟區符合 NVMe 1.0e 規格。您可以將這些指NVMe令與NVMe磁碟區搭配使用。如果使用 Amazon Linux，您可以使用 `nvme-cli` 命令，從儲存庫安裝 yum install 套件。使用其他支援的 Linux 版本時，如果映像中未提供，您可以下載 `nvme-cli` 套件。

Windows 執行個體

AMIs適用於下列作業系統的最新 AWS Windows 包含用於與SSD執行個體存放區磁碟區互動的 AWS NVMe驅動程式，這些磁碟區會公開為區NVMe塊裝置，以提高效能：

- Windows Server 2022

- Windows Server 2019
- Windows Server 2016
- Windows Server 2012 R2

連線到執行個體後，您可以確認您在 [磁碟管理員] 中看到磁碟NVMe區。在任務列上開啟 Windows 標誌的內容 (按右鍵) 選單，然後選擇 Disk Management (磁碟管理)。

Amazon AMIs 提供的 AWS 窗口包括 AWS NVMe驅動程序。如果您沒有使用最新的 AWS WindowsAMIs，則可以安裝當前的 AWS NVMe驅動程序。

非NVMeSSD磁碟區

下列執行個體支援使用非提供高隨機 I/O 效能的執行個體儲存磁碟區：C3、I2、M3、R3 和 X1。NVMe SSDs關於每種執行個體類型所支援的執行個體存放磁碟區，詳細資訊請參閱Amazon EC2 執行個體的執行個體存放量限制。

SSD型執行個體儲存磁碟區 I/O 效能

當您填滿執行個體的SSD基於執行個體的執行個體存放區磁碟區時，可達到IOPS的寫入次數會減少。這是由於SSD控制器必須執行額外的工作來尋找可用空間、重寫現有資料，以及清除未使用的空間，以便重寫。這個記憶體回收程序會導致內部寫入放大SSD，表示為寫入作業與使用者SSD寫入作業的比率。假如寫入操作不是 4,096 位元組的倍數或未符合 4,096 位元組的邊界，效能下降的幅度會更明顯。如果您編寫的字節或字節數量較小，則SSD控制器必須讀取周圍的數據並將結果存儲在新位置。此模式會造成寫入放大大幅增加，提高延遲，因而使 I/O 效能巨幅下降。

SSD控制器可以使用多種策略來減少寫入放大的影響。其中一種策略是在執行個SSD體儲存體中保留空間，以便控制器能夠更有效率地管理可用於寫入作業的空間。此策略稱為過度佈建。提供給執行個體的SSD基於執行個體的執行個體儲存磁碟區沒有任何保留空間以進行過度佈建 為了減少寫入放大，我們建議您將 10% 的磁碟區保留未分割，以便SSD控制器可以將其用於過度佈建。這樣做雖然會減少可用的儲存空間，但可提高效能，即使磁碟即將用完所有容量。

例如支援的儲存磁碟區TRIM，您可以在不再需要寫入的資料時使用TRIM指令來通知SSD控制器。這會讓控制器有更多的可用空間能夠使用，可減少寫入放大現象，並提升效能。如需詳細資訊，請參閱TRIM支援執行個體儲存區。

TRIM支援執行個體儲存區

某些執行個體類型支援SSD磁碟區TRIM。如需詳細資訊，請參閱Amazon EC2 執行個體的執行個體存放量限制。

> **Note**
>
> (僅適用於 Windows 執行個體) 從 AWS 光伏驅動程式 7.3.0 版起，執行視窗伺服器 2012 R2 支援TRIM的執行個體。執行舊版 Windows 伺服器的執行個體不支援TRIM。

支援的執行個體儲存磁碟區TRIM會在配置到您的執行個體之前完全修剪。在執行個體啟動時，這些磁碟區不會格式化為任何檔案系統，因此您必須先將這些磁碟區格式化之後，才能掛載和使用這些磁碟區。若要更快速地存取這些磁碟區，請在格式化磁碟區時略過此TRIM作業。

(Windows 執行個體) 若要在初始格式化期間暫時停用TRIM支援，請使用`fsutil behavior set DisableDeleteNotify 1`指令。格式化完成後，請使用重新啟用TRIM`fsutil behavior set DisableDeleteNotify 0`支援。

使用支援的執行個體儲存磁碟區TRIM，您可以使用TRIM指令在不再需要寫入的資料時通知SSD控制器。這會讓控制器有更多的可用空間能夠使用，可減少寫入放大現象，並提升效能。在 Linux 執行個體上，使用`fstrim`命令定期啟用TRIM。在 Windows 執行個體上，使用`fsutil behavior set DisableDeleteNotify 0`指令確保在正常作業期間啟用TRIM支援。

將執行個體存放磁碟區新增至 Amazon EC2 執行個

對於具有執行個NVMe體存放磁碟區的執行個體類型，所有支援的執行個體存放磁碟區都會在啟動時自動附加至執行個 執行個體啟動時，系統會自動列舉這些執行個體儲存體磁碟區並指派裝置名稱。

對於具有非執NVMe行個體儲存磁碟區的執行個體類型 (例如 C1、C3、M1、M2、R3、D2、H1、I2、X1 和 X1e)，您必須手動為要在啟動時連接的執行個體儲存磁碟區指定區塊裝置對映。您可以在執行個體啟動要求中或在AMI用來啟動執行個體的中指定區塊裝置對應。區塊型裝置映射包含裝置名稱和對應的磁碟區。如需詳細資訊，請參閱 Amazon EC2 執行個體上磁碟區的區塊裝置對映

> **Important**
>
> 您只能在啟動執行個體時，將執行個體儲存體磁碟區連接至執行個體。啟動執行個體之後，即無法將執行個體存放磁碟區連結到執行個體。

在啟動執行個體之後，您必須先確定執行個體的執行個體存放磁碟區已經過格式化並掛載，才能使用這些磁碟區。以執行個體存放區為後端的執行個體，其根磁碟區會自動掛載。

根磁碟區的考量

區塊型設備映射一律會指定執行個體的根磁碟區。根磁碟區一律會自動掛載。

Linux 執行個體 — 根磁碟區可以是 Amazon EBS 磁碟區或執行個體存放區磁碟區。對於具有根磁碟區執行個體儲存磁碟區的執行個體，此磁碟區的大小會有所不同AMI，但最大大小為 10 GB。如需詳細資訊，請參閱Root device type (根設備類型)。

視窗執行個體 — 根磁碟區必須是 Amazon EBS 磁碟區。根磁碟區不支援執行個體存放區。

目錄

- 將實例存儲卷添加到 Amazon EC2 AMI
- 在啟動期間將執行個體存放磁碟區新增至 Amazon EC2 執行
- 讓可在 Amazon 執行個體上使用的執行個EC2體存放區磁碟區

將實例存儲卷添加到 Amazon EC2 AMI

您可以AMI使用包含執行個體存放區磁碟區的區塊裝置對應來建立。

如果您使用指定執行個體儲存磁碟區區塊裝置對映來啟動支援非執行AMI個體儲存磁碟區的執行個體，則執行個體會包含執行個體儲存磁碟區。NVMe如果中的執行個體儲存磁碟區塊裝置對映數目AMI超過執行個體可用的執行個體存放磁碟區數目，則會忽略額外的執行個體儲存磁碟區塊裝置對應。

如果您使用指定執行個體儲存磁碟區塊裝置對映來啟動支援NVMe執行AMI個體儲存磁碟區的執行個體，則會忽略執行個體存放區磁碟區塊裝置對映。支NVMe援執行個體儲存磁碟區的執行個體會取得所有支援的執行個體儲存磁碟區，無論執行個體啟動要求中指定的區塊裝置對應為何，以及AMI.

考量事項

- 對於 M3 執行個體，請在執行個體的區塊裝置對映中指定執行個體儲存磁碟區，而不是指定AMI. Amazon EC2 可能會忽略中的執行個體存放區磁碟區塊裝置對應AMI。
- 啟動執行個體時，您可以省略區AMI塊裝置對應中指定的非NVMe執行個體儲存磁碟區，或新增執行個體儲存磁碟區。

Console

使用控制台將實例存儲卷添EBS加到 AMI Amazon 支持

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。

2. 在導覽窗格中，選取 Instances (執行個體)，然後選取該執行個體。
3. 選擇 Actions (動作)、Image and templates (映像和範本)、Create image (建立映像)。
4. 在 建立影像 頁面，為您的影像輸入有意義的名稱和說明。
5. 針對每個要新增的執行個體存放區的磁碟區，選取 Add New Volume (新增新磁碟區)、從 Volume Type (磁碟區類型) 選取一個執行個體存放區的磁碟區，並從 Device (裝置) 選取一個裝置名稱。(如需詳細資訊，請參閱「Amazon EC2 執行個體上磁碟區的裝置名稱」。) 可用執行個體存放磁碟區的數量，會視執行個體類型而定。對於具有NVMe執行個體儲存磁碟區的執行個體，這些磁碟區的裝置對應取決於作業系統列舉磁碟區的順序。
6. 選擇 Create image (建立映像)。

AWS CLI

若要AMI使用指令列將執行個體儲存磁碟區新增至

您可以使用下列其中一個命令。如需關於這些命令列介面的詳細資訊，請參閱 訪問 Amazon EC2。

- create-image 或 register-image (AWS CLI)
- New-EC2Image和 Register-EC2Image（AWS Tools for Windows PowerShell）

在啟動期間將執行個體存放磁碟區新增至 Amazon EC2 執行

當您啟動具有非執行個體儲存磁碟區的NVMe執行個體類型時，例如 C1、C3、M2、M2、R3、D2、H1、I2、X1 和 X1e，您必須為要在啟動時連接的執行個體存放磁碟區指定區塊裝置對映。必須在執行個體啟動要求或AMI用於啟動執行個體的中指定區塊裝置對應。

如果AMI包含執行個體儲存磁碟區的區塊裝置對應，則您不需要在執行個體啟動要求中指定區塊裝置對應，除非您需要的執行個體儲存磁碟區數量超過包含在中AMI。

如果AMI不包含執行個體儲存磁碟區的區塊裝置對映，則您必須在執行個體啟動要求中指定區塊裝置對應。

對於具有執行個NVMe體存放磁碟區的執行個體類型，所有支援的執行個體存放磁碟區都會在啟動時自動附加至執行個

考量事項

- 如果是 M3 執行個體，即使未在執行個體的區塊型設備映射中指定這些執行個體，您可能也會獲得執行個體存放磁碟區。

若要在執行個體啟動要求中指定區塊型裝置映射，請使用下列任一方法。

Amazon EC2 console

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在儀表板中，選擇啟動執行個體。
3. 在「應用程式和作業系統映像」區段中，選取AMI要使用的。
4. 在設定儲存體區段中，執行個體儲存體磁碟區區段會列出可連結至該執行個體的執行個體儲存體磁碟區。可用執行個體存放磁碟區的數量，會視執行個體類型而定。
5. 對於要連接的每個執行個體儲存體磁碟區，針對裝置名稱，請選取要使用的裝置名稱。
6. 根據需要進行剩餘的執行個體設定，然後選擇啟動執行個體。

Command line

您可以使用下列任一命令，搭配對應的選項。

- `--block-device-mappings` 搭配 run-instances (AWS CLI)
- `-BlockDeviceMapping`與 New-EC2Instance（AWS Tools for Windows PowerShell）

讓可在 Amazon 執行個體上使用的執行個EC2體存放區磁碟區

啟動具有連接執行個體儲存體磁碟區的執行個體之後，您必須先掛接磁碟區，才能存取這些磁碟區。

Linux 執行個體

啟動執行個體後，您可以使用您選擇的檔案系統來格式化磁碟區。

您可以依照下列程序所述檢視和掛接執行個體儲存磁碟區。

在 Linux 上讓執行個體存放磁碟區變成可用

1. 使用用SSH戶端 Connect 至執行個體。如需詳細資訊，請參閱使用 Connect 到您的 Linux 執行個體 SSH。
2. 使用 `df -h` 指令來檢視磁碟區是否已格式化和掛載。

```
$ df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        3.8G   72K  3.8G   1% /dev
tmpfs           3.8G     0  3.8G   0% /dev/shm
```

```
/dev/nvme0n1p1  7.9G  1.2G  6.6G  15% /
```

3. 使用 `lsblk` 來檢視是否有任何磁碟區已在啟動時映射，但尚未格式化和掛載。

```
$ lsblk
NAME          MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
nvme0n1       259:1    0    8G  0 disk
##nvme0n1p1   259:2    0    8G  0 part /
##nvme0n1p128 259:3    0    1M  0 part
nvme1n1       259:0    0 69.9G  0 disk
```

4. 若要只針對映射的執行個體存放磁碟區來進行格式化和掛載，請執行下列動作：

   a. 使用 `mkfs` 指令來在裝置上建立檔案系統。

   ```
   $ sudo mkfs -t xfs /dev/nvme1n1
   ```

   b. 使用 `mkdir` 指令來建立用來掛載裝置的目錄。

   ```
   $ sudo mkdir /data
   ```

   c. 使用 `mount` 指令，將裝置掛載到新建立的目錄上。

   ```
   $ sudo mount /dev/nvme1n1 /data
   ```

Windows 執行個體

對於 Windows 執行個體，我們會使用NTFS檔案系統重新格式化執行個體儲存磁碟區。

您可以使用 Windows 磁碟管理來檢視執行個體儲存磁碟區。如需詳細資訊，請參閱列出非NVMe磁碟。

手動掛載執行個體存放區磁碟區

1. 選擇開始，輸入電腦管理，然後按 Enter。
2. 在左側面板中，選擇磁碟管理。
3. 如果系統提示您初始化磁碟區，請選取要初始化的磁碟區，根據您的使用案例選取所需的磁碟分割類型，然後選取 確定。
4. 在磁碟區清單中，以滑鼠右鍵按一下要裝載的磁碟區，然後選擇新增簡單磁碟區。
5. 在精靈上，選擇下一步。

6. 在「指定磁碟區大小」畫面上，選擇下一步以使用最大磁碟區大小。或者，選擇介於最小和最大磁碟空間之間的磁碟區大小。
7. 在「指派磁碟機代號或路徑」畫面上，執行下列其中一項作業，然後選擇下一步。
   - 若要使用磁碟機代號掛接磁碟區，請選擇指派下列磁碟機代號，然後選擇要使用的磁碟機代號。
   - 若要將磁碟區掛載為資料夾，請在下列空白資NTFS料夾中選擇 [掛載]，然後選擇 [瀏覽] 建立或選取要使用的資料夾。
   - 若要在沒有磁碟機代號或路徑的情況下掛接磁碟區，請選擇不要指派磁碟機代號或磁碟機路徑。
8. 在「格式化磁碟分割」畫面上，指定是否要格式化磁碟區。如果您選擇格式化磁碟區，請選擇所需的檔案系統和單位大小，然後指定磁碟區標籤。
9. 依序選擇下一步，完成。

有關如何在重新開機後自動掛接連接的磁碟區的說明，請參閱 Amazon EBS 使用者指南中的重新開機後自動掛接連接的磁碟區。

為 Amazon EC2 M1 和 C1 執行個體啟用執行個體存放區交換磁碟區

> **Note**
>
> 本主題僅適用`c1.medium`於 `m1.small` Linux 執行個體。

`c1.medium`和`m1.small`執行個體類型的實體記憶體數量有限。因此，它們會在啟動時獲得 900 MiB 交換磁碟區，以作為 Linux 系統的虛擬記憶體或交換空間。當系統需要比實體配置更多的記憶體時，可以使用 Linux 的置換空間。啟用置換空間時，Linux 系統可以將不常使用的記憶體分頁，從實體記憶體換到置換空間 (專用磁區或是現有檔案系統中的調換檔)，以釋出空間，給需要高速存取的記憶體分頁使用。

> **Note**
>
> - 使用交換空間進行內存分頁不如使用RAM快速或有效。如果您的工作負載定期將記憶體分頁到交換空間，您應該考慮遷移到具有更多RAM空間的較大執行個體類型。如需詳細資訊，請參閱Amazon EC2 實例類型更改。
> - 雖然 Linux 核心將此置換空間視為根設備上的分割區，但該空間實際上為獨立的執行個體存放磁碟區，而不論根設備磁碟區是何種類型。

Amazon Linux 會自動啟用並使用此交換空間，但您AMI可能需要一些額外的步驟來識別和使用此交換空間。若要查看執行個體是否正在使用置換空間，您可以使用 swapon -s 命令。

```
[ec2-user ~]$ swapon -s
Filename                                Type            Size    Used    Priority
/dev/xvda3                              partition       917500  0       -1
```

上述的執行個體連結和啟用了 900 MiB 的智慧磁碟區。如果使用此指令未列出置換磁碟區，可能需要啟用該設備的置換空間。使用 lsblk 命令來檢查可用的磁碟。

```
[ec2-user ~]$ lsblk
NAME  MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
xvda1 202:1    0    8G  0 disk /
xvda3 202:3    0  896M  0 disk
```

在此處，置換磁碟區 `xvda3` 可供執行個體使用，但是並未啟用 (請注意，`MOUNTPOINT` 欄位為空白)。您可以利用 swapon 命令來啟用置換磁碟區。

> **Note**
>
> 您必須在 `/dev/` 命令所列出的裝置名稱前方，加上 lsblk。裝置可能需有不同的名稱，例如 `sda3`、`sde3` 或 `xvde3`。在下列的指令中使用系統的裝置名稱。

```
[ec2-user ~]$ sudo swapon /dev/xvda3
```

現在置換空間應該會顯示 lsblk 和 swapon -s 輸出中。

```
[ec2-user ~]$ lsblk
NAME  MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
xvda1 202:1    0    8G  0 disk /
xvda3 202:3    0  896M  0 disk [SWAP]
[ec2-user ~]$ swapon -s
Filename                                Type            Size    Used    Priority
/dev/xvda3                              partition       917500  0       -1
```

您也需要編輯 `/etc/fstab` 檔案，讓此一置換空間在每次系統開機時自動啟用。

```
[ec2-user ~]$ sudo vim /etc/fstab
```

將下列的指令行附加到 /etc/fstab 檔案之後 (使用系統的置換裝置名稱)：

```
/dev/xvda3       none    swap    sw  0       0
```

將執行個體存放磁碟區做為置換空間使用

任何執行個體存放磁碟區皆可做為置換空間使用。例如，m3.medium執行個體類型包含適合交換空間的 4 GB SSD 執行個體存放磁碟區。如果執行個體存放磁碟區的容量大上許多 (例如 350 GB)，您可以考慮將磁碟區分割出一個較小的 4 到 8 GB 置換磁區，其他的部分則做為資料磁碟區。

> **Note**
>
> 此程序僅適用於支援執行個體儲存體的執行個體類型。如需支援的執行個體類型清單，請參閱 Amazon EC2 執行個體的執行個體存放量限制。

1. 列出連結至執行個體的區塊型儲存設備，以取得執行個體存放磁碟區的裝置名稱。

   ```
   [ec2-user ~]$ lsblk -p
   NAME       MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
   /dev/xvdb  202:16   0   4G  0 disk /media/ephemeral0
   /dev/xvda1 202:1    0   8G  0 disk /
   ```

   在此範例中，執行個體存放磁碟區為 /dev/xvdb。因為這是 Amazon Linux 執行個體，因此執行個體存放磁碟區會經過格式化並掛載於 /media/ephemeral0；不是所有的 Linux 作業系統都會自動執行這項動作。

2. (選用) 如果執行個體存放磁碟區已掛載 (會在 MOUNTPOINT 命令輸出中列出 lsblk)，請利用下列指令來卸載該磁碟區。

   ```
   [ec2-user ~]$ sudo umount /dev/xvdb
   ```

3. 使用 mkswap 命令，在裝置上設定 Linux 置換區。

   ```
   [ec2-user ~]$ sudo mkswap /dev/xvdb
   mkswap: /dev/xvdb: warning: wiping old ext3 signature.
   Setting up swapspace version 1, size = 4188668 KiB
   no label, UUID=b4f63d28-67ed-46f0-b5e5-6928319e620b
   ```

4. 啟用新的置換空間。

```
[ec2-user ~]$ sudo swapon /dev/xvdb
```

5. 確認新的置換空間正在使用。

```
[ec2-user ~]$ swapon -s
Filename    Type  Size Used Priority
/dev/xvdb                               partition 4188668 0 -1
```

6. 編輯 /etc/fstab 檔案，讓此一置換空間在每次系統開機時自動啟用。

```
[ec2-user ~]$ sudo vim /etc/fstab
```

如果 /etc/fstab 檔案具有 /dev/xvdb (或 /dev/sdb) 的項目，請修改為和下列行相同；如果該檔案不具有此裝置的項目，請將下列行附加於 /etc/fstab 檔案之後 (讓系統使用置換裝置的名稱)：

```
/dev/xvdb       none    swap    sw  0       0
```

> **Important**
>
> 當執行個體停止或休眠時，執行個體存放磁碟區上的資料會遺失；這包括在 Step 3 中所建立的執行個體存放區置換空間格式。如果停止和重新啟動執行個體，而此執行個體已設定使用執行個體存放區置換空間，則您必須在新的執行個體存放磁碟區上，重複執行 Step 1 到 Step 5。

初始化 Amazon EC2 實例上的實例存儲卷

> **Note**
>
> 本主題僅適用於 Linux 執行個體。

由於 Amazon EC2 虛擬化磁碟的方式，第一次寫入某些執行個體存放磁碟區上任何位置的執行速度會比後續寫入的速度要慢。對大多數應用程式而言，在執行個體的整個生命週期內分攤此效能減損是可以接受的。不過，如果要求高度磁碟效能，建議在正式生產使用之前，藉由寫入每個磁碟位置一次，來將磁碟初始化。

> **Note**
>
> 具有直接連結固態硬碟 (SSD) 和TRIM支援的執行個體類型可在啟動時提供最大效能，而無需初始化。關於適用於每種執行個體類型的執行個體存放區，詳細資訊請參閱 Amazon EC2 執行個體的執行個體存放量限制。

如果您需要更大的延遲或輸送量彈性，我們建議您使用 Amazon EBS。

若要將執行個體存放磁碟區初始化，請根據要初始化的存放區 (例如 dd 或 /dev/sdb)，使用下列 /dev/nvme1n1 指令來進行。

> **Note**
>
> 請務必在執行此指令之前卸載磁碟。
> 初始化作業可能需要很長的時間來完成 (如果是超大型的執行個體，約需要 8 小時)。

若要將執行個體存放磁碟區初始化，請針對 m1.large、m1.xlarge、c1.xlarge、m2.xlarge、m2.2xlarge 和 m2.4xlarge 執行個體類型使用下列指令：

```
dd if=/dev/zero of=/dev/sdb bs=1M
dd if=/dev/zero of=/dev/sdc bs=1M
dd if=/dev/zero of=/dev/sdd bs=1M
dd if=/dev/zero of=/dev/sde bs=1M
```

若要在所有的執行個體存放磁碟區上同時進行初始化，請使用下列指令：

```
dd if=/dev/zero bs=1M|tee /dev/sdb|tee /dev/sdc|tee /dev/sde > /dev/sdd
```

配置驅動器通過寫入每個驅動器位置RAID初始化它們。設定以軟體為基礎的時候RAID，請務必變更最低重建速度：

```
echo $((30*1024)) > /proc/sys/dev/raid/speed_limit_min
```

Amazon EC2 執行個體的根磁碟區

當您啟動執行個體時，系統會為執行個體建立根磁碟區。根磁碟區內含用來啟動執行個體的映像。每個執行個體都有一個根磁碟區。您可以在啟動期間或之後將儲存磁碟區新增至執行個體。

您用來啟動執行個體的決定了根磁碟區的類型。AMI您可以從支援 Amazon 的執行個體 AMI (Linux 和 Windows 執行個體) 或存放區EBS支援的執行個體 AMI (僅限 Linux 執行個體) 啟動執行個體。您可以對每種類型執行的操作之間存在顯著差異AMI。如需這些差異的詳細資訊，請參閱 Root device type (根設備類型)。

我們建議您使用 Amazon AMIs 支援EBS，因為這些執行個體的啟動速度更快，並使用持續性儲存。

系統會為根磁碟區保留特定的裝置名稱。如需詳細資訊，請參閱Amazon EC2 執行個體上磁碟區的裝置名稱。

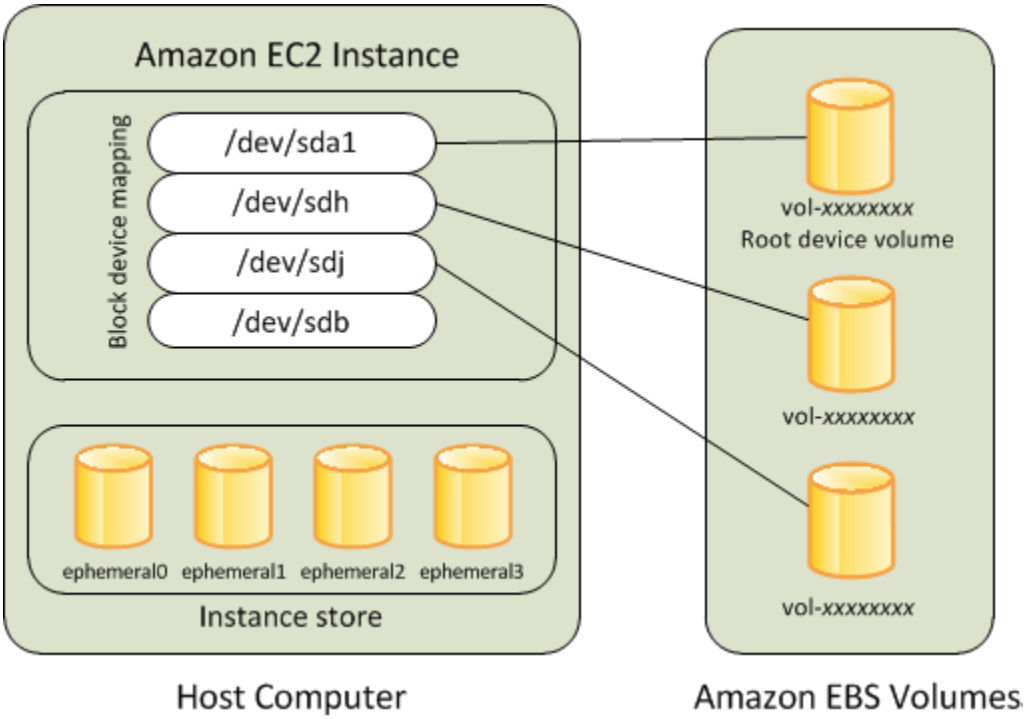
目錄

- Amazon EBS 支持的實例
- 執行個體商店支援執行個體 (僅限 Linux 執行個體)
- 在 Amazon 實EC2例終止後保留 Amazon EBS 根卷
- 取代 Amazon EC2 執行個體的根磁碟區而不停止它

Amazon EBS 支持的實例

使用 Amazon 作EBS為根磁碟區的執行個體會自動附加一個 Amazon EBS 磁碟區。當您啟動 Amazon EBS 支援的執行個體時，我們會為AMI您使用的每個 Amazon EBS 快照建立一個 Amazon EBS 磁碟區。視執行個體類型而定，您可以選擇使用其他 Amazon EBS 磁碟區或執行個體存放區磁碟區。

Amazon EBS 支援的執行個體可以停止並稍後重新啟動，而不會影響儲存在連接磁碟區中的資料。Amazon EBS 支援的執行個體處於停止狀態時，您可以執行各種執行個體和磁碟區相關的任務。例如，您可修改執行個體的屬性、變更其大小、更新其正使用的核心，或者您可將根磁碟區連接至另一個運作中執行個體以進行除錯或其他用途。如需詳細資訊，請參閱 Amazon EBS 磁碟區。



限制

您無法使用`st1`或`sc1`EBS磁碟區做為根磁碟區。

執行個體失敗

如果 Amazon EBS 備份的執行個體失敗，您可以按照下列其中一種方法恢復工作階段：

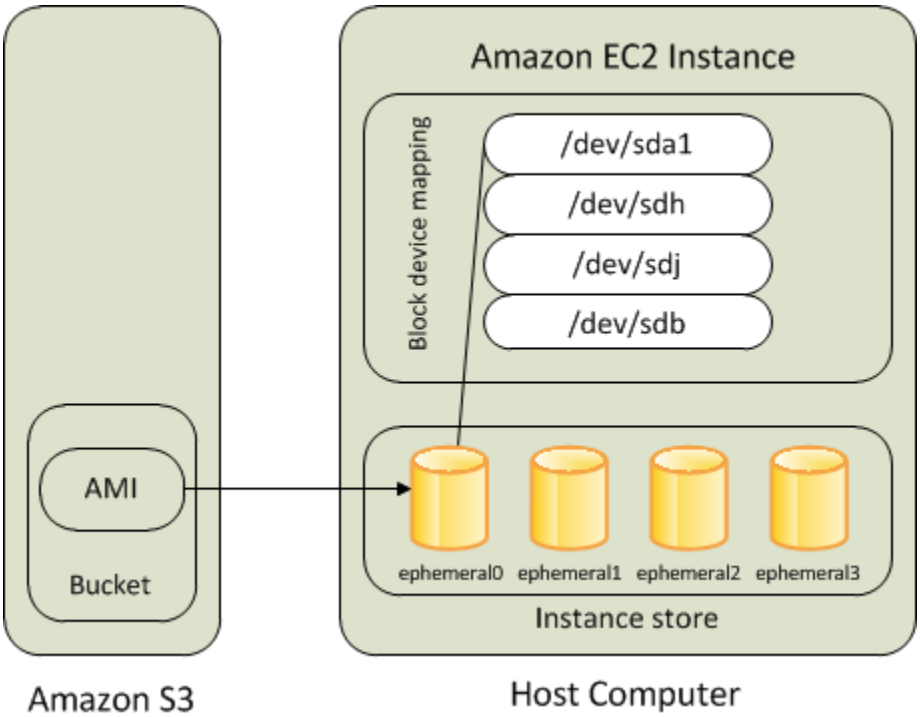
- 停止然後再次啟動 (請先嘗試此方法)。
- 自動為所有相關磁碟區建立快照，並建立新的AMI。如需詳細資訊，請參閱創建一個 Amazon 支EBS持 AMI。
- 採取下列步驟將磁碟區連接至新的執行個體：
  1. 建立根磁碟區的快照。
  2. AMI使用快照註冊新的。
  3. 從新執行個體啟動新執行個體AMI。
  4. 從舊執行個體中分離剩餘的 Amazon EBS 磁碟區。
  5. 將 Amazon EBS 磁碟區重新連接到新執行個體。

執行個體商店支援執行個體 (僅限 Linux 執行個體)

> **Note**
>
> Windows 執行個體不支援執行個體儲存支援的根磁碟區。

將執行個體儲存體用於根磁碟區的執行個體，自動具備一個或多個可用的執行個體儲存體磁碟區，其中一個磁碟區會當成根磁碟區使用。執行個體啟動時，用於啟動執行個體的映像會複製到根磁碟區。請注意，您可根據執行個體類型選擇使用其他執行個體存放區磁碟區。

只要執行個體持續運作，執行個體存放區磁碟區上的資料就會永久保存，但若執行個體被終止 (執行個體後端執行個體不支援 Stop (停止) 動作) 或故障 (如底層磁碟區出現問題)，這些資料就會被刪除。如需詳細資訊，請參閱Amazon EC2 執行個體的執行個體存放臨時區塊儲。



支援的執行個體類型

只有下列執行個體類型支援執行個體儲存磁碟區做為根磁碟區：
C1、C3、D2、I2、M2、M2、M3、R3 和 X1。

執行個體失敗

執行個體後端執行個體故障或終止後，將無法還原。如果您打算使用 Amazon EC2 執行個體商店支援的執行個體，我們強烈建議您將執行個體存放區上的資料分配到多個可用區域。您也應將執行個體存放區磁碟區上的重要資料，定期備份為持久性儲存。

在 Amazon 實EC2例終止後保留 Amazon EBS 根卷

依預設，執行個體終止時，會刪除執行個體的 Amazon EBS 根磁碟區。您可以變更預設行為，以確保 Amazon EBS 根磁碟區在執行個體終止後仍然存在。若要變更預設行為，請將`DeleteOnTermination`屬性設定為`false`。您可以在執行個體啟動時或稍後執行此操作。

任務

- 將根磁碟區設定為執行個體啟動期間持續存在
- 將根磁碟區設定為若為現有執行個體則持續存在
- 確認根磁碟區已設定為持續存在

將根磁碟區設定為執行個體啟動期間持續存在

您可以將根磁碟區設定為在啟動執行個體時持續存在。

Console

將根磁碟區設定為您使用主控台啟動執行個體時持續存在

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)，然後選擇 Launch instances (啟動執行個體)。
3. 選擇 Amazon Machine 映像 (AMI)、選擇和執行個體類型、選擇 key pair，以及設定網路設定。
4. 針對設定儲存，選擇進階。
5. 展開根磁碟區。
6. 針對在終止時刪除，選擇否。
7. 當您完成對執行個體的設定後，選擇啟動執行個體。

AWS CLI

若要將根磁碟區設定為在使用 AWS CLI

使用 run-instances 命令，並包括將 `DeleteOnTermination` 屬性設定為 `false` 的區塊型設備映射。

```
aws ec2 run-instances --block-device-mappings file://mapping.json ...other
 parameters...
```

在 `mapping.json` 中指定下列內容。

```
[
    {
        "DeviceName": "/dev/sda1",
        "Ebs": {
            "DeleteOnTermination": false
        }
    }
]
```

Tools for Windows PowerShell

若要將根磁碟區設定為在使用 Windows 的工具啟動執行個體時持續存在 PowerShell

使用指New-EC2Instance令並包括將`DeleteOnTermination`屬性設定為的圖塊設備對映`false`。

```
C:\> $ebs = New-Object Amazon.EC2.Model.EbsBlockDevice
C:\> $ebs.DeleteOnTermination = $false
C:\> $bdm = New-Object Amazon.EC2.Model.BlockDeviceMapping
C:\> $bdm.DeviceName = "dev/xvda"
C:\> $bdm.Ebs = $ebs
C:\> New-EC2Instance -ImageId ami-0abcdef1234567890 -BlockDeviceMapping
 $bdm ...other parameters...
```

將根磁碟區設定為若為現有執行個體則持續存在

您可以將根磁碟區設定為持續執行中的執行個體。請注意，您無法使用 Amazon EC2 主控台完成此任務。

AWS CLI

若要將根磁碟區設定為保留現有執行個體，請使用 AWS CLI

將指modify-instance-attribute令與將`DeleteOnTermination`屬性設定為的圖塊設備對映搭配使用`false`。

```
aws ec2 modify-instance-attribute --instance-id i-1234567890abcdef0 --block-device-
mappings file://mapping.json
```

在 `mapping.json` 中指定下列內容。

```
[
    {
        "DeviceName": "/dev/xvda",
        "Ebs": {
            "DeleteOnTermination": false
        }
    }
]
```

Tools for Windows PowerShell

使用 AWS Tools for Windows PowerShell將根磁碟區設定為若為現有執行個體則持續存在

將指Edit-EC2InstanceAttribute令與將`DeleteOnTermination`屬性設定為的圖塊設備對映搭配使用`false`。

```
C:\> $ebs = New-Object Amazon.EC2.Model.EbsInstanceBlockDeviceSpecification
C:\> $ebs.DeleteOnTermination = $false
C:\> $bdm = New-Object Amazon.EC2.Model.InstanceBlockDeviceMappingSpecification
C:\> $bdm.DeviceName = "/dev/xvda"
C:\> $bdm.Ebs = $ebs
C:\> Edit-EC2InstanceAttribute -InstanceId i-1234567890abcdef0 -BlockDeviceMapping
 $bdm
```

確認根磁碟區已設定為持續存在

您可以使用 Amazon EC2 主控台或命令列工具確認根磁碟區設定為持續存在。

Console

使用 Amazon EC2 主控台確認根磁碟區設定為持續存在

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。

2. 在導覽窗格中，選取 Instances (執行個體)，然後選取執行個體。
3. 在 Storage (儲存體) 標籤的 Block devices (區塊型裝置) 下，找出根磁碟區的項目。如果 Delete on termination (終止時刪除) 為 No，則磁碟區會設定為持續存在。

AWS CLI

若要確認根磁碟區已設定為持續使用 AWS CLI

使用 describe-instance 命令，並驗證 `DeleteOnTermination` 回應元素中的 `BlockDeviceMappings` 屬性是否已設定為 `false`。

```
aws ec2 describe-instances --instance-id i-1234567890abcdef0
```

```
...
    "BlockDeviceMappings": [
    {
        "DeviceName": "/dev/sda1",
        "Ebs": {
            "Status": "attached",
            "DeleteOnTermination": false,
            "VolumeId": "vol-1234567890abcdef0",
            "AttachTime": "2013-07-19T02:42:39.000Z"
        }
    }
...
```

Tools for Windows PowerShell

若要確認根磁碟區已設定為持續使用 AWS Tools for Windows PowerShell

使用Get-EC2Instance並確認`BlockDeviceMappings`回應元素中的`DeleteOnTermination`屬性是否設定為`false`。

```
C:\> (Get-EC2Instance -InstanceId i-
i-1234567890abcdef0).Instances.BlockDeviceMappings.Ebs
```

取代 Amazon EC2 執行個體的根磁碟區而不停止它

Amazon EC2 可讓您替換執行中執行個體的根 Amazon EBS 磁碟區，同時保留下列項目：

- 儲存在執行個體存放區磁碟區上的資料 - 在還原根磁碟區後，執行個體存放區磁碟區仍會保持連接至執行個體。
- 存放在資料 (非根) Amazon EBS 磁碟區上的資料 — 還原根EBS磁碟區後，非根 Amazon 磁碟區會保持連接至執行個體。
- 網路設定 — 所有網路介面會保持連接至執行個體，並保留其 IP 位址、識別碼和附件IDs。當執行個體可用時，會清除所有擱置中的網路流量。此外，執行個體會保留在相同的實體主機上，因此它會保留其公有和私有 IP 位址和DNS名稱。
- IAM原則 — 會保留並強制執行與執行個體相關聯的IAM設定檔和策略 (例如以標籤為基礎的策略)。

目錄

- 根磁碟區取代的運作方式
- 考量事項
- 取代根磁碟區

根磁碟區取代的運作方式

當您取代執行個體的根磁碟區時，我們會建立根磁碟區取代工作。原始根磁碟區會從執行個體中分離，新根磁碟區會連接至其位置中的執行個體。執行個體的區塊型裝置映射已更新，以反映取代根磁碟區的 ID。

取代執行個體的根磁碟區時，您必須指定新磁碟區的快照來源。以下是可能的選項。

將根磁碟區還原至其原始狀態

此選項將目前的根磁碟區取代為根據用來建立該磁碟區的快照的磁碟區。

使用啟動狀態的考量

取代根磁碟區會獲得與原始根磁碟區相同的類型、大小和「在終止時刪除」屬性。

使用快照取代根磁碟區

此選項會以您指定的快照為基礎的取代磁碟區，取代目前的根磁碟區。例如，您先前從此根磁碟區建立的特定快照。如果您需要從根磁碟區損毀或客體作業系統中的網路組態錯誤所造成的問題中復原，此功能非常有用。

取代根磁碟區會獲得與原始根磁碟區相同的類型、大小和「在終止時刪除」屬性。

使用快照時的考量事項

- 您只能使用與目前根磁碟區屬於相同歷程的快照。
- 您無法使用從根磁碟區擷取的快照中建立的快照複本。
- 成功更換根磁碟區之後，您仍然可以使用從原始根磁碟區擷取的快照來取代新的 (取代) 根磁碟區。

使用取代根磁碟區 AMI

此選項會使用您指定的來取代目前的根磁碟區。AMI如果您需要執行作業系統和應用程式的修補或升級，這會很有用。AMI必須具有與執行個體相同的產品代碼、帳單資訊、架構類型和虛擬化類型。

如果執行個體已啟用ENA或 sriov-net，則您必須使用支援這些功能的執行AMI個體。如果執行個體未針對ENA或 sriov-net 啟用，則您可以選取不支援這些功能的執行個AMI體，也可以在選取支援或 sriov-net 時自動新增支援。AMI ENA

如果針對 Nitro 啟用執行個體TPM，則您必須使用已啟用 Nitro AMI TPM 的執行個體。如果執行個體未設定 Nitro TPM 支援，則不會啟用 Nitro 支援，無論您選取哪一AMI個執行個體。

您可以選取AMI與執行個體不同的開機模式，只要執行個體支援的啟動模式AMI。如果執行個體不支援該開機模式，則請求會失敗。如果執行個體支援開機模式，則新的開機模式會傳播至執行個體，並相應地更新其UEFI資料。如果您手動修改了開機順序，或新增私密 UEFI Secure Boot 金鑰來載入私有核心模組，則在取代根磁碟區期間變更將會遺失。

取代根磁碟區會取得與原始根磁碟區相同的磁碟區類型和終止時刪除屬性，並取得AMI根磁碟區區塊裝置對應的大小。

> **Note**
>
> AMI根磁碟區塊裝置對映的大小必須等於或大於原始根磁碟區的大小。如果AMI根磁碟區塊裝置對映的大小小於原始根磁碟區的大小，則要求會失敗。

在根磁碟區取代工作完成之後，當您使用主控台描述執行個體時，會反映下列新增和更新的資訊，AWS CLI 或 AWS SDKs：

- 新AMI識別碼
- 根磁碟區的新磁碟區 ID
- 更新的啟動模式配置（如果由更改AMI）

- 更新硝基TPM配置（如果啟用AMI）
- 已更新的ENA組態 (如果已啟用AMI)
- 已更新 sriov-net 組態 (如果已啟用) AMI

新 AMI ID 也會反映在執行個體中繼資料中。

使用以下注意事項AMI：

- 如果您使用具AMI有多個區塊裝置對映的，則只會使用的AMI根磁碟區。其他 (非根) 磁碟區會被忽略。
- 只有在您擁有AMI及其關聯根磁碟區快照的權限時，才能使用此功能。您無法搭配使用此功能 AWS Marketplace AMIs。
- 只有在執行個體AMI沒有產品代碼的情況下，才能使用沒有產品代碼的。
- AMI根磁碟區塊裝置對映的大小必須等於或大於原始根磁碟區的大小。如果AMI根磁碟區塊裝置對映的大小小於原始根磁碟區的大小，則要求會失敗。
- 執行個體的執行個體身分文件會自動更新。
- 如果執行個體支援 NitroTPM，則會重設執行個體的 Nitro TPM 資料，並產生新的金鑰。

您可以選擇在根磁碟區取代程序完成後是否要保留原始根磁碟區。如果您選擇在取代程序完成之後刪除原始根磁碟區，則原始根磁碟區會自動刪除並變得無法復原。如果您選擇在程序完成後保留原始根磁碟區，該磁碟區仍會在您的帳戶中佈建；您必須在不再需要磁碟區時手動刪除磁碟區。

根磁碟區取代工作會在下列狀態中轉換：

- `pending`— 正在建立取代體積塊。
- `in-progress`— 正在分離原始磁碟區，且正在連接取代磁碟區。
- `succeeded`— 取代磁碟區已成功附加至執行個體，且可使用執行個體。
- `failing`— 取代任務正在失敗的過程中。
- `failed`— 取代任務失敗，但根磁碟區仍然連接。
- `failing-detached`— 取代任務正在失敗的過程中，執行個體可能沒有附加根磁碟區。
- `failed-detached`— 取代工作失敗且執行個體沒有連接根磁碟區。

如果根磁碟區取代任務失敗，則執行個體會重新啟動，而原始根磁碟區會保持連接到執行個體。

考量事項

在開始之前，請考慮下列事項。

要求

- 執行個體必須處於 `running` 狀態。
- 執行個體會在過程中自動重新啟動。重新開機期間會清除記憶體 (RAM) 的內容。不需要手動重新啟動。
- 如果根磁碟區是執行個體存放區磁碟區，則無法取代根磁碟區。僅支援具有 Amazon EBS 根磁碟區的執行個體。
- 您可以替換所有虛擬化執行個體類型和 EC2 Mac 裸機執行個體的根磁碟區。不支援其他裸機執行個體類型。
- 您只能使用屬於相同關係的快照作為執行個體以前的根磁碟區。
- 如果您的帳戶在目前區域中預設啟用 Amazon EBS 加密，則無論指定快照的加密狀態或指定的根磁碟區的加密狀態為何，根磁碟區取代任務建立的替換根磁碟區都會一律加密AMI。

加密結果

下表摘要提供了可能的加密結果的摘要。

	原始根磁碟區	指定的快照或 AMI	預設加密	取代根磁碟區	用於取代根磁碟區的加密金鑰
將取代根磁碟區還原至初始啟動狀態	Encrypted	不適用	不考慮	Encrypted	與原始根磁碟區相同的KMS金鑰
	未加密	不適用	已停用	未加密	不適用
	未加密	不適用	已啟用	Encrypted	Amazon EBS 加密的帳戶默認密KMS鑰

	原始根磁碟區	指定的快照或AMI	預設加密	取代根磁碟區	用於取代根磁碟區的加密金鑰
從快照或還原取代根磁碟區AMI	Encrypted	未加密	不考慮	Encrypted	與原始根磁碟區相同的KMS金鑰
	Encrypted	Encrypted	不考慮	Encrypted	與原始根磁碟區相同的KMS金鑰
	未加密	未加密	已停用	未加密	不適用
	未加密	未加密	已啟用	Encrypted	Amazon EBS加密的帳戶默認密KMS鑰
	未加密	Encrypted	不考慮	Encrypted	如果AMI或快照屬於帳戶，則取代磁碟區會使用AMI或快照的KMS金鑰加密。如果與帳戶共用AMI或快照，則會使用帳戶的預設KMS金鑰加密替換磁碟區，以進行Amazon EBS加密。

取代根磁碟區

取代執行個體的根磁碟區時，會建立根磁碟區取代任務。您可以使用根磁碟區取代任務來監視取代過程的進度和結果。

Console

取代根磁碟區

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取要取代根磁碟區的執行個體，然後依序選取 Actions (動作)、Monitor and troubleshoot (監控與疑難排解)、Replace root volume (取代根磁碟區)。

> **Note**
> 如果選定的執行個體不在 `running` 狀態，則會停用取代根磁碟區操作。

4. 在「取代根磁碟區」畫面中，對於「還原」，選擇下列其中一個選項：
   - 啟動狀態 — 從用來建立目前根磁碟區的快照還原取代根磁碟區。
   - 快照 — 將取代根磁碟區還原至您指定的快照。針對快照，選取要使用的快照。
   - 映像 — 使用您指定的還原取代根磁碟區。AMI對於「影像」，選取AMI要使用的。
5. (選擇性) 若要刪除要取代的根磁碟區，請選取刪除已取代的根磁碟區。
6. 選擇 [建立取代工作]。
7. 若要監視取代工作，請選擇執行個體的 [儲存] 索引標籤，然後展開 [最近的根磁碟區取代工作]

AWS CLI

若要將取代根磁碟區還原至啟動狀態

使用 create-replace-root-volume-任務命令。若為 `--instance-id`，請指定要對其取代根磁碟區的執行個體 ID。省略 `--snapshot-id` 和 `--image-id` 參數。若要在取代原始根磁碟區之後將其刪除，請包含 `--delete-replaced-root-volume` 並指定 `true`。

```
$ aws ec2 create-replace-root-volume-task \
--instance-id i-1234567890abcdef0 \
--delete-replaced-root-volume
```

若要將取代根磁碟區還原至特定快照

使用 create-replace-root-volume-任務命令。若為 `--instance-id`，請指定要對其取代根磁碟區的執行個體 ID。若為 `--snapshot-id`，請指定要使用的快照 ID。若要在取代原始根磁碟區之後將其刪除，請包含 `--delete-replaced-root-volume` 並指定 `true`。

```
$ aws ec2 create-replace-root-volume-task \
--instance-id i-1234567890abcdef0 \
--snapshot-id snap-9876543210abcdef0 \
--delete-replaced-root-volume
```

使用還原取代根磁碟區 AMI

使用 create-replace-root-volume-任務命令。若為 `--instance-id`，請指定要對其取代根磁碟區的執行個體 ID。對於`--image-id`，指定AMI要使用的 ID。若要在取代原始根磁碟區之後將其刪除，請包含 `--delete-replaced-root-volume` 並指定 `true`。

```
$ aws ec2 create-replace-root-volume-task \
--instance-id i-01234567890abcdef \
--image-id ami-09876543210abcdef \
--delete-replaced-root-volume
```

檢視根磁碟區取代任務的狀態

使用 describe-replace-root-volume-tasks 指令並指定要檢視IDs的根磁碟區取代工作。

```
$ aws ec2 describe-replace-root-volume-tasks \
--replace-root-volume-task-ids replacevol-1234567890abcdef0
```

```
{
  "ReplaceRootVolumeTasks": [
  {
    "ReplaceRootVolumeTaskId": "replacevol-1234567890abcdef0",
    "InstanceId": "i-1234567890abcdef0",
    "TaskState": "succeeded",
    "StartTime": "2020-11-06 13:09:54.0",
    "CompleteTime": "2020-11-06 13:10:14.0",
    "SnapshotId": "snap-01234567890abcdef",
    "DeleteReplacedRootVolume": "True"
  }]
}
```

或者，指定要依執行個體篩選結果的 `instance-id` 篩選條件。

```
$ aws ec2 describe-replace-root-volume-tasks \
--filters Name=instance-id,Values=i-1234567890abcdef0
```

Tools for Windows PowerShell

若要將取代根磁碟區還原至啟動狀態

使用指New-EC2ReplaceRootVolumeTask令。若為 `-InstanceId`，請指定要對其取代根磁碟區的執行個體 ID。省略 `-SnapshotId` 和 `-ImageId` 參數。若要在取代原始根磁碟區之後將其刪除，請包含 `-DeleteReplacedRootVolume` 並指定 `$true`。

```
PS C:\> New-EC2ReplaceRootVolumeTask -InstanceId i-1234567890abcdef0 -
DeleteReplacedRootVolume $true
```

若要將取代根磁碟區還原至特定快照

使用指New-EC2ReplaceRootVolumeTask令。若為 `--InstanceId`，請指定要對其取代根磁碟區的執行個體 ID。若為 `-SnapshotId`，請指定要使用的快照 ID。若要在取代原始根磁碟區之後將其刪除，請包含 `-DeleteReplacedRootVolume` 並指定 `$true`。

```
PS C:\> New-EC2ReplaceRootVolumeTask -InstanceId i-1234567890abcdef0 -
SnapshotId snap-9876543210abcdef0 -DeleteReplacedRootVolume $true
```

使用還原取代根磁碟區 AMI

使用指New-EC2ReplaceRootVolumeTask令。若為 `-InstanceId`，請指定要對其取代根磁碟區的執行個體 ID。對於`-ImageId`，指定AMI要使用的 ID。若要在取代原始根磁碟區之後將其刪除，請包含 `-DeleteReplacedRootVolume` 並指定 `$true`。

```
PS C:\> New-EC2ReplaceRootVolumeTask -InstanceId i-1234567890abcdef0 -
ImageId ami-09876543210abcdef -DeleteReplacedRootVolume $true
```

檢視根磁碟區取代任務的狀態

使用指Get-EC2ReplaceRootVolumeTask令並指定要檢視IDs的根磁碟區取代工作。

```
PS C:\> Get-EC2ReplaceRootVolumeTask -
ReplaceRootVolumeTaskIds replacevol-1234567890abcdef0
```

或者，指定要依執行個體篩選結果的 `instance-id` 篩選條件。

```
PS C:\> Get-EC2ReplaceRootVolumeTask -Filters @{Name = 'instance-id'; Values =
 'i-1234567890abcdef0'} | Format-Table
```

Amazon EC2 執行個體上磁碟區的裝置名稱

當您將磁碟區連接到您的執行個體時，您會在其中包含磁碟區的裝置名稱。此設備名稱由 Amazon 使用EC2。執行個體的區塊裝置驅動程式會在掛接磁碟區時指派實際的磁碟區名稱，而指派的名稱可能與 Amazon EC2 使用的名稱不同。

您執行個體可以支援的磁碟區數目取決於作業系統。如需詳細資訊，請參閱 Amazon EC2 實例的EBS數量限制。

目錄

- 可用裝置名稱
- 裝置名稱考量

可用裝置名稱

Linux 執行個體

Linux 執行個體有兩種虛擬化類型：半虛擬化 (PV) 和硬體虛擬機器 (HVM)。執行個體的虛擬化類型是由AMI用來啟動執行個體的決定。支援所有執行個體類型HVMAMIs。上一代的某些執行個體類型支援 PV AMIs。請務必記下您的虛擬化類型，AMI因為您可以使用的建議和可用裝置名稱取決於執行個體的虛擬化類型。如需詳細資訊，請參閱 虛擬化類型。

下表列出可在區塊裝置對映中或貼附EBS磁碟區時指定的可用裝置名稱。

虛擬化類型	可用性	預留供根磁碟區使用	建議用於EBS磁碟區	執行個體存放區磁碟區
全虛擬化	/dev/sd[a-z] /dev/sd[a-z][1-15]	/dev/sda1	/dev/sd[f-p] /dev/sd[f-p][1-6]	/dev/sd[b-e]

虛擬化類型	可用性	預留供根磁碟區使用	建議用於EBS磁碟區	執行個體存放區磁碟區
	/dev/hd[a-z] /dev/hd[a-z][1-15]			
HVM	/dev/sd[a-z] /開發/xvd [a-d][a-x] /dev/xvd[e-z]	不同 AMI /dev/sda1 或 /dev/xvda	/dev/sd[f-p] *	/dev/sd[b-e] /dev/sd[b-h] (h1.16xlarge) /dev/sd[b-y] (d2.8xlarge) /dev/sd[b-i] (i2.8xlarge) **

* 您為區塊裝置對應中的NVMeEBS磁碟區指定的裝置名稱會使用NVMe裝置名稱 (`/dev/nvme[0-26]n1`) 重新命名。區塊裝置驅動程式可以使用與您在區塊NVMe裝置對應中為磁碟區指定的不同順序指派裝置名稱。

** NVMe 實例存儲卷會自動枚舉並分配NVMe設備名稱。

Windows 執行個體

Windows AMIs 使用下列其中一組驅動程式來允許存取虛擬化硬體： AWS PV、Citrix PV 和 RedHat PV。如需詳細資訊，請參閱 the section called “窗戶光伏驅動器”。

下表列出可在區塊裝置對映中或貼附EBS磁碟區時指定的可用裝置名稱。

驅動器類型	可用性	預留供根磁碟區使用	建議用於EBS磁碟區	執行個體儲存體磁碟區
AWS 光伏、思杰光伏	xvd[b-z] xvd[b-c][a-z]	/dev/sda1	xvd[f-z] *	xvdc[a-x] xvd[a-e]

驅動器類型	可用性	預留供根磁碟區使用	建議用於EBS磁碟區	執行個體儲存體磁碟區
	/dev/sda1 /dev/sd[b-e]			**
Red Hat PV	xvd[a-z] xvd[b-c][a-z] /dev/sda1 /dev/sd[b-e]	/dev/sda1	xvd[f-p]	xvdc[a-x] xvd[a-e]

* 對於 Citrix PV 和紅帽光伏，如果您以名稱對映EBS磁碟區`xvda`，Windows 將無法辨識該磁碟區 (AWS PV 或磁碟區可見 AWS NVMe)。

** NVMe 執行個體儲存磁碟區會自動列舉並指派 Windows 磁碟機代號。

如需執行個體存放區磁碟區的詳細資訊，請參閱Amazon EC2 執行個體的執行個體存放臨時區塊儲。如需有關NVMeEBS磁碟區 (Nitro-based 執行個體) 的詳細資訊，包括如何識別EBS裝置，請參閱 Amazon EBS 和 Amazon EBS 使用者指南NVMe中的。

裝置名稱考量

當您選取裝置名稱時，請注意下列項目：

- 您使用的裝置名稱結尾部分不應重疊，因為啟動執行個體時可能會造成問題。例如，避免使用附加至相同`xvdf`例證的體積塊等組合。`/dev/xvdf`
- 雖然您可以使用用來連接執行個體儲存EBS磁碟區的裝置名稱來連接磁碟區，但我們強烈建議您不要這樣做，因為行為可能無法預測。
- 執行個NVMe體的執行個體儲存磁碟區數量取決於執行個體的大小。NVMe執行個體存放區磁碟區會自動列舉並指派NVMe裝置名稱 (Linux 執行個體) 或 Windows 磁碟機代號 (Windows 執行個體)。
- (Windows 執行個體) AWS Windows AMIs 隨附額外的軟體，可在首次啟動時準備執行個體。這是無論是EC2Config服務（視窗服務器 2016 年AMIs之前）或EC2Launch（視窗服務器 2016 及更高版本）。在裝置映射到磁碟機後，便會初始化並掛載。根磁碟機會初始化並掛載為 `C:\`。根據預設，當EBS磁碟區連接至 Windows 執行個體時，它可以顯示為執行個體上的任何磁碟機代號。您可以根據您的規格，變更設定以設定磁碟區的磁碟機代號。對於實例存儲卷，默認值取決於驅動程序。

AWS PV 驅動程式和 Citrix PV 驅動程式會指派執行個體儲存磁碟區磁碟區代號，從 Z: 到 A:。Red Hat 驅動程式會為執行個體存放區磁碟區指派 D: 到 Z: 的磁碟機代號。如需詳細資訊，請參閱 視窗啟動代理程式 EC2 和 將磁碟映射到連接到 Amazon EC2 Windows 執行個體的磁碟區。

- (Linux 執行個體) 視核心的區塊裝置驅動程式而定，裝置可能會使用與您指定的名稱不同。例如，假設您指定了裝置名稱 /dev/sdh，您的裝置可以重新命名為 /dev/xvdh 或 /dev/hdh。在大部分情況下，結尾的字母會維持相同。在 Red Hat Enterprise Linux (及其變體，例如 CentOS) 的某些版本中，結尾的字母可能會變更 (/dev/sda 可能會變成 /dev/xvde)。在這些情況下，各裝置名稱的結尾字母會以相同的次數遞增。例如，如果 /dev/sdb 重新命名為 /dev/xvdf，則 /dev/sdc 重新命名為 /dev/xvdg。Amazon Linux 會為您指定給已重新命名裝置的名稱建立符號連結。其他作業系統可能會有不同的行為。
- （Linux 實例）HVMAMIs不支持在設備名稱上使用尾隨數字/dev/sda1，除了為根設備保留的，和/dev/sda2。雖然可以使/dev/sda2用，但我們不建議將此裝置對應與HVM執行個體搭配使用。
- (Linux 執行個體) 使用 PV 時AMIs，您無法附加共用相同裝置字母的磁碟區，且不含尾隨位數字。例如，若您將磁碟區連接為 /dev/sdc，並將另一個磁碟區連接為 /dev/sdc1，執行個體只會看見 /dev/sdc。若要在裝置名稱中使用尾碼，您必須在所有共享相同基礎代號的裝置名稱上使用尾碼 (例如 /dev/sdc1、/dev/sdc2、/dev/sdc3)。
- (Linux 執行個體) 某些自訂核心可能有限制使用/dev/sd[f-p]或/dev/sd[f-p][1-6]。若您在使用 /dev/sd[q-z] 或 /dev/sd[q-z][1-6] 上發生問題，請嘗試切換為 /dev/sd[f-p] 或 /dev/sd[f-p][1-6]。

在您指定已選取的裝置名稱之前，請確認該名稱是否可用。否則，您將收到設備名稱已在使用中的錯誤。若要檢視磁碟裝置及其掛接點，請使用lsblk指令 (Linux 執行個體) 或「磁碟管理」公用程式或diskpart指令 (Windows 執行個體)。

Amazon EC2 執行個體上磁碟區的區塊裝置對映

您啟動的每個執行個體都有一個關聯的根裝置磁碟區，也就是 Amazon EBS 磁碟區或執行個體存放區磁碟區。您可以使用區塊裝置對應來指定其他EBS磁碟區或執行個體儲存磁碟區，以便在執行個體啟動時連接至執行個體。您也可以將其他EBS磁碟區附加至執行中的執行個體。但是，唯一能將執行個體存放磁碟區連接至執行個體的方式，就是使用區塊型儲存設備映射來連接磁碟區，如同執行個體啟動時一樣。

目錄

- 區塊型儲存設備映射概念

- 將區塊裝置對應新增至 AMI
- 將區塊裝置對應新增至 Amazon EC2 執行個體

區塊型儲存設備映射概念

區塊型儲存設備是一種儲存設備，它可依位元組或位元 (區塊) 的順序移動資料。這些裝置支援隨機存取，一般使用緩衝 I/O。例如硬碟、CD-磁ROM碟機和快閃磁碟機。區塊型設備可實際連接至電腦，或如同實際連接至電腦一般進行遠端存取。

Amazon EC2 支持兩種類型的塊設備：

- 執行個體存放磁碟區 (基礎硬體實際連接到執行個體主機電腦的虛擬設備)
- EBS磁碟區 (遠端儲存裝置)

區塊裝置對應會定義要連接至執行個體的區塊裝置 (執行個體儲存EBS磁碟區和磁碟區)。您可以指定區塊裝置對映作為建立的一部分，以AMI便從中啟動的所有執行個體都使用該對應AMI。或者，您也可以在啟動執行個體時指定區塊裝置對應，因此此對應會覆寫您AMI從中啟動執行個體時所指定的對應。請注意，NVMe執行個體類型支援的所有執行個體存放區磁碟區都會在執行個體啟動時自動列舉並指派裝置名稱；將它們包含在區塊裝置對應中並無作用。

目錄

- 區塊型儲存設備映射項目
- 區塊型儲存設備映射執行個體存放區 Caveat
- 區塊型儲存設備映射範例
- 如何在作業系統中將設備變為可用

區塊型儲存設備映射項目

建立區塊型設備映射時，您需為要連接至執行個體的每個區塊型儲存設備指定下列資訊：

- Amazon 中使用的設備名稱EC2。執行個體的區塊型儲存設備驅動程式，會在掛載磁碟區時指派實際的磁碟區名稱。分配的名稱可以與 Amazon EC2 推薦的名稱不同。如需詳細資訊，請參閱Amazon EC2 執行個體上磁碟區的裝置名稱。

對於執行個體存放區磁碟區，您也可以指定下列資訊：

- 虛擬裝置：`ephemeral[0-23]`。請注意，可供執行個體使用的執行個體存放磁碟區數量和大小，根據執行個體類型而異。

對於NVMe執行個體儲存磁碟區，下列資訊也適用：

- 這些磁碟區會自動列舉，並獲指派設備名稱；將這些磁碟區加入區塊型設備映射中並無作用。

對於EBS磁碟區，您也可以指定下列資訊：

- 用於建立區塊型儲存設備之快照的 ID (snap-xxxxxxxx)。若已指定磁碟區大小，則此值是選用的。無法指定已封存快照的 ID。
- 磁碟區的大小 (GiB)。指定的大小必須大於或等於指定快照的大小。
- 是否在執行個體終止時刪除磁碟區 (`true` 或 `false`)。根設備磁碟區的預設值為 `true`，連接磁碟區的預設值為 `false`。建立時AMI，其區塊裝置對應會從執行個體繼承此設定。當您啟動執行個體時，執行個體會從繼承此設定。AMI
- 磁碟區類型，可以是gp2和gp3適用於一般用途SSD、io1已佈建IOPSSSD、st1輸送量最佳化HDDHDD、sc1冷或磁帶standard的磁碟區類型。io2
- 磁碟區支援的每秒輸入/輸出作業數 (IOPS)。(僅與 io1 和io2 磁碟區搭配使用。)

區塊型儲存設備映射執行個體存放區 Caveat

啟動在區塊裝置對映中具有執行個體儲存磁碟區AMIs的執行個體時，需要考慮幾個警告。

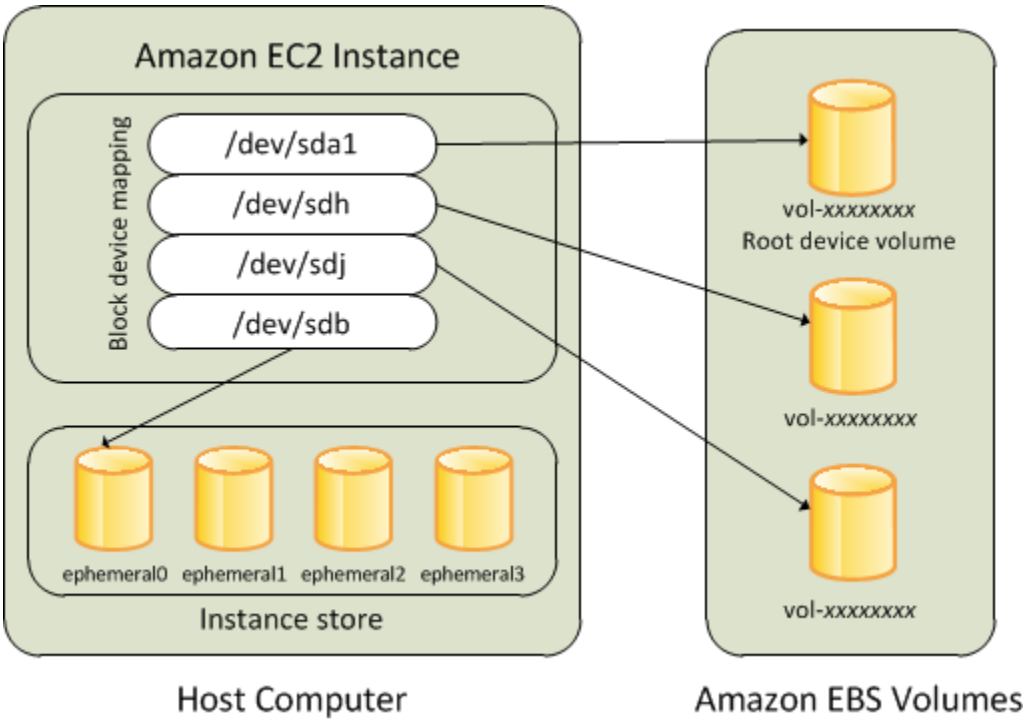
- 部分執行個體類型包含的執行個體存放磁碟區較多，有些執行個體類型則完全沒有執行個體存放磁碟區。如果您的執行個體類型支援一個執行個體存放區磁碟區，且您AMI有兩個執行個體存放磁碟區的對應，則執行個體會以一個執行個體存放區磁碟
- 執行個體存放磁碟區只能在啟動時映射。您無法停止沒有執行個體存放磁碟區的執行個體 (例如 `t2.micro`)，請將執行個體變更為支援執行個體存放磁碟區的類型，然後重新啟動含執行個體存放磁碟區的執行個體。不過，您可以AMI從執行個體建立並在支援執行個體儲存磁碟區的執行個體類型上啟動該執行個體，然後將這些執行個體儲存磁碟區對應至執行個體。
- 如果啟動有執行個體存放磁碟區映射的執行個體，接著停止執行個體，將其變更為執行個體存放磁碟區較少的執行個體類型，並重新啟動執行個體，則一開始啟動的執行個體存放磁碟區映射仍會出現在執行個體中繼資料內。但是，只有該執行個體類型支援的執行個體存放磁碟區數量上限可供執行個體使用。

> **Note**
>
> 停止執行個體時，執行個體存放磁碟區上的所有資料都會遺失。

- 根據啟動時的執行個體儲存容量，M3 執行個體可能會在啟動時忽略AMI執行個體儲存區塊裝置對應，除非在啟動時指定它們。您應該在啟動時指定執行個體儲存區塊裝置對映，即使AMI您要啟動的執行個體儲存區磁碟區已對映於中AMI，以確保執行個體啟動時可使用執行個體存放區磁碟區。

區塊型儲存設備映射範例

此圖顯示了EBS支援的執行個體的區塊裝置對應範例。它會`/dev/sdb`對映至`ephemeral0`兩個EBS磁碟區，一個磁碟區，另一個磁碟區對應至`/dev/sdj`。`/dev/sdh`它也會顯示根裝置EBS磁碟區的磁碟區`/dev/sda1`。



請注意，此範例區塊裝置對應用於範例指令和本主題APIs中。您可以在指定區塊裝置對應 AMI和中找到範例指令APIs，並建立區塊裝置對應啟動執行個體時更新區塊型儲存裝置對應。

如何在作業系統中將設備變為可用

設備名稱喜歡`/dev/sdh`和`xvdh`被 Amazon 用EC2來描述塊設備。Amazon 使用區塊裝置對應EC2來指定要連接到EC2執行個體的區塊裝置。區塊型儲存設備連接至執行個體之後，作業系統必須掛載儲存

設備，接著方可供存取。區塊型儲存設備與執行個體分離之後，作業系統會卸載儲存設備，於是便無法再存取。

Linux 執行個體 — 區塊裝置對應中指定的裝置名稱會在執行個體第一次開機時對應至其對應的區塊裝置。執行個體類型將決定預設要格式化及掛載的執行個體存放磁碟區。您可在啟動時掛載額外的執行個體存放磁碟區，只要不超過執行個體類型可用的執行個體存放磁碟區數量即可。如需詳細資訊，請參閱Amazon EC2 執行個體的執行個體存放臨時區塊儲。執行個體的區塊型儲存設備驅動程式決定格式化及掛載磁碟區時要使用的設備。

Windows 執行個體 — 區塊裝置對應中指定的裝置名稱會在執行個體第一次開機時對應至其對應的區塊裝置，然後 EC2Config 服務會初始化並掛接磁碟機。根設備磁碟區將掛載為 `C:\`。執行個體存放磁碟區將掛載為 `Z:\`、`Y:\`，並以此類推。當加EBS密卷被掛接時，它可以使用任何可用的驅動器號進行安裝。不過，您可以設定磁碟機代號指定給EBS磁碟區的方式；如需詳細資訊，請參閱the section called “啟動代理程式”。

將區塊裝置對應新增至 AMI

每個區塊裝置對應都AMI有一個區塊裝置對應，指定從啟動執行個體時要連接至執行個體的區塊裝置AMI。要將更多塊設備添加到AMI，您必須創建自己的塊設備AMI。

目錄

- 指定區塊裝置對應 AMI
- 檢視EBS區AMI塊裝置對應中的磁碟區

指定區塊裝置對應 AMI

建立根磁碟區時，除了指定根磁碟區之外，還有兩種方法可以指定磁碟區AMI。如果在從執行個體建立執行個體之前，已將磁碟區連結至執行AMI中的執行個體，則該磁碟區的區塊裝置對應會AMI包含這些相同的磁碟區。對於EBS磁碟區，現有資料會儲存到新的快照，而這是區塊裝置對應中指定的新快照。如為執行個體存放磁碟區，資料將不會保留。

對於EBS支援的EBS磁碟區AMI，您可以使用區塊裝置對應來新增磁碟區和執行個體儲存磁碟區。對於執行個體存放區支援AMI，您只能透過在註冊映像時修改映像資訊清單檔案中的區塊裝置對應項目來新增執行個體存放區磁碟區。

> **Note**
>
> 如為 M3 執行個體，您必須在啟動時在執行個體的區塊型設備映射中指定執行個體存放磁碟區。啟動 M3 執行個體時，如果未指定為執行個體區塊裝置對映的一部分，則在區塊裝置對映中指定的執行個體儲存磁碟區AMI可能會被忽略。

Console

AMI使用控制台將磁碟區新增至

1. 打開 Amazon EC2 控制台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體並選取 Actions (動作)、Image and templates (影像和範本)、Create image (建立影像)。
4. 輸入影像的名稱和說明。
5. 執行個體磁碟區會出現在 Instance volumes (執行個體磁碟區) 下。若要新增其他磁碟區，請選擇 Add volume (新增磁碟區) 。
6. 針對 Volume Type (磁碟區類型)，選擇一種磁碟區類型。對於Device (裝置)，請選擇裝置名稱。對於EBS磁碟區，您可以指定其他詳細資料，例如快照、磁碟區大小、磁碟區類型和加密狀態。IOPS
7. 選擇 Create image (建立映像)。

Command line

AMI使用指令行將磁碟區加入至

使用「建立映像」指 AWS CLI 令指定- backed 的區塊裝置對應。EBS AMI使用寄存器映像檔指 AWS CLI 令，為執行個體存放區支援指定區塊裝置對應。AMI

使用 `--block-device-mappings` 參數指定區塊型儲存設備映射。編碼中的引數JSON可以直接在命令行上提供，也可以通過對文件的引用提供：

```
--block-device-mappings [mapping, ...]
--block-device-mappings [file://mapping.json]
```

若要新增執行個體存放磁碟區，請使用下列映射。

```
{
    "DeviceName": "device_name",
    "VirtualName": "ephemeral0"
}
```

若要新增空的 100 GiB gp2 磁碟區，請使用下列映射。

```
{
    "DeviceName": "device_name",
    "Ebs": {
      "VolumeSize": 100
    }
}
```

若要根據快照新增EBS磁碟區，請使用下列對映。

```
{
    "DeviceName": "device_name",
    "Ebs": {
      "SnapshotId": "snap-xxxxxxxx"
    }
}
```

若要省略設備的映射，請使用下列映射。

```
{
    "DeviceName": "device_name",
    "NoDevice": ""
}
```

或者，您可用 `-BlockDeviceMapping` 參數結合下列命令 (AWS Tools for Windows PowerShell)：

- New-EC2Image
- Register-EC2Image

檢視EBS區AMI塊裝置對應中的磁碟區

您可以輕鬆地列舉區塊裝置對應中的EBS磁碟區。AMI

Console

若要AMI使用控制台檢視EBS磁碟區

1. 打開 Amazon EC2 控制台。
2. 在導覽窗格中，選擇AMIs。
3. 從過濾器列表中選擇EBS圖像以獲取 EBS-backed AMIs 的列表。
4. 選擇所需的AMI，然後查看「詳細信息」選項卡。將至少顯示根儲存設備的以下資訊：
   - Root Device Type (根設備類型)`ebs` ()
   - Root Device Name (根設備名稱) (例如，`/dev/sda1`)
   - Block Devices (區塊型儲存設備) (例如，`/dev/sda1=snap-1234567890abcdef0:8:true`)

   如果AMI是使用區塊裝置對映使用其他EBS磁碟區建立的，則「區塊裝置」欄位也會顯示這些其他磁碟區的對映。(此畫面不會顯示執行個體存放磁碟區。)

Command line

AMI使用指令行檢視EBS磁碟區的步驟

使用 describe-image (AWS CLI) 指令或 Get-EC2Image(AWS Tools for Windows PowerShell) 指令列舉區塊裝置對映中的EBS磁碟區。AMI

將區塊裝置對應新增至 Amazon EC2 執行個體

根據預設，您啟動的執行個體會包含您AMI從中啟動執行個體的區塊裝置對應中指定的任何儲存裝置。您可以在啟動執行個體時指定對其區塊裝置對應的變更，而這些更新會覆寫或與的區塊裝置對應合併AMI。

限制

- 如為根磁碟區，只能修改：磁碟區大小、磁碟區類型和 Delete on Termination (在終止時刪除) 旗標。
- 修改EBS卷宗時，無法減小其大小。因此，您必須指定大小等於或大於在的區塊裝置對映中指定的快照大小的快照AMI。

目錄

- 啟動執行個體時更新區塊型儲存裝置對應
- 更新執行中執行個體的區塊型儲存裝置對應
- 檢視執行個體EBS區塊裝置對應中的磁碟區
- 檢視執行個體區塊型儲存裝置對應的執行個體存放區的磁碟區

啟動執行個體時更新區塊型儲存裝置對應

啟動執行個體時，您可以將EBS磁碟區和執行個體存放磁碟區新增至執行個體。請注意，更新執行個體的區塊裝置對應並不會對啟動執行個體的AMI來源區塊裝置對應進行永久變更。

Console

使用主控台新增磁碟區至執行個體

1. 打開 Amazon EC2 控制台。
2. 在儀表板中，選擇 Launch Instance (啟動執行個體)。
3. 在選擇 Amazon 機器映像 (AMI) 頁面上，選取AMI要使用的，然後選擇選取。
4. 遵循精靈的指示，來完成 Choose an Instance Type (選擇執行個體類型) 和 Configure Instance Details (設定執行個體的詳細資訊) 頁面上的操作。
5. 您可以在 [新增儲存體] 頁面上修改根EBS磁碟區、磁碟區和執行個體儲存磁碟區，如下所示：
   - 若要變更根磁碟區的大小，請在 Type (類型) 資料行中尋找 Root (根) 磁碟區，並變更其 Size (大小) 欄位的值。
   - 若要隱藏由AMI用於啟動執行個體的EBS區塊裝置對映所指定的磁碟區，請找出該磁碟區並按一下其刪除圖示。
   - 若要新增EBS磁碟區，請選擇 [新增磁碟區]，EBS從 [類型] 清單中選擇，然後填入欄位 ([裝置]、[快照] 等)。
   - 若要隱藏由AMI用於啟動執行個體的區塊裝置對應所指定的執行個體儲存磁碟區，請找出該磁碟區，然後選擇其刪除圖示。
   - 若要新增執行個體存放磁碟區，請選取 Add New Volume (新增新磁碟區)，選取 Type (類型) 清單中的 Instance Store (執行個體存放區)，然後從 Device (設備) 中選取設備名稱。
6. 完成剩餘的精靈頁面，然後選擇 Launch (啟動)。

Command line

若要使用新增磁碟區至執行個體 AWS CLI

使用 run-instance 指 AWS CLI 令搭配`--block-device-mappings`選項來指定啟動時執行個體的區塊裝置對映。

例如，假設 EBS-backed 為 Linux 實例AMI指定了以下塊設備映射：

- `/dev/sdb` = `ephemeral0`
- `/dev/sdh` = `snap-1234567890abcdef0`
- `/dev/sdj` = `100`

若要防止附加至`/dev/sdj`從此啟動的執行個體AMI，請使用下列對應。

```
{
    "DeviceName": "/dev/sdj",
    "NoDevice": ""
}
```

若要增加`/dev/sdh`到的大小`300 GiB`，請指定下列對映。請注意，您不需要指定 `/dev/sdh` 的快照 ID，因為指定設備名稱足以找出磁碟區。

```
{
    "DeviceName": "/dev/sdh",
    "Ebs": {
      "VolumeSize": 300
    }
}
```

若要在執行個體啟動時增加根磁碟區的大小，請先呼叫 ID 為的描述映像檔，AMI以驗證根磁碟區的裝置名稱。例如：`"RootDeviceName": "/dev/xvda"`。若要覆寫根磁碟區的大小，請指定根裝置的裝置名稱，以AMI及新磁碟區大小。

```
{
    "DeviceName": "/dev/xvda",
    "Ebs": {
      "VolumeSize": 100
    }
}
```

若要連接額外的執行個體存放磁碟區 /dev/sdc，請指定下列映射。如果執行個體類型不支援多重執行個體存放磁碟區，此映射將沒有效果。如果執行個體支援NVMe執行個體儲存磁碟區，系統會自動列舉這些磁碟區並指派NVMe裝置名稱。

```
{
    "DeviceName": "/dev/sdc",
    "VirtualName": "ephemeral1"
}
```

若要使用新增磁碟區至執行個體 AWS Tools for Windows PowerShell

透過New-EC2Instance指令使用-BlockDeviceMapping參數 (AWS Tools for Windows PowerShell)。

更新執行中執行個體的區塊型儲存裝置對應

您可以使用指modify-instance-attribute AWS CLI 令來更新執行中執行個體的區塊裝置對應。您不需要為了變更屬性而停止執行個體。

```
aws ec2 modify-instance-attribute --instance-id i-1a2b3c4d --block-device-mappings
 file://mapping.json
```

例如，若要在執行個體終止時保留根磁碟區，請在 mapping.json 中指定下列項目。

```
[
  {
    "DeviceName": "/dev/sda1",
    "Ebs": {
      "DeleteOnTermination": false
    }
  }
]
```

或者，您可以將-BlockDeviceMapping參數與Edit-EC2InstanceAttribute指令 (AWS Tools for Windows PowerShell) 搭配使用。

檢視執行個體EBS區塊裝置對應中的磁碟區

您可以輕鬆列舉對應至執行個體的EBS磁碟區。

> **Note**
>
> 對於 2009-10-31 發行之前啟動的執行個體API， AWS 無法顯示區塊裝置對應。您必須分離並重新連接磁碟區，才 AWS 能顯示區塊裝置對映。

Console

使用主控台檢視執行個體的EBS磁碟區

1. 打開 Amazon EC2 控制台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 在搜尋方塊中，輸入根裝置類型，然後選擇EBS。這會顯示EBS備份執行個體的清單。
4. 選取所要的執行個體，並查看 Storage (儲存) 索引標籤中顯示的詳細資訊。將至少顯示根儲存設備的以下資訊：
   - 根裝置類型 (例如，EBS)
   - Root device name (根裝置名稱) (例如：`/dev/xvda`)
   - Block devices (區塊型儲存裝置) (例如：`/dev/xvda`、`/dev/sdf` 和 `/dev/sdj`)

   如果執行個體是使用EBS區塊裝置對應的其他磁碟區啟動，則它們會顯示在「區塊裝置」下方。任何執行個體存放區的磁碟區都不會顯示在此索引標籤上。
5. 若要顯示有關EBS磁碟區的其他資訊，請選擇其磁碟區 ID 以前往磁碟區頁面。

Command line

使用指令行檢視執行個體的EBS磁碟區

使用 describe-instance (AWS CLI) 指令或 Get-EC2Instance(AWS Tools for Windows PowerShell) 指令列舉執行個體的區塊裝置對映中的EBS磁碟區。

檢視執行個體區塊型儲存裝置對應的執行個體存放區的磁碟區

執行個體類型決定執行個體可用的執行個體儲存磁碟區數量和類型。如果區塊型設備映射中的執行個體存放磁碟區的數量，超過執行個體可用的執行個體存放磁碟區，請忽略額外的磁碟區。若要檢視執行個體的執行個體存放區磁碟區，請執行lsblk指令 (Linux 執行個體) 或開啟 Windows 磁碟管理 (Windows

執行個體)。要了解每個執行個體類型支援多少個執行個體存放區磁碟區，請參閱 Amazon EC2 執行個體類型規格。

當您檢視執行個體的區塊裝置對應時，您只能看到EBS磁碟區，而不能看到執行個體存放區磁碟區。您用來檢視執行個體的執行個體存放區磁碟區的方法取決於磁碟區類型。

NVMe執行個體儲存體

Linux 執行個體

您可以使用指NVMe令列套件 nvme-cli 來查詢區塊裝置對應中的NVMe執行個體儲存磁碟區。在您的執行個體上下載並安裝套件，然後執行下列命令。

```
[ec2-user ~]$ sudo nvme list
```

以下是執行個體的範例輸出。「模型」(Model) 欄中的文字指出磁碟區是EBS磁碟區還是例證儲存磁碟區。在本範例中，`/dev/nvme1n1` 和 `/dev/nvme2n1` 是執行個體存放區磁碟區。

```
Node             SN                   Model
 Namespace
---------------- -------------------- ----------------------------------------
 ---------
/dev/nvme0n1     vol06afc3f8715b7a597 Amazon Elastic Block Store              1

/dev/nvme1n1     AWS2C1436F5159EB6614 Amazon EC2 NVMe Instance Storage        1

/dev/nvme2n1     AWSB1F4FF0C0A6C281EA Amazon EC2 NVMe Instance Storage        1
  ...
```

Windows 執行個體

您可以使用磁碟管理或 PowerShell 同時列出EBS和實例存儲NVMe卷。如需詳細資訊，請參閱the section called “將NVME磁碟對映至磁碟區”。

HDD或SSD實例存儲卷

您可以使用執行個體中繼資料來查詢區塊裝置對應中的HDD或SSD執行個體儲存磁碟區。NVMe不包括執行個體儲存磁碟區。

執行個體中繼資料的所有要求的基URI礎為`http://169.254.169.254/latest/`。如需詳細資訊，請參閱使用執行個體中繼資料管理EC2執行個。

Linux 執行個體

首先，連接至執行中的執行個體。從執行個體使用此查詢，取得其區塊型設備映射。

IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/block-device-mapping/
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/block-device-mapping/
```

回應將包含執行個體的區塊型儲存設備名稱。例如，執行個體存放區支援的 `m1.small` 執行個體的輸出看起來如下。

```
ami
ephemeral0
root
swap
```

`ami` 設備並非執行個體所看見的根設備。執行個體存放磁碟區將命名為 `ephemeral[0-23]`。`swap` 設備用於分頁檔。如果您也已對映EBS磁碟區，它們會顯示為ebs1ebs2、等等。

若要取得區塊型設備映射中個別區塊型儲存設備的詳細資訊，請將其名稱附加到先前的查詢，如下所示。

IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/block-device-mapping/ephemeral0
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/block-device-mapping/ephemeral0
```

Windows 執行個體

首先，連接至執行中的執行個體。從執行個體使用此查詢，取得其區塊型設備映射。

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/block-device-
mapping/
```

回應將包含執行個體的區塊型儲存設備名稱。例如，執行個體存放區支援的 `m1.small` 執行個體的輸出看起來如下。

```
ami
ephemeral0
root
swap
```

`ami` 設備並非執行個體所看見的根設備。執行個體存放磁碟區將命名為 `ephemeral[0-23]`。`swap` 設備用於分頁檔。如果您也已對映EBS磁碟區，它們會顯示為`ebs1``ebs2`、等等。

若要取得區塊型設備映射中個別區塊型儲存設備的詳細資訊，請將其名稱附加到先前的查詢，如下所示。

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/block-device-
mapping/ephemeral0
```

將磁碟映射到連接到 Amazon EC2 Windows 執行個體的磁碟區

> **Note**
>
> 本主題僅適用於 Windows 執行個體。

您的 Windows 執行個體隨附一個作為根EBS磁碟區的磁碟區。如果您的 Windows 執行個體使用 AWS PV 或 Citrix 光伏驅動程式，您可以選擇性地新增最多 25 個磁碟區，總共計 26 個磁碟區。如需詳細資訊，請參閱Amazon EC2 實例的EBS數量限制。

取決於執行個體的類型，執行個體將可使用 0 到 24 個可能的執行個體存放磁碟區。若要使用執行個體可用的任何執行個體存放區磁碟區，您必須在建立AMI或啟動執行個體時指定這些磁碟區。您也可以在建立AMI或啟動執行個體時新增EBS磁碟區，或在執行個體執行時附加磁碟區。

將磁碟區新增至執行個體時，請指定 Amazon EC2 使用的裝置名稱。如需詳細資訊，請參閱Amazon EC2 執行個體上磁碟區的裝置名稱。 AWS Windows Amazon 機器映像（AMIs）包含一組驅動程序，Amazon 使用這些驅動程序EC2將實例存儲和EBS卷映射到 Windows 磁盤和驅動器代號。如果您從使用 AWS PV 或 Citrix PV 驅動程式的 Windows AMI 啟動執行個體，則可以使用本頁所述的關係，將 Windows 磁碟對應至執行個體存放EBS區和磁碟區。如果您的 Windows AMI 使用 Red Hat PV 驅動程式，您可以更新執行個體以使用 Citrix 驅動程式。如需詳細資訊，請參閱the section called “升級 PV 驅動程式”。

主題

- 將 Amazon EC2 視窗執行個體上的NVMe磁碟映射到
- 將 Amazon EC2 視窗執行個體上的非NVMe磁碟對應至磁碟區

將 Amazon EC2 視窗執行個體上的NVMe磁碟映射到

本主題說明如何檢視執行個體上 Windows 作業系統可用的NVMe磁碟。它也會示範如何將這些NVMe磁碟對應至基礎 Amazon 磁碟EBS區，以及為 Amazon 使用的區塊裝置對映指定的裝置名稱EC2。

主題

- 列出NVMe磁碟
- 將NVMe磁碟對映至磁碟區

列出NVMe磁碟

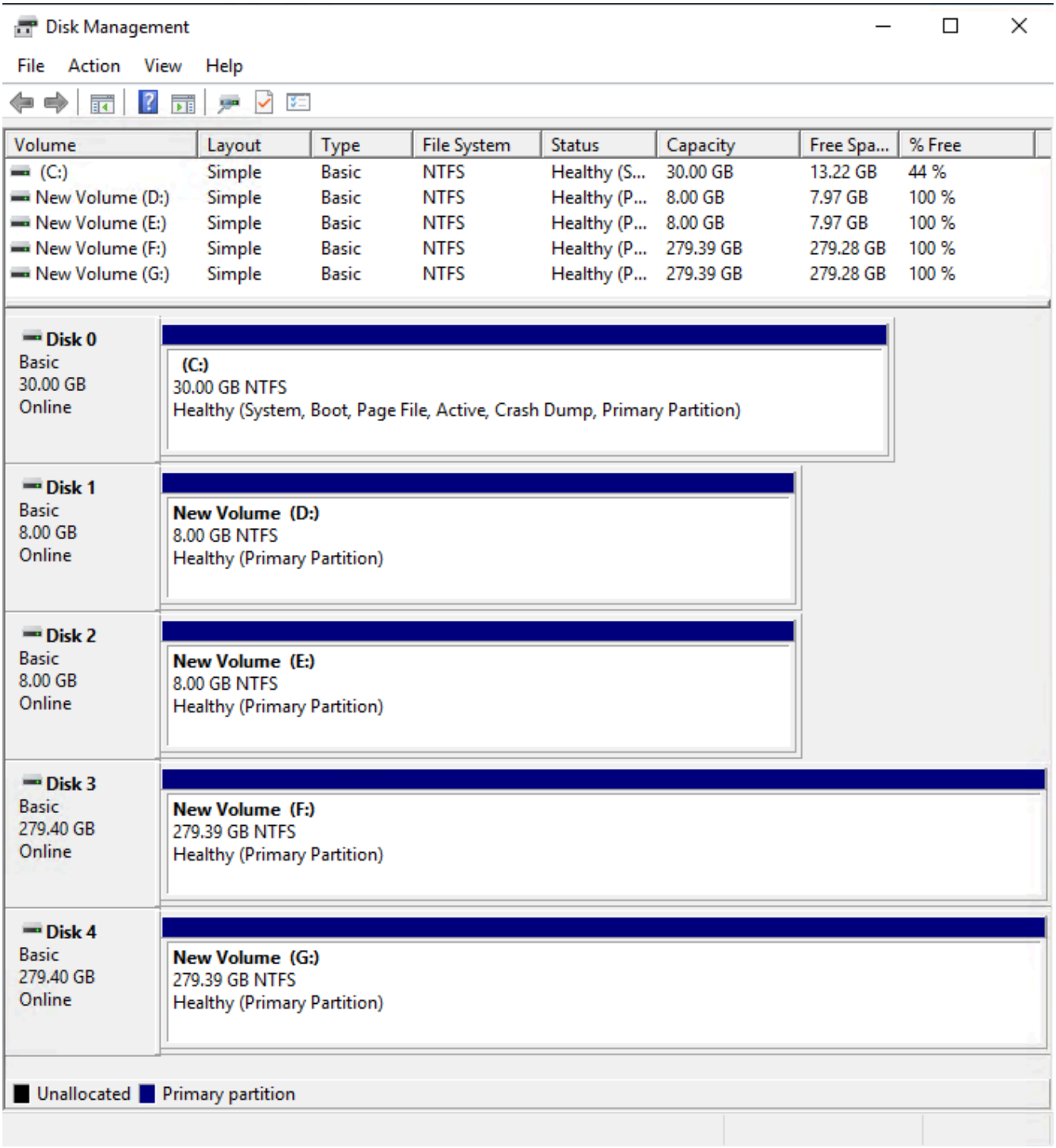
您可以使用磁碟管理公用程式或 Powershell，來搜尋 Windows 執行個體上的磁碟。

Disk Management

搜尋 Windows 執行個體上的磁碟

1. 使用遠端桌面登入 Windows 執行個體。如需詳細資訊，請參閱 使用 Connect 至您的 Windows 執行個體 RDP。
2. 啟動磁碟管理公用程式。
3. 檢閱磁碟。根磁碟區是掛接為的EBS磁碟區`C:\`。如果沒有顯示其他磁碟，表示您在建立AMI或啟動執行個體時並未指定其他磁碟區。

   以下範例顯示如果您啟動具有另外兩個磁碟區的`r5d.4xlarge`執行個體時，可使用的磁EBS碟。



AWS CLI

下列 PowerShell 指令碼會列出每個磁碟及其對應的裝置名稱和磁碟區。它適用於在 AWS Nitro System 上建置的執行個體，這些執行個體使用NVMeEBS和執行個體儲存磁碟區。

Connect 至您的 Windows 執行個體，並執行下列命令以啟用指 PowerShell 令碼執行。

```
Set-ExecutionPolicy RemoteSigned
```

複製下列指令碼，並在您的 Windows 執行個體上將其儲存為 `mapping.ps1`。

```
# List the disks for NVMe volumes

function Get-EC2InstanceMetadata {
    param([string]$Path)
    (Invoke-WebRequest -Uri "http://169.254.169.254/latest/$Path").Content
}

function GetEBSVolumeId {
    param($Path)
    $SerialNumber = (Get-Disk -Path $Path).SerialNumber
    if($SerialNumber -clike 'vol*'){
        $EbsVolumeId = $SerialNumber.Substring(0,20).Replace("vol","vol-")
    }
    else {
       $EbsVolumeId = $SerialNumber.Substring(0,20).Replace("AWS","AWS-")
    }
    return $EbsVolumeId
}

function GetDeviceName{
    param($EbsVolumeId)
    if($EbsVolumeId -clike 'vol*'){

        $Device  = ((Get-EC2Volume -VolumeId $EbsVolumeId ).Attachment).Device
        $VolumeName = ""
    }
     else {
        $Device = "Ephemeral"
        $VolumeName = "Temporary Storage"
    }
    Return $Device,$VolumeName
}

function GetDriveLetter{
    param($Path)
    $DiskNumber =  (Get-Disk -Path $Path).Number
    if($DiskNumber -eq 0){
```

```
        $VirtualDevice = "root"
        $DriveLetter = "C"
        $PartitionNumber = (Get-Partition -DriveLetter C).PartitionNumber
    }
    else
    {
        $VirtualDevice = "N/A"
        $DriveLetter = (Get-Partition -DiskNumber $DiskNumber).DriveLetter
        if(!$DriveLetter)
        {
            $DriveLetter = ((Get-Partition -DiskId $Path).AccessPaths).Split(",")[0]
        }
        $PartitionNumber = (Get-Partition -DiskId $Path).PartitionNumber
    }

    return $DriveLetter,$VirtualDevice,$PartitionNumber

}

$Report = @()
foreach($Path in (Get-Disk).Path)
{
    $Disk_ID = ( Get-Partition -DiskId $Path).DiskId
    $Disk = ( Get-Disk -Path $Path).Number
    $EbsVolumeId  = GetEBSVolumeId($Path)
    $Size =(Get-Disk -Path $Path).Size
    $DriveLetter,$VirtualDevice, $Partition = (GetDriveLetter($Path))
    $Device,$VolumeName = GetDeviceName($EbsVolumeId)
    $Disk = New-Object PSObject -Property @{
      Disk          = $Disk
      Partitions    = $Partition
      DriveLetter   = $DriveLetter
      EbsVolumeId   = $EbsVolumeId
      Device        = $Device
      VirtualDevice = $VirtualDevice
      VolumeName= $VolumeName
    }
 $Report += $Disk
}

$Report | Sort-Object Disk | Format-Table -AutoSize -Property Disk, Partitions,
 DriveLetter, EbsVolumeId, Device, VirtualDevice, VolumeName
```

執行指令碼，如下所示：

```
PS C:\> .\mapping.ps1
```

以下是具有根磁碟區、兩個磁碟區和兩個執行個體儲存EBS磁碟區的執行個體輸出範例。

```
Disk Partitions DriveLetter EbsVolumeId           Device    VirtualDevice VolumeName
---- ---------- ----------- -----------           ------    ------------- ----------
   0          1 C           vol-03683f1d861744bc7 /dev/sda1 root
   1          1 D           vol-082b07051043174b9 xvdb      N/A
   2          1 E           vol-0a4064b39e5f534a2 xvdc      N/A
   3          1 F           AWS-6AAD8C2AEEE1193F0 Ephemeral N/A           Temporary
 Storage
   4          1 G           AWS-13E7299C2BD031A28 Ephemeral N/A           Temporary
 Storage
```

如果您未在 Windows 執行個體 PowerShell 上設定 Windows 適用工具的認證，指令碼將無法取得EBS磁碟區識別碼，並在`EbsVolumeId`欄中使用 N/A。

將NVMe磁碟對映至磁碟區

在 AWS Nitro 系統上建置的執行個體時，EBS磁碟區會以NVMe裝置的形式公開。您可以使用「取得磁碟」指令將 Windows 磁碟號碼對映至EBS磁碟區。IDs

```
PS C:\> Get-Disk
Number Friendly Name Serial Number                    HealthStatus
 OperationalStatus      Total Size Partition

                      Style
------ ------------- -------------                    ------------
 -----------------      ---------- ----------
3      NVMe Amazo... AWS6AAD8C2AEEE1193F0_00000001.   Healthy              Online
             279.4 GB MBR
4      NVMe Amazo... AWS13E7299C2BD031A28_00000001.   Healthy              Online
             279.4 GB MBR
2      NVMe Amazo... vol0a4064b39e5f534a2_00000001.   Healthy              Online
                  8 GB MBR
0      NVMe Amazo... vol03683f1d861744bc7_00000001.   Healthy              Online
                 30 GB MBR
1      NVMe Amazo... vol082b07051043174b9_00000001.   Healthy              Online
                  8 GB MBR
```

您也可以執行指ebsnvme-id令，將NVMe磁碟號碼對應至磁碟EBS區IDs和裝置名稱。

```
PS C:\> C:\PROGRAMDATA\Amazon\Tools\ebsnvme-id.exe
Disk Number: 0
Volume ID: vol-03683f1d861744bc7
Device Name: sda1

Disk Number: 1
Volume ID: vol-082b07051043174b9
Device Name: xvdb

Disk Number: 2
Volume ID: vol-0a4064b39e5f534a2
Device Name: xvdc
```

將 Amazon EC2 視窗執行個體上的非NVMe磁碟對應至磁碟區

本主題說明如何檢視執行個體上 Windows 作業系統可用的非NVMe磁碟。它也會示範如何將這些非NVMe磁碟對應至基礎 Amazon 磁碟EBS區，以及為 Amazon 使用的區塊裝置對映指定的裝置名稱EC2。

主題

- 列出非NVMe磁碟
- 將非磁碟對映至NVMe磁碟區

列出非NVMe磁碟

您可以使用磁碟管理或在 Windows 執行個體上找到磁碟 PowerShell。

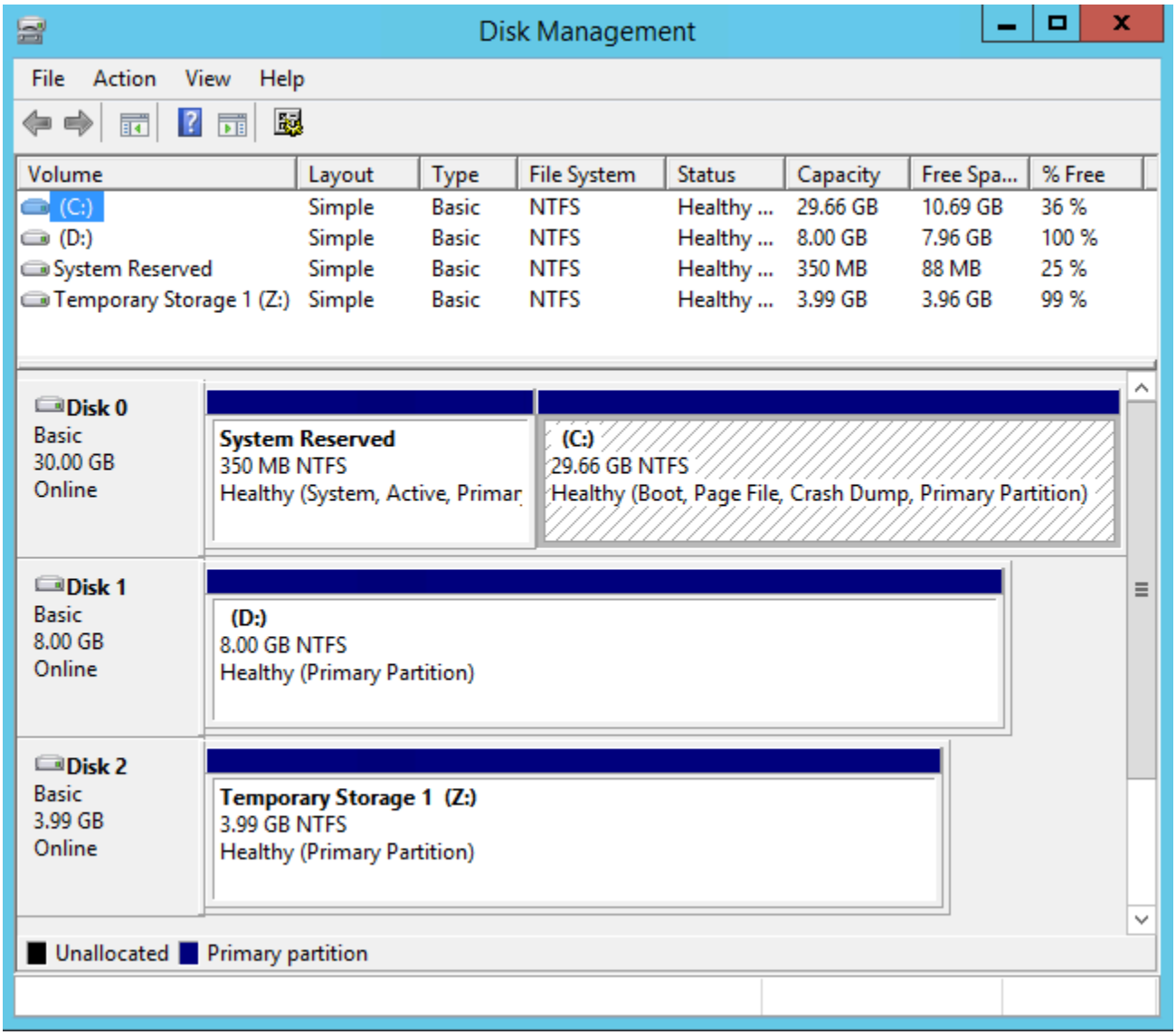
Disk Management

搜尋 Windows 執行個體上的磁碟

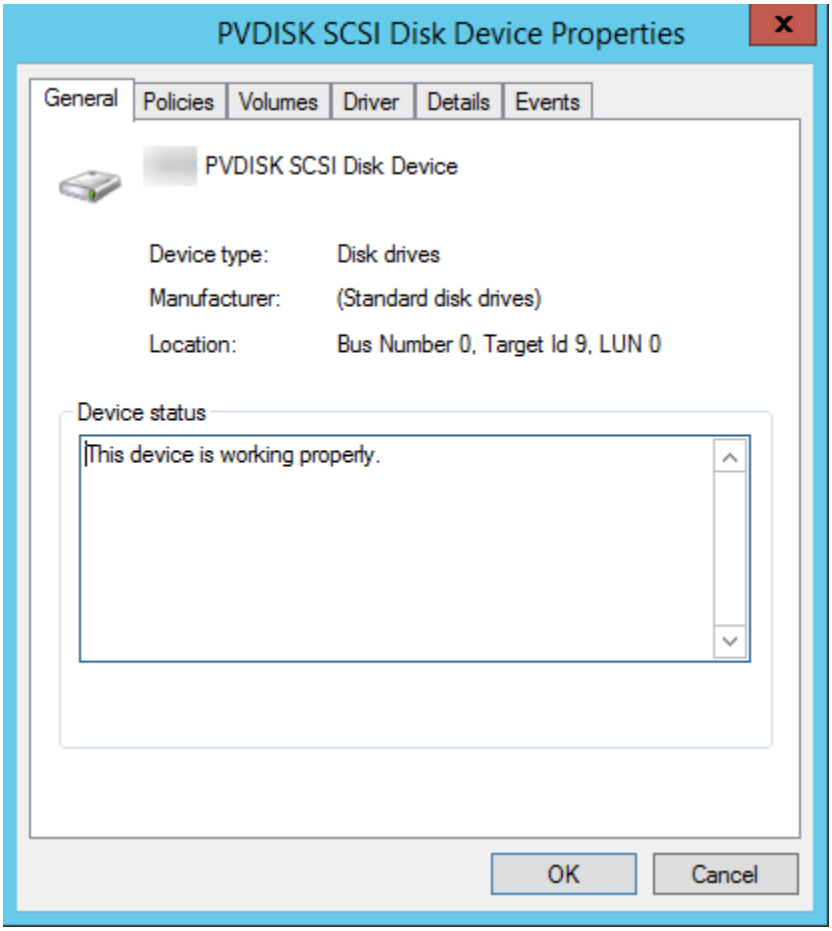
1. 使用遠端桌面登入 Windows 執行個體。如需詳細資訊，請參閱 使用 Connect 至您的 Windows 執行個體 RDP。
2. 啟動磁碟管理公用程式。

   在工作列上，以滑鼠右鍵按一下 Windows 標誌，然後選擇 [磁碟管理]。
3. 檢閱磁碟。根磁碟區是掛接為的EBS磁碟區`C:\`。如果沒有顯示其他磁碟，表示您在建立AMI或啟動執行個體時並未指定其他磁碟區。

以下範例顯示如果您使用`m3.medium`執行個體存放磁碟區 (磁碟 2) 和額外磁碟區 (磁碟 1) 啟動執行個體時，可使用的磁EBS碟。



4. 在以灰色窗格標示的磁碟 1 上，按一下滑鼠右鍵，然後選取 Properties (屬性)。記下 Location (位置) 的值，然後在將非磁碟對映至NVMe磁碟區的表格中查閱。例如，下列磁碟具有位置匯流排號碼 0、目標識別碼 9、LUN 0。根據EBS磁碟區的表格，此位置的裝置名稱為`xvdj`。



PowerShell

下列 PowerShell 指令碼會列出每個磁碟及其對應的裝置名稱和磁碟區。

要求與限制

- 需要 Windows Server 2012 或更新版本。
- 需要憑證才能取得EBS磁碟區 ID。您可以使用的 [工具] 設定設定檔 PowerShell，或將IAM角色附加至執行個體。
- 不支援NVMe磁碟區。
- 不支援動態磁碟。

Connect 至您的 Windows 執行個體，並執行下列命令以啟用指 PowerShell 令碼執行。

```
Set-ExecutionPolicy RemoteSigned
```

複製下列指令碼，並在您的 Windows 執行個體上將其儲存為 `mapping.ps1`。

```
# List the disks
function Convert-SCSITargetIdToDeviceName {
  param([int]$SCSITargetId)
  If ($SCSITargetId -eq 0) {
    return "sda1"
  }
  $deviceName = "xvd"
  If ($SCSITargetId -gt 25) {
    $deviceName += [char](0x60 + [int]($SCSITargetId / 26))
  }
  $deviceName += [char](0x61 + $SCSITargetId % 26)
  return $deviceName
}

[string[]]$array1 = @()
[string[]]$array2 = @()
[string[]]$array3 = @()
[string[]]$array4 = @()

Get-WmiObject Win32_Volume | Select-Object Name, DeviceID | ForEach-Object {
  $array1 += $_.Name
  $array2 += $_.DeviceID
}

$i = 0
While ($i -ne ($array2.Count)) {
  $array3 += ((Get-Volume -Path $array2[$i] | Get-Partition | Get-
Disk).SerialNumber) -replace "_[^ ]*$" -replace "vol", "vol-"
  $array4 += ((Get-Volume -Path $array2[$i] | Get-Partition | Get-
Disk).FriendlyName)
  $i ++
}

[array[]]$array = $array1, $array2, $array3, $array4

Try {
  $InstanceId = Get-EC2InstanceMetadata -Category "InstanceId"
  $Region = Get-EC2InstanceMetadata -Category "Region" | Select-Object -
ExpandProperty SystemName
}
Catch {
```

```
  Write-Host "Could not access the instance Metadata using AWS Get-
EC2InstanceMetadata CMDLet.
Verify you have AWSPowershell SDK version '3.1.73.0' or greater installed and
 Metadata is enabled for this instance." -ForegroundColor Yellow
}
Try {
  $BlockDeviceMappings = (Get-EC2Instance -Region $Region -Instance
 $InstanceId).Instances.BlockDeviceMappings
  $VirtualDeviceMap = (Get-EC2InstanceMetadata -Category
 "BlockDeviceMapping").GetEnumerator() | Where-Object { $_.Key -ne "ami" }
}
Catch {
  Write-Host "Could not access the AWS API, therefore, VolumeId is not available.
Verify that you provided your access keys or assigned an IAM role with adequate
 permissions." -ForegroundColor Yellow
}

Get-disk | ForEach-Object {
  $DriveLetter = $null
  $VolumeName = $null
  $VirtualDevice = $null
  $DeviceName = $_.FriendlyName

  $DiskDrive = $_
  $Disk = $_.Number
  $Partitions = $_.NumberOfPartitions
  $EbsVolumeID = $_.SerialNumber -replace "_[^ ]*$" -replace "vol", "vol-"
  if ($Partitions -ge 1) {
    $PartitionsData = Get-Partition -DiskId $_.Path
    $DriveLetter = $PartitionsData.DriveLetter | Where-object { $_ -notin @("",
 $null) }
    $VolumeName = (Get-PSDrive | Where-Object { $_.Name -in
 @($DriveLetter) }).Description | Where-object { $_ -notin @("", $null) }
  }
  If ($DiskDrive.path -like "*PROD_PVDISK*") {
    $BlockDeviceName = Convert-SCSITargetIdToDeviceName((Get-WmiObject -Class
 Win32_Diskdrive | Where-Object { $_.DeviceID -eq ("\\.\PHYSICALDRIVE" +
 $DiskDrive.Number) }).SCSITargetId)
    $BlockDeviceName = "/dev/" + $BlockDeviceName
    $BlockDevice = $BlockDeviceMappings | Where-Object { $BlockDeviceName -like "*"
 + $_.DeviceName + "*" }
    $EbsVolumeID = $BlockDevice.Ebs.VolumeId
    $VirtualDevice = ($VirtualDeviceMap.GetEnumerator() | Where-Object { $_.Value -
eq $BlockDeviceName }).Key | Select-Object -First 1
```

```
  }
  ElseIf ($DiskDrive.path -like "*PROD_AMAZON_EC2_NVME*") {
    $BlockDeviceName = (Get-EC2InstanceMetadata -Category
 "BlockDeviceMapping")."ephemeral$((Get-WmiObject -Class Win32_Diskdrive | Where-
Object { $_.DeviceID -eq ("\\.\PHYSICALDRIVE" + $DiskDrive.Number) }).SCSIPort - 2)"
    $BlockDevice = $null
    $VirtualDevice = ($VirtualDeviceMap.GetEnumerator() | Where-Object { $_.Value -
eq $BlockDeviceName }).Key | Select-Object -First 1
  }
  ElseIf ($DiskDrive.path -like "*PROD_AMAZON*") {
    if ($DriveLetter -match '[^a-zA-Z0-9]') {
      $i = 0
      While ($i -ne ($array3.Count)) {
        if ($array[2][$i] -eq $EbsVolumeID) {
          $DriveLetter = $array[0][$i]
          $DeviceName = $array[3][$i]
        }
        $i ++
      }
    }
    $BlockDevice = ""
    $BlockDeviceName = ($BlockDeviceMappings | Where-Object { $_.ebs.VolumeId -eq
 $EbsVolumeID }).DeviceName
  }
  ElseIf ($DiskDrive.path -like "*NETAPP*") {
    if ($DriveLetter -match '[^a-zA-Z0-9]') {
      $i = 0
      While ($i -ne ($array3.Count)) {
        if ($array[2][$i] -eq $EbsVolumeID) {
          $DriveLetter = $array[0][$i]
          $DeviceName = $array[3][$i]
        }
        $i ++
      }
    }
    $EbsVolumeID = "FSxN Volume"
    $BlockDevice = ""
    $BlockDeviceName = ($BlockDeviceMappings | Where-Object { $_.ebs.VolumeId -eq
 $EbsVolumeID }).DeviceName
  }
  Else {
    $BlockDeviceName = $null
    $BlockDevice = $null
  }
```

```
  New-Object PSObject -Property @{
    Disk          = $Disk;
    Partitions    = $Partitions;
    DriveLetter   = If ($DriveLetter -eq $null) { "N/A" } Else { $DriveLetter };
    EbsVolumeId   = If ($EbsVolumeID -eq $null) { "N/A" } Else { $EbsVolumeID };
    Device        = If ($BlockDeviceName -eq $null) { "N/A" } Else
 { $BlockDeviceName };
    VirtualDevice = If ($VirtualDevice -eq $null) { "N/A" } Else { $VirtualDevice };
    VolumeName    = If ($VolumeName -eq $null) { "N/A" } Else { $VolumeName };
    DeviceName    = If ($DeviceName -eq $null) { "N/A" } Else { $DeviceName };
  }
} | Sort-Object Disk | Format-Table -AutoSize -Property Disk, Partitions,
 DriveLetter, EbsVolumeId, Device, VirtualDevice, DeviceName, VolumeName
```

執行指令碼，如下所示：

```
PS C:\> .\mapping.ps1
```

下列為範例輸出。

```
Disk  Partitions  DriveLetter   EbsVolumeId             Device      VirtualDevice
 DeviceName              VolumeName
----  ----------  -----------   -----------             ------      -------------
 ----------              ----------
   0           1            C   vol-0561f1783298efedd   /dev/sda1   N/A
 NVMe Amazon Elastic B   N/A
   1           1            D   vol-002a9488504c5e35a   xvdb        N/A
 NVMe Amazon Elastic B   N/A
   2           1            E   vol-0de9d46fcc907925d   xvdc        N/A
 NVMe Amazon Elastic B   N/A
```

如果您未在 Windows 執行個體上提供認證，指令碼將無法取得EBS磁碟區識別碼，並在`EbsVolumeId`欄中使用 N/A。

將非磁碟對映至NVMe磁碟區

執行個體的區塊型儲存設備驅動程式，會在掛載磁碟區時指派實際的磁碟區名稱。

映射項目

- 執行個體儲存體磁碟區

- EBS磁碟區

執行個體儲存體磁碟區

下表說明 Citrix 光伏和光 AWS 伏驅動程式如何將非NVMe執行個體儲存磁碟區對應至 Windows 磁碟區。可用執行個體存放磁碟區的數量，會取決於執行個體的類型。如需詳細資訊，請參閱 Amazon EC2 執行個體的執行個體存放量限制。

位置	裝置名稱
公交車編號 0，目標識別碼 78，LUN0	xvdca
匯流排編號 0，目標識別碼 79，LUN0	xvdcb
匯流排編號 0，目標識別碼 80，LUN0	xvdcc
匯流排編號 0，目標識別碼 81，LUN0	xvdcd
匯流排編號 0，目標識別碼 82，LUN0	xvdce
匯流排編號 0，目標識別碼 83，LUN0	xvdcf
匯流排編號 0，目標識別碼 84，LUN0	xvdcg
匯流排編號 0，目標識別碼 85，LUN0	xvdch
匯流排編號 0，目標識別碼 86，LUN0	xvdci
匯流排編號 0，目標識別碼 87，LUN0	xvdcj
公交車編號 0，目標識別碼 88，LUN0	xvdck
公交車編號 0，目標識別碼 89，LUN0	xvdcl

EBS磁碟區

下表說明 Citrix 光伏和光 AWS 伏驅動程式如何將非磁碟區對應至 Windows NVME EBS 磁碟區。

位置	裝置名稱
匯流排編號 0，目標識別碼 0，LUN0	/dev/sda1
匯流排編號 0，目標識別碼 1，LUN0	xvdb
匯流排編號 0，目標識別碼 2，LUN0	xvdc
匯流排編號 0，目標識別碼 3，LUN0	xvdd
匯流排編號 0，目標識別碼 4，LUN0	xvde
匯流排編號 0，目標識別碼 5，LUN0	xvdf
匯流排編號 0，目標識別碼 6，LUN0	xvdg
匯流排編號 0，目標識別碼 7，LUN0	xvdh
匯流排編號 0，目標識別碼 8，LUN0	xvdi
匯流排編號 0，目標識別碼 9，LUN0	xvdj
匯流排編號 0，目標識別碼 10，LUN0	xvdk
匯流排編號 0，目標識別碼 11，LUN0	xvdl
匯流排編號 0，目標識別碼 12，LUN0	xvdm
匯流排編號 0，目標識別碼 13，LUN0	xvdn
匯流排編號 0，目標識別碼 14，LUN0	xvdo
匯流排編號 0，目標識別碼 15，LUN0	xvdp
匯流排編號 0，目標識別碼 16，LUN0	xvdq
匯流排編號 0，目標識別碼 17，LUN0	xvdr
匯流排編號 0，目標識別碼 18，LUN0	xvds
匯流排編號 0，目標識別碼 19，LUN0	xvdt

位置	裝置名稱
匯流排編號 0，目標識別碼 20，LUN0	xvdu
匯流排編號 0，目標識別碼 21，LUN0	xvdv
匯流排編號 0，目標識別碼 22，LUN0	xvdw
匯流排編號 0，目標識別碼 23，LUN0	xvdx
匯流排編號 0，目標識別碼 24，LUN0	xvdy
匯流排編號 0，目標識別碼 25，LUN0	xvdz

Amazon EC2 Linux 實例上的寫入防止撕裂

> **Note**
>
> 僅 Linux 執行個體支援防止寫入損毀。

防寫是一項區塊儲存功能，其設計目的是改善 I/O 密集型關聯式資料庫工作負載的效能，並在不影響資料復原能力的情況下減少延遲。 AWS 使用 InnoDB 或 XtraDB 作為資料庫引擎的關聯式資料庫，例如我SQL和 MariaDB，將受益於寫入防止撕裂。

一般而言，使用大於儲存裝置電源故障不可分割性分頁的關聯式資料庫會使用資料記錄機制來預防未完成的寫入。MariaDB 和我SQL使用雙寫緩衝區文件將數據寫入數據表之前記錄數據。在寫入不完整或未完成的情況下，由於作業系統當機或寫入交易期間斷電，資料庫可以從雙寫緩衝區復原資料。與寫入雙寫緩衝區相關聯的額外 I/O 負荷會影響資料庫效能和應用程式延遲，並減少每秒可處理的交易數量。有關雙寫緩衝區的更多信息，請參閱 MariaDB 和我的文檔。SQL

透過防止寫入損毀，資料會all-or-nothing寫入寫入交易中的儲存體，因此無需使用雙寫緩衝區。這可避免在寫入交易期間發生作業系統當機或斷電時，將部分或未完成的資料寫入儲存區。每秒處理的交易數量最多可增加 30%，且寫入延遲最多可減少 50%，並且不會影響工作負載的彈性。

定價

使用預防未完成的寫入不會產生額外成本。

目錄

- 區塊大小可防止 Amazon 上的寫入撕裂 EC2
- 在 Amazon 上使用防寫防止撕裂的要求 EC2
- 檢查 Amazon EC2 執行個體支援是否防止寫入損毀
- 在 Amazon 上設定您的工作負載EC2以防止寫入損毀

區塊大小可防止 Amazon 上的寫入撕裂 EC2

預防未完成的寫入支援 4 KiB、8 KiB 和 16 KiB 資料區塊的寫入作業。數據塊起始邏輯塊地址（LBA）必須與 4 KiB，8 KiB 或 16 KiB 的相應塊邊界大小對齊。例如，對於 16 KiB 寫入作業，資料區塊起點 LBA必須與 16 KiB 的區塊邊界大小對齊。

下表顯示各種儲存體和執行個體類型的支援。

	4 KiB 區塊	8 KiB 區塊	16 KiB 區塊
執行個體儲存體磁碟區	所有NVMe執行個體儲存磁碟區連接至目前一代 I 系列執行個體	硝基支持的 i4i，IM4GN 和 IS4gen 實例。 AWS SSD	
Amazon EBS 卷	連接到 AWS Nitro 系統上建置執行個體的所有 Amazon EBS 磁碟區。		

若要確認您的執行個體和磁碟區是否支援預防未完成的寫入，請查詢執行個體是否支援預防未完成的寫入和其他詳細資訊，例如支援的區塊和邊界大小。如需詳細資訊，請參閱檢查 Amazon EC2 執行個體支援是否防止寫入損毀。

在 Amazon 上使用防寫防止撕裂的要求 EC2

若要預防未完成的寫入功能正常運作，I/O 作業必須符合 NTWPU、NTWGU、NTWBU 欄位中指定的大小、對齊和邊界要求。您必須設定作業系統，以確保特定的儲存子系統 (檔案系統RAID等) 在提交至裝置之前，不會修改儲存堆疊下的 I/O 屬性，包括區塊合併、分割或區塊位址重新定位。LVM

已透過下列組態測試預防未完成的寫入：

- 支援所需區塊大小的執行個體類型和儲存體類型。
- 核心版本 5.10 或更高版本的 Amazon Linux 2。

- 已啟用 `bigalloc` 且叢集大小為 16 KiB 的 ext4，以及最新的 ext4 公用程式 (e2fsprogs 1.46.5 或更高版本)。
- `O_DIRECT` 檔案存取模式繞過 Linux 核心緩衝區快取。

> **Note**
> 您不需要為「我的」SQL 和「MariaDB」工作負載停用 I/O 合併。

檢查 Amazon EC2 執行個體支援是否防止寫入損毀

若要確認您的執行個體和磁碟區是否支援寫入防止損毀，以及檢視包含失效寫入防護資訊的NVMe命名空間廠商特定資料，請使用下列命令。

```
$ sudo nvme id-ns -v device_name
```

> **Note**
> 該命令以十六進制返回供應商特定的信息，並帶有ASCII解釋。您可能需要在可讀取和剖析輸出的應用程式中建置類似於 `ebsnvme-id` 的工具。

例如，下列命令會傳回的NVMe命名空間廠商特定資料，其中包含的撕毀寫入防護資訊`/dev/nvme1n1`。

```
$ sudo nvme id-ns -v /dev/nvme1n1
```

如果您的執行個體和磁碟區支援損毀的寫入防護，它會在NVMe命名空間廠商特定資料中傳回下列 AWS 撕裂的寫入防護資訊。

> **Note**
> 下表中的位元組表示從NVMe命名空間供應商特定資料開頭算起的偏移量 (以位元組為單位)。

位元組	描述
0:31	例如，裝置配件掛載點的名稱，例如 /dev/xvda 。您可以在磁碟區附件請求期間提供此項目，Amazon EC2 執行個體可用來建立指向區NVMe塊裝置的符號連結 (nvmeXn1)。
32:63	磁碟區 ID。例如：vol01234567890abcdef 。此欄位可用於將NVMe裝置對應至連接的磁碟區。
64:255	保留以供日後使用。
256:257	命名空間撕毀寫入防止單位大小 (NTWPU)。此欄位指出在電源故障或錯誤情況期間保證以原子方式寫入的寫入作業的命名空NVM間特定大小。此欄位邏輯區塊 (用以零為基礎的值表示) 中指定。
258:259	命名空間撕毀寫入防止粒度大小 (NTWPG)。此欄位指出在電源故障或錯誤情況期間，保證以原子方NTWPU式寫入寫入作業的命名空NVM間特定大小增量。也就是說，大小應為 NTWPG * n <= NTWPU，其中 n 是正整數。寫操作LBA偏移也必須與此字段對齊。此欄位邏輯區塊 (用以零為基礎的值表示) 中指定。
260:263	命名空間撕裂寫入防止邊界大小 (NTWPB)。此欄位指出此 NTWPU 值的不可分割邊界大小。寫入此命名空間，跨越原子邊界並不保證在電源故障或錯誤情況NVM期間以原子方式寫入。0h 的值表示停電或錯誤情況沒有不可分割邊界。所有其他值會使用與 NTWPU 欄位相同的編碼方式，以邏輯區塊來指定大小。

在 Amazon 上設定您的工作負載EC2以防止寫入損毀

支援的執行個體類型 (含支援的磁碟區) 預設啟用預防未完成的寫入。您不需要啟用任何其他設定，即可啟用磁碟區或執行個體，進而預防未完成的寫入。

> **Note**
>
> 不會對不支援預防未完成的寫入的工作負載造成效能影響。您不需要為這些工作負載進行任何變更。

確實支援預防未完成的寫入，但未設定為使用它的工作負載，會繼續使用雙寫緩衝區，而且不會獲得任何效能優勢。

若要設定 My SQL 或 MariaDB 軟體堆疊以停用雙重寫入緩衝區並使用損毀的寫入預防，請完成以下步驟：

1. 配置您的磁碟區以使用 ext4 檔案系統與 BigAlloc 選項，並將叢集大小設定為 4 KiB、8 KiB 或 16 KiB。使 BigAlloc 用叢集大小為 4 KiB、8 KiB 或 16 KiB，可確保檔案系統配置與相應邊界對齊的檔案。

```
$  mkfs.ext4 -O bigalloc -C 4096|8192|16384 device_name
```

> **Note**
>
> 對於我SQL和 MariaDB，您必須使用`-C 16384`來匹配數據庫頁面大小。將分配精密度設定為頁面大小倍數以外的值，可能會導致分配與儲存裝置的預防未完成的寫入邊界不符。

例如：

```
$  mkfs.ext4 -O bigalloc -C 16384 /dev/nvme1n1
```

2. 將 InnoDB 設定為使用 `O_DIRECT` 排清方法並關閉 InnoDB 雙寫。使用您偏好的文字編輯器開啟 `/etc/my.cnf`，並更新 `innodb_flush_method` 和 `innodb_doublewrite` 參數，如下所示：

```
innodb_flush_method=O_DIRECT
innodb_doublewrite=0
```

> **Important**
>
> 如果您使用的是邏輯磁碟區管理員 (LVM) 或其他儲存虛擬化層，請確定磁碟區的起始偏移量已對齊 16 KiB 倍數。這與基礎NVMe儲存裝置相關，以說明儲存虛擬化層所使用的中繼資料標頭和超級區塊。如果您新增LVM實體磁碟區的偏移量，可能會導致檔案系統配置與NVMe裝置的位移之間不對齊，這會使寫入防止撕裂失效。如需詳細資訊，請參閱 Linux 手冊頁面中的 `--dataalignmentoffset`。

應用程式一致的視窗VSS型 Amazon EBS 快照

您可以使用執AWS Systems Manager 行命令，為連接到 Amazon EC2 Windows 執行個體的所有 Amazon EBS 磁碟區拍攝應用程式一致的快照。快照處理程序使用 Windows 磁碟區陰影複製服務 (VSS) 來取得VSS感知應用程式的EBS磁碟區層級備份。快照包括來自這些應用程式和磁碟之間擱置中交易的資料。若需要備份所有連接的磁碟區，您不需要關閉執行個體或中斷其連結。

使用VSS基於EBS快照不會產生額外費用。您只需為備份程序建立的EBS快照付費。如需詳細資訊，請參閱 Amazon EBS 快照如何計費？

> **Note**
>
> 以 Windows VSS 為基礎的應用程式一致快照僅支援 Windows 執行個體。

目錄

- 什麼是VSS？
- VSS基於 Amazon EBS 快照解決方案的運作方式
- 建立以 Windows 為VSS基礎之EBS快照的必要
- 為您的 EC2 Windows 執行個體建VSS立EBS快照
- 疑難排解 Windows VSS EBS 快照
- 從基礎快照還原 Windows 執行個體VSS的EBS磁碟區
- AWS VSS解決方案的版本

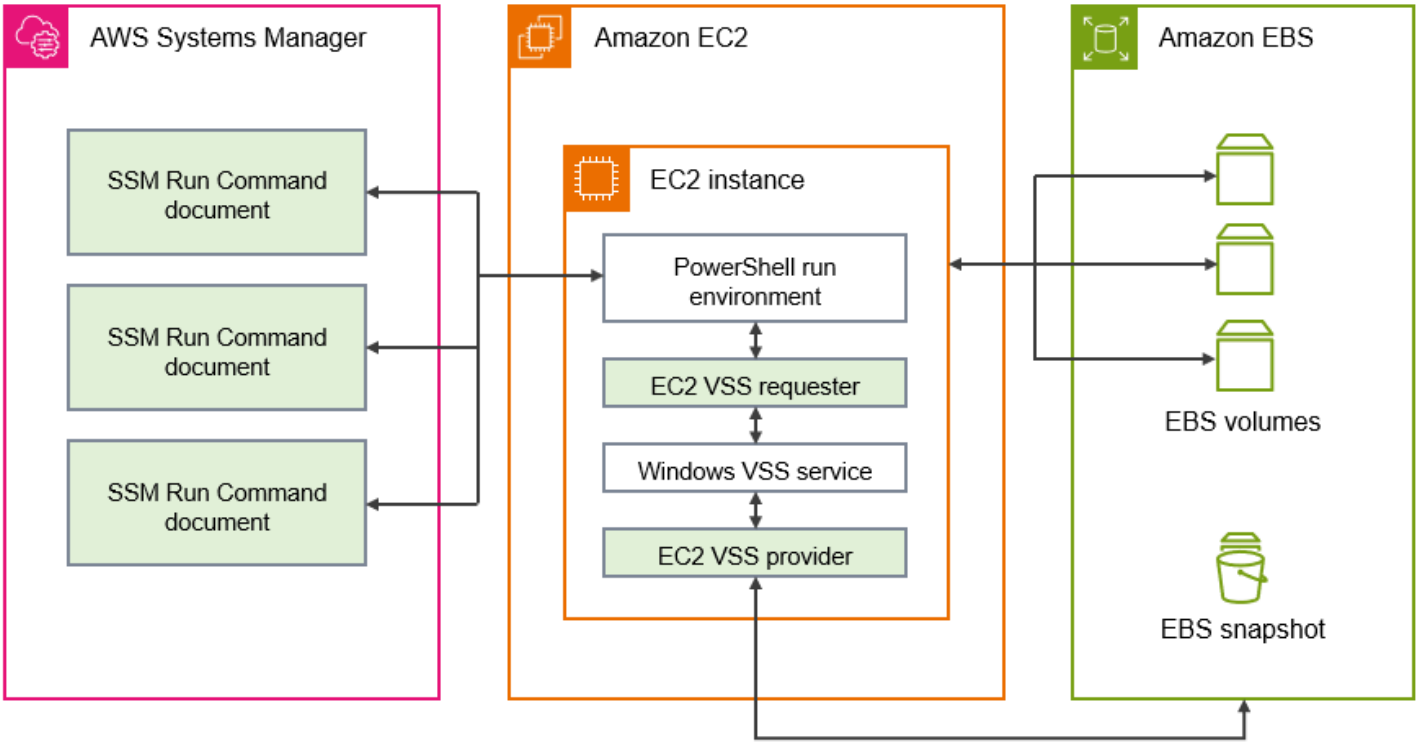
什麼是VSS？

磁碟區快照複製服務 (VSS) 是一項包含在 Microsoft Windows 中的備份與復原技術。當正在使用電腦檔案或磁碟區時，其可建立其備份複本或快照。如需詳細資訊，請參閱磁碟區陰影複製服務。

若要建立應用程式一致性快照，會涉及以下軟體元件。

- VSS服務 — 視窗作業系統的一部分
- VSS要求者 — 要求建立陰影複製的軟體
- VSSwriter — 通常作為應用程序的一部分提供，例如SQL服務器，以確保一致的數據集備份
- VSS提供者 — 建立基礎磁碟區陰影複本的元件

以 Windows 為VSS基礎的 Amazon EBS 快照解決方案包含多個可協助備份建立的 Systems Manager (SSM) 執行命令文件，以及一個系統管理員代理商套件 (稱為)`AwsVssComponents`，其中包括EC2VSS請求EC2VSS者和供應商。此`AwsVssComponents`套件必須安裝在 EC2 Windows 執行個體上，才能擷取應用程式一致的EBS磁碟區快照。下圖說明這些軟體元件之間的關係。



VSS基於 Amazon EBS 快照解決方案的運作方式

取得應用程式一致VSS性EBS快照指令碼的程序包含下列步驟。

1. 完成建立以 Windows 為VSS基礎之EBS快照的必要。
2. 輸入`AWSEC2-VssInstallAndSnapshotSSM`文件的參數，並使用執行命令來執行此文件。如需詳細資訊，請參閱執行 AWSEC2-命VssInstallAndSnapshot 令文件 (建議使用)。
3. 執行個體上的 Windows VSS 服務會協調執行應用程式的所有進行中 I/O 作業。
4. 系統會排清所有 I/O 緩衝區，並暫時暫停所有 I/O 操作。暫停時間最多持續 10 秒。
5. 在暫停期間，系統會為連接至執行個體的所有磁碟區建立快照。
6. 系統會取消暫停，並恢復 I/O 操作。
7. 系統會將所有新建立的快照新增至快照清單。EBS系統會使用:AppConsistenttrue 標記此程序成功建立的所有VSS基礎EBS快照。

8. 如果您需要從快照還原，可以使用從快照建立磁碟區的標準EBS程序，或使用範例指令碼將所有磁碟區還原至執行個體，如中所述從基礎快照還原 Windows 執行個體VSS的EBS磁碟區。

建立以 Windows 為VSS基礎之EBS快照的必要

您可VSS以使用 Systems Manager 執行命令或 Amazon Data Lifecycle Manager 建立EBS快照。AWS Backup以下先決條件適用所有解決方案。

系統要求

請確定您的 EC2 Windows 執行個體符合建立VSS快照的所有系統需求，包括支援的 Windows 作業系統版本。 NET架構、PowerShell AWS Tools for Windows PowerShell、和 AWS Systems Manager 代理程式。

IAM權限

附加到 Amazon EC2 Windows 執行個體的IAM角色必須具有建立應用程式一致快照的VSS權限。若要授與必要權限，您可以將`AWSEC2VssSnapshotPolicy`受管理政策附加至執行個體設定檔。

VSS元件

若要在 Windows 作業系統上建立應用程式一致快照，則必須在執行個體上安裝 `AwsVssComponents` 套件。該軟件包包含一個作為VSS請求者的實例EC2VSS代理程序以及卷的提EC2VSS供EBS者。

系統要求

安裝系統管理員代理程式

VSS是由系統管理員代理程式使用 PowerShell. 請確定您已在EC2執行個體上安裝 SSM Agent 版本`3.0.502.0`或更新版本。如果您已使用舊版SSM代理程式，請使用「執行命令」進行更新。如需詳細資訊，請參閱使用者指南中的為 Amazon EC2 執行個體設定 Systems Manager 和在適用於 Windows Server 的 Amazon EC2 執行個體上AWS Systems Manager 使用SSM代理程式。

Amazon EC2 窗口實例要求

VSS執行 Windows 伺服器 2016 及更新版本的執行個體支援以EBS快照為基礎。

。 NET框架版本

該`AwsVssComponents`軟件包需要。 NET框架版本`4.6`或更高版本。視窗伺服器 2016 之前的 Windows 作業系統版本預設為. NET框架。如果您的執行個體使用舊版的. NET框架中，您必須使用 Windows 更新安裝版本`4.6`或更高版本。

AWS Tools for Windows PowerShell 版本

確定您的執行個體執行 AWS Tools for Windows PowerShell 版本`3.3.48.0`或更新版本。若要檢查您的版本，請在執行個體的 PowerShell 終端機中執行下列命令。

```
C:\> Get-AWSPowerShellVersion
```

如果您需要更新 AWS Tools for Windows PowerShell 執行個體，請參閱《AWS Tools for Windows PowerShell 使用者指南》 AWS Tools for Windows PowerShell中的〈安裝〉。

視窗 PowerShell 版

確定您的執行個體執行的是 Windows PowerShell 主要版本`34`、或`5`。若要檢查您的版本，請在執行個體的 PowerShell 終端機中執行下列命令。

```
C:\> $PSVersionTable.PSVersion
```

PowerShell 語言模式

確定您的執行個體已將 PowerShell 語言模式設定為`FullLanguage`。如需詳細資訊，請參閱 Microsoft 文件中的 about_Language_Modes。

使用受IAM管理的原則授與快照VSS的權限

受AWSEC2VssSnapshotPolicy管理的原則可讓 Systems Manager 在 Windows 執行個體上執行下列動作：

- 建立和標記EBS快照
- 創建和標記 Amazon 機器映像（AMIs）
- 將中繼資料 (例如裝置 ID) 附加至VSS建立的預設快照標籤。

本主題說明VSS受管政策的權限詳細資料，以及如何將其附加到您的EC2執行個體設定檔IAM角色。

目錄

- AWSEC2VssSnapshotPolicy受管理策略詳細
- 將VSS快照管理政策附加到您的執行個體設定檔角色

AWSEC2VssSnapshotPolicy受管理策略詳細

受 AWS 管政策是 Amazon 為 AWS 客戶提供的獨立政策。 AWS 受管理的原則是為了授與常見使用案例的權限而設計。您無法變更受 AWS 管理原則中定義的權限。不過，您可以複製政策，並將其用作您使用案例特定之客戶管理策略的基準。

如需有關 AWS 受管理策略的詳細資訊，請參閱IAM使用指南中的AWS 受管理策略。

若要使用AWSEC2VssSnapshotPolicy受管理的原則，您可以將其附加到 EC2 Windows 執行個體所附加的IAM角色。此政策可讓EC2VSS解決方案建立和新增標籤至 Amazon 機器映像 (AMIs) 和EBS快照。若要附加原則，請參閱將VSS快照管理政策附加到您的執行個體設定檔角色。

AWSEC2VssSnapshotPolicy 授予的許可

該AWSEC2VssSnapshotPolicy政策包括以下 Amazon EC2 許可，以允許 Amazon EC2 代表您創建和管理VSS快照。您可以將此受管理政策附加到IAM您用於 EC2 Windows 執行個體的執行個體設定檔角色。

- ec2: CreateTags — 將標籤新增至EBS快照，AMIs以協助識別和分類資源。
- ec2: DescribeInstanceAttribute — 擷取連接到目標執行個體的EBS磁碟區和對應的區塊裝置對映。
- ec2: CreateSnapshots — 建立EBS磁碟區的快照。
- ec2：CreateImage— AMI 從正在運行的EC2實例創建一個。
- ec2: DescribeImages — 擷取EC2AMIs和快照的資訊。
- ec2: DescribeSnapshots — 決定快照的建立時間和狀態，以驗證應用程式的一致性。

> **Note**
>
> 若要檢視此原則的權限詳細資料，請參閱AWS 受管理AWSEC2VssSnapshotPolicy的策略參考資料中的。

簡化特定使用案例的權限-進階

受`AWSEC2VssSnapshotPolicy`管理的原則包含您可以建立VSS快照的所有方式的權限。您可以建立僅包含所需權限的自訂原則。

使用案例：建立AMI，使用案例：使用 AWS Backup 服務

如果您只使用此選`CreateAmi`項，或者僅透過 AWS Backup 服務建立以快照為VSS基礎，則可以依照下列方式簡化原則陳述式。

- 省略下列陳述式 IDs (SIDs) 所識別的原則陳述式：
  - `CreateSnapshotsWithTag`
  - `CreateSnapshotsAccessInstance`
  - `CreateSnapshotsAccessVolume`
- 調整`CreateTagsOnResourceCreation`語句，如下所示：
  - `arn:aws:ec2:*:*:snapshot/*`從資源中移除。
  - `CreateSnapshots`從`ec2:CreateAction`條件中移除。
- 調整`CreateTagsAfterResourceCreation`陳述式以`arn:aws:ec2:*:*:snapshot/*`從資源中移除。
- 調整`DescribeImagesAndSnapshots`陳述式以`ec2:DescribeSnapshots`從陳述式動作中移除。

使用案例：僅限快照

如果您不使用該`CreateAmi`選項，則可以按照以下方式簡化政策聲明。

- 省略下列陳述式 IDs (SIDs) 所識別的原則陳述式：
  - `CreateImageAccessInstance`
  - `CreateImageWithTag`
- 調整`CreateTagsOnResourceCreation`語句，如下所示：
  - `arn:aws:ec2:*:*:image/*`從資源中移除。
  - `CreateImage`從`ec2:CreateAction`條件中移除。
- 調整`CreateTagsAfterResourceCreation`陳述式以`arn:aws:ec2:*:*:image/*`從資源中移除。
- 調整`DescribeImagesAndSnapshots`陳述式以`ec2:DescribeImages`從陳述式動作中移除。

> **Note**
>
> 為了確保您的自訂政策能如預期般執行，建議您定期檢閱受管理原則並加入更新。

將VSS快照管理政策附加到您的執行個體設定檔角色

若要授與 EC2 Windows 執行個VSS體快照的權限，您可以將AWSEC2VssSnapshotPolicy受管理原則附加至執行個體設定檔角色，如下所示。確保您的執行個體符合所有項目是非常重要的系統要求。

> **Note**
>
> 若要使用受管理政策，您的執行個體必須安裝`AwsVssComponents`套件版本`2.3.1`或更新版本。如需版本歷程記錄，請參閱AwsVssComponents 套件版本。

1. 在開啟IAM主控台https://console.aws.amazon.com/iam/。
2. 在導覽窗格中，選擇 [角色] 以查看您有權存取的IAM角色清單。
3. 選取連結至執行個體之角色的角色名稱連結。這會開啟角色詳細資料頁面。
4. 若要附加受管理的策略，請選擇位於清單面板右上角的 [新增權限]。然後從下拉式清單中選取 [附加原則]。
5. 若要簡化結果，請在搜尋列 (`AWSEC2VssSnapshotPolicy`) 中輸入策略名稱。
6. 選取要附加之原則名稱旁的核取方塊，然後選擇 [新增權限]。

管理以 Windows 為VSS基礎的EBS快照的VSS元件套件

建VSS立EBS快照之前，請確定 Windows 執行個體上已安裝最新版本的VSS元件套件。有幾種方法可以將`AwsVssComponents`套件安裝到現有的執行個體上，如下所示：

- (建議) 執行 AWSEC2-命VssInstallAndSnapshot 令文件 (建議使用)。這會在每次執行時視需要自動安裝或更新。
- 在 EC2 Windows 執行個體上手動安裝VSS元件.
- 更新 EC2 Windows 執行個體上的VSS元件套件.

您也可以使用`aws-vss-components-windows`受管理的元件來安裝EC2映像`AwsVssComponents`套件，建立AMI具有 Image Builder 的影像產生器。受管理元件會使用「 AWS Systems Manager 散發者」來安裝套件。在 Image Builder 建立映像檔之後，您從相關聯的每個執行個體啟動的執行個體都AMI會安裝VSS套件。如需如何在已安裝VSS套件的情況下建立套件AMI的詳細資訊，請參閱 EC2Image Builder 使用指南中的 Windows 適用散發者套件管理元件。

目錄

- 在 EC2 Windows 執行個體上手動安裝VSS元件
- 更新 EC2 Windows 執行個體上的VSS元件套件

在 EC2 Windows 執行個體上手動安裝VSS元件

您的 EC2 Windows 執行個體必須先安裝VSS元件，才能使用 Systems Manager 建立應用程式一致的快照集。如果您不執行 `AWSEC2-VssInstallAndSnapshot` 命令文件，以便在每次建立應用程式一致快照時自動安裝或更新套件，則必須手動安裝套件。

如果您打算使用下列其中一種方法從EC2執行個體建立應用程式一致的快照，也必須手動安裝。

- 使用建立VSS快照 AWS Backup
- 使用 Amazon Data Lifecycle Manager 建立VSS快

如果您需要執行手動安裝，建議您使用最新的 AWS VSS元件套件，以改善 EC2 Windows 執行個體上應用程式一致性快照的可靠性和效能。

> **Note**
>
> 若要在建立應用程式一致快照時自動安裝或更新 `AwsVssComponents` 套件，建議您使用 Systems Manager 來執行 `AWSEC2-VssInstallAndSnapshot` 文件。如需詳細資訊，請參閱執行 AWSEC2-命VssInstallAndSnapshot 令文件 (建議使用)。

若要在 Amazon EC2 Windows 執行個體上安裝VSS元件，請依照您偏好環境的步驟執行。

Console

使用SSM代理商安裝VSS元件

1. 在開啟 AWS Systems Manager 主控台https://console.aws.amazon.com/systems-manager/。
2. 在導覽窗格中，選擇 執行命令。
3. 選擇 執行命令。
4. 對於「命令」文件，請選擇「AWS-C」旁邊的按鈕onfigureAWSPackage。
5. 在 Command parameters (命令參數) 中，執行以下操作：
   a. 驗證 Action (動作) 設為 Install (安裝)。
   b. 對於名稱，輸入 `AwsVssComponents`。

c. 在版本中，輸入版本或將欄位留空，以便 Systems Manager 安裝最新版本。

6. 在 Targets (目標) 中，手動指定標籤或選取執行個體，以識別您要執行這項操作的執行個體。

> **Note**
>
> 如果您選取手動選取執行個體，但您預期看到的執行個體並未出現在清單上，請參閱 https://docs.aws.amazon.com/systems-manager/latest/userguide/troubleshooting-remote-commands.html#where-are-instances 使用者指南中的AWS Systems Manager 我的執行個體在哪裡？以取得故障診斷秘訣。

7. 對於 Other parameters (其他參數)：

   - (選用) 在 Comment (註解) 中，輸入此命令的相關資訊。
   - 在 Timeout (seconds) (逾時 (秒)) 中，指定在命令執行全面失敗之前，系統要等候的秒數。

8. (選用) 在 Rate control (速率控制) 中：

   - 在 Concurrency (並行) 中，指定可同時執行命令的執行個體數目或百分比。

   > **Note**
   >
   > 如果您透過選擇 Amazon 標EC2籤來選取目標，但不確定有多少執行個體使用選取的標籤，請透過指定百分比來限制可同時執行文件的執行個體數量。

   - 在 Error threshold (錯誤閾值) 中，指定在特定數目或百分比之執行個體上的命令失敗之後，停止在其他執行個體上執行命令。例如，如果您指定三個錯誤，則 Systems Manager 會在收到第四個錯誤時停止傳送命令。仍在處理命令的執行個體也可能會傳送錯誤。

9. (選用) 在 Output options (輸出選項) 區段中，如果您想要將命令輸出儲存至檔案，請選取 Enable writing to an S3 bucket (啟用寫入至 S3 儲存貯體) 旁的按鈕。指定儲存貯體和 (選用) 字首 (資料夾) 名稱。

   > **Note**
   >
   > 授予能力以將資料寫入至 S3 儲存貯體的 S3 許可，會是指派給執行個體之執行個體描述檔的許可，而不是執行此工作之使用者的許可。如需詳細資訊，請參閱AWS Systems Manager 使用指南中的「為 Systems Manager 建立IAM執行個體設定檔」。

10. (選擇性) 指定SNS通知的選項。

如需為執行命令設定 Amazon SNS 通知的相關資訊，請參閱為 AWS Systems Manager. SNS

11. 選擇執行。

AWS CLI

透過以下程序，即可使用 `AwsVssComponents` 中的執行命令來下載 AWS CLI套件，並在執行個體上進行安裝。該軟件包安裝兩個組件：VSS請求者和VSS提供程序。系統會將這些元件複製到執行個體上的目錄，然後將提供者註冊DLL為VSS提供者。

若要使用安VSS裝套件 AWS CLI

- 執行下列命令以下載並安裝 Systems Manager 所需的VSS元件。

```
aws ssm send-command \
--document-name "AWS-ConfigureAWSPackage" \
--instance-ids "i-01234567890abcdef" \
--parameters '{"action":["Install"],"name":["AwsVssComponents"]}'
```

PowerShell

使用下列程序，使用 Windows 工具中的執行命令，在執行個體上下載並安裝`AwsVssComponents`套件 PowerShell。該軟件包安裝兩個組件：VSS請求者和VSS提供程序。系統會將這些元件複製到執行個體上的目錄，然後將提供者註冊DLL為VSS提供者。

若要使用安裝VSS套件 AWS Tools for Windows PowerShell

- 執行下列命令以下載並安裝 Systems Manager 所需的VSS元件。

```
Send-SSMCommand -DocumentName AWS-ConfigureAWSPackage -InstanceId
 "i-01234567890abcdef" -Parameter
 @{'action'='Install';'name'='AwsVssComponents'}
```

驗證 AWS VSS元件上的簽章

請使用下列程序驗證 `AwsVssComponents` 套件上的簽章。

1. 連接至 Windows 執行個體。如需詳細資訊，請參閱使用 Connect 至您的 Windows 執行個體 RDP。

2. 瀏覽至 C:\Program 檔案\ Amazon\ AwsVssComponents。
3. 開啟 `ec2-vss-agent.exe` 的內容選單 (按一下右鍵)，然後選擇屬性。
4. 導覽至數位簽章索引標籤，並確認簽署者的名稱是否為 Amazon Web Services Inc.
5. 使用上述步驟驗證 `Ec2VssInstaller` 和 `Ec2VssProvider.dll` 上的簽章。

更新 EC2 Windows 執行個體上的VSS元件套件

我們建議您使用最新的建議版本保持VSS元件更新。在新版 `AwsVssComponents` 套件發行時，您可以透過多種不同方式更新元件。

更新方法

- 您可以重複發行 AWS VSS元件的新版本在 EC2 Windows 執行個體上手動安裝VSS元件時所述的步驟。
- 您可以設定「Systems Manager 狀態管理員」關聯，以便在`AwsVssComponents`套件可用時自動下載並安裝新的或更新的VSS元件。
- 您可以在使用 Systems Manager 執行 `AWSEC2-VssInstallAndSnapshot` 文件時，隨時在建立應用程式一致快照時自動安裝或更新 `AwsVssComponents` 套件。

> **Note**
>
> 建議您使用 Systems Manager 執行 `AwsVssComponents` 命令文件，其會在建立應用程式一致快照時，自動安裝或更新 `AWSEC2-VssInstallAndSnapshot` 套件。如需詳細資訊，請參閱執行 AWSEC2-命VssInstallAndSnapshot 令文件 (建議使用)。

若要建立 Systems Manager State Manager 關聯，請依照適用於您偏好之環境的步驟執行。

Console

使用主控台建立 State Manager 關聯

1. 在開啟 AWS Systems Manager 主控台https://console.aws.amazon.com/systems-manager/。
2. 在導覽窗格中，選擇 State Manager (狀態管理員)。

   或者，如果 Systems Manager 首頁先開啟，請開啟導覽窗格，然後選擇 State Manager。
3. 選擇 Create association (建立關聯)。

4. 在 Name (名稱) 欄位中，輸入描述性名稱。
5. 在「文件」清單中，選擇「AWS-C」onfigureAWSPackage。
6. 在 Parameters (參數) 區段中，從 Action (動作) 清單中選擇 Install (安裝)。
7. 對於 Installation type (安裝類型)，選擇 Uninstall and reinstall (解除安裝並重新安裝)。
8. 在 Name (名稱) 欄位中，輸入 `AwsVssComponents`。可將 Version (版本) 和 Additional Arguments (其他引數) 保留為空白。
9. 在 Targets (目標) 區段中，選擇一個選項。

> **Note**
>
> 如果您選擇使用標籤將執行個體設為目標，且指定映射到 Linux 執行個體的標籤，則關聯會在 Windows 執行個體上執行成功，但在 Linux 執行個體上執行失敗。關聯的整體狀態會顯示為 Failed (失敗)。

10. 在 Specify schedule (指定排程) 區段中，選擇一個選項。
11. 在 Advanced options (進階選項) 區段中，針對 Compliance severity (合規嚴重性)，選擇關聯的嚴重性等級。如需詳細資訊，請參閱關於 State Manager 關聯合規性。針對變更行事曆，請選取預先設定的變更行事曆。如需詳細資訊，請參閱 AWS Systems Manager 變更行事曆。
12. 針對比率控制，請執行下列操作：
    - 在 Concurrency (並行) 中，指定可同時執行命令的受管節點數目或百分比。
    - 在 Error threshold (錯誤閾值) 中，指定在特定數目或百分比之節點上的命令失敗之後，停止在其他受管節點上執行命令。
13. (選用) 針對輸出選項，若要將命令輸出儲存至檔案，請選取啟用將輸出寫入 S3。在方塊中輸入儲存貯體和字首 (資料夾) 名稱。
14. 選擇 Create association (建立關聯)，接著選擇 Close (關閉)。系統會嘗試在執行個體上建立關聯，並立即套用狀態。

> **Note**
>
> 如果 Windows Server 的EC2執行個體顯示「失敗」的狀態，請確認該執行個體上是否正在執行SSM代理程式，並確認執行個體已設定「Systems Manager」的 AWS Identity and Access Management (IAM) 角色。如需詳細資訊，請參閱設定 AWS Systems Manager。

AWS CLI

您可以執行建立關聯 AWS CLI 命令，按排程更新散發者套件，而不必讓關聯的應用程式離線。只會取代套件中的新檔案或更新檔案。

若要使用建立「狀態管理員」關聯 AWS CLI

1. 安裝和配置 AWS CLI, 如果你還沒有. 如需相關資訊，請參閱安裝或更新最新版本的 AWS CLI。
2. 執行下列命令以建立關聯。`--name` 值 (文件名稱) 一律為 `AWS-ConfigureAWSPackage`。以下命令會使用索引鍵 `InstanceIds` 來指定目標執行個體。

```
aws ssm create-association \
--name "AWS-ConfigureAWSPackage" \
--parameters '{"action":["Install"],"installationType":["Uninstall and
 reinstall"],"name":["AwsVssComponents"]}' \
--targets [{\"Key\":\"InstanceIds\",\"Values\":[\"i-01234567890abcdef\",
\"i-000011112222abcde\"]}]
```

若要取得有關可與指令配合使用的其他選項的資訊，請參閱《`create-association`指令參考》一 AWS Systems Manager 節中的〈建立關聯〉。 AWS CLI

為您的 EC2 Windows 執行個體建VSS立EBS快照

完成所有項目之後建立以 Windows 為VSS基礎之EBS快照的必要，您可以使用下列任何一種方法，從EC2執行個體建立VSS基於快照。

AWS Systems Manager 命令文件

使用 Systems Manager 命令文件以建立VSS基於快照。

若要自動備份，您可以建立使用`AWSEC2-VssInstallAndSnapshot`指令文件的 AWS Systems Manager 維護時段工作。如需詳細資訊，請參閱《AWS Systems Manager 使用者指南》中的 使用維護時段 Windows (主控台)。

AWS Backup

您可以在使用時 AWS Backup 透過在主控台或VSS中啟用來建立VSS備份CLI。如需詳細資訊，請參閱AWS Backup 開發人員指南中的建立 Windows VSS 備份。

> **Note**
>
> AWS Backup 不會自動在您的執行個體上安裝`AwsVssComponents`套件。您必須在執行個體上執行手動安裝。如需詳細資訊，請參閱在 EC2 Windows 執行個體上手動安裝VSS元件。

Amazon Data Lifecycle Manager

您可以在VSS快照生命週期政策中啟用指令碼前和後置指令碼，使用 Amazon Data 生命週期管理員建立快照。如需詳細資訊，請參閱 Amazon EBS 使用者指南中的自動化應用程式一致性快照。

> **Note**
>
> Amazon Data Lifecycle Manager 不會在您的執行個體上自動安裝 `AwsVssComponents` 套件。您必須在執行個體上執行手動安裝。如需詳細資訊，請參閱在 EC2 Windows 執行個體上手動安裝VSS元件。

使用 Systems Manager 指令文件建VSS立快照

您可以使用 AWS Systems Manager 指令文件建立VSS基於快照。下列內容會介紹可用的命令文件，以及文件用來建立快照的執行期參數。

在您使用任何 Systems Manager 命令文件之前，請確保您已符合所有 建立以 Windows 為VSS基礎之EBS快照的必要。

主題

- Systems Manager VSS 快照文件的參數
- 執行 Systems Manager VSS 快照命令文件

Systems Manager VSS 快照文件的參數

建立VSS快照的「Systems Manager」文件都使用下列參數，除非另有說明：

ExcludeBootVolume（字符串，可選）

若您建立快照，此設定會從備份程序中排除開機磁碟區。若要從快照中排除開機磁碟區，ExcludeBootVolume請`True`將開機磁碟區設定CreateAmi為和`False`。

如果您AMI為備份建立了一個，此參數應設定為`False`。此參數的預設值為 `False`。

NoWriters（字符串，可選）

若要從快照處理VSS作業中排除應用程式寫入器，請將此參數設定為`True`。排除應用程式VSS撰寫器可協助您解決與第三方VSS備份元件的衝突。此參數的預設值為 `False`。

CopyOnly（字符串，可選）

如果您除了使用原生SQL伺服器備份之外 AWS VSS，執行僅複製備份可防止 AWS VSS中斷原生差異備份鏈。若要執行僅限複製備份操作，請將此參數設定為 `True`。

此參數的預設值為`False`，這會導 AWS VSS致執行完整備份作業。

CreateAmi（字符串，可選）

若要建立VSS基於 Amazon 機器映像 (AMI) 以備份您的執行個體，請將此參數設定為`True`。這個參數的預設值是`False`，而是使用EBS快照來備份您的執行個體。

如需從執行個體建立AMI的詳細資訊，請參閱創建一個 Amazon 支EBS持 AMI。

AmiName（字符串，可選）

如果選CreateAmi項設定為`True`，請指定備份所AMI建立的名稱。

description (字串，選用)

指定此程序所建立之快照或映像的描述。

tags (字串，選用)

我們建議您為快照和映像加上標籤，以協助您尋找和管理資源，例如從快照清單還原磁碟區。系統會新增`Name`金鑰，其中有一個空白值，您可以在其中指定要套用至輸出快照或影像的名稱。

如果要指定其他標籤，請在兩者之間以分號分隔標籤。例如：`Key=Environment,Value=Test;Key=User,Value=TestUser1`。

依預設，系統會為VSS基礎的快照和映像新增下列保留標籤。

- 裝置 — 對VSS於以快照為基礎，這是快照擷取之EBS磁碟區的裝置名稱。
- AppConsistent— 此標籤表示成功建立VSS基於快照或AMI。
- AwsVssConfig— 這會識別快照AMIs，並在VSS啟用的情況下建立。該標籤包括元信息，例如`AwsVssComponents`版本。

> **Warning**
> 在參數清單中指定這些保留標籤中的任何一個都會導致錯誤。

executionTimeout（字符串，可選）

指定在執行處理上執行快照建立處理作業或AMI從執行處理建立快照集的時間上限 (以秒為單位)。增加此逾時時間可讓命令等待更長的時間，VSS以便開始凍結並完成其建立的資源標記。此逾時僅適用於快照或AMI建立步驟。安裝或更新 `AwsVssComponents` 套件的初始步驟不包含在逾時中。

CollectDiagnosticLogs（字符串，可選）

若要在快照和AMI建立步驟期間收集更多資訊，請將此參數設定為 `True` ""。此參數的預設值為 "`False`"。合併的診斷記錄會以`.zip`格式封存格式儲存在執行個體的下列位置：

```
C:\ProgramData\Amazon\AwsVss\Logs\timestamp.zip
```

VssVersion（字符串，可選）

在使用 `AWSEC2-VssInstallAndSnapshot` 文件的情況下，您可以指定 `VssVersion` 參數以在執行個體上安裝特定版本的 `AwsVssComponents` 套件。將此參數保留空白，以安裝建議的預設版本。

如果已安裝指定版本的 `AwsVssComponents` 套件，則指令碼會略過安裝步驟，並繼續進行備份步驟。如需 `AwsVssComponents` 套件版本和操作支援的清單，請參閱AWS VSS解決方案的版本。

執行 Systems Manager VSS 快照命令文件

您可以使用 AWS Systems Manager 命令文檔創建VSS基於EBS快照，如下所示。

執行 AWSEC2-命VssInstallAndSnapshot 令文件 (建議使用)

當您使用執 AWS Systems Manager 行`AWSEC2-VssInstallAndSnapshot`文件時，指令碼會執行下列步驟。

1. 該指令碼會先在執行個體上安裝或更新 `AwsVssComponents` 套件，具體取決於是否已安裝執行個體。
2. 指令碼會在完成第一個步驟後，建立應用程式一致快照。

若要執行 `AWSEC2-VssInstallAndSnapshot` 文件，請依照適用於您偏好之環境的步驟執行。

Console

從主控台建立VSSEBS快照

1. 在開啟 AWS Systems Manager 主控台https://console.aws.amazon.com/systems-manager/。
2. 從導覽窗格中選取執行命令。這會顯示您帳戶中目前正在執行的命令清單 (若適用)。
3. 選擇 執行命令。這會開啟您具有存取權限的命令文件清單。
4. 從命令文件清單中選取 `AWSEC2-VssInstallAndSnapshot`。若要簡化結果，可輸入全部或部分文件名稱。您也可以依擁有者、平台類型或標籤進行篩選。

   在您選取命令文件時，系統會將詳細資訊填入清單下方。
5. 從文件版本清單中選取 `Default version at runtime`。
6. 設定 Command 參數`AWSEC2-VssInstallAndSnapshot`，以定義如何使用快VSS照或AMI. `AwsVssComponents` 如需參數詳細資訊，請參閱 Systems Manager VSS 快照文件的參數。
7. 針對目標選取，手動指定標籤或選取執行個體，以識別要執行這項操作的執行個體。

   > **Note**
   > 如果您手動選取執行個體，但預期看到的執行個體並未出現在清單中，請參閱我的執行個體在哪裡？，以取得疑難排解秘訣。
8. 如需定義 Systems Manager Run Command 行為的其他參數 (例如速率控制)，請依照從主控台執行命令中的說明輸入值。
9. 選擇執行。

   如果成功，指令會將新快照填入EBS快照清單中。您可以搜尋指定的標籤或搜尋，在EBS快照清單中尋找這些快照`AppConsistent`。如果命令執行失敗，請檢視 Systems Manager 命令輸出的詳細資訊，以了解執行失敗的原因。如果命令成功完成，但特定的磁碟區備份失敗，您可以疑難排解EBS磁碟區清單中的失敗問題。

AWS CLI

您可以在中執行下列指令 AWS CLI 來建VSS立EBS快照並取得快照建立的狀態。

建VSS立EBS快照

執行下列命令以建立VSS基礎EBS快照。若要建立快照，您必須使用 `--instance-ids` 參數來識別執行個體。如需您可以查看之其他參數的詳細資訊，請參閱 Systems Manager VSS 快照文件的參數。

```
aws ssm send-command \
 --document-name "AWSEC2-VssInstallAndSnapshot" \
 --instance-ids "i-01234567890abcdef" \
 --parameters '{"ExcludeBootVolume":["False"],"description":["Description"],"tags":
["Key=key_name,Value=tag_value"],"VssVersion":[""]}'
```

如果成功，命令文件會以新EBS快照填入快照清單。您可以搜尋指定的標籤或搜尋，在EBS快照清單中尋找這些快照`AppConsistent`。如果命令執行失敗，請檢視命令輸出的詳細資訊，以了解執行失敗的原因。

取得命令狀態

若要取得快照目前的狀態，請使用從 send-command 傳回的命令 ID 執行下列命令。

```
aws ssm get-command-invocation
 --instance-ids "i-01234567890abcdef" \
 --command-id "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111" \
 --plugin-name "CreateVssSnapshot"
```

PowerShell

運行以下命令 AWS Tools for Windows PowerShell 以創建VSS基於EBS快照，並獲取創建輸出的當前運行時狀態。指定前面清單中描述的參數，以修改快照處理程序的行為。

使用適用VSS於 Windows 的工具建立EBS快照 PowerShell

執行下列命令以建VSS立EBS快照或AMIs。

```
Send-SSMCommand -DocumentName "AWSEC2-VssInstallAndSnapshot" -InstanceId
 "i-01234567890abcdef" -Parameter
 @{'ExcludeBootVolume'='False';'description'='a_description'
 ;'tags'='Key=key_name,Value=tag_value';'VssVersion'=''}
```

取得命令狀態

若要取得快照目前的狀態，請使用從 Send-SSMCommand 傳回的命令 ID 執行下列命令。

```
Get-SSMCommandInvocationDetail -InstanceId "i-01234567890abcdef" -CommandId
 "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111" -PluginName "CreateVssSnapshot"
```

如果成功，指令會將新快照填入EBS快照清單中。您可以搜尋指定的標籤或搜尋，在EBS快照清單中尋找這些快照`AppConsistent`。如果命令執行失敗，請檢視命令輸出的詳細資訊，以了解執行失敗的原因。

運行 AWSEC 2-CreateVssSnapshot 命令文檔

若要執行 `AWSEC2-CreateVssSnapshot` 文件，請依照適用於您偏好之環境的步驟執行。

Console

從主控台建立VSSEBS快照

1. 在開啟 AWS Systems Manager 主控台https://console.aws.amazon.com/systems-manager/。
2. 從導覽窗格中選取執行命令。這會顯示您帳戶中目前正在執行的命令清單 (若適用)。
3. 選擇 執行命令。這會開啟您具有存取權限的命令文件清單。
4. 從命令文件清單中選取 `AWSEC2-CreateVssSnapshot`。若要簡化結果，可輸入全部或部分文件名稱。您也可以依擁有者、平台類型或標籤進行篩選。

   在您選取命令文件時，系統會將詳細資訊填入清單下方。
5. 從文件版本清單中選取 `Default version at runtime`。
6. 設定 Command 參數，以定義如何`AWSEC2-CreateVssSnapshot`使用VSS快照或AMI. 如需參數詳細資訊，請參閱 Systems Manager VSS 快照文件的參數。
7. 針對目標選取，手動指定標籤或選取執行個體，以識別要執行這項操作的執行個體。

   > **Note**
   > 如果您手動選取執行個體，但預期看到的執行個體並未出現在清單中，請參閱我的執行個體在哪裡？，以取得疑難排解秘訣。
8. 如需定義 Systems Manager Run Command 行為的其他參數 (例如速率控制)，請依照從主控台執行命令中的說明輸入值。
9. 選擇執行。

如果成功，指令會將新快照填入EBS快照清單中。您可以搜尋指定的標籤或搜尋，在EBS快照清單中尋找這些快照`AppConsistent`。如果命令執行失敗，請檢視 Systems Manager 命令輸出的詳細資訊，以了解執行失敗的原因。如果命令成功完成，但特定的磁碟區備份失敗，您可以疑難排解EBS磁碟區清單中的失敗問題。

AWS CLI

您可以在中執行下列指令 AWS CLI 來建立以建立VSS基礎的EBS快照。

建VSS立EBS快照

執行下列命令以建立VSS基礎EBS快照。若要建立快照，您必須使用 `--instance-ids` 參數來識別執行個體。如需您可以查看之其他參數的詳細資訊，請參閱 Systems Manager VSS 快照文件的參數。

```
aws ssm send-command \
 --document-name "AWSEC2-CreateVssSnapshot" \
 --instance-ids "i-01234567890abcdef" \
 --parameters '{"ExcludeBootVolume":["False"],"description":["Description"],"tags":
["Key=key_name,Value=tag_value"]}'
```

如果成功，命令文件會以新EBS快照填入快照清單。您可以搜尋指定的標籤或搜尋，在EBS快照清單中尋找這些快照`AppConsistent`。如果命令執行失敗，請檢視命令輸出的詳細資訊，以了解執行失敗的原因。

PowerShell

使用執行下列命令 AWS Tools for Windows PowerShell 以建立VSS基於EBS快照。

使用適用VSS於 Windows 的工具建立EBS快照 PowerShell

執行下列命令以建立VSS基礎EBS快照。若要建立快照，您必須使用 `InstanceId` 參數來識別執行個體。您可以指定多個執行個體，藉此為其建立快照。如需您可以查看之其他參數的詳細資訊，請參閱 Systems Manager VSS 快照文件的參數。

```
Send-SSMCommand -DocumentName AWSEC2-CreateVssSnapshot -InstanceId
 "i-01234567890abcdef" -Parameter
 @{'ExcludeBootVolume'='False';'description'='a_description'
 ;'tags'='Key=key_name,Value=tag_value'}
```

如果成功，指令會將新快照填入EBS快照清單中。您可以搜尋指定的標籤或搜尋，在EBS快照清單中尋找這些快照`AppConsistent`。如果命令執行失敗，請檢視命令輸出的詳細資訊，以了解執行失敗的原因。如果命令成功完成，但特定磁碟區備份失敗，您可以疑難排解EBS快照清單中的失敗問題。

針對具有共用EBS儲存區的 Windows 容錯移轉叢集執行命令文件

您可以使用上一節所述的任何命令行程序來建立VSS基於快照。命令文件 (`AWSEC2-VssInstallAndSnapshot` 或 `AWSEC2-CreateVssSnapshot`) 必須在叢集中的主節點上執行。次要節點上的文件將會失敗，因為其無法存取共用磁碟。如果您的主要和次要動態變更，您可以在多個節點上 AWS Systems Manager 執行 Run Command 文件，並期望指令在主要節點上成功並在次要節點上失敗。

> **Note**
>
> 若要自動備份，您可以建立使用`AWSEC2-VssInstallAndSnapshot`文件的 AWS Systems Manager 維護時段工作。如需詳細資訊，請參閱《AWS Systems Manager 使用者指南》中的使用維護時段 Windows (主控台)。

疑難排解 Windows VSS EBS 快照

在嘗試任何其他疑難排解步驟之前，建議您先確認下列詳細資料。

- 確保您符合所有 建立以 Windows 為VSS基礎之EBS快照的必要。
- 確認您使用您作業系統適用的 `AwsVssComponents` 套件最新 支援 Windows OS 版本。您觀察到的問題可能已在較新版本中得到解決。

主題

- 檢查記錄檔
- 收集其他診斷記錄
- 在已設定 Proxy 的執行個體VSS上使用
- 錯誤:解凍管連接逾時、解凍時發生錯誤、等待VSS凍結逾時或其他逾時錯誤
- 錯誤：無法叫用方法。僅在此語言模式下的核心類型上支援方法調用

檢查記錄檔

如果您在建立VSSEBS快照時遇到問題或收到錯誤訊息，您可以在 Systems Manager 主控台中檢視命令輸出。

對於建立VSS快照的 Systems Manager 文件，您可以在執行時間將`CollectDiagnosticLogs`參數設定為 `True` ""。當`CollectDiagnosticLogs`參數設定為 "`True`「時，VSS會收集額外的記錄檔以協助偵錯。如需詳細資訊，請參閱收集其他診斷記錄。

如果您收集診斷記錄檔，Systems Manager 文件會將它們儲存在您的執行個體中，位於下列位置：`C:\ProgramData\Amazon\AwsVss\Logs\`*`timestamp`*`.zip``CollectDiagnosticLogs`參數的預設值為 "`False`"。

> **Note**
>
> 如需其他偵錯說明，您可以將`.zip`檔案傳送至 AWS Support。

無論您是否收集診斷記錄，都可以使用下列其他記錄檔：

- `%ProgramData%\Amazon\SSM\InstanceData\`*`InstanceID`*`\document\orchestration\`*`SSMCommandID`*`\awsrunPowerShellScript\runPowerShellScript\stdout`
- `%ProgramData%\Amazon\SSM\InstanceData\`*`InstanceID`*`\document\orchestration\`*`SSMCommandID`*`\awsrunPowerShellScript\runPowerShellScript\stderr`

您也可以開啟事件檢視器 Windows 應用程式，然後選擇 Windows 日誌 、應用程式來檢視其他日誌。若要查看特別來自 EC2 Windows VSS 提供者和磁碟區陰影複製服務的事件，請根據條款**`Ec2VssSoftwareProvider`**和來源篩選**`VSS`**。

如果您將 Systems Manager 與VPC端點搭配使用，且 Systems Manager SendCommandAPI動作 (在主控台中執行命令) 失敗，請確認您已正確設定下列端點：com.amazonaws。***`region`***.ec2。

如果未定義 Amazon EC2 端點，則列舉連接EBS磁碟區的呼叫會失敗，進而導致 Systems Manager 命令失敗。如需有關使用 Systems Manager 設定VPC端點的詳細資訊，請參閱AWS Systems Manager 使用手冊中的建立虛擬私有雲端端點。

收集其他診斷記錄

若要在使用 Systems Manager send 命令執行VSS快照文件時收集其他診斷記錄，請在執行階段將`CollectDiagnosticLogs`輸入參數設定為 `True` ""。建議您在疑難排解時將此參數設定為 `True` ""。

若要查看指令行範例，請選取下列其中一個標籤。

AWS CLI

下列範例會在中執行 `AWSEC2-CreateVssSnapshot` Systems Manager 文件 AWS CLI：

```
aws ssm send-command \
--document-name "AWSEC2-CreateVssSnapshot" \
--instance-ids "i-1234567890abcdef0" \
--parameters '{"description":["Example - create diagnostic logs at
 runtime."],"tags":["Key=tag_name,Value=tag_value"],"CollectDiagnosticLogs":
["True"]}'
```

PowerShell

下列範例會在中執行 `AWSEC2-CreateVssSnapshot` Systems Manager 文件 PowerShell：

```
Send-SSMCommand -DocumentName AWSEC2-CreateVssSnapshot -InstanceId
 "i-1234567890abcdef0" -Parameter @{'description'='Example - create diagnostic logs
 at runtime.';'tags'='Key=tag_name,Value=tag_value';'CollectDiagnosticLogs'='True'}
```

在已設定 Proxy 的執行個體VSS上使用

如果您在使用 Proxy 連線EC2端點的執行個體上建立VSSEBS快照時遇到問題，請確認執行個體上的下列設定：

- 驗證 Proxy 是否已設定為執行個體區域中的EC2服務端點，並IMDS可透過執行身分 AWS Tools for Windows PowerShell 執行來連線SYSTEM。
- 若要支援使用系統設定的 Win HTTP Proxy，請確定您已在執行個體上安裝最新`AwsVssComponents`版本。如需有關設定 Win HTTP Proxy 的詳細資訊，請參閱 Microsoft 網站上適用於 Windows 超文字傳輸通訊協定的 Netsh 命令 (WINHTTP)。

錯誤:解凍管連接逾時、解凍時發生錯誤、等待VSS凍結逾時或其他逾時錯誤

EC2Windows 提VSS供者可能會因為執行個體上的活動或服務而逾時，防止以VSS基礎的快照集及時進行。Windows VSS 架構提供了一個不可設定的 10 秒視窗，在此期間暫停與檔案系統的通訊。在這段期間，`AWSEC2-CreateVssSnapshot` 會對磁碟區執行快照。

下列問題可能會導致 EC2 Windows VSS 提供者在快照期間遇到時間限制：

- 磁碟區的過多 I/O
- 實例EC2API上的響應速度慢
- 分段的磁碟區
- 與某些防毒軟體不相容
- VSS應用程式撰寫程式的問題
- 為大量模組啟用「模 PowerShell 組記錄」時，可能會導致 PowerShell 指令碼執行速度變慢

執行 `AWSEC2-CreateVssSnapshot` 命令文件時遇到的大部分逾時問題是與備份時執行個體的工作負載過高相關。下列動作可協助您成功建立快照：

- 重試 `AWSEC2-CreateVssSnapshot` 命令來查看快照是否成功。若在某些情況下重試成功，減少執行個體負載可能使快照更加成功。
- 等候一段時間，待執行個體上的工作負載減少，再重試 `AWSEC2-CreateVssSnapshot` 命令。或者，您可以在執行個體處於低壓力時嘗試快照。
- 當系統上的防毒軟體關閉時，請嘗試VSS快照。如果這樣可以解決問題，請參閱防毒軟體指示，並將其設定為允許VSS快照。
- 如果您的帳戶在執行快照的相同區域內有大量 Amazon EC2 API 呼叫，API節流可能會延遲快照作業。若要減少節流影響，請使用最新`AwsVssComponents`的套件。此套件會利用此`EC2CreateSnapshots`API動作來減少變更動作的數量，例如每個磁碟區快照建立和標記。
- 如果您同時執行多個 `AWSEC2-CreateVssSnapshot` 命令指令碼，您可以採取下列步驟來減少並行問題。
  - 請考慮在較低API活動期間排程快照。
  - 如果您Run Command在 Systems Manager 主控台中使用 (或SendCommand在API) 中執行命令指令碼，您可以使用 Systems Manager 速率控制來減少並行性。

    您也可以使用 Systems Manager 速率控制來減少使用 Systems Manager 執行命令指令碼之類 AWS Backup 服務的並行性。

- 在 Shell 中執行命令 `vssadmin list writers`，並查看是否有回報系統上任何寫入器的上次錯誤欄位中任何錯誤。如有任何寫入器回報逾時錯誤，請考慮在執行個體負載較少時重試快照。
- 當您使用較小的實例類型時 *t2 | t3 | t3a*. 納米或 *t2 | t3 | t3a*.micro，可能會發生由於內存和CPU約束而導致的超時。下列動作可能有助於減少逾時問題。
  - 在拍攝快照之前，請先關閉記憶體或CPU密集型應用
  - 嘗試在執行個體活動較低期間建立快照。

錯誤：無法叫用方法。僅在此語言模式下的核心類型上支援方法調用

PowerShell 語言模式未設定為時，您將會遇到此錯誤`FullLanguage`。文`AWSEC2-CreateVssSnapshotSSM`件需 PowerShell 要設定為`FullLanguage`模式。

若要驗證語言模式，請在PowerShell 主控台的執行個體上執行下列命令：

```
$ExecutionContext.SessionState.LanguageMode
```

如需語言模式的詳細資訊，請參閱 Microsoft 文件中的 about_Language_Modes。

從基礎快照還原 Windows 執行個體VSS的EBS磁碟區

您可以使用指`RestoreVssSnapshotSampleScript.ps1`令碼從EBS快照還原執行個體上VSS的磁碟區。此指令碼會執行下列任務：

- 停止執行個體
- 移除執行個體上的所有現有磁碟機 (若已排除開機磁碟區，則其將不會涵蓋在內)
- 從快照建立新的磁碟區
- 使用快照上的裝置 ID 標記來將磁碟區連接至執行個體
- 重新啟動執行個體

> **Important**
>
> 下列指令碼會分離所有連接至執行個體的磁碟區，然後從快照建立新的磁碟區。請確認您已正確地備份執行個體。系統不會刪除舊的磁碟區。如果您想刪除舊磁碟區，則可編輯指令碼。

若要從VSSEBS快照還原磁碟區

1. 下載 RestoreVssSnapshotSampleScript.zip 檔案並解壓縮檔案內容。
2. `RestoreVssSnapshotSampleScript.ps1`在文字編輯器中開啟，並使用有效的執行個EC2體 ID 和EBS快照 ID 編輯指令碼底部的範例呼叫，然後從中執行指令碼 PowerShell。

AWS VSS解決方案的版本

此頁面包含 AWS VSS元件套件的版本說明，以及每個支援 Windows Server 版本的元件和指令碼版本需求。

主題

- AwsVssComponents 套件版本
- 支援 Windows OS 版本

AwsVssComponents 套件版本

下表說明 AWS VSS元件套件的已發行版本。

版本	詳細資訊	發行日期
2.3.3	已更新VSS代理程式，以確保在`Ec2VssProvider` 建立快照期間使用。	2024年6月25日
2.3.2	修正解除安裝時未移除VSS提供者註冊的情況。	2024年5月9日
2.3.1	添加了新的默認標籤`AwsVssConfig` 來識別快照和AMIs創建者 AWS VSS。	2024年3月7日
2.2.1	• 增加了對使用 `DescribeInstanceAttribute` API. • 錯誤修正與可靠性改進。 •	2024年1月18日

版本	詳細資訊	發行日期
	對視窗服務器 2012 年和 2012 R2 不推薦使用的支持。 AWS VSS視窗伺服器 2012 和 2012 R2 上的元件版本 2.2.1 安裝將會失敗。 AWS VSS組件版本 2.1.0 是最後一個版本，以支持視窗服務器 2012 年和 2012 R2。	
2.1.0	增加了對使用 `CreateSnapshots` API.	2023 年 11 月 6 日
2.0.1	增加了對使用 Win HTTP 代理設置的支持。	2023 年 10 月 26 日
2.0.0	已新增 AWS VSS元件建立快照的功能AMIs，並可與 PowerShell 模組記錄、指令碼區塊記錄和轉錄功能相容。	2023 年 4 月 28 日
1.3.2.0	修復了未正確報告安裝失敗的情況。	2022 年 5 月 10 日
1.3.1.0	• 修正與NTDSVSS寫入器記錄錯誤相關的網域控制站上快照失敗的問題。 • 卸載 1.0 版VSS提供VSS程序時修復了代理錯誤。	2020 年 2 月 6 日
1.3.00	• 降低不需要的詳細資訊以改善記錄。 • 修正安裝期間的區域化問題。 • 修正某些提供者註冊錯誤狀況的傳回代碼。 • 修正各種安裝問題。	2019 年 3 月 19 日

版本	詳細資訊	發行日期
1.2.00	• 為代理程式新增命令列參數 `-nw` (無寫入器) 和 `-copy` (僅限複製)。 • 修 EventLog 復了內存分配調用不當導致的錯誤。	2018 年 11 月 15 日
1.1	修正錯誤作為預設 Windows Backup 與還原提供者使用的 AWS VSS元件。	2017 年 12 月 12 日
1.0	初始版本。	2017 年 11 月 20 日

支援 Windows OS 版本

下表顯示您應該在 Amazon 上每個版本的 Windows 伺服器上執行哪些 AWS VSS解決方案版本EC2。

Windows Server 版本	AwsVssComponents 版本	AWSEC2-VssInstallAndSnapshot 版本名稱	AWSEC2-CreateVssSnapshot 版本名稱
Windows Server 2022	預設	預設	預設
Windows Server 2019	預設	預設	預設
Windows Server 2016	預設	預設	預設

Windows Server 版本	AwsVssComponents 版本	AWSEC2-VssInstallAndSnapshot 版本名稱	AWSEC2-CreateVssSnapshot 版本名稱
Windows Server 2012 R2	2.1.0	不支援	2012R2
Windows Server 2012	2.1.0	不支援	2012R2
Windows Server 2008 R2	1.3.1.0	不支援	2008R2

Amazon 上的物件儲存、檔案儲存和檔案快取 EC2

雲端檔案儲存是一種在雲端存放資料的方式，提供伺服器和應用程式透過共享檔案系統存取資料。這種相容性使得雲端檔案儲存非常適合用於倚賴共享檔案系統的工作負載，而且提供簡易的整合，無須程式碼變更。

目前存在許多檔案儲存解決方案，從運算執行個體上的單一節點檔案伺服器使用區塊儲存做為沒有延展性的基礎，或是保護資料的冗餘性很少，到 do-it-yourself 叢集解決方案，到完全管理的解決方案。以下內容介紹一些由 AWS Amazon EC2 執行個體提供的儲存服務。

目錄

- 使用 Amazon S3 與 Amazon EC2 實例
- 使用 Amazon EFS 與 Amazon EC2 Linux 實例
- 使用 Amazon FSx 與 Amazon EC2 實例
- 搭配 Amazon EC2 執行個體使用 Amazon 檔案快取

使用 Amazon S3 與 Amazon EC2 實例

Amazon Simple Storage Service (Amazon S3) 是一項物件儲存服務，提供領先業界的可擴展性、資料可用性、安全性和效能。您可以使用 Amazon S3 存放和擷取各種使用案例的任意數量資料，例如資料湖、網站、備份和大數據分析、從 Amazon EC2 執行個體或網際網路上的任何地方。如需詳細資訊，請參閱什麼是 Amazon S3？

物件是存放在 Amazon S3 中的基本實體。Amazon S3 中存放的每個物件都包含在儲存貯體中。儲存貯體在最高層級組織 Amazon S3 命名空間，並識別負責該儲存體的帳戶。Amazon S3 儲存貯體與網際網路網域名稱類似。儲存在值區中的物件具有唯一索引鍵值，可使用URL. 例如，如果具有索引鍵值`/photos/mygarden.jpg`的物件儲存在`amzn-s3-demo-bucket1`值區中，則可使用. URL `https://amzn-s3-demo-bucket1.s3.amazonaws.com/photos/mygarden.jpg` 如需詳細資訊，請參閱 Amazon S3 的運作方式。

使用範例

鑑於 Amazon S3 對儲存的好處，您可以決定使用此服務存放檔案和資料集，以便與EC2執行個體搭配使用。有幾種方法可在 Amazon S3 與執行個體之間移動資料。除了底下討論的範例之外，您也能從電腦或執行個體使用其他人所編寫的多種工具來存取 Amazon S3 中的資料。其中一些常見工具的討論請參閱 AWS 論壇。

如果您有許可，您可用下列其中一種方式在 Amazon S3 和執行個體之間複製檔案。

GET or wget (Linux)

> **Note**
>
> 此方法僅適用於公有物件。如果物件非公有，您會收到 `ERROR 403: Forbidden` 訊息。如果收到此錯誤，則必須使用 Amazon S3 主控台 AWS CLI、、 AWS API AWS SDK、或 AWS Tools for Windows PowerShell，且必須擁有必要的許可。如需詳細資訊，請參閱 Amazon S3 使用者指南中的 Amazon S3 中的身分和存取管理及下載物件。

該wget實用程序是一個HTTP和FTP客戶端，可讓您從 Amazon S3 下載公共對象。其預設安裝在 Amazon Linux 和多數其他的發行版本上，也可供 Windows 下載。若要下載 Amazon S3 物件，請使用下列命令，取代要下載URL的物件。

```
[ec2-user ~]$ wget https://my_bucket.s3.amazonaws.com/path-to-file
```

AWS Tools for Windows PowerShell (Windows)

Windows 執行個體的優點為使用圖形瀏覽器，可讓您用來直接存取 Amazon S3 主控台；不過，在編寫指令碼時，Windows 使用者也可使用 AWS Tools for Windows PowerShell 在 Amazon S3 之間移動物件。

請用下列命令將 Amazon S3 物件複製至 Windows 執行個體。

```
PS C:\> Copy-S3Object -BucketName my_bucket -Key path-to-file -
LocalFile my_copied_file.ext
```

AWS CLI (Linux and Windows)

AWS Command Line Interface (AWS CLI) 是用來管理您的 AWS 服務的統一工具。 AWS CLI 可讓使用者進行自我身分驗證，以及從 Amazon S3 下載受限制的項目，還有上傳項目。如需詳細資訊，包括如何安裝與設定工具，請參閱 AWS Command Line Interface 詳細資訊頁面。

aws s3 cp 命令結果類似以下命令：Unix cp。可讓您從 Amazon S3 複製檔案到執行個體，從執行個體複製檔案到 Amazon S3，以及從某一 Amazon S3 位置複製檔案到其他位置。

請用下列命令從 Amazon S3 將物件複製至執行個體。

```
aws s3 cp s3://my_bucket/my_folder/my_file.ext my_copied_file.ext
```

請用下列命令從執行個體將物件複製回 Amazon S3。

```
aws s3 cp my_copied_file.ext s3://my_bucket/my_folder/my_file.ext
```

aws s3 sync 命令可將整個 Amazon S3 儲存貯體同步至本機目錄位置。這對於下載資料集以及將本機副本與遠端設定一 up-to-date 起保留很有幫助。如果您擁有 Amazon S3 儲存貯體的適當許可，還能將命令中的來源與目的地位置對調，於完成後將本機目錄備份推送至雲端。

請用下列命令將整個 Amazon S3 儲存貯體下載至執行個體上的本機目錄。

```
aws s3 sync s3://remote_S3_bucket local_directory
```

Amazon S3 API

如果您是開發人員，可以使用存API取 Amazon S3 中的資料。您可以使用此功能API來協助開發應用程式，並將其與其他APIs和整合SDKs。如需詳細資訊，請參閱 Amazon S3 使用者指南 AWS SDKs中使用的 Amazon S3 程式碼範例。

使用 Amazon EFS 與 Amazon EC2 Linux 實例

> **Note**
>
> 視窗執行個體不支援 Amazon EFS。

Amazon EFS 提供可擴展的文件存儲與 Amazon 使用EC2。您可以使用EFS檔案系統做為在多個執行個體上執行的工作負載和應用程式的通用資料來源。如需詳細資訊，請參閱 Amazon Elastic File System 產品頁面。

本教學說明如何在執行個體啟動期間使用 Amazon EFS 快速建立精靈建立和連接 Amazon EFS 檔案系統。如需如何使用 Amazon EFS 主控台建立檔案系統的教學課程，請參閱 Amazon Elastic File System 使用者指南中的 Amazon Elastic File System 入門。

> **Note**
>
> 當您使用「快速建立EFS」(EFSQuick Create) 建立檔案系統時，會使用下列服務建議的設定來建立檔案系統：
>
> - 啟用自動備份。
> - 在選取的每個預設子網路中掛載目標VPC。
> - 一般用途效能模式。
> - 大量批量輸送量模式。
> - 使用 Amazon EFS (`aws/elasticfilesystem`) 的預設金鑰啟用靜態資料加密。
> - 使用 30 天政策啟用 Amazon EFS 生命週期管理。

任務

- 使用 Amazon EFS 快速建立EFS檔案系統
- 測試檔EFS案系統
- 刪除檔EFS案系統

使用 Amazon EFS 快速建立EFS檔案系統

當您使用 Amazon 啟動執行個體精靈的 Amazon EFS 快速建立功能啟動執行個體時，您可以建立EFS檔案系統並將其掛接到執行個體。EC2

若要使用 Amazon EFS 快速建立EFS檔案系統

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 選擇啟動執行個體。
3. (選用) 在 Name and tags (名稱和標籤) 下，針對 Name (名稱)，輸入用於識別執行個體的名稱。
4. 在應用程式和作業系統映像 (Amazon 機器映像) 下，選擇 Linux 作業系統，然後針對 Amazon 機器映像 (AMI) 選取 Linux AMI。
5. 在 Instance type (執行個體類型) 下，針對 Instance type (執行個體類型)，選取執行個體類型或保留預設值。
6. 在 Key pair (login) (金鑰對 (登入)) 下，針對 Key pair name (金鑰對名稱)，選擇現有的金鑰對或建立新的金鑰對。
7. 在 Network settings (網路設定) 下，選取 Edit (編輯) (在右側)，然後針對 Subnet (子網)，選取一個子網。

   > **Note**
   > 您必須先選取子網路，才能新增EFS檔案系統。

8. 在 Configure storage (設定儲存) 下，選擇 Edit (編輯) (在右下角)，然後執行下列動作：
   a. 若為 [檔案系統]，請確定EFS已選取，然後選擇 [建立新的共用檔案系統]。
   b. 針對檔案系統名稱，輸入 Amazon EFS 檔案系統的名稱，然後選擇 [建立檔案系統]。
   c. 對於「掛載點」，請指定自訂掛載點或保留預設掛載點。
   d. 若要啟用檔案系統的存取權，請選取 Automatically create and attach security groups (自動建立和連接安全群組)。選取此核取方塊後，會自動建立下列安全群組，並將其附加至檔案系統的執行個體和掛載目標：
      - 執行個體安全性群組 — 包含允許 NFS 2049 連接埠流量但不包含輸入規則的輸出規則。
      - 檔案系統掛載目標安全性群組 — 包括允許來自執行個體安全性群組 (如上所述) 透過 NFS 2049 連接埠流量的輸入規則，以及允許 NFS 2049 連接埠流量的輸出規則。

> **Note**
>
> 或者，您也可以手動建立並連結安全性群組。如果您要手動建立並連接安全群組，請清除 Automatically create and attach the required security groups (自動建立並連接必要的安全群組)。

e. 若要在執行個體啟動時自動掛載共用檔案系統，請選取 Automatically mount shared file system by attaching required user data script (透過連接所需的使用者指令碼自動掛載共用檔案系統)。若要檢視自動產生的使用者資料，請展開 Advanced details (進階詳細資訊)，然後向下捲動至 User data (使用者資料)。

> **Note**
>
> 如果您在選取此核取方塊之前加入了使用者資料，原始使用者資料將被自動產生的使用者資料覆寫。

9. 視需要進行任何其他執行個體組態設定。
10. 在 Summary (摘要) 面板中，檢閱您的執行個體組態，然後選擇 Launch instance (啟動執行個體)。如需詳細資訊，請參閱 使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體。

測試檔EFS案系統

您可以連線至執行個體，並確認將檔案系統掛載至您指定的目錄 (例如 /mnt/efs)。

確認檔案系統已掛載

1. 連線到您的執行個體。如需詳細資訊，請參閱 使用 Connect 到您的 Linux 執行個體 SSH。
2. 在執行個體的終端機視窗中，執行df -T命令以確認EFS檔案系統是否已掛載。

```
$ df -T
Filesystem     Type              1K-blocks    Used          Available Use% Mounted
 on
/dev/xvda1     ext4                8123812 1949800            6073764  25% /
devtmpfs       devtmpfs            4078468      56            4078412   1% /dev
tmpfs          tmpfs               4089312       0            4089312   0% /dev/shm
efs-dns        nfs4       9007199254740992       0   9007199254740992   0% /mnt/efs
```

請注意，文件系統的名稱，在示例輸出中顯示為 *efs-dns*，具有以下形式。

```
file-system-id.efs.aws-region.amazonaws.com:/
```

3. (選用) 在檔案系統中從執行個體建立檔案，然後確認您可以從另一個執行個體檢視檔案。

   a. 從執行個體中，執行下列命令來建立檔案。

   ```
   $ sudo touch /mnt/efs/test-file.txt
   ```

   b. 從另一個執行個體中，執行下列命令來檢視檔案。

   ```
   $ ls /mnt/efs
   test-file.txt
   ```

刪除檔EFS案系統

若您不再需要檔案系統，可以刪除它。

刪除檔案系統

1. 開啟 Amazon Elastic File System 主控台，位於https://console.aws.amazon.com/efs/。
2. 選取要刪除的檔案系統。
3. 選擇 Actions (動作)、Delete file system (刪除檔案系統)。
4. 出現確認提示時，請輸入檔案系統 ID，然後選擇 Delete file system (刪除檔案系統)。

使用 Amazon FSx 與 Amazon EC2 實例

Amazon 服務系FSx列可讓您輕鬆啟動、執行和擴展由熱門商業和開放原始碼檔案系統提供支援的共用儲存。您可以使用新的啟動執行個體精靈，在啟動時自動將下列類型的 Amazon FSx 檔案系統附加到 Amazon EC2 執行個體：

- Amazon FSx for 在 AWS 雲端 NetApp ONTAP提供全受管共用儲存，並具備的常用資料存取和管理功能 NetApp ONTAP。
- Amazon FSx for Open ZFS 提供全受管、符合成本效益的共用儲存，由熱門的開放式ZFS檔案系統提供

> **Note**
>
> - 此功能僅在新的啟動執行個體精靈中可用。如需詳細資訊，請參閱 使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體
> - Amazon FSx 視窗檔案伺服器和 Amazon FSx 的 Lustre 檔案系統無法在啟動時掛載。啟動後，必須手動掛載這些檔案系統。

您可以選擇掛載先前建立的現有檔案系統，也可以建立新的檔案系統，以便在啟動時掛載到執行個體。

主題

- 安全群組和使用者資料指令碼
- 啟動時掛載 Amazon FSx 檔案系統

安全群組和使用者資料指令碼

使用啟動執行個體精靈將 Amazon FSx 檔案系統掛接到執行個體時，您可以選擇是否自動建立並附加啟用檔案系統存取權所需的安全群組，以及是否自動包含掛載檔案系統所需的使用者資料指令碼並使其可供使用。

主題

- 安全群組
- 使用者資料指令碼

安全群組

如果您選擇自動建立啟用存取檔案系統所需的安全群組，啟動執行個體精靈將建立並連接兩個安全群組：一個安全群組連接到執行個體，另一個連接到檔案系統。如需有關安全群組要求的詳細資訊，請參閱使FSx用 Amazon 進行ONTAP檔案系統存取控制，以VPC及FSx如何使用 Amazon 開啟ZFS檔案系統存取控制VPC。

我們會將標籤 `Name=instance-sg-`*`1`* 新增至建立並附加至執行個體的安全群組。每次啟動執行個體精靈為 Amazon FSx 檔案系統建立安全群組時，標籤中的值都會自動增加。

安全群組包含下列輸出規則，但不包含輸入規則。

傳出規則

通訊協定類型	連接埠號碼	目的地
UDP	111	*file system security group*
UDP	二零零一至二零零三年	*file system security group*
UDP	4049	*file system security group*
UDP	2049	*file system security group*
UDP	635	*file system security group*
UDP	4045 - 4046	*file system security group*
TCP	4049	*file system security group*
TCP	635	*file system security group*
TCP	2049	*file system security group*
TCP	111	*file system security group*
TCP	4045 - 4046	*file system security group*
TCP	二零零一至二零零三年	*file system security group*
全部	全部	*file system security group*

建立並附加至檔案系統的安全群組會以 `Name=fsx-sg-1` 標記。每次啟動執行個體精靈為 Amazon FSx 檔案系統建立安全群組時，標籤中的值都會自動增加。

安全群組包括以下規則。

傳入規則

通訊協定類型	連接埠號碼	來源
UDP	2049	*instance security group*
UDP	二零零一至二零零三年	*instance security group*
UDP	4049	*instance security group*
UDP	111	*instance security group*
UDP	635	*instance security group*
UDP	4045 - 4046	*instance security group*
TCP	4045 - 4046	*instance security group*
TCP	635	*instance security group*
TCP	2049	*instance security group*
TCP	4049	*instance security group*
TCP	二零零一至二零零三年	*instance security group*
TCP	111	*instance security group*

傳出規則

通訊協定類型	連接埠號碼	目的地
全部	全部	0.0.0.0/0

使用者資料指令碼

如果您選擇自動連接使用者資料指令碼，啟動執行個體精靈會將以下使用者資料新增到執行個體中。此指令碼將安裝必要的套件、掛載檔案系統並更新您的執行個體設定，以便在執行個體重新啟動時，自動重新掛載檔案系統。

```
#cloud-config
package_update: true
package_upgrade: true
runcmd:
- yum install -y nfs-utils
- apt-get -y install nfs-common
- svm_id_1=svm_id
- file_system_id_1=file_system_id
- vol_path_1=/vol1
- fsx_mount_point_1=/mnt/fsx/fs1
- mkdir -p "${fsx_mount_point_1}"
- if [ -z "$svm_id_1" ]; then printf "\n${file_system_id_1}.fsx.eu-
north-1.amazonaws.com:/${vol_path_1} ${fsx_mount_point_1} nfs4
 nfsvers=4.1,rsize=1048576,wsize=1048576,hard,timeo=600,retrans=2,noresvport,_netdev
 0 0\n" >> /etc/fstab; else printf "\n${svm_id_1}.${file_system_id_1}.fsx.eu-
north-1.amazonaws.com:/${vol_path_1} ${fsx_mount_point_1} nfs4
 nfsvers=4.1,rsize=1048576,wsize=1048576,hard,timeo=600,retrans=2,noresvport,_netdev 0
 0\n" >> /etc/fstab; fi
- retryCnt=15; waitTime=30; while true; do mount -a -t nfs4 defaults; if [ $? = 0 ] ||
 [ $retryCnt -lt 1 ]; then echo File system mounted successfully; break; fi; echo File
 system not available, retrying to mount.; ((retryCnt--)); sleep $waitTime; done;
```

啟動時掛載 Amazon FSx 檔案系統

若要在啟動時掛載新的或現有的 Amazon FSx 檔案系統

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)，接著選擇 Launch instance (啟動執行個體) 來開啟啟動執行個體精靈。
3. 在「應用程式和作業系統映像」區段中，選取AMI要使用的。
4. 在 Instance type (執行個體類型) 區段中，選取執行個體類型。
5. 在 Key pair (金鑰對) 區段中，選取現有的金鑰對或建立新金鑰對。
6. 在 Network settings (網路設定) 區段中，執行下列動作：

a. 選擇編輯。

b. 如果您想要掛載現有的檔案系統，請針對 Subnet (子網) 選擇檔案系統的偏好子網。我們推薦您在與檔案系統之偏好子網路相同的可用區域中啟動執行個體，以便最佳化效能。

   如果您想要建立新的檔案系統以掛載到執行個體，請針對 Subnet (子網) 選擇要在其中啟動執行個體的子網。

> **Important**
>
> 您必須選取子網路，才能在新啟動執行個體精靈中啟用 Amazon FSx 功能。如果沒有選取子網，則無法掛載現有檔案系統或建立新的檔案系統。

7. 在 Storage (儲存) 區段中，執行下列動作：

   a. 根據需要設定磁碟區。

   b. 展開 [檔案系統] 區段，然後選取FSx。

   c. 選擇 Add shared file system (新增共用檔案系統)。

   d. 針對 File system (檔案系統)，選取要掛載的檔案系統。

> **Note**
>
> 此清單會顯示所選區域中您帳戶中所有FSx用於開啟ZFS檔案系統的 Amazon for NetApp ONTAP 和 Amazon。FSx

   e. 若要自動建立並連接啟用存取檔案系統所需的安全群組，請選取 Automatically create and attach security groups (自動建立並連接安全群組)。如果您希望手動建立安全群組，請清除核取方塊。如需詳細資訊，請參閱 安全群組。

   f. 若要自動連接掛載檔案系統所需的使用者資料指令碼，請選取 Automatically mount shared file system by attaching required user data script (透過連接所需的使用者資料指令碼自動掛載共享檔案系統)。如果您希望手動提供使用者資料指令碼，請清除核取方塊。如需詳細資訊，請參閱 使用者資料指令碼。

8. 在 Advanced (進階) 區段中，根據需要編輯其他執行個體設定。

9. 選擇啟動。

搭配 Amazon EC2 執行個體使用 Amazon 檔案快取

Amazon File Cache 提供全受管的高速快取 AWS ，無論資料存放在何處，都能更輕鬆地處理檔案資料。Amazon 檔案快取可做為暫存、高效能的儲存位置，儲存在現場部署檔案系統、 AWS 檔案系統和 Amazon Simple Storage Service (Amazon S3) 存貯體中的資料。您可以使用此功能， AWS 透過統一檢視和高速 (低於一毫秒的延遲和高輸送量)，將分散的資料集提供給以檔案為基礎的應用程式。如需詳細資訊，請參閱 Amazon File Cache 使用者指南。

Amazon 檔案快取可搭配最常用的 Linux 使用AMIs，並且與 x86 型執行個體類型和引力子執行個體類型相容。您可以使用開放原始碼 Lustre 用戶端，從 Amazon EC2 執行個體存取快取。您可以掛載快取，然後使用標準 Linux 指令處理快取中的檔案和目錄。Amazon EC2 執行個體可以從相同虛擬私有雲 (VPC) 內的其他可用區域存取您的快取，前提是您的網路組態允許在中的子網路進行存取。VPC您也可以在共用中建立快取VPC。

若要開始使用，請參閱 Amazon 檔案快取使用者指南中的 Amazon 檔案快取入門。

管理您的 Amazon EC2 資源

資源是您可以使用的實體。Amazon EC2 會在您使用服務的功能時建立資源。例如，Amazon EC2 資源包括映像檔、執行個體、叢集、金鑰配對和安全群組。所有 Amazon EC2 資源類型都包含描述資源的屬性。例如，名稱、說明、資源識別碼和 Amazon 資源名稱 (ARN)。

Amazon EC2 資源專屬於它們所在的區 AWS 域或區域。例如，Amazon Machine Image (AMI) 是特定於某個 AWS 區域，但您從某個啟動的執行AMI個體專屬於您啟動該映像的區域。您可以使用其在許可政策中指定 Amazon EC2 資源ARN。

您 AWS 帳戶 有 Amazon 的預設配額EC2。這些配額定義了您可以建立的資源數目上限。例如，您所有執行中執行個體的最大vCPUs數量都有配額。如果啟動執行個體或啟動停止的執行個體會導致您超出配額，則作業會失敗。

您可以使用資源IDs或標籤， AWS 帳戶 依地區搜尋特定資源。若要跨多個區域搜尋特定資源或資源類型，請使用 Amazon EC2 全球檢視。

目錄

- 為您的 Amazon EC2 資源選擇一個區域
- 尋找您的 Amazon EC2 資源
- 使用 Amazon EC2 全球視圖查看跨區域的資源
- 標記您的 Amazon EC2 資源
- Amazon EC2 服務配額

為您的 Amazon EC2 資源選擇一個區域

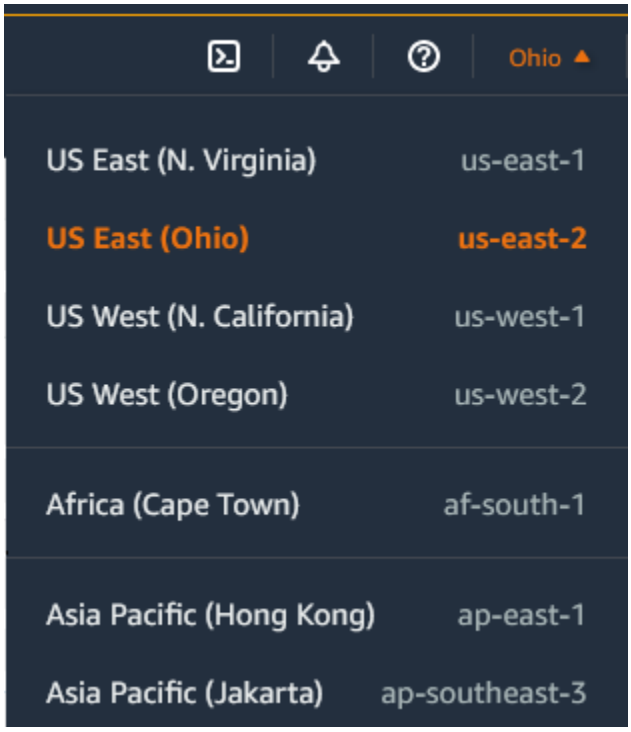
Amazon EC2 資源專屬於它們所在的區 AWS 域或區域。建立 Amazon EC2 資源時，請選取資源的區域。

考量事項

部分 AWS 資源可能無法在所有區域使用。在啟動 Amazon EC2 執行個體之前，請確保您可以在選定區域中建立所需的所有 AWS 資源。

若要使用主控台選取資源的區域

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 從導覽列中，選取 Regions (區域) 選取器，然後選取區域。



3. 「地區」選取器包含所有可用於您的 AWS 帳戶. 選擇清單底部附近加底線的文字，以檢視您帳戶未啟用的地區。若要啟用未啟用的AWS 地區，請參閱《AWS Account Management 參考指南》中的「指定您的帳戶可以使用的區域」。

尋找您的 Amazon EC2 資源

您可以使用 Amazon EC2 控制台獲取某些類型的資源列表。您可以使用其對應的命令或API動作獲取每種類型的資源的列表。如果您有許多資源，您可篩選結果，只包含或排除符合特定條件的資源。

內容

- 使用主控台列出和篩選資源
- 使用和來列出CLI和篩選 API
- 使用 Amazon EC2 全球視圖查看跨區域的資源

使用主控台列出和篩選資源

內容

- 使用主控台列出資源
- 使用主控台篩選資源

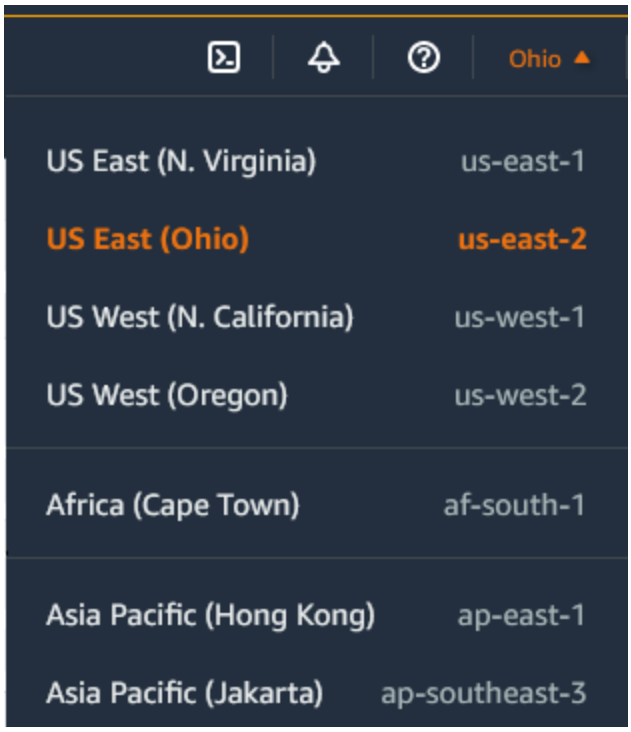
- 支援的篩選條件

使用主控台列出資源

您可以使用主控台檢視最常見的 Amazon EC2 資源類型。若要檢視其他資源，請使用指令行介面或 API動作。

若要使用主控台列出EC2資源

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. Amazon EC2 資源特定於 AWS 區域. 在導覽列中，從「區域」選取器中選擇「區域」。



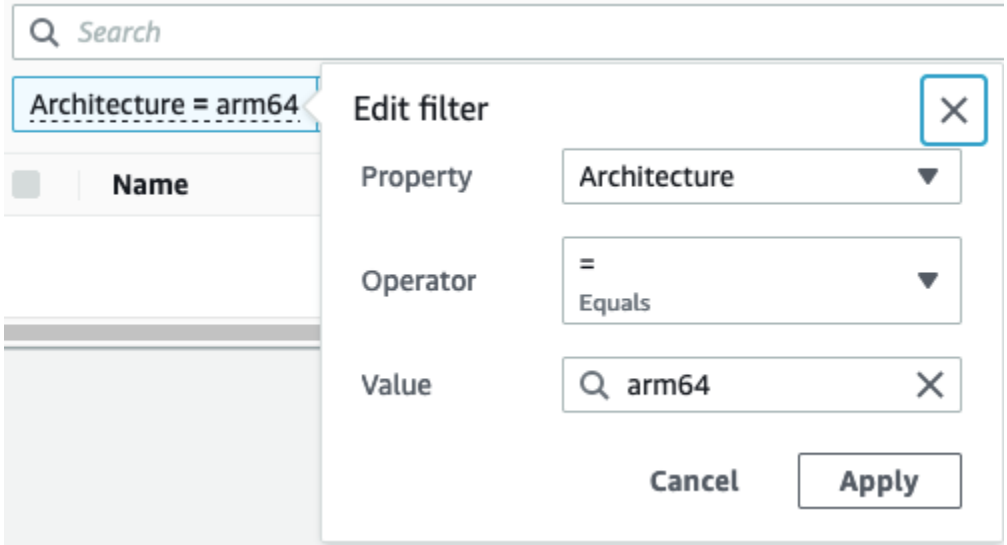
3. 在導覽窗格中選擇對應資源類型的選項。例如，若要列出所有執行個體，請選擇 [執行個體]。

使用主控台篩選資源

篩選資源清單

1. 在導覽窗格中，選取資源類型 (例如，Instances (執行個體))。
2. 選擇搜尋欄位。
3. 從清單中選取篩選條件。

4. 選取運算子，例如 = (等於)。某些屬性有更多可用的運算子可供選取。請注意，並非所有畫面均支援選取運算子。
5. 選取篩選條件值。
6. 若要編輯選取的篩選條件，請選取篩選條件字符 (藍色框)，進行所需的編輯，然後選取 Apply (套用)。請注意，並非所有畫面均支援編輯選取的篩選條件。



7. 完成後，請移除篩選器。

支援的篩選條件

Amazon 主EC2控台支援兩種篩選類型。

- API過濾發生在服務器端。篩選會套用於API呼叫，這會減少伺服器傳回的資源數目。它允許快速篩選大量資源，並且可以減少伺服器和瀏覽器之間傳輸資料的時間和成本。API過濾支持 = (等於) 和 : (包含) 運算符，並且始終區分大小寫。
- 用戶端篩選發生在用戶端。它可讓您篩選瀏覽器中已經可用的資料 (換句話說，已由傳回的資料API)。用戶端篩選可與API篩選器搭配使用，以便在瀏覽器中篩選到較小的資料集。除了 = (等於) 和 : (包含) 運算子，用戶端篩選也可以支援範圍運算子，例如 >= (大於或等於) 和否定 (反向) 運算子，例如 != (不等於)。

Amazon 主EC2控台支援以下類型的搜尋：

以關鍵字搜尋

以關鍵字搜尋是一種任意文字搜尋，可讓您在所有資源屬性或標籤中搜尋值，而不需指定要搜尋的屬性或標籤金鑰。

> **Note**
>
> 所有關鍵字搜尋都使用用戶端篩選。

要以關鍵字搜尋，請在搜尋方塊中輸入或貼上您要找的內容，然後選擇 Enter。例如，搜尋123符合其任何屬性中具有 123 的所有執行個體，例如 IP 位址、執行個體 VPC ID、AMI ID 或 ID，或其任何標籤 (例如名稱)。如果任意文字搜尋傳回非預期的相符項目，請套用其他篩選器。

依屬性搜尋

依屬性搜尋可讓您在所有資源中搜尋特定屬性。

> **Note**
>
> 根據選取的屬性，屬性搜尋會使用過API濾或用戶端篩選。執行屬性搜尋時，屬性會相應地分組。

例如，您可以搜尋所有執行個體的 Instance state (執行個體狀態) 屬性，以僅傳回處於 `stopped` 狀態的執行個體。若要執行此作業：

1. 在執行個體畫面的搜尋欄位中，開始輸入 `Instance state`。當您輸入字元時，執行個體狀態會顯示兩種篩選器類型：API篩選器和用戶端篩選器。
2. 若要在伺服器端進行搜尋，請在「API篩選器」下選擇「執行個體 若要在用戶端側搜尋，請在 Client filters (用戶端篩選條件) 下選擇 Instance state (client) (執行個體狀態 (用戶端))。

   此時會顯示所選屬性的可能運算子清單。
3. 選擇 = (等於) 運算子。

   此時會顯示所選屬性和運算子的可能值清單。
4. 從清單中選擇 Stopped (已停止)。

依標籤搜尋

依標籤搜尋，您可以透過標籤金鑰或標籤值篩選目前顯示的表中的資源。

標籤搜尋會使用API過濾或用戶端過濾，視「偏好設定」視窗中的設定而定。

確保API篩選標籤

1. 開啟 Preferences (偏好設定) 視窗。

2. 清除 Use regular expression matching (使用常規運算式比對) 核取方塊。如果選取此核取方塊，則會執行用戶端篩選。
3. 選取 Use case sensitive matching (使用區分大小寫比對) 核取方塊。如果清除此核取方塊，則會執行用戶端篩選。
4. 選擇確認。

依標籤搜尋時，可以使用下列值：

- (empty) ((空白)) – 尋找具有指定標籤金鑰的所有資源，但必須沒有標籤值。
- All values (所有值) – 尋找具有指定標籤金鑰和任何標籤值的所有資源。
- Not tagged (未標記) – 尋找不含指定標籤金鑰的索引資源。
- 顯示的值 – 尋找具有指定標籤金鑰和指定標籤值的所有資源。

您可以使用下列技巧來強化或精簡搜尋：

反向搜尋

反向搜尋可讓您搜尋不符合指定值的資源。在實例和AMIs畫面中，反向搜尋是透過選取! = (不等於) 或! : (不包含) 運算符，然後選擇一個值。在其他畫面中，反向搜尋是透過在搜尋關鍵字前面加上驚嘆號 (!) 字元來執行。

> **Note**
> 只有用戶端篩選條件上的關鍵字搜尋和屬性搜尋才支援反向搜尋。API篩選器的屬性搜尋不支援此功能。

例如，您可以搜尋所有執行個體的 Instance state (執行個體狀態) 屬性，以排除所有處於 `terminated` 狀態的執行個體。若要執行此作業：

1. 在執行個體畫面的搜尋欄位中，開始輸入 `Instance state`。當您輸入字元時，執行個體狀態會顯示兩種篩選器類型：API篩選器和用戶端篩選器。
2. 在 Client filters (用戶端篩選條件) 下選擇 Instance state (client) (執行個體狀態 (用戶端))。只有用戶端篩選條件支援反向搜尋。

   此時會顯示所選屬性的可能運算子清單。
3. 選擇 != (不等於)，然後選擇 terminated (終止)。

若要根據執行個體狀態屬性來篩選執行個體，您還可以使用 Instance state (執行個體狀態) 欄中的搜尋圖示

( )。

帶有加號 (+) 的搜尋圖示會顯示符合該屬性的所有執行個體。帶有減號 (-) 的搜尋圖示會排除符合該屬性的所有執行個體。

以下是使用反向搜尋的另一個範例：若要列出未獲指派名稱為 `launch-wizard-1` 之安全群組的所有執行個體，請在 Client filters (用戶端篩選條件) 下，依 Security group name (安全群組名稱) 屬性搜尋，選擇 !=，並在搜尋列中輸入 `launch-wizard-1`。

部分搜尋

透過部分搜尋，您可以搜尋部分字串值。若要執行部分搜尋，請僅輸入您要搜尋的關鍵字部分。在「執行個體」和AMIs畫面上，只能使用:(包含) 運算子執行部分搜尋。在其他畫面上，可以選取用戶端篩選條件屬性，並立即僅輸入您要搜尋的關鍵字部分。例如，在 Instance type (執行個體類型) 畫面上，若要搜尋所有 `t2.micro`、`t2.small` 和 `t2.medium` 執行個體，請依 Instance Type (執行個體類型) 屬性搜尋，並輸入關鍵字 `t2`。

常規運算式搜尋

若要使用常規運算式搜尋，必須選取 Preferences (偏好設定) 視窗中的 Use regular expression matching (使用常規運算式比對) 核取方塊。

當您需要將欄位中的值與特定模式匹配，就可以使用常規表達式。例如，若要搜尋開頭為 `s` 的值，請搜尋 `^s`。若要搜尋結尾為 `xyz` 的值，請搜尋 `xyz$`。或者，若要搜尋以數字開頭後跟一個或多個字元的值，請搜尋 `[0-9]+.*`。

> **Note**
>
> 只有用戶端篩選器上的關鍵字搜尋和屬性搜尋才支援常規運算式搜尋。API篩選器的屬性搜尋不支援此功能。

區分大小寫的搜尋

若要使用區分大小寫的搜尋，必須選取 Preferences (偏好設定) 視窗中的 Use case sensitive matching (使用區分大小寫比對) 核取方塊。區分大小寫的偏好設定僅適用於用戶端和標籤篩選條件。

> **Note**
> API過濾器始終區分大小寫。

萬用字元搜尋

使用 * 萬用字元來比對零個或多個字元。使用 ? 萬用字元來比對零個或一個字元。例如，如果您有一個具有值 prod、prods 和 production 的資料集，則搜尋 prod* 會比對出所有值，而 prod? 僅會比對出 prod 和 prods。若要使用常值，請使用反斜線 (\) 將其逸出。例如，"prod*" 會符合 prod*。

> **Note**
> 只有API篩選器的屬性和標籤搜尋才支援萬用字元搜尋。用戶端篩選器上的關鍵字搜尋和屬性及標籤搜尋不支援此功能。

組合搜尋

一般而言，具有相同屬性的多個篩選器會自動使用 OR 合併。例如，搜尋 Instance State : Running 和 Instance State : Stopped 會傳回執行中「或」已停止的所有執行個體。若要使用 AND 合併搜尋，請跨不同的屬性進行搜尋。例如，搜尋Instance State : Running並只Instance Type : c4.large傳回處於執行中狀態c4.largeAND之類型的執行個體。

使用和來列出CLI和篩選 API

每個資源類型都有一個對應的CLI命令和API動作，可用來列出該類型的資源。產生的資源清單可能很長，因此若要加快速度並提高實用性，請篩選結果，只包含符合特定條件的資源。

篩選考量

- 您可以在單一要求中指定多達 50 個篩選條件和每個篩選器最多 200 個值。
- 篩選字串的長度最多可達 255 個字元。
- 您可以使用萬用字元搭配篩選條件值。星號 (*) 代表零或更多字元，問號 (?) 代表零或一個字元。
- 篩選條件值區分大小寫。
- 您的搜尋可包含萬用字元的文字值；您只需要在字元前加一個反斜線區隔它們即可。例如，*amazon\?\\ 值會搜尋 *amazon?\ 文字字串。

支援的篩選條件

若要查看每個 Amazon EC2 資源支援的篩選器，請參閱下列文件：

- AWS CLI：命`describe`令參考-EC2 Amazon 中的AWS CLI 命令。
- 適用於 Windows 的工具 PowerShell：指`Get`令程式中的命AWS Tools for PowerShell 令參考 EC2-Amazon。
- 查詢API：Amazon EC2 API 參考中的`Describe`API操作。

Example 範例：指定單一篩選條件

您可以使用列出 Amazon EC2 實例describe-instances。在沒有篩選條件的情況下，回應會包含您所有資源的資訊。您可以使用下列命令，只在輸出中包含執行中的執行個體。

```
aws ec2 describe-instances --filters Name=instance-state-name,Values=running
```

若只要列出執行中執行個體IDs的執行個體，請依照下列步驟新增`--query`參數。

```
aws ec2 describe-instances --filters Name=instance-state-name,Values=running --query
 "Reservations[*].Instances[*].InstanceId" --output text
```

下列為範例輸出。

```
i-0ef1f57f78d4775a4
i-0626d4edd54f1286d
i-04a636d18e83cfacb
```

Example 範例：指定多個篩選條件或篩選條件值

如果您指定多個篩選條件或多個篩選條件值，資源必須符合所有要包含在結果中的篩選條件。

您可以使用下列命令，列出類型為 `m5.large` 或 `m5d.large` 的所有執行個體。

```
aws ec2 describe-instances --filters Name=instance-type,Values=m5.large,m5d.large
```

您可以使用下列命令，列出類型為 `t2.micro` 的所有已停止的執行個體。

```
aws ec2 describe-instances --filters Name=instance-state-name,Values=stopped
 Name=instance-type,Values=t2.micro
```

Example 範例：在篩選條件值中使用萬用字元

如果您在使用描述EBS快照描述快照描述時指定database為description篩選器的篩選值，命令只會傳回描述為「Database」的快照集。

```
aws ec2 describe-snapshots --filters Name=description,Values=database
```

* 萬用字元符合零或更多字元。如果您將 *database* 指定為篩選條件值，命令只會傳回描述中包含 database (資料庫) 此單字的快照。

```
aws ec2 describe-snapshots --filters Name=description,Values=*database*
```

? 萬用字元確切符合 1 個字元。如果您指定 database? 做為篩選條件值，命令只會傳回描述為「database」或「database」後面接著一個字元的快照。

```
aws ec2 describe-snapshots --filters Name=description,Values=database?
```

如果您指定 database????，命令只會傳回描述為「資料庫」後面接著最多四個字元的快照。它會排除具有「資料庫」後面接著五個或更多字元的描述。

```
aws ec2 describe-snapshots --filters Name=description,Values=database????
```

Example 範例：根據日期篩選

使用 AWS CLI，您可以使用運算式JMESPath來篩選結果。例如，下列describe-snapshots指令會顯示由您建立IDs的所有快照 AWS 帳戶 (以下列方式表示) *123456789012*) 在指定日期之前 (以下列方式表示) *2020-03-31*). 如果您未指定擁有者，結果會包含所有公用快照集。

```
aws ec2 describe-snapshots --filters Name=owner-id,Values=123456789012 --query
 "Snapshots[?(StartTime<='2020-03-31')].[SnapshotId]" --output text
```

下列指令會顯示在指定日期範圍內建立IDs的所有快照。

```
aws ec2 describe-snapshots --filters Name=owner-id,Values=123456789012 --query
 "Snapshots[?(StartTime>='2019-01-01') && (StartTime<='2019-12-31')].[SnapshotId]" --
output text
```

根據標籤篩選

如需如何根據標籤篩選資源清單的範例，請參閱按標籤過濾 Amazon EC2 資源。

使用 Amazon EC2 全球視圖查看跨區域的資源

Amazon EC2 全球檢視可讓您在單一主控台中同時檢視EC2和搜尋單一 AWS 區域或跨多個區域的 Amazon 和 Amazon VPC 資源。如需詳細資訊，請參閱使用 Amazon EC2 全球視圖查看跨區域的資源。

使用 Amazon EC2 全球視圖查看跨區域的資源

Amazon EC2 全球檢視可讓您在單一主控台中檢視單一區域或跨多個 AWS 區域的部分 Amazon EC2 和 Amazon VPC 資源。Amazon EC2 全域檢視也提供全域搜尋功能，可讓您同時在多個區域中搜尋特定資源或特定資源類型。

Amazon EC2 全球視圖不允許您以任何方式修改資源。

支援的資源

使用 Amazon EC2 全域檢視，您可以檢視已啟用您 AWS 帳戶 的所有區域的下列資源的全域摘要。

- Auto Scaling 群組
- DHCP選項集
- 輸出限定網際網路閘道
- 彈性 IPs
- 端點服務
- 執行個體
- 網際網路閘道
- 受管理的字首清單
- NAT閘道
- 網絡 ACLs
- 網路介面
- 路由表
- 安全群組
- 子網
- 磁碟區

- VPCs
- VPC端點
- VPC對等連接

所需的許可

使用者必須具有下列許可才能使用 Amazon EC2 全域檢視。

```
{
  "Version": "2012-10-17",
  "Statement": [
  {
    "Effect": "Allow",
    "Action": [
    "autoscaling:DescribeAutoScalingGroups",
    "ec2:DescribeRegions",
    "ec2:DescribeDhcpOptions",
    "ec2:DescribeEgressOnlyInternetGateways",
    "ec2:DescribeAddresses",
    "ec2:DescribeVpcEndpointServices",
    "ec2:DescribeInstances",
    "ec2:DescribeInternetGateways",
    "ec2:DescribePrefixLists",
    "ec2:DescribeNatGateways",
    "ec2:DescribeNetworkAcls",
    "ec2:DescribeNetworkInterfaces",
    "ec2:DescribeRouteTables",
    "ec2:DescribeSecurityGroups",
    "ec2:DescribeSubnets",
    "ec2:DescribeVolumes",
    "ec2:DescribeVpcs",
    "ec2:DescribeVpcEndpoints",
    "ec2:DescribeVpcPeeringConnections"
  ],
  "Resource": "*"
  }]
}
```

若要使用 Amazon EC2 全球檢視

在https://console.aws.amazon.com/ec2globalview/家中打開 Amazon EC2 全球視圖控制台。

> **Important**
>
> 您不能使用 Firefox 中的私人窗口訪問 Amazon EC2 全球視圖。

主控台包含下列項目：

- Region explorer (區域總管) - 此索引標籤包含下列部分：
  - 摘要：提供所有區域資源的高階概觀。

    「啟用區域」表示您 AWS 帳戶 已啟用的「區域」數目。其餘欄位表示您目前在這些區域中擁有的資源數目。選擇任一連結，即可檢視所有區域中該類型的資源。例如，如果 Instances (執行個體) 標籤下的連結是 29 in 10 Regions (10 個區域中的 29 個)，則表示您目前跨 10 個區域擁有 29 個執行個體。選擇該連結以檢視全部 29 個執行個體的清單。
  - 資源區域計數：列出所有 AWS 區域 (包括未啟用您帳戶的區域)，並針對每個區域提供每種資源類型的總計。

    選擇區域名稱，以檢視該特定區域所有類型的所有資源。例如，選擇 af-south-1 的非洲 (開普敦)VPCs，即可檢視該區域中的所有子網路、執行個體、安全群組、磁碟區和 Auto Scaling 群組。或者，選取一個區域，然後選取 View resources for selected Region (檢視所選區域的資源)。

    選擇特定區域中特定資源類型的值，以僅檢視該區域中該類型的資源。例如，選擇非洲 (開普敦) af-south-1 以僅檢視該區域中的執行個體。
- Global search (全域搜尋) - 此索引標籤可讓您跨單一區域或多個區域搜尋特定資源或特定資源類型。它也可讓您檢視特定資源的詳細資訊。

  若要搜尋資源，請在方格前面的欄位中輸入搜尋條件。您可以依區域、資源類型以及指派給資源的標籤進行搜尋。

  若要檢視特定資源的詳細資訊，請在方格中選取該資源。您也可以選擇要在該各個主控台中開啟的資源 ID。例如，選擇執行個體 ID 以在 Amazon EC2 主控台中開啟執行個體，或選擇子網路 ID 以在 Amazon VPC 主控台中開啟子網路。

> **Tip**
>
> 如果您只使用特定區域或資源類型，則可以自訂 Amazon EC2 全域檢視以僅顯示那些區域和資源類型。若要自訂顯示的區域和資源類型，請在導覽面板中選擇設定，然後在資源和區域索引標籤上選取您不想在 Amazon EC2 Global View 中顯示的區域和資源類型。

標記您的 Amazon EC2 資源

為了協助您管理執行個體、映像和其他 Amazon EC2 資源，您可以以標籤形式將自己的中繼資料指派給每個資源。標籤可讓您以不同的方式對 AWS 資源進行分類，例如，依目的、擁有者或環境。當您有許多相同類型的資源時，這將會很有用，因為—您可以依據先前指派的標籤，快速識別特定的資源。本主題說明標籤並示範如何建立它們。

> **Warning**
>
> 標籤鍵和它們的值由許多不同的API調用返回。拒絕存取`DescribeTags`並不會自動拒絕存取其他APIs人傳回的標籤。根據最佳實務，建議您不要在標籤中包含敏感資料。

內容

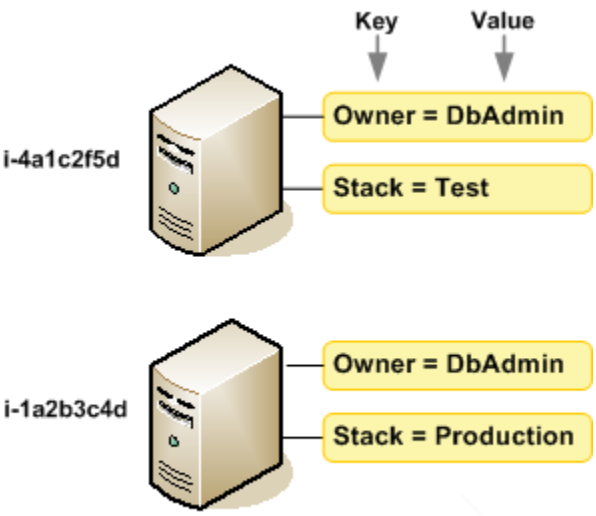
- 標籤基本概念
- 標記您的 資源
- 標籤限制
- 標籤與存取管理
- 標記您的資源以便計費
- 授予在建立期間標記 Amazon EC2 資源的權限
- 為 Amazon EC2 資源添加和刪除標籤
- 按標籤過濾 Amazon EC2 資源
- 使用執行個體中繼資料檢視EC2執行個體的

標籤基本概念

標籤是指派給 AWS 資源的標籤。每個標籤皆包含由您定義的一個金鑰與一個選用值。

標籤可讓您以不同的方式對 AWS 資源進行分類，例如，依目的、擁有者或環境。例如，您可以為帳戶的 Amazon EC2 執行個體定義一組標籤，以協助您追蹤每個執行個體的擁有者和堆疊層級。

下圖說明標籤建立的運作方式。在此範例中，您已在每個執行個體上指派兩個標籤 – 一個標籤的金鑰為 `Owner`，另一個標籤的金鑰為 `Stack`。每個標籤也都有一個關聯值。



我們建議您為每種資源類型建立符合您需求的標籤金鑰。使用一致的標籤金鑰組可讓您更輕鬆管理您的資源。您可以根據您新增的標籤搜尋和篩選資源。如需有關如何實作有效資源標記策略的詳細資訊，請參閱標記最佳做法 AWS 白皮書。

標籤對 Amazon 沒有任何語義意義，EC2並嚴格解釋為一串字符。此外，標籤不會自動指派給您的資源。您可以編輯標籤金鑰和值，並且可以隨時從資源移除標籤。您可以將標籤的值設為空白字串，但您無法將標籤的值設為 Null。若您將與現有標籤具有相同鍵的標籤新增到該資源，則新值會覆寫舊值。如果您刪除資源，也會刪除任何該資源的標籤。

> **Note**
>
> 刪除資源後，其標籤可能會在控制台中保持可見API，並在短時間內CLI輸出。這些標籤將逐漸取消與資源的關聯，並永久刪除。

標記您的 資源

使用 Amazon EC2 主控台時，可以使用相關資源畫面上的「標籤」索引標籤將標籤套用至資源，也可以使用 AWS Resource Groups 主控台中的「標籤編輯器」。有些資源畫面可讓您在建立資源時指定資

源的標籤；例如，具有 Name 鍵和您指定之值的標籤。在大多數的案例中，主控台會立即在建立資源後套用標籤 (而非在資源建立過程時)。控制台可能會根據Name標籤來組織資源，但是此標籤對 Amazon EC2 服務沒有任何語義意義。

如果您使用的是 Amazon EC2 API、 AWS CLI、或，則 AWS SDK可以使用CreateTagsEC2API動作將標籤套用至現有資源。此外，有些資源建立動作可讓您在建立資源時指定資源的標籤。若標籤無法在資源建立時套用，我們會轉返資源建立程序。這可確保資源不是具有標籤建立，就是不會建立，因此無論何時都不會有不具有標籤的資源。藉由在建立時為資源建立標籤，您可以消除在資源建立後執行自訂標籤指令碼的必要。如需有關讓使用者在建立時為資源加上標籤的詳細資訊，請參閱 授予在建立期間標記 Amazon EC2 資源的權限。

您可以在IAM政策中將以標籤為基礎的資源層級許可套用至支援建立標記的 Amazon EC2 API 動作，以便對可在建立時標記資源的使用者和群組實作精細控制。您的資源從建立時便已獲得適當保全，由於標籤會立即套用到您的資源，因此控制使用資源的任何標籤式資源層級許可都會立即生效。您可以更準確的追蹤和報告您的資源。您可以強制新資源使用標籤，並控制哪些標籤金鑰和值會在您的資源上設定。

您也可以將資源層級許可套用至IAM政策中的CreateTags和 DeleteTags Amazon EC2 API 動作，以控制在現有資源上設定哪些標籤金鑰和值。如需詳細資訊，請參閱範例：標籤資源。

如需為您的資源建立標籤以便計費的詳細資訊，請參閱 AWS Billing 使用者指南中的 Using cost allocation tags (使用成本分配標籤)。

標籤限制

以下基本限制適用於標籤：

- 每一資源最多標籤數 – 50
- 對於每一個資源，每個標籤金鑰必須是唯一的，且每個標籤金鑰只能有一個值。
- 最大金鑰長度 — 128 個萬國碼字元 UTF -8
- 最大值長度 — 256 個萬國碼字元 UTF -8
- 允許的字元
  - 雖然EC2允許其標籤中的任何字元，但其他 AWS 服務的限制性更強。所有 AWS 服務中允許的字元包括：字母 (a-z,A-Z)、數字 (0-9) 和 UTF -8 中可表示的空格，以及下列字元:+ - = . _ : / @.
  - 若啟用執行個體中繼資料中的執行個體標籤，則執行個體標籤鍵值只能使用字母 (a-z、A-Z)、數字 (0-9) 和下列字元：+ - = . , _ : @。執行個體標籤鍵不可含有空格或 /，而且不能僅包

含 `.` (一個句點)、`..` (兩個句點) 或 `_index`。如需詳細資訊，請參閱 使用執行個體中繼資料檢視EC2執行個體的。

- 標籤金鑰與值皆區分大小寫。
- 前`aws:`綴保留供 AWS 使用。如果標籤具有此字首的標籤金鑰，則您無法編輯或刪除標籤的金鑰或值。具 `aws:` 字首的標籤，不算在受資源限制的標籤計數內。

您無法僅根據標籤終止、停止或刪除資源。您必須指定資源識別符。例如，若要刪除您套用稱為 `DeleteMe` 標籤金鑰的快照，您必須搭配快照的資源識別符 (例如 `DeleteSnapshots`) 使用 `snap-1234567890abcdef0` 動作。

當您標記公開或共用資源時，您指派的標籤僅適用於您的 AWS 帳戶；沒有其他 AWS 帳戶可以存取這些標籤。對於以標籤為基礎的共用資源存取控制，每個 AWS 帳號都必須指派自己的標籤集，以控制對資源的存取。

標籤與存取管理

如果您使用 AWS Identity and Access Management (IAM)，您可以控制 AWS 帳戶中哪些使用者有權建立、編輯或刪除標籤。如需詳細資訊，請參閱授予在建立期間標記 Amazon EC2 資源的權限。

您也可以使用資源標籤來實作以屬性為基礎的控制項 () ABAC。您可以建立允許根據資源標籤進行作業的IAM策略。如需詳細資訊，請參閱使用以屬性為基礎的存取控制存取。

標記您的資源以便計費

您可以使用標籤來組織帳 AWS 單，以反映自己的成本結構。為此，請註冊以獲取包含標籤鍵值的 AWS 帳戶賬單。如需有關使用標籤設定成本分配報告的詳細資訊，請參閱 AWS Billing 使用者指南中的每月成本分配報告。若想要查看合併資源的成本，您可根據具有相同標籤金鑰值的資源來整理您的帳單資訊。例如，您可以使用特定應用程式名稱來標記數個資源，然後整理帳單資訊以查看該應用程式跨數項服務的總成本。如需詳細資訊，請參閱 AWS Billing 使用者指南中的使用成本分配標籤。

> **Note**
>
> 若您才剛啟用報告，目前月份的資料會在 24 小時之後提供檢視。

成本分配標籤可用來表示哪些資源產生成本，但刪除或停用這些資源不一定能降低成本。例如，由另一個快照參考的快照資料會進行保留，即使包含原始資料的快照遭到刪除也不例外。如需詳細資訊，請參閱 AWS Billing 使用者指南中的 Amazon Elastic Block Store 磁碟區及快照。

> **Note**
>
> 已標記的彈性 IP 地址，不會顯示在您的成本分配報告中。

授予在建立期間標記 Amazon EC2 資源的權限

某些資源建立 Amazon EC2 API 動作可讓您在建立資源時指定標籤。您可以使用資源標籤來實作以屬性為基礎的控制項 () ABAC。如需詳細資訊，請參閱 標記您的 資源 和 使用以屬性為基礎的存取控制存取。

使用者若要在建立時標記資源，他們必須具備建立資源動作 (如 `ec2:RunInstances` 或 `ec2:CreateVolume`) 的使用許可。若標籤於資源建立動作指定，Amazon 會針對 `ec2:CreateTags` 動作執行其他授權，以確認使用者具備建立標籤的許可。因此，使用者必須同時具備使用 `ec2:CreateTags` 動作的明確許可。

在`ec2:CreateTags`動作的IAM策略定義中，使用帶有`ec2:CreateAction`條件索引鍵的`Condition`元素，為建立資源的動作授與標記權限。

下列範例示範一個政策，其允許使用者啟動執行個體，並在啟動期間將標籤套用至執行個體與磁碟區。使用者沒有標記現有資源的權限 (他們不能直接呼叫 `ec2:CreateTags` 動作)。

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
         "ec2:RunInstances"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
         "ec2:CreateTags"
      ],
      "Resource": "arn:aws:ec2:region:account:*/*",
      "Condition": {
         "StringEquals": {
             "ec2:CreateAction" : "RunInstances"
          }
       }
```

```
    }
  ]
}
```

同樣的，下列政策允許使用者建立磁碟區，並在磁碟區建立期間將標籤套用至該磁碟區。使用者沒有標記現有資源的權限 (他們不能直接呼叫 `ec2:CreateTags` 動作)。

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
         "ec2:CreateVolume"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
         "ec2:CreateTags"
      ],
      "Resource": "arn:aws:ec2:region:account:*/*",
      "Condition": {
         "StringEquals": {
             "ec2:CreateAction" : "CreateVolume"
          }
       }
    }
  ]
}
```

只有在資源建立動作中套用了標籤時，才評估 `ec2:CreateTags` 動作。因此，在沒有標記條件的情況下，若請求中未指定標籤，則具備資源建立許可的使用者不需要使用 `ec2:CreateTags` 動作的許可。然而，若該使用者試圖建立具有標籤的資源卻未具備使用 `ec2:CreateTags` 動作的許可，則該請求會失敗。

如果在啟動範本中提供標籤，也會評估 `ec2:CreateTags` 動作。如需政策範例，請參閱 啟動範本中的標籤。

控制對特定標籤的存取

您可以在IAM原則的`Condition`元素中使用其他條件來控制可套用至資源的標籤索引鍵和值。

下列條件金鑰機可與前一節中的範例搭配使用：

- aws:RequestTag：表示請求中必須存在特定標籤金鑰或標籤金鑰與值。請求內亦可指定其他標籤。
  - 搭配 StringEquals 條件運算子使用，以強制結合特定標籤金鑰與值，例如強制執行標籤 cost-center=cc123：

    ```
    "StringEquals": { "aws:RequestTag/cost-center": "cc123" }
    ```

  - 搭配 StringLike 條件運算子使用，以在請求中強制執行特定標籤金鑰，例如強制執行標籤金鑰 purpose：

    ```
    "StringLike": { "aws:RequestTag/purpose": "*" }
    ```

- aws:TagKeys：強制執行請求中使用的標籤金鑰。
  - 搭配 ForAllValues 修飾詞使用，若請求內提供特定標籤金鑰，將強制加以執行 (若請求內指定標籤，則僅允許特定標籤金鑰，不允許其他標籤)。例如，允許標籤金鑰 environment 或 cost-center：

    ```
    "ForAllValues:StringEquals": { "aws:TagKeys": ["environment","cost-center"] }
    ```

  - 搭配 ForAnyValue 修飾詞使用，以強制要求請求內至少具有一個指定的標籤金鑰。例如，請求內必須出現至少下列標籤金鑰 environment 或 webserver 之一：

    ```
    "ForAnyValue:StringEquals": { "aws:TagKeys": ["environment","webserver"] }
    ```

這些條件鍵可套用於支援標記的資源建立動作，以及 ec2:CreateTags 和 ec2:DeleteTags 動作。若要了解 Amazon EC2 API 動作是否支援標記，請參閱 Amazon 的動作、資源和條件金鑰EC2。

若要強制使用者在建立資源時指定標籤，您必須在資源建立動作內，搭配 aws:RequestTag 修飾詞使用 aws:TagKeys 條件金鑰或 ForAnyValue 條件金鑰。若使用者未針對資源建立動作指定標籤，則不會評估 ec2:CreateTags 動作。

以條件而言，條件金鑰不區分大小寫，而條件值會區分大小寫。因此，欲強制標籤鍵區分大小寫，請使用 aws:TagKeys 條件索引鍵，其中標籤鍵指定為條件值。

如需IAM原則範例，請參閱控制 Amazon 訪問的示例政策 EC2 API。如需有關多值條件的詳細資訊，請參閱《IAM使用指南》中的〈建立測試多個索引鍵值的條件〉。

為 Amazon EC2 資源添加和刪除標籤

當您建立 Amazon EC2 資源 (例如 Amazon EC2 執行個體) 時，您可以指定要新增至資源的標籤。您也可以使用 Amazon EC2 主控台顯示特定 Amazon EC2 資源的標籤。您也可以在現有的 Amazon EC2 資源中新增或移除標籤。

您可以使用 AWS Resource Groups 主控台中的標籤編輯器來檢視、新增或移除所有區域中所有 AWS 資源的標籤。您可以同時套用或移除多種資源類型的標籤。若要取得更多資訊，請參閱〈標記 AWS 資源使用指南〉。

任務

- 使用主控台新增和移除標籤
- 使用新增標籤 AWS CLI
- 使用新增標籤 CloudFormation

使用主控台新增和移除標籤

您可以直接從資源頁面管理現有資源的標籤。

若要管理現有資源的標籤

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽列中，選取資源所在的區域。
3. 在導覽窗格中，選取資源類型 (例如，Instances (執行個體))。
4. 從清單中選取資源。
5. 從「標籤」標籤中，選擇「管理標籤」。
6. 若要新增標籤，請選擇「新增標籤」，然後輸入標籤的金鑰和值。若要移除標籤，請選擇 Remove (移除)。
7. 選擇 Save (儲存)。

使用新增標籤 AWS CLI

下面的例子說明如何使用 create-tags 命令將標籤新增到現有資源。

Example 範例：將標籤新增至資源

以下命令將標籤添加**Stack=production**到指定的圖像，或覆蓋標籤鍵所AMI在的現有標籤。**Stack**如果命令成功，則不會傳回任何輸出。

```
aws ec2 create-tags \
    --resources ami-78a54011 \
    --tags Key=Stack,Value=production
```

Example 範例：將標籤新增至多個資源

此範例會為AMI和執行個體新增 (或覆寫) 兩個標籤。其中一個標籤只包含金鑰 (**webserver**)，沒有值 (我們會將值設定為空字串)。其他標籤包含金鑰 (**stack**) 與值 (**Production**)。如果命令成功，則不會傳回任何輸出。

```
aws ec2 create-tags \
    --resources ami-1a2b3c4d i-1234567890abcdef0 \
    --tags Key=webserver,Value=  Key=stack,Value=Production
```

Example 範例：新增具有特殊字元的標籤

此範例會將標籤 **[Group]=test** 新增至執行個體。中括號 (**[** 和 **]**) 是特殊字元，必須將其逸出。

如果您使用 Linux 或 OS X，若要逸出特殊字元，請以雙引號 (**"**) 括住具有特殊字元的元素，然後使用單引號 (**'**) 括住整個金鑰和值結構。

```
aws ec2 create-tags \
    --resources i-1234567890abcdef0 \
    --tags 'Key="[Group]",Value=test'
```

如果您使用 Windows，若要逸出特殊字元，請以雙引號 (") 括住具有特殊字元的元素，然後在每個雙引號字元前面加上反斜線 (****)，如下所示：

```
aws ec2 create-tags ^
    --resources i-1234567890abcdef0 ^
    --tags Key=\"[Group]\",Value=test
```

如果您使用 Windows PowerShell，若要逸出特殊字元，請以雙引號 (**"**) 括住具有特殊字元的值，在每個雙引號字元前加上反斜線 (****)，然後以單引號 (**'**) 括住整個索引鍵和值結構，如下所示：

```
aws ec2 create-tags `
    --resources i-1234567890abcdef0 `
    --tags 'Key=\"[Group]\",Value=test'
```

使用新增標籤 CloudFormation

使用 Amazon EC2 資源類型時，您可以使用`Tags`或`TagSpecifications`屬性指定標籤。

下列範例會使用其`Tags`屬性**`Stack=Production`**將標籤加入AWSEC2:::: Instance。

Example 範例：中的標籤 YAML

```
Tags:
  - Key: "Stack"
    Value: "Production"
```

Example 範例：中的標籤 JSON

```
"Tags": [
    {
        "Key": "Stack",
        "Value": "Production"
    }
]
```

下列範例會LaunchTemplate LaunchTemplateData使用其`TagSpecifications`屬性**`Stack=Production`**將標籤加入AWSEC2::::。

Example 範例： TagSpecifications YAML

```
TagSpecifications:
  - ResourceType: "instance"
    Tags:
    - Key: "Stack"
      Value: "Production"
```

Example 範例： TagSpecifications JSON

```
"TagSpecifications": [
```

```
    {
        "ResourceType": "instance",
        "Tags": [
            {
                "Key": "Stack",
                "Value": "Production"
            }
        ]
    }
]
```

按標籤過濾 Amazon EC2 資源

新增標籤後，您可以根據標籤金鑰和標籤值篩選 Amazon EC2 資源。

使用 Amazon EC2 主控台依標籤篩選資源

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選取資源類型 (例如，Instances (執行個體))。
3. 選擇搜尋欄位。
4. 在清單中的標籤下，選擇標籤金鑰。
5. 從清單中選擇對應的標籤值。
6. 完成後，請移除篩選器。

如需在 Amazon EC2 主控台中使用篩選器的詳細資訊，請參閱尋找您的 Amazon EC2 資源。

若要使用標籤篩選資源 AWS CLI

下列範例說明如何將篩選與 describe-instances 搭配使用，以檢視具有特定標籤的執行個體。所有EC2描述命令都使用此語法在單個資源類型中按標籤進行過濾。或者，您可以使用 describe-tags 命令，跨EC2資源類型按標籤進行篩選。

Example 範例 1：說明具有指定標籤金鑰的執行個體

下列命令會描述具有 **Stack** 標籤的執行個體，無論標籤的值為何。

```
aws ec2 describe-instances \
    --filters Name=tag-key,Values=Stack
```

Example 範例 2：說明具有指定標籤的執行個體

下列命令會描述具有 **Stack=production** 標籤的執行個體。

```
aws ec2 describe-instances \
    --filters Name=tag:Stack,Values=production
```

Example 範例 3：說明具有指定標籤值的執行個體

下列命令會描述具有值為 **production** 標籤的執行個體，無論標籤金鑰為何。

```
aws ec2 describe-instances \
    --filters Name=tag-value,Values=production
```

Example 範例 4：使用指定標籤描述所有EC2資源

以下命令描述了帶有標籤的所有EC2資源**Stack=Test**。

```
aws ec2 describe-tags \
    --filters Name=key,Values=Stack Name=value,Values=Test
```

使用執行個體中繼資料檢視EC2執行個體的

您可以從執行個體中繼資料存取執行個體的標籤。透過從執行個體中繼資料存取標籤，您不再需要使用`DescribeInstances`或`DescribeTags`API呼叫來擷取標籤資訊，這樣可以減少每秒的API交易次數，並可讓標籤擷取隨您控制的執行個體數量調整。此外，在執行個體上執行的本機程序可以直接從執行個體中繼資料檢視執行個體的標籤資訊。

根據預設，不可從執行個體中繼資料取得標籤；您必須明確允許存取。您可以在執行個體啟動時允許存取，或在執行中或已停止執行個體上啟動後允許存取。您也可以透過在啟動範本中指定此許可來允許標籤的存取。使用範本啟動的執行個體允許存取執行個體中繼資料中的標籤。

若新增或移除執行個體標籤，會在執行個體執行期間更新執行個體中繼資料，而無需停止然後啟動執行個體。

任務

- 允許存取執行個體中繼資料中的標籤
- 從執行個體中繼資料擷取標籤

- 關閉對執行個體中繼資料中標籤的存取

允許存取執行個體中繼資料中的標籤

根據預設，無法存取執行個體中繼資料中的執行個體標籤。對於每個執行個體，您必須使用下列其中一種方法來明確允許存取。

允許使用主控台存取執行個體中繼資料中的標籤

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後依次選取 Actions (動作)、Instance settings (執行個體設定)、Allow tags in instance metadata (允許執行個體中繼資料中的標籤)。
4. 要允許存取執行個體中繼資料中的標籤，請選取 Allow (允許) 核取方塊。
5. 選擇 Save (儲存)。

使用 AWS CLI允許在啟動時存取執行個體中繼資料中的標籤

使用 run-instances 命令並將 `InstanceMetadataTags` 設定為 `enabled`。

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --instance-type c3.large \
    ...
    --metadata-options "InstanceMetadataTags=enabled"
```

使用 AWS CLI允許存取執行中或已停止執行個體上的的執行個體中繼資料中的標籤

使用指modify-instance-metadata-options令並`--instance-metadata-tags`將設定為`enabled`。

```
aws ec2 modify-instance-metadata-options \
    --instance-id i-123456789example \
    --instance-metadata-tags enabled
```

從執行個體中繼資料擷取標籤

當您允許存取執行個體中繼資料中的執行個體標籤之後，就可以從執行個體中繼資料存取`tags/instance`類別。如需詳細資訊，請參閱訪問實例的實EC2例元數據。

執行個體中繼資料服務第 2 版

在 Amazon 執行EC2個體上執行下列範例，以擷取的執行個體中繼資料IMDSv2。

cURL

此範例會取得執行個體的所有標籤金鑰。

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/tags/instance
Name
Environment
```

此範例會取得先前範例中所取得之Name金鑰的值。要IMDSv2求會使用前一個範例中使用命令建立的已儲存 Token。令牌不得過期。

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/tags/instance/Name
MyInstance
```

PowerShell

此範例會取得執行個體的所有標籤金鑰。

```
PS C:\> $token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/meta-data/tags/instance
Name
Environment
```

此範例會取得先前範例中所取得之Name金鑰的值。要IMDSv2求會使用前一個範例中使用命令建立的已儲存 Token。令牌不得過期。

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/meta-data/tags/instance/Name
MyInstance
```

執行處理中繼資料服務版

在 Amazon 執行EC2個體上執行下列範例，以擷取的執行個體中繼資料IMDSv1。

cURL

此範例會取得執行個體的所有標籤金鑰。

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/tags/instance
Name
Environment
```

此範例會取得先前範例中所取得之Name金鑰的值。

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/tags/instance/Name
MyInstance
```

PowerShell

此範例會取得執行個體的所有標籤金鑰。

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/tags/instance
Name
Environment
```

此範例會取得先前範例中所取得之Name金鑰的值。

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/tags/
instance/Name
MyInstance
```

關閉對執行個體中繼資料中標籤的存取

若要關閉對執行個體中繼資料中的中繼資料標籤的存取，請使用下列其中一種方法。您不需要在啟動時關閉對執行個體中繼資料上的執行個體標籤的存取，因為根據預設，它處於關閉狀態。

使用主控台關閉對執行個體中繼資料中標籤的存取

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。

2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後依次選取 Actions (動作)、Instance settings (執行個體設定)、Allow tags in instance metadata (允許執行個體中繼資料中的標籤)。
4. 要允許存取執行個體中繼資料中的標籤，請選擇 Allow (允許) 核取方塊。
5. 選擇 Save (儲存)。

若要關閉執行個體中繼資料中繼資料中標籤的存取權 AWS CLI

使用指modify-instance-metadata-options令並`--instance-metadata-tags`將設定為`disabled`。

```
aws ec2 modify-instance-metadata-options \
    --instance-id i-123456789example \
    --instance-metadata-tags disabled
```

若要檢視是否允許存取執行個體中繼資料中的標籤，請使用 AWS CLI

使用 describe-instances 命令並指定執行個體 ID。使用`--query`參數可在結果中僅顯示執行個體中繼資料選項。

```
aws ec2 describe-instances \
    --instance-ids i-1234567890abcdef0 \
    --query "Reservations[*].Instances[*].MetadataOptions"
```

下列為範例輸出。的值`InstanceMetadataTags`表示是否允許存取執行個體中繼資料中的標籤。如果值是`enabled`，則允許。如果該值是`disabled`，則不允許。

```
[
    [
        {
            "State": "applied",
            "HttpTokens": "required",
            "HttpPutResponseHopLimit": 2,
            "HttpEndpoint": "enabled",
            "HttpProtocolIpv6": "disabled",
            "InstanceMetadataTags": "enabled"
        }
    ]
]
```

Amazon EC2 服務配額

當您建立您的時候 AWS 帳戶，我們會根據每個地區對您的 AWS 資源設定預設配額 (也稱為限制)。如果您嘗試超出資源的配額，則要求會失敗。例如，某個區域中可以為隨需執行個體佈建的 Amazon EC2 vCPUs 數量上限。如果您嘗試在區域中啟動執行個體，且此要求會導致使用量超過此配額，則要求會失敗。如果發生這種情況，您可以減少資源使用量或要求增加配額。

Service Quotas 主控台是一個集中的位置，您可以在其中檢視和管理 AWS 服務配額，並要求提高您使用的許多資源配額。使用我們提供的配額資訊來管理您的 AWS 基礎架構。計劃在您需要的時候，先行請求提高配額。

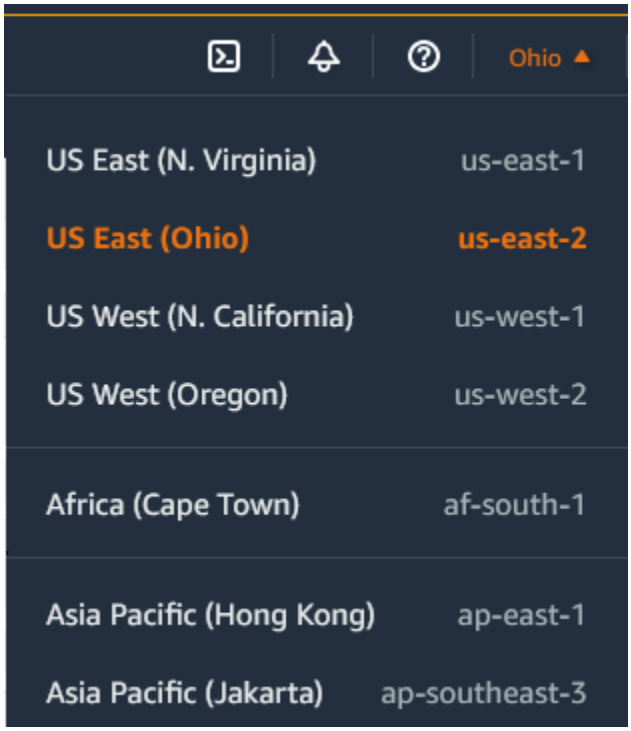
如需詳細資訊，請參閱中的 Amazon EBS 端點和配額以及 Amazon Web Services 一般參考. EC2

檢視您目前的配額

您可以使用 Ser vice Quotas 主控台檢視每個區域的配額。

使用 Service Quotas 主控台檢視目前的配額

1. 在「家庭服務 /ec2/Quotas https://console.aws.amazon.com/servicequotas//」開啟「Service Quotas」主控台。
2. 從導覽列 (位於畫面頂部) 中，選取一個區域。



3. 使用篩選欄位，按照資源名稱篩選清單。例如，輸入 **On-Demand** 找出隨需執行個體的配額。

4. 若要檢視詳細資訊，請選擇配額名稱以開啟配額的詳細資訊頁面。

請求增加

您可以請求增加每個區域的配額。

使用 Service Quotas 主控台請求提高

1. 在「家庭服務 /ec2/Quotas https://console.aws.amazon.com/servicequotas//」開啟「Service Quotas」主控台。
2. 從導覽列 (位於畫面頂部) 中，選取一個區域。
3. 使用篩選欄位，按照資源名稱篩選清單。例如，輸入 **On-Demand** 找出隨需執行個體的配額。
4. 如果配額可調整，請選取配額，然後選擇請求提高配額。
5. 對於變更配額值，輸入新配額值。
6. 選擇請求。
7. 若要檢視主控台中任何擱置或最近解決的請求，請從導覽窗格中選擇儀表板。對於擱置的請求，請選擇請求狀態以開啟請求回條。請求的初始狀態為 Pending (待定)。狀態變更為「要求配額」後，您會看到案例編號與 AWS Support。選擇案例編號，為請求開啟票證。

如需詳細資訊，包括如何使用 AWS CLI 或SDKs要求提高配額，請參閱《Service Quotas 使用者指南》中的要求增加配額。

使用通訊埠 25 傳送的電子郵件限制

在所有執行個體上，Amazon 預設會將輸出EC2流量限制為透過連接埠 25 的公有 IP 地址。您可以請求移除此限制。如需詳細資訊，請參閱如何從 Amazon EC2 執行個體或 Lambda 函數移除連接埠 25 的限制？

> **Note**
>
> 此限制不會適用於在連接埠 25 上傳送至下列地址的輸出流量：
>
> - 原始網路介面所VPC在之主要CIDR區塊中的 IP 位址。
> - 在一九一RFC八、RFC六五九八和 RFC 4193 中所CIDRs定義的 IP 位址。

監控 Amazon EC2 資源

監控是維護 Amazon EC2 執行個體和 AWS 解決方案的可靠性、可用性和效能的重要組成部分。您應該從 AWS 解決方案中的所有部分收集監視資料，以便在發生多點故障時更輕鬆地對多點失敗進行除錯。

AWS 提供了可用於監控 Amazon 的各種工具EC2。Amazon EC2 和 CloudWatch 主控台儀表板可讓您 at-a-glance 檢視 Amazon EC2 環境的狀態。此外，我們還提供以下內容：

- 系統狀態檢查 — 監控使用執行個體所需的 AWS 系統，以確保它們正常運作。這些檢查會偵測執行個體需要 AWS 參與修復的問題。系統狀態檢查失敗時，您可以選擇等待 AWS 修正問題，也可以自行解決問題 (例如停止並重新啟動，或終止並取代執行個體)。導致系統狀態檢查失敗的問題範例包括：
  - 網路連線中斷
  - 系統電力中斷
  - 實體主機的軟體問題
  - 實體主機上會影響網路連線的硬體問題

  如需詳細資訊，請參閱 Amazon EC2 執行個體的狀態檢查。
- 執行個體狀態檢查 — 監控個別執行個體的軟體和網路組態。這些檢查會偵測需要您介入修復的問題。一般來說，當執行個體狀態檢查失敗時，您需要自行處理問題 (例如重新啟動執行個體，或在作業系統中進行修改)。可能導致執行個體狀態檢查失敗的問題範例包括：
  - 系統狀態檢查失敗
  - 網路或啟動組態的設定不正確
  - 記憶體用盡
  - 檔案系統毀損
  - 核心不相容

  如需詳細資訊，請參閱Amazon EC2 執行個體的狀態檢查。
- Amazon CloudWatch 警示 — 觀看您指定期間內的單一指標，並根據指定臨界值在多個時段內相對於指定閾值的指標值執行一或多個動作。動作是傳送至 Amazon 簡單通知服務 (AmazonSNS) 主題或 Amazon EC2 Auto Scaling 政策的通知。警示只會呼叫持續狀態變更的動作。 CloudWatch 警示不會僅因為處於特定狀態而叫用動作；狀態必須已變更並維持指定數目的期間。如需詳細資訊，請參閱使用監控執行個體 CloudWatch。

- Amazon EventBridge 事件 — 自動化您的 AWS 服務並自動回應系統事件。 AWS 服務中的事件會以近乎即時 EventBridge 的方式傳送到，您可以指定當事件符合您撰寫的規則時要採取的自動化動作。如需詳細資訊，請參閱the section called “使用自動化 EventBridge”。
- AWS CloudTrail 日誌 — 擷取有關向 Amazon 發出的呼叫的詳細資訊，EC2API並將它們以日誌檔的形式存放在 Amazon S3 中。您可以使用 CloudTrail 記錄來確定撥打了哪些呼叫、呼叫的來源 IP 位址、撥打電話的人員以及撥打電話的時間。如需詳細資訊，請參閱the section called “使用記錄API呼叫 CloudTrail”。
- CloudWatch 代理程式 — 從EC2執行個體和內部部署伺服器上的主機和來賓收集記錄檔和系統層級指標。如需詳細資訊，請參閱 Amazon 使用者指南中的使用 CloudWatch 代理程式從 Amazon EC2 執行個體和現場部署伺服器收集 CloudWatch 指標和日誌。

監控 Amazon EC2 執行個體的狀態

您可以透過檢視您執行個體的狀態檢查和排程事件，監控您執行個體的狀態。

狀態檢查可為您提供 Amazon 執行自動化檢查所產生的資訊EC2。這些自動化檢查會偵測是否有特定問題正在影響您的執行個體。狀態檢查資訊以及 Amazon 提供的資料 CloudWatch，為您提供每個執行個體的詳細操作能見度。

您也可以查看為您的執行個體排程之特定事件的狀態。事件的狀態提供為您的執行個體計畫之近期活動的相關資訊，例如重新開機或淘汰。也提供每個事件的排程開始時間和結束時間。

目錄

- Amazon EC2 執行個體的狀態檢查
- Amazon EC2 執行個體的狀態變更事件
- Amazon EC2 執行個體的排程事件

Amazon EC2 執行個體的狀態檢查

透過執行個體狀態監控，您可以快速判斷 Amazon 是EC2否偵測到任何可能導致執行個體無法執行應用程式的問題。Amazon EC2 對每個執行中的執行個體執行自動化檢查，以識別硬體和軟體問題。EC2您可以檢視這些狀態檢查的結果，以找出特定及可偵測的問題。事件狀態資料可增強 Amazon EC2 已提供的有關每個執行個體狀態的資訊 (例如`pending`、`running`、`stopping`) 以及 Amazon CloudWatch 監控的使用率指標 (CPU使用率、網路流量和磁碟活動)。

狀態檢查會每分鐘執行一次，並會傳回通過或失敗狀態。如果所有檢查都通過，執行個體的整體狀態即為 OK (正常)。若一或多個檢查失敗，整體狀態便會是 impaired (故障)。狀態檢查內建於 Amazon 中 EC2，因此無法停用或刪除它們。

當狀態檢查失敗時，狀態檢查的對應 CloudWatch 測量結果就會增加。如需詳細資訊，請參閱狀態檢查指標。您可以使用這些指標來建立根據狀態檢查結果觸發的 CloudWatch警示。例如，您可以建立警示，在狀態檢查於特定執行個體上失敗時警告您。如需詳細資訊，請參閱為失敗狀態檢查的 Amazon EC2 執行個體建立 CloudWatch 警示。

您也可以建立 Amazon CloudWatch 警示來監控 Amazon 執行個體，並在EC2執行個體因潛在問題而受損時自動復原該執行個體。如需詳細資訊，請參閱執行個體恢復。

內容

- 狀態檢查類型
- 檢視 Amazon EC2 執行個體的狀態檢查
- 為失敗狀態檢查的 Amazon EC2 執行個體建立 CloudWatch 警示

狀態檢查類型

有三種類型的狀態檢查。

- 系統狀態檢查
- 執行個體狀態檢查
- 附加EBS狀態檢查

系統狀態檢查

系統狀態檢查會監控執行個體執行的 AWS 系統。這些檢查會偵測您執行個體需要 AWS 介入修復的基礎問題。系統狀態檢查失敗時，您可以選擇等 AWS 待修正問題，也可以自行解決。對於 Amazon 支援的執行個體EBS，您可以自行停止和啟動執行個體，這在大多數情況下會導致執行個體遷移到新主機。針對執行個體存放區支援的 Linux 執行個體，您可以終止並取代執行個體。對於 Windows 執行個體，根磁碟區必須是 Amazon EBS 磁碟區；根磁碟區不支援執行個體存放區。請注意，執行個體存放區是暫時的，當執行個體停止時，所有資料都會丟失。

下列為可能導致系統狀態檢查失敗的問題範例：

- 網路連線中斷

- 系統電力中斷
- 實體主機的軟體問題
- 實體主機上會影響網路連線的硬體問題

如果系統狀態檢查失敗，我們會增加 StatusCheckFailed_System 量度。

裸機執行個體

如果您在裸機執行個體上從作業系統執行重新啟動，則系統狀態檢查可能會暫時傳回失敗狀態。當執行個體變得可用時，系統狀態檢查應傳回通過狀態。

執行個體狀態檢查

Instance status checks (執行個體狀態檢查) 監控個別執行個體的軟體和網路組態。Amazon EC2 會將地址解析通訊協定 (ARP) 請求傳送至網路界面 (NIC)，以檢查執行個體的運作狀態。這些檢查會偵測需要您介入修復的問題。當執行個體狀態檢查失敗時，通常您需要自行處理問題 (例如重新開機執行個體或對執行個體的組態進行變更)。

> **Note**
>
> 最近用`systemd-networkd`於網路配置的 Linux 發行版可能會與早期發行版不同，報告健康狀態檢查的報告。在開機程序期間，這種類型的網路可能會提前啟動，並且可能會在其他也會影響執行個體健康狀態的啟動工作之前完成。依賴網路可用性的狀態檢查可在其他工作完成之前報告狀況良好的狀態。

下列為可能導致執行個體狀態檢查失敗的問題範例：

- 系統狀態檢查失敗
- 網路或啟動組態不正確
- 記憶體用盡
- 檔案系統毀損
- 核心不相容
- [Windows 執行個體] 在執行個體重新啟動期間或搭配 Windows 執行個體存放區支援的執行個體時，執行個體狀態檢查會報告失敗，直到執行個體再次可用為止。

如果執行個體狀態檢查失敗，我們會增加 StatusCheckFailed_Instance 量度。

裸機執行個體

如果您在裸機執行個體上從作業系統執行重新啟動，則執行個體狀態檢查可能會暫時傳回失敗狀態。當執行個體變得可用時，執行個體狀態檢查應傳回通過狀態。

附加EBS狀態檢查

附加EBS狀態檢查會監控連接至執行個體的 Amazon EBS 磁碟區是否可存取，以及是否能夠完成 I/O 操作。此`StatusCheckFailed_AttachedEBS`測量結果是二進位值，表示連接至執行個體的一或多個EBS磁碟區無法完成 I/O 作業時會造成損害。這些狀態檢查可偵測運算或 Amazon 基EBS礎設施的潛在問題。當連接的EBS狀態檢查指標失敗時，您可以等待 AWS 解決問題，也可以採取動作，例如更換受影響的磁碟區或停止再重新啟動執行個體。

以下是可能導致附加EBS狀態檢查失敗的問題範例：

- 磁碟區底層儲存子系統上的硬體或軟體問題 EBS
- 影響磁碟區可連接性的實體主機上的硬體問題 EBS
- 執行個體和EBS磁碟區之間的連線問題

可以使用 `StatusCheckFailed_AttachedEBS` 指標來協助改善工作負載的彈性。您可以使用此指標建立根據狀態檢查結果觸發的 Amazon CloudWatch 警示。例如，偵測到長期影響時，可容錯移轉至次要執行個體或可用區域。或者，您可以使用EBS CloudWatch 指標來偵測和更換受損的磁碟區，監控每個連接磁碟區的 I/O 效能。如果您的工作負載並未驅動連接至執行個體的任何EBS磁碟區的 I/O，且連接EBS狀態檢查顯示有損壞，您可以停止並啟動執行個體，以解決影響磁碟區可達性的實體主機的問題。EBS有關更多信息，請參閱 Amazon Amazon CloudWatch 指標 EBS

> **Note**
>
> - 附加的EBS狀態檢查指標僅適用於 Nitro 實例。
> - 您可以根據指標建立 CloudWatch 警示，來監視附加的EBS狀態檢查指`StatusCheckFailed_AttachedEBS`標。您無法使用describe-instance-status AWS CLI 命令來檢視此狀態檢查。

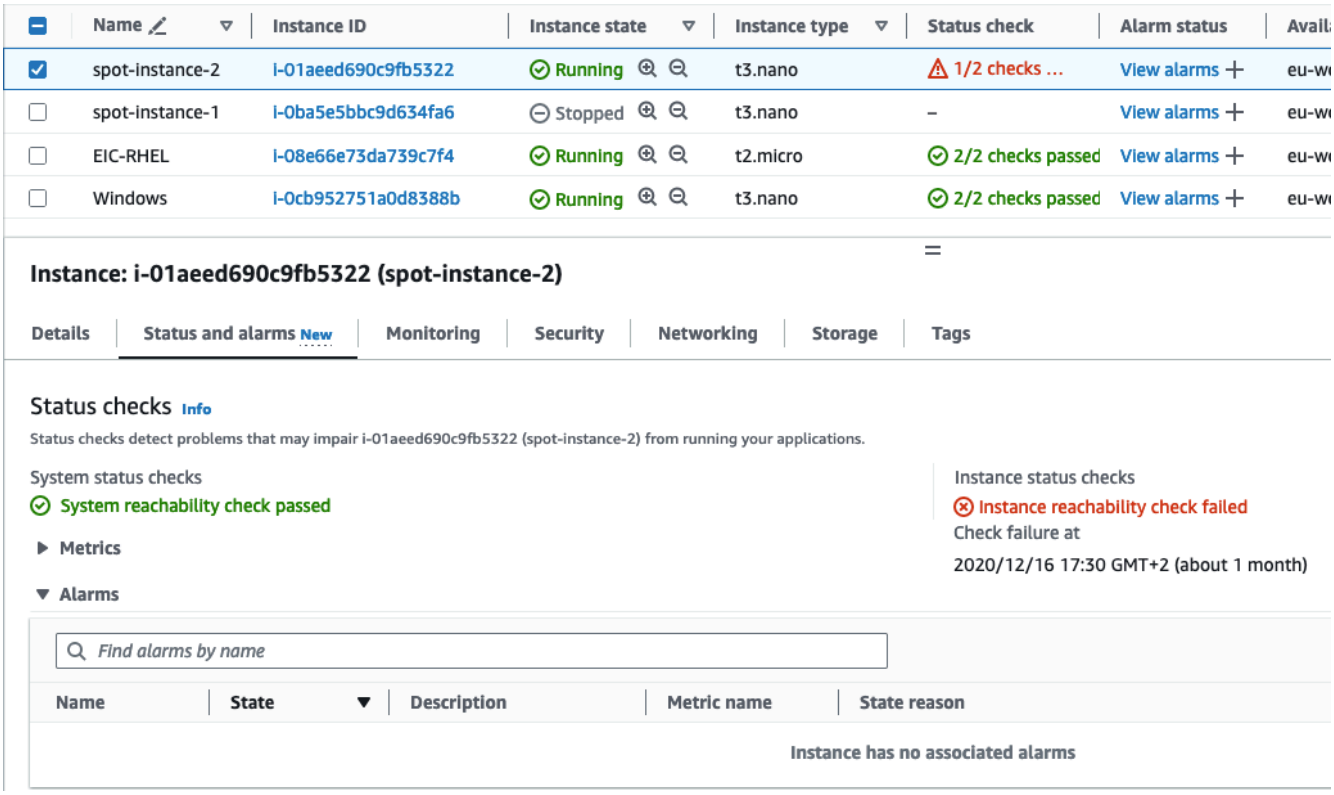
檢視 Amazon EC2 執行個體的狀態檢查

若要檢視狀態檢查，請使用以下方法之一。

Console

檢視狀態檢查

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 在執行個體頁面上，狀態檢查資料欄會列出每個執行個體的操作狀態。
4. 若要檢視特定執行個體的狀態，請選取執行個體，然後選擇狀態和警示索引標籤。



如果執行個體狀態檢查失敗，通常就需要您自行處理問題 (例如重新啟動執行個體或對執行個體的組態進行變更)。若要排解 Linux 執行個體上的系統或執行個體狀態檢查失敗，請參閱使用失敗的狀態檢查進行 Amazon EC2 Linux 執行個體。

5. 若要檢閱狀態檢查的 CloudWatch 量度，請在狀態與警示索引標籤上展開量度，以查看下列量度的圖表：
   - 系統狀態檢查失敗
   - 執行個體狀態檢查失敗

如需詳細資訊，請參閱the section called “狀態檢查指標”。

Command line

您可以使用 describe-instance-status(AWS CLI) 命令來檢視執行中執行個體的狀態檢查。

若要檢視所有執行個體的狀態，請使用下列命令。

```
aws ec2 describe-instance-status
```

若要取得執行個體狀態為 `impaired` 的所有執行個體狀態，請使用下列命令。

```
aws ec2 describe-instance-status \
--filters Name=instance-status.status,Values=impaired
```

若要取得單一執行個體的狀態，請使用下列命令。

```
aws ec2 describe-instance-status \
--instance-ids i-1234567890abcdef0
```

或者，使用下列 命令：

- Get-EC2InstanceStatus (AWS Tools for Windows PowerShell)
- DescribeInstanceStatus（Amazon EC2 查詢API）

如果您的 Linux 執行個體具有失敗狀態檢查，請參閱使用失敗的狀態檢查進行 Amazon EC2 Linux 執行個體。

為失敗狀態檢查的 Amazon EC2 執行個體建立 CloudWatch 警示

您可以使用狀態檢查測量結果來建立 CloudWatch 警示，以便在執行處理狀態檢查失敗時通知您。

> **Important**
>
> 如果缺少指標資料點，狀態檢查和狀態檢查警示可能會暫時進入不足的資料狀態。雖然很少見，但是當指標報告系統發生中斷時，即使執行個體健康狀態良好，也可能發生這種情況。我們建議您將此狀態視為遺失資料，而非狀態檢查失敗或警示洩漏，尤其是在執行個體上執行停止、終止、重新開機或復原動作以回應時。

若要建立狀態檢查警示，請使用以下方法之一：

Console

使用下列程序來設定警示，在執行個體的狀態檢查失敗時，透過電子郵件通知您、或者停止、終止或回復執行個體。

建立狀態檢查警示

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，選擇狀態檢查標籤，然後選擇動作 > 建立狀態檢查警示。
4. 在 [管理 CloudWatch 鬧鐘] 頁面的 [新增或編輯鬧鐘] 底下，選擇 [建立鬧鐘]。
5. 對於警示通知，請開啟開關以設定 Amazon 簡單通知服務 (AmazonSNS) 通知。選取現有的 Amazon SNS 主題或輸入名稱以建立新主題。

   如果您在收件人清單中新增電子郵件地址或建立新主題，Amazon SNS 會向每個新地址傳送訂閱確認電子郵件訊息。每個收件人都必須選擇該訊息中包含的連結以確認訂閱。僅會寄送提醒通知至確認過的地址。
6. 針對警示動作，開啟切換開關，以指定觸發警示時要採取的動作。選取動作。
7. 針對 Alarm thresholds (警示閾值)，指定警示的指標和準則。

   您可以保留 Group samples by (分組範例條件) (Average (平均)) 和 Type of data to sample (要作為範例的資料類型) (Status check failed:either (狀態檢查失敗：其中一個)) 的預設設定，或者您可以根據您的需求進行變更。

   針對連續期間，設定要評估的期間數，然後在期間中，輸入在觸發提醒及傳送電子郵件前的評估期間長度。
8. (選用) 針對 Sample metric data (範例指標資料)，選擇 Add to dashboard (新增至儀表板)。
9. 選擇 Create (建立)。

若您需要對執行個體狀態警示進行任何變更，您可以編輯它。

編輯狀態檢查警示

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體並選擇 [動作]、[監控]、[管理 CloudWatch 警示]
4. 在 [管理 CloudWatch 鬧鐘] 頁面的 [新增或編輯鬧鐘] 底下，選擇 [編輯鬧鐘]。

5. 針對 Search for alarm (搜尋警示)，選擇警示。
6. 完成變更後，請選擇 Update (更新)。

Command line

在下列範例中，當執行個體檢查失敗或系統狀態檢查至少兩個連續期間時`arn:aws:sns:us-west-2:111122223333:my-sns-topic`，警示會向SNS主題發出通知。使用的 CloudWatch 量度為`StatusCheckFailed`。

若要使用建立狀態檢查警示 AWS CLI

1. 選取現有SNS主題或建立新主題。如需詳細資訊，請參閱使 AWS CLI 用AWS Command Line Interface 者指南SNS中的搭配 Amazon 使用。
2. 使用以下列表指標命令查看 Amazon 的可用 Amazon 指 CloudWatch 標。EC2

```
aws cloudwatch list-metrics --namespace AWS/EC2
```

3. 使用下列put-metric-alarm指令建立鬧鐘。

```
aws cloudwatch put-metric-alarm \
--alarm-name StatusCheckFailed-Alarm-for-i-1234567890abcdef0 \
--metric-name StatusCheckFailed \
--namespace AWS/EC2 \
--statistic Maximum \
--dimensions Name=InstanceId,Value=i-1234567890abcdef0 \
--unit Count \
--period 300 \
--evaluation-periods 2 \
--threshold 1 \
--comparison-operator GreaterThanOrEqualToThreshold \
--alarm-actions arn:aws:sns:us-west-2:111122223333:my-sns-topic
```

此期間是收集 Amazon CloudWatch 指標的時間範圍 (以秒為單位)。此範例使用 300，即 60 秒乘以 5 分鐘。評估期間為針對指標的值與閾值進行比較的連續期間數。此範例使用 2。警示動作為觸發此警示時要執行的動作。此範例會將警示設定為使用 Amazon 傳送電子郵件。SNS

Amazon EC2 執行個體的狀態變更事件

Amazon EC2 會在執行`EC2 Instance State-change Notification`個體狀態變更 EventBridge 時向 Amazon 傳送事件。

以下是此事件的範例資料。在本範例中，執行個體進入 `pending` 狀態。

```
{
   "id":"7bf73129-1428-4cd3-a780-95db273d1602",
   "detail-type":"EC2 Instance State-change Notification",
   "source":"aws.ec2",
   "account":"123456789012",
   "time":"2021-11-11T21:29:54Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:ec2:us-east-1:123456789012:instance/i-abcd1111"
   ],
   "detail":{
      "instance-id":"i-abcd1111",
      "state":"pending"
   }
}
```

`state` 的可能值為：

- `pending`
- `running`
- `stopping`
- `stopped`
- `shutting-down`
- `terminated`

啟動或開始執行個體時，其會變成 `pending` 狀態，然後再變成 `running` 狀態。停止執行個體時，其會變成 `stopping` 狀態，然後再變成 `stopped` 狀態。終止執行個體時，其會變成 `shutting-down` 狀態，然後再變成 `terminated` 狀態。

建立警示，在 Amazon EC2 執行個體變更狀態時傳送電子郵件

若要在執行個體狀態變更時接收電子郵件通知，請建立 Amazon SNS 主題，然後為`EC2 Instance State-change Notification`事件建立 EventBridge 規則。

建立 SNS 主題

1. 在 https://console.aws.amazon.com/sns/v3/ home 上打開 Amazon SNS 控制台。
2. 在導覽窗格中，選擇主題。
3. 請選擇建立主題。
4. 針對類型，選擇標準。
5. 在 Name (名稱) 中，輸入主題名稱。
6. 請選擇建立主題。
7. 選擇 Create subscription (建立訂閱)。
8. 對於通訊協定，選擇電子郵件。
9. 在 Endpoint (端點) 中，輸入接收通知的電子郵件地址。
10. 選擇 Create subscription (建立訂閱)。
11. 您會收到帶有下列主旨行的電子郵件訊息：AWS Notification - Subscription Confirmation。請依照指示來確認訂閱。

若要建立 EventBridge 規則

1. 在打開 Amazon EventBridge 控制台https://console.aws.amazon.com/events/。
2. 選擇建立規則。
3. 在 Name (名稱) 中，輸入規則名稱。
4. 針對規則類型，選擇具有事件模式的規則。
5. 選擇 Next (下一步)。
6. 針對 Event pattern (事件模式)，請執行下列動作：
   a. 在 Event source (事件來源)，選擇 AWS 服務。
   b. 對於 AWS 服務，選擇EC2。
   c. 對於事件類型，請選擇EC2執行處理狀態變更通知。
   d. 根據預設，我們會傳送任何執行個體的任何狀態變更通知。如果願意，可以選擇特定狀態或特定執行個體。

7. 選擇 Next (下一步)。
8. 如下所示指定目標：
   a. 對於 Target types (目標類型)，選擇 AWS 服務。
   b. 針對 [選取目標]，選擇 [SNS主題]。
   c. 在「主題」中，選擇您在上一個程序中建立的SNS主題。
9. 選擇 Next (下一步)。
10. (選用) 將標籤新增至您的規則。
11. 選擇 Next (下一步)。
12. 選擇建立規則。
13. 為了測試您的規則，請啟動狀態變更。例如，開始已停止的執行個體、停止執行中的執行個體或啟動執行個體。您會收到帶有下列主旨行的電子郵件訊息：AWS Notification Message。電子郵件內文包含事件資料。

Amazon EC2 執行個體的排程事件

AWS 可以為您的執行個體排程事件，例如重新開機、停止/啟動或淘汰。這些事件不會頻繁發生。如果您的其中一個執行個體會受到排程事件的影響，請在排定的事件發生前，將電子郵件 AWS 傳送至與您 AWS 帳戶相關聯的電子郵件地址。此電子郵件提供事件的詳細資訊，包括開始和結束日期。視事件而定，您可能可以採取行動來控制事件的時間。 AWS 還會傳送 AWS Health 事件，您可以使用 Amazon 事件監控和管理 CloudWatch 事件。如需有關使用監視 AWS Health 事件的詳細資訊 CloudWatch，請參閱使用 AWS Health 事件監視 CloudWatch 事件。

排程事件由管理 AWS；您無法為執行個體排程事件。您可以檢視排程的事件 AWS、自訂排定的事件通知以包含或移除電子郵件通知中的標籤，以及在執行個體排定為重新開機、淘汰或停止時執行動作。

若要更新您帳戶的聯絡資訊，確保您會收到排程事件的通知，請前往帳戶設定頁面。

> **Note**
>
> 如果執行個體受到排程事件的影響，且該執行個體屬於 Auto Scaling 群組，Amazon EC2 Auto Scaling 最終會將其取代為其運作狀態檢查的一部分，您無需進一步採取任何動作。如需 Amazon Auto Scaling 執行的運作 Health 態檢查的詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南中的自 Au EC2 to Scaling 執行個體運作狀態檢查。

排程事件類型

Amazon EC2 可以為您的執行個體建立下列類型的事件，事件會在排定的時間發生：

- Instance stop (執行個體停止)：在排程時間，執行個體停止。當您再次啟動時，它會遷移到新的主機。僅適用於由 Amazon 支援的執行個體EBS。
- 執行個體淘汰：在排定的時間，如果執行個體受 Amazon 支援，則會停止執行個體；如果執行個體存放區支援EBS，則會終止執行個體。
- Instance reboot (執行個體重新開機)：在排程時間，執行個體重新開機。
- System reboot (系統重新開機)：在排程時間，執行個體的主機重新開機。
- System maintenance (系統維護)：在排程時間，執行個體可能會因網路維護或電源維護而暫時受到影響。

目錄

- 受排程事件影響之 Amazon EC2 執行個體的建議動作
- 檢視影響 Amazon EC2 執行個體的排程事件
- 為影響 Amazon EC2 執行個體的排程事件自訂電子郵件通知
- 重新排程影響 Amazon EC2 執行個體的排程事件
- 為影響 Amazon EC2 執行個體的排程事件建立自訂事件視窗

受排程事件影響之 Amazon EC2 執行個體的建議動作

下列主題說明 Amazon EC2 執行個體受到排程事件影響時應採取的動作。

主題

- 排定停止或淘汰的執行個體動作
- 排定重新開機的執行個體動作
- 排定維護的執行個體動作

排定停止或淘汰的執行個體動作

當 AWS 偵測到執行個體的基礎主機無法修復的故障時，會根據執行個體的根裝置類型，排程執行個體停止或終止執行個體。如果根裝置是EBS磁碟區，則執行個體會排程停止。若根設備為執行個體存放區磁碟區，則執行個體會排程終止。如需詳細資訊，請參閱 執行個體淘汰。

> **Important**
>
> 任何存放在執行個體存放區磁碟區的資料都會在執行個體停止、休眠或終止時遺失。這包括連接到具有磁碟區做為根裝置的執行個體的執行個體儲存磁EBS碟區。請確認您已在執行個體停止、休眠或終止前，從您的執行個體存放區磁碟區儲存您稍後需要的資料。

Amazon 支援的執行個體採取的動作 EBS

您可以等待執行個體依照排程停止。或者，您也可以自行停止並啟動執行個體，這會將它遷移到新的主機。如需停止執行個體的詳細資訊 (除了在停止時執行個體組態有何變更的資訊)，請參閱停止並啟動 Amazon EC2 執行個體。

您可以將立即停止並啟動的作業自動化，以回應排程的執行個體停止事件。如需詳細資訊，請參閱AWS Health 使用者指南中的自動化 Amazon EC2 執行個體動作。

執行個體存放區後端執行個體的動作

我們建議您從最新的執行個體啟動替代執行個體，AMI並在排定終止執行個體之前，將所有必要的資料移轉到替代執行個體。接著，您便可以終止原始執行個體，或是等待執行個體依照排程終止。

排定重新開機的執行個體動作

當 AWS 必須執行諸如安裝更新或維護基礎主機之類的工作時，它可以排程執行個體或基礎主機進行重新開機。您可以重新排程大部分重新開機事件，讓您的執行個體在適合您的特定日期和時間重新開機。

檢視重新開機事件類型

您可以使用下列其中一種方法，檢視重新開機事件是執行個體重新開機還是系統重新開機。

Console

檢視排程重新開機事件的類型

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Events (事件)。
3. 從篩選清單中選擇 Resource type: instance (資源類型：執行個體)。
4. 對於每個執行個體，檢視 Event type (事件類型) 欄中的值。此值為 system-reboot 或 instance-reboot。

AWS CLI

檢視排程重新開機事件的類型

使用指describe-instance-status令。

```
aws ec2 describe-instance-status \
    --instance-id i-1234567890abcdef0
```

對於排程的重新開機事件，Code 的值為 system-reboot 或 instance-reboot。下列範例輸出顯示 system-reboot 事件。

```
[
    "Events": [
                {
                    "InstanceEventId": "instance-event-0d59937288b749b32",
                    "Code": "system-reboot",
                    "Description": "The instance is scheduled for a reboot",
                    "NotAfter": "2019-03-14T22:00:00.000Z",
                    "NotBefore": "2019-03-14T20:00:00.000Z",
                    "NotBeforeDeadline": "2019-04-05T11:00:00.000Z"
                }
              ]
]
```

執行個體重新開機動作

在其排程維護時段內，您可以等待執行個體重新啟動。重新排程執行個體重新啟動至適合您的日期和時間，或在方便時親自重新啟動執行個體。

在您的執行個體重新開機後，排程事件便會取消，事件的描述也會更新。基礎主機的擱置維護已完成，您可以在完全啟動後再次開始使用您的執行個體。

系統重新開機動作

您無法自行重新開機系統。您可以等待在排程的維護時段視窗內，讓系統重新開機，或將系統重新開機重新排程在適合您的日期和時間。系統重新開機通常在幾分鐘內完成。系統重新開機後，執行個體會保留其 IP 位址和DNS名稱，並保留本機執行個體儲存磁碟區上的所有資料。在系統重新開機完成後，執行個體的排程事件便會清除，並且您可以確認您執行個體上的軟體正以預期的方式運作中。

或者，如果需要在不同的時間維護執行個體，而您無法重新排程系統重新開機，則可以停止並啟動 Amazon EBS 支援的執行個體，將其遷移到新主機。但是，本機執行個體存放區磁碟區上的資料不會保留。您也可以將立即執行個體停止並啟動的作業自動化，以回應排程的系統重新啟動事件。如需詳細資訊，請參閱AWS Health 使用指南中的自動EC2執行個體動作。對於執行個體商店支援的執行個體，如果您無法重新排定系統重新啟動，您可以從最近啟動替代執行個體AMI，在排定的維護時段之前將所有必要的資料移轉至替代執行個體，然後終止原始執行個體。

排定維護的執行個體動作

當 AWS 必須維護執行個體的基礎主機時，就會排程執行個體進行維護。有兩種維護事件類型：網路維護及電源維護。

在網路維護期間，排程執行個體會暫時遺失網路連線能力。維護完成後，便會還原您執行個體的一般網路連線能力。

在電源維護期間，排程執行個體會暫時離線，然後重新開機。當執行重新開機時，會保留所有您執行個體的組態設定。

在您的執行個體重新開機後 (通常需要數分鐘)，確認您的應用程式正以預期的方式正常運作。此時，您的執行個體應該不會再有任何相關聯的排程事件，或若是如此，則排程事件描述的開頭為 [Completed] ([已完成])。有時需要最多 1 個小時，才會更新執行個體狀態描述。完成的維護事件最多會顯示在 Amazon EC2 主控台儀表板上一週。

Amazon 支援的執行個體採取的動作 EBS

您可以等待維護依照排程執行。或者，您可以停止並啟動執行個體，這會將它遷移到新的主機。如需停止執行個體的詳細資訊 (除了在停止時執行個體組態有何變更的資訊)，請參閱停止並啟動 Amazon EC2 執行個體。

您可以將立即停止並啟動的作業自動化，以回應排程的維護事件。如需詳細資訊，請參閱AWS Health 使用指南中的自動EC2執行個體動作。

執行個體存放區後端執行個體的動作

您可以等待維護依照排程執行。或者，如果您想要在排定的維護時段期間維持正常作業，您可以從最近啟動取代執行個體AMI，在排定的維護時段之前，將所有必要的資料移轉至取代執行個體，然後終止原始執行個體。

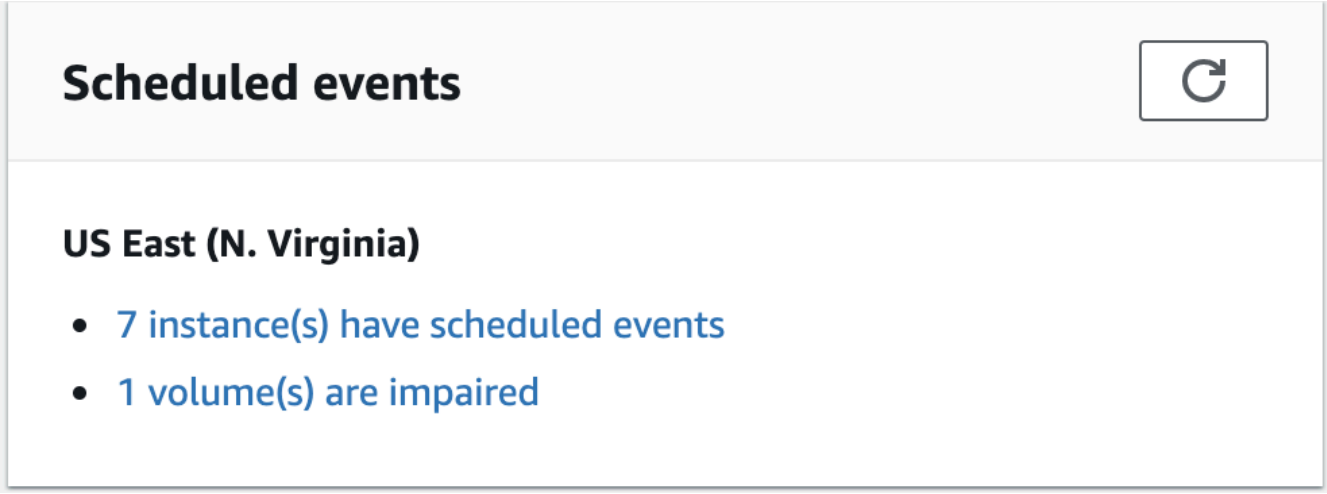
檢視影響 Amazon EC2 執行個體的排程事件

除了在電子郵件中接收排程事件的通知，您可以使用以下其中一個方法檢查排程事件。

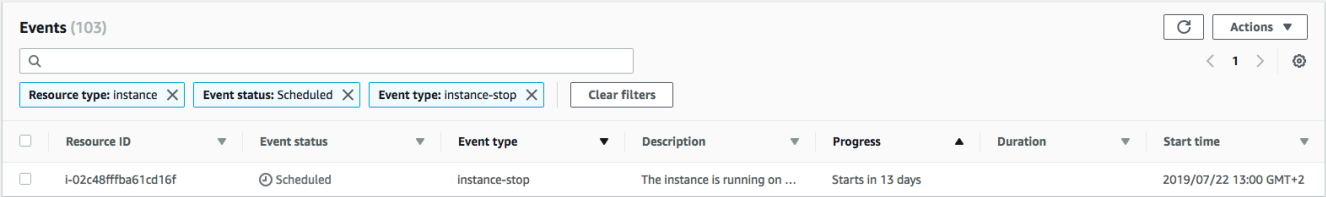
Console

檢視執行個體的排程事件

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 儀表板會在排程事件下方顯示任何與事件相關聯的資源。



3. 如需更多詳細資訊，在導覽窗格中選擇事件。此時會顯示任何與事件相關聯的資源。您可以依特性 (例如事件類型、資源類型和可用區域) 進行篩選。



AWS CLI

檢視執行個體的排程事件

使用指describe-instance-status令。

```
aws ec2 describe-instance-status \
    --instance-id i-1234567890abcdef0 \
    --query "InstanceStatuses[].Events"
```

下列範例輸出顯示重新開機事件。

```
[
    "Events": [
```

```
        {
            "InstanceEventId": "instance-event-0d59937288b749b32",
            "Code": "system-reboot",
            "Description": "The instance is scheduled for a reboot",
            "NotAfter": "2019-03-15T22:00:00.000Z",
            "NotBefore": "2019-03-14T20:00:00.000Z",
            "NotBeforeDeadline": "2019-04-05T11:00:00.000Z"
         }

    ]
]
```

下列為顯示執行個體淘汰事件的範例輸出。

```
[
    "Events": [
        {
            "InstanceEventId": "instance-event-0e439355b779n26",

            "Code": "instance-stop",
            "Description": "The instance is running on degraded hardware",
            "NotBefore": "2015-05-23T00:00:00.000Z"
        }
    ]
]
```

PowerShell

使用 AWS Tools for Windows PowerShell檢視您執行個體的排程事件

使用下列 Get-EC2InstanceStatus 命令。

```
PS C:\> (Get-EC2InstanceStatus -InstanceId i-1234567890abcdef0).Events
```

下列為顯示執行個體淘汰事件的範例輸出。

```
Code         : instance-stop
Description  : The instance is running on degraded hardware
NotBefore    : 5/23/2015 12:00:00 AM
```

Instance metadata

使用執行個體中繼資料檢視您執行個體的排程事件

您可以使用執行個體中繼資料服務第 2 版或執行個體中繼資料服務第 1 版，從執行個體中繼資料擷取有關執行個體之作用中維護事件的資訊。

IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/events/maintenance/scheduled
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/events/maintenance/scheduled
```

以下是範例輸出，其中包含排程系統重新開機事件的相關資訊 (格式為JSON格式)。

```
[
  {
    "NotBefore" : "21 Jan 2019 09:00:43 GMT",
    "Code" : "system-reboot",
    "Description" : "scheduled reboot",
    "EventId" : "instance-event-0d59937288b749b32",
    "NotAfter" : "21 Jan 2019 09:17:23 GMT",
    "State" : "active"
  }
]
```

使用執行個體中繼資料檢視執行個體已完成或已取消事件的事件歷史記錄

您可以使用執行個體中繼資料服務第 2 版或執行個體中繼資料服務第 1 版，從執行個體中繼資料擷取有關已完成或已取消之執行個體事件的資訊。

IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/events/maintenance/history
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/events/maintenance/
history
```

以下是範例輸出，其中包含已取消之系統重新開機事件的相關資訊，以及已完成的系統重新開機事件 (格式為JSON格式)。

```
[
  {
    "NotBefore" : "21 Jan 2019 09:00:43 GMT",
    "Code" : "system-reboot",
    "Description" : "[Canceled] scheduled reboot",
    "EventId" : "instance-event-0d59937288b749b32",
    "NotAfter" : "21 Jan 2019 09:17:23 GMT",
    "State" : "canceled"
  },
  {
    "NotBefore" : "29 Jan 2019 09:00:43 GMT",
    "Code" : "system-reboot",
    "Description" : "[Completed] scheduled reboot",
    "EventId" : "instance-event-0d59937288b749b32",
    "NotAfter" : "29 Jan 2019 09:17:23 GMT",
    "State" : "completed"
  }
]
```

AWS Health

您可以使用 AWS Health Dashboard 來瞭解可能會影響執行個體的事件。會將 AWS Health Dashboard 問題分為三個群組：未結問題、排程變更及其他通知。排定的變更包含進行中或即將進行的項目。

如需詳細資訊，請參閱《 指南》中的《AWS Health AWS Health Dashboard入門》。

為影響 Amazon EC2 執行個體的排程事件自訂電子郵件通知

您可以自訂排程事件通知，以便在電子郵件通知中包含標籤。如此可以更輕鬆地識別受影響的資源 (執行個體或 專用執行個體)，並為即將到來的事件排定動作的優先順序。

當您自訂事件通知以包含標籤時，您可以選擇包含：

- 與受影響資源關聯的所有標籤

- 僅與受影響資源關聯的特定標籤

例如，假設您將 `application`、`costcenter`、`project`，和 `owner` 標籤指派給所有執行個體。您可以選擇在事件通知中包含所有標籤。或者，如果您只想在事件通知中看到 `owner` 和 `project` 標籤，則可以選擇只包含這些標籤。

選取要包含的標籤之後，事件通知將包含資源 ID (執行個體 ID 或 專用執行個體 ID)，以及與受影響資源關聯的標籤金鑰和值組。

任務

- 在事件通知中包含標籤
- 從事件通知中移除標籤
- 檢視要包含在事件通知中的標籤

在事件通知中包含標籤

您選擇要包含的標籤會套用至所選區域的所有資源 (執行個體和 專用執行個體)。若要自訂其他區域中的事件通知，請先選取所需的區域，然後執行下列步驟。

您可以使用下列其中一種方法在事件通知中包含標籤。

Console

在事件通知中包含標籤

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Events (事件)。
3. 選擇 Actions (動作)、Manage event notifications (管理事件通知)。
4. 開啟在事件通知中包含標籤。
5. 根據您要包含在事件通知中的標籤，執行下列其中一項動作：
   - 若要包含與受影響執行個體或專用執行個體相關聯的所有標籤，請選取包含所有資源標籤。
   - 如需選取要包含的標籤，請選取選擇要包含的標籤，然後選取或輸入標籤索引鍵。
6. 選擇 Save (儲存)。

AWS CLI

在事件通知中包含所有標籤

使用 register-instance-event-notification-屬性 AWS CLI 指令並將`IncludeAllTagsOfInstance`參數設定為`true`。

```
aws ec2 register-instance-event-notification-attributes \
    --instance-tag-attribute "IncludeAllTagsOfInstance=true"
```

在事件通知中包含特定標籤

使用 register-instance-event-notification-attributes 指 AWS CLI 令，並使用`InstanceTagKeys`參數指定要包含的標籤。

```
aws ec2 register-instance-event-notification-attributes \
    --instance-tag-attribute 'InstanceTagKeys=["tag_key_1", "tag_key_2",
 "tag_key_3"]'
```

從事件通知中移除標籤

您可以使用下列其中一種方法將標籤從事件通知中移除。

Console

從事件通知中移除標籤

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Events (事件)。
3. 選擇 Actions (動作)、Manage event notifications (管理事件通知)。
4. 若要從事件通知中移除所有標籤，請關閉在事件通知中包含資源標籤。
5. 若要從事件通知中移除特定標籤，請為對應的標籤索引鍵選擇 X)。
6. 選擇 Save (儲存)。

AWS CLI

從事件通知中移除所有標籤

使用 deregister-instance-event-notification-屬性 AWS CLI 指令並將`IncludeAllTagsOfInstance`參數設定為`false`。

```
aws ec2 deregister-instance-event-notification-attributes \
    --instance-tag-attribute "IncludeAllTagsOfInstance=false"
```

從事件通知中移除特定標籤

使用 deregister-instance-event-notification-attributes 指 AWS CLI 令，並使用`InstanceTagKeys`參數指定要移除的標籤。

```
aws ec2 deregister-instance-event-notification-attributes \
    --instance-tag-attribute 'InstanceTagKeys=["tag_key_1", "tag_key_2",
 "tag_key_3"]'
```

檢視要包含在事件通知中的標籤

您可以使用下列其中一種方法來檢視要包含在事件通知中的標籤。

Console

檢視要包含在事件通知中的標籤

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Events (事件)。
3. 選擇 Actions (動作)、Manage event notifications (管理事件通知)。

AWS CLI

檢視要包含在事件通知中的標籤

使用 describe-instance-event-notification-屬性 AWS CLI 命令。

```
aws ec2 describe-instance-event-notification-attributes
```

重新排程影響 Amazon EC2 執行個體的排程事件

您可以重新排程事件，使其發生在適合您的特定日期和時間。只有具有截止日期的活動才能重新排程。有其他重新排程事件的限制。

您可以使用下列其中一種方法來重新排程事件。

Console

重新排程事件

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Events (事件)。
3. 從篩選清單中選擇 Resource type: instance (資源類型：執行個體)。
4. 選取一或多個執行個體，然後選擇 Actions (動作)、Schedule event (排程事件)。

   只能重新排程有事件截止日期 (由 Deadline (截止日期) 的值指出) 的事件。如果其中一個選取的事件沒有截止日期，則會停用 Actions (動作)、Schedule event (排程事件)。
5. 對於 New start time (新增開始時間)，輸入事件的新日期和時間。新的日期和時間必須發生在 Event deadline (事件截止日期) 之前。
6. 選擇 Save (儲存)。

   更新後的事件開始時間可能需要一兩分鐘才會反映在主控台中。

AWS CLI

重新排程事件

1. 只能重新排程有事件截止日期 (由 `NotBeforeDeadline` 的值指出) 的事件。使用describe-instance-status指令檢視`NotBeforeDeadline`參數值。

```
aws ec2 describe-instance-status \
    --instance-id i-1234567890abcdef0
```

   以下範例輸出顯示 `system-reboot` 事件，因為 `NotBeforeDeadline` 有值，所以可重新排程此事件。

```
[
    "Events": [
                {
                    "InstanceEventId": "instance-event-0d59937288b749b32",
                    "Code": "system-reboot",
                    "Description": "The instance is scheduled for a reboot",
                    "NotAfter": "2019-03-14T22:00:00.000Z",
```

```
                    "NotBefore": "2019-03-14T20:00:00.000Z",
                    "NotBeforeDeadline": "2019-04-05T11:00:00.000Z"
                }
            ]
]
```

2. 若要重新排程事件，請使用 modify-instance-event-starttime 命令。使用 `not-before` 參數指定新的事件開始時間。新的事件開始時間必須落在 `NotBeforeDeadline` 之前。

```
aws ec2 modify-instance-event-start-time \
    --instance-id i-1234567890abcdef0 \
    --instance-event-id instance-event-0d59937288b749b32 \
    --not-before 2019-03-25T10:00:00.000
```

describe-instance-status命令會傳回更新的`not-before`參數值之前，可能需要一兩分鐘的時間。

限制

- 只能重新排程有事件截止日期的事件。事件最晚可以重新排程到事件截止日期。主控台中的「截止日期`NotBeforeDeadline`」欄和中的欄位會 AWS CLI 指出事件是否有截止日期。
- 只能重新排程尚未開始的事件。主控台中的「開始時間`NotBefore`」欄和中的欄位 AWS CLI 表示事件開始時間。已排程在接下來 5 分鐘內開始的事件不能重新排程。
- 新的事件開始時間與目前時間至少必須相隔 60 分鐘。
- 如果您使用主控台重新排程多個事件，則由具有最早事件截止日期的事件決定事件截止日期。

為影響 Amazon EC2 執行個體的排程事件建立自訂事件視窗

您可以為重新開機、停止或終止 Amazon EC2 執行個體的排定事件定義每週重複的自訂事件視窗。您可以將一或多個執行個體關聯到事件時段。如果已計劃這些執行個體的排程事件， AWS 會在相關聯的事件時段中排程事件。

您可以使用事件時段，透過指定在工作負載離峰期間出現的事件時段，將工作負載可用性最大化。您也可以調整事件時段與內部維護排程。

可以透過指定一組時間範圍來定義事件時段。最短時間範圍為 2 小時。合併的時間範圍總計必須至少 4 小時。

您可以使用執行個體或執行個體標籤，將一IDs或多個執行個體與事件視窗產生關聯。您也可以使用主機 ID，將專用執行個體與事件時段建立關聯。

> **Warning**
>
> 事件時段僅適用於停止、重新啟動或終止執行個體的排程事件。
> 事件時段不適用於：
>
> - 快速排程事件和網路維護事件。
> - 不定期的維護，例如意外 AutoRecovery 重新啟動。

使用事件時段

- 考量事項
- 建立事件時段
- 檢視事件時段
- 修改事件時段
- 刪除事件時段
- 標記事件時段

考量事項

- 所有事件窗口時間都在UTC。
- 最短每週事件時段為 4 小時。
- 事件時段內的時間範圍必須至少為 2 小時。
- 只有一個目標類型 (執行個體 ID、專用執行個體 ID 或執行個體標籤) 可以與事件時段關聯。
- 一個目標類型 (執行個體 ID、專用執行個體 ID 或執行個體標籤) 只能與一個事件時段關聯。
- 一個事件視窗最多可以關聯 100 個執行個體或 50 個專用主機IDs，或 50 個執行個體標籤。IDs執行個體標籤可與任意數量的執行個體關聯。
- 每個 AWS 區域最多可建立 200 個事件視窗。
- 與事件時段相關聯的多個執行個體可能會同時發生排程事件。
- 如果 AWS 已排程事件，修改事件視窗不會變更已排程事件的時間。如果事件有截止日期，您可以重新排程事件。

- 您可以在排程事件之前停止並啟動執行個體，這樣會將執行個體遷移到新主機，而排程的事件將不再發生。

建立事件時段

您可以建立一或多個事件時段。對於每個事件時段，您可以指定一個或多個時間段。例如，您可以建立一個事件時段，其時間段為每天凌晨 4 點開始，持續 2 小時。或者，您可以建立一個事件時段，其時間段為星期日凌晨 2 點到凌晨 4 點和星期三凌晨 3 點到凌晨 5 點。

如需事件時段限制，請參閱本主題前面的考量事項。

事件時段會每週重複出現，直到您刪除為止。

使用下列其中一種方法來建立事件時段。

Console

建立事件時段

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Events (事件)。
3. 選擇 Actions (動作)、Manage event windows (管理事件時段)。
4. 選擇 Create instance event window (建立執行個體事件時段)。
5. 對於 Event window name (事件時段名稱)，輸入事件時段的描述性名稱。
6. 對於 Event window schedule (事件時段排程)，透過使用 cron 排程建置器或指定時間範圍，進行選擇以在事件時段中指定時間段。
   - 如果選擇 Cron schedule builder (Cron 排程建置器)，請指定以下內容：
     1. 對於 D ays (UTC)，請指定發生事件視窗的星期幾。
     2. 對於「開始時間」(UTC)，請指定事件視窗開始的時間。
     3. 對於 Duration (持續時間)，在事件時段中指定時間段的持續時間。每個時間段的最短持續時間為 2 小時。事件時段的最短持續時間總計必須等於或超過 4 小時。所有的時間都在 UTC。
   - 如果選擇 Time ranges (時間範圍)，請選擇 Add new time range (新增時間範圍)，然後指定開始日期和時間以及結束日期和時間。對每個時間範圍重複此步驟。每個時間範圍的最短持續時間為 2 小時。所有時間範圍合併的最小持續時間總計必須等於或超過 4 小時。

7. (選用) 對於 Target details (目標詳細資訊)，請將一個或多個執行個體與事件時段關聯，如果執行個體已排程進行維護，則排程的事件會在關聯的事件時段中發生。您可以使用執行個體或執行個體標籤，將一或多個執IDs行個體與事件視窗產生關聯。您可以使用主機 ID，將專用執行個體與事件時段關聯。

   請注意，您可以建立事件時段，而無需將目標與時段建立關聯。稍後，您可以修改時段以關聯一個或多個目標。
8. (選用) 對於 Event window tags (事件時段標籤)，選擇 Add tag (新增標籤)，然後輸入標籤的金鑰和值。針對每個標籤，重複執行上述動作。
9. 選擇 Create event window (建立事件時段)。

AWS CLI

若要使用建立事件視窗 AWS CLI，請先建立事件視窗，然後將一或多個目標與事件視窗相關聯。

建立事件時段

您可以在建立事件時段時定義一組時間範圍或 cron 表達式，但不能同時定義兩者。

建立具有時間範圍的事件時段

使用指create-instance-event-window令並指定`--time-range`參數。您不能指定 `--cron-expression` 參數。

```
aws ec2 create-instance-event-window \
    --region us-east-1 \
    --time-range StartWeekDay=monday,StartHour=2,EndWeekDay=wednesday,EndHour=8 \
    --tag-specifications "ResourceType=instance-event-
window,Tags=[{Key=K1,Value=V1}]" \
    --name myEventWindowName
```

預期的輸出結果

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "TimeRanges": [
            {
                "StartWeekDay": "monday",
                "StartHour": 2,
```

```
                "EndWeekDay": "wednesday",
                "EndHour": 8
            }
        ],
        "Name": "myEventWindowName",
        "State": "creating",
        "Tags": [
            {
                "Key": "K1",
                "Value": "V1"
            }
        ]
    }
}
```

透過 cron 表達式建立事件時段

使用指create-instance-event-window令並指定`--cron-expression`參數。您不能指定 `--time-range` 參數。

```
aws ec2 create-instance-event-window \
    --region us-east-1 \
    --cron-expression "* 21-23 * * 2,3" \
    --tag-specifications "ResourceType=instance-event-
window,Tags=[{Key=K1,Value=V1}]" \
    --name myEventWindowName
```

預期的輸出結果

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "Name": "myEventWindowName",
        "CronExpression": "* 21-23 * * 2,3",
        "State": "creating",
        "Tags": [
            {
                "Key": "K1",
                "Value": "V1"
            }
        ]
    }
```

```
}
```

將目標與事件時段關聯

您只能將一種類型的目標 (執行個體IDs、專用主機IDs或執行個體標記) 與事件視窗建立關聯。

將執行個體標籤與事件時段關聯

使用指associate-instance-event-window令並指定`instance-event-window-id`參數以指定事件視窗。若要關聯執行個體標籤，請指定 `--association-target` 參數，並針對參數值指定一個或多個標籤。

```
aws ec2 associate-instance-event-window \
    --region us-east-1 \
    --instance-event-window-id iew-0abcdef1234567890 \
    --association-target "InstanceTags=[{Key=k2,Value=v2},{Key=k1,Value=v1}]"
```

預期的輸出結果

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "Name": "myEventWindowName",
        "CronExpression": "* 21-23 * * 2,3",
        "AssociationTarget": {
            "InstanceIds": [],
            "Tags": [
                {
                    "Key": "k2",
                    "Value": "v2"
                },
                {
                    "Key": "k1",
                    "Value": "v1"
                }
            ],
            "DedicatedHostIds": []
        },
        "State": "creating"
    }
}
```

將一個或多個執行個體與事件時段關聯

使用指associate-instance-event-window令並指定instance-event-window-id參數以指定事件視窗。若要關聯例證，請指定--association-target參數，並為參數值指定一個或多個例證IDs。

```
aws ec2 associate-instance-event-window \
    --region us-east-1 \
    --instance-event-window-id iew-0abcdef1234567890 \
    --association-target "InstanceIds=i-1234567890abcdef0,i-0598c7d356eba48d7"
```

預期的輸出結果

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "Name": "myEventWindowName",
        "CronExpression": "* 21-23 * * 2,3",
        "AssociationTarget": {
            "InstanceIds": [
                "i-1234567890abcdef0",
                "i-0598c7d356eba48d7"
            ],
            "Tags": [],
            "DedicatedHostIds": []
        },
        "State": "creating"
    }
}
```

將專用執行個體與事件時段關聯

使用指associate-instance-event-window令並指定instance-event-window-id參數以指定事件視窗。若要關聯專用主機，請指定--association-target參數，並為參數值指定一或多個專用主機IDs。

```
aws ec2 associate-instance-event-window \
    --region us-east-1 \
    --instance-event-window-id iew-0abcdef1234567890 \
    --association-target "DedicatedHostIds=h-029fa35a02b99801d"
```

預期的輸出結果

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "Name": "myEventWindowName",
        "CronExpression": "* 21-23 * * 2,3",
        "AssociationTarget": {
            "InstanceIds": [],
            "Tags": [],
            "DedicatedHostIds": [
                "h-029fa35a02b99801d"
            ]
        },
        "State": "creating"
    }
}
```

檢視事件時段

您可以使用下列其中一種方法來檢視事件時段。

Console

檢視事件時段

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Events (事件)。
3. 選擇 Actions (動作)、Manage event windows (管理事件時段)。
4. 選取事件時段以檢視其詳細資訊。

AWS CLI

描述所有事件時段

使用指describe-instance-event-windows令。

```
aws ec2 describe-instance-event-windows \
    --region us-east-1
```

預期的輸出結果

```
{
    "InstanceEventWindows": [
        {
            "InstanceEventWindowId": "iew-0abcdef1234567890",
            "Name": "myEventWindowName",
            "CronExpression": "* 21-23 * * 2,3",
            "AssociationTarget": {
                "InstanceIds": [
                    "i-1234567890abcdef0",
                    "i-0598c7d356eba48d7"
                ],
                "Tags": [],
                "DedicatedHostIds": []
            },
            "State": "active",
            "Tags": []
        }

        ...

    ],
    "NextToken": "9d624e0c-388b-4862-a31e-a85c64fc1d4a"
}
```

描述所有特定事件時段

使用指describe-instance-event-windows令搭配`--instance-event-window-id`參數來描述特定事件視窗。

```
aws ec2 describe-instance-event-windows \
    --region us-east-1 \
    --instance-event-window-id iew-0abcdef1234567890
```

描述符合一個或多個篩選條件的事件時段

將describe-instance-event-windows指令與`--filters`參數搭配使用。在以下範例中，`instance-id` 篩選條件用於描述與指定執行個體相關聯的所有事件時段。

使用篩選條件時，它會執行直接比對。不過，`instance-id` 篩選條件是不同的。如果執行個體 ID 沒有直接匹配項，則會返回到與事件時段的間接關聯，例如執行個體的標籤或專用執行個體 ID (如果執行個體位於專用執行個體上)。

如需支援的篩選器清單，請參閱〈AWS CLI 參考〉describe-instance-event-windows中的〈〉。

```
aws ec2 describe-instance-event-windows \
    --region us-east-1 \
    --filters Name=instance-id,Values=i-1234567890abcdef0 \
    --max-results 100 \
    --next-token <next-token-value>
```

預期的輸出結果

在下列範例中，執行個體位於與事件時段相關聯的專用執行個體上。

```
{
    "InstanceEventWindows": [
        {
            "InstanceEventWindowId": "iew-0dbc0adb66f235982",
            "TimeRanges": [
                {
                    "StartWeekDay": "sunday",
                    "StartHour": 2,
                    "EndWeekDay": "sunday",
                    "EndHour": 8
                }
            ],
            "Name": "myEventWindowName",
            "AssociationTarget": {
                "InstanceIds": [],
                "Tags": [],
                "DedicatedHostIds": [
                    "h-0140d9a7ecbd102dd"
                ]
            },
            "State": "active",
            "Tags": []
        }
    ]
}
```

修改事件時段

您可以修改事件時段的所有欄位，但其 ID 除外。例如，當日光節約時間開始時，您可能想要修改事件時段排程。對於現有事件時段，您可能會想要新增或移除目標。

使用下列其中一種方法來修改事件時段。

Console

修改事件時段

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Events (事件)。
3. 選擇 Actions (動作)、Manage event windows (管理事件時段)。
4. 選取要修改的事件時段，然後選擇 Actions (動作)、Modify instance event window (修改執行個體事件時段)。
5. 修改事件時段中的欄位，然後選擇 Modify event window (修改事件時段)。

AWS CLI

若要使用修改事件視窗 AWS CLI，您可以修改時間範圍或 Cron 運算式，以及將一或多個目標與事件視窗產生關聯或取消關聯。

修改事件時段時間

您可以在修改事件時段時修改時間範圍或 cron 表達式，但不能同時修改兩者。

修改事件時段的時間範圍

使用指modify-instance-event-window令並指定要修改的事件視窗。指定 `--time-range` 參數來修改時間範圍。您不能指定 `--cron-expression` 參數。

```
aws ec2 modify-instance-event-window \
    --region us-east-1 \
    --instance-event-window-id iew-0abcdef1234567890 \
    --time-range StartWeekDay=monday,StartHour=2,EndWeekDay=wednesday,EndHour=8
```

預期的輸出結果

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "TimeRanges": [
            {
```

```
                "StartWeekDay": "monday",
                "StartHour": 2,
                "EndWeekDay": "wednesday",
                "EndHour": 8
            }
        ],
        "Name": "myEventWindowName",
        "AssociationTarget": {
            "InstanceIds": [
                "i-0abcdef1234567890",
                "i-0be35f9acb8ba01f0"
            ],
            "Tags": [],
            "DedicatedHostIds": []
        },
        "State": "creating",
        "Tags": [
            {
                "Key": "K1",
                "Value": "V1"
            }
        ]
    }
}
```

修改事件時段的一組時間範圍

使用指modify-instance-event-window令並指定要修改的事件視窗。指定 `--time-range` 參數來修改時間範圍。您不能指定相同呼叫中的 `--cron-expression` 參數。

```
aws ec2 modify-instance-event-window \
    --region us-east-1 \
    --instance-event-window-id iew-0abcdef1234567890 \
    --time-range '[{"StartWeekDay": "monday", "StartHour": 2, "EndWeekDay":
 "wednesday", "EndHour": 8},
   {"StartWeekDay": "thursday", "StartHour": 2, "EndWeekDay": "friday",
 "EndHour": 8}]'
```

預期的輸出結果

```
{
    "InstanceEventWindow": {
```

```
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "TimeRanges": [
            {
                "StartWeekDay": "monday",
                "StartHour": 2,
                "EndWeekDay": "wednesday",
                "EndHour": 8
            },
            {
                "StartWeekDay": "thursday",
                "StartHour": 2,
                "EndWeekDay": "friday",
                "EndHour": 8
            }
        ],
        "Name": "myEventWindowName",
        "AssociationTarget": {
            "InstanceIds": [
                "i-0abcdef1234567890",
                "i-0be35f9acb8ba01f0"
            ],
            "Tags": [],
            "DedicatedHostIds": []
        },
        "State": "creating",
        "Tags": [
            {
                "Key": "K1",
                "Value": "V1"
            }
        ]
    }
}
```

修改事件時段的 cron 表達式

使用指modify-instance-event-window令並指定要修改的事件視窗。指定 `--cron-expression` 參數來修改 cron 表達式。您不能指定 `--time-range` 參數。

```
aws ec2 modify-instance-event-window \
    --region us-east-1 \
    --instance-event-window-id iew-0abcdef1234567890 \
    --cron-expression "* 21-23 * * 2,3"
```

預期的輸出結果

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "Name": "myEventWindowName",
        "CronExpression": "* 21-23 * * 2,3",
        "AssociationTarget": {
            "InstanceIds": [
                "i-0abcdef1234567890",
                "i-0be35f9acb8ba01f0"
            ],
            "Tags": [],
            "DedicatedHostIds": []
        },
        "State": "creating",
        "Tags": [
            {
                "Key": "K1",
                "Value": "V1"
            }
        ]
    }
}
```

修改與事件時段相關聯的目標

您可以將其他目標與事件時段關聯。您也可以取消現有目標與事件時段的關聯。不過，只能將一種類型的目標 (執行個體IDsIDs、專用主機或執行個體標籤) 與事件視窗產生關聯。

將其他目標與事件時段關聯

如需如何將目標與事件時段產生關聯的指示，請參閱Associate a target with an event window。

解除執行個體標籤與事件時段的關聯

使用指disassociate-instance-event-window令並指定`instance-event-window-id`參數以指定事件視窗。若要解除執行個體標籤的關聯，請指定 `--association-target` 參數，並針對參數值指定一個或多個標籤。

```
aws ec2 disassociate-instance-event-window \
    --region us-east-1 \
```

```
    --instance-event-window-id iew-0abcdef1234567890 \
    --association-target "InstanceTags=[{Key=k2,Value=v2},{Key=k1,Value=v1}]"
```

預期的輸出結果

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "Name": "myEventWindowName",
        "CronExpression": "* 21-23 * * 2,3",
        "AssociationTarget": {
            "InstanceIds": [],
            "Tags": [],
            "DedicatedHostIds": []
        },
        "State": "creating"
    }
}
```

解除一個或多個執行個體與事件時段的關聯

使用指disassociate-instance-event-window令並指定`instance-event-window-id`參數以指定事件視窗。若要取消例證的關聯，請指定`--association-target`參數，並為參數值指定一個或多個例證IDs。

```
aws ec2 disassociate-instance-event-window \
    --region us-east-1 \
    --instance-event-window-id iew-0abcdef1234567890 \
    --association-target "InstanceIds=i-1234567890abcdef0,i-0598c7d356eba48d7"
```

預期的輸出結果

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "Name": "myEventWindowName",
        "CronExpression": "* 21-23 * * 2,3",
        "AssociationTarget": {
            "InstanceIds": [],
            "Tags": [],
            "DedicatedHostIds": []
```

```
        },
        "State": "creating"
    }
}
```

解除專用執行個體與事件時段的關聯

使用指disassociate-instance-event-window令並指定instance-event-window-id參數以指定事件視窗。若要取消專用主機的關聯，請指定--association-target參數，並為參數值指定一或多個專用主機IDs。

```
aws ec2 disassociate-instance-event-window \
    --region us-east-1 \
    --instance-event-window-id iew-0abcdef1234567890 \
    --association-target DedicatedHostIds=h-029fa35a02b99801d
```

預期的輸出結果

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "Name": "myEventWindowName",
        "CronExpression": "* 21-23 * * 2,3",
        "AssociationTarget": {
            "InstanceIds": [],
            "Tags": [],
            "DedicatedHostIds": []
        },
        "State": "creating"
    }
}
```

刪除事件時段

您可以使用下列其中一種方法，一次刪除一個事件時段。

Console

刪除事件時段

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。

2. 在導覽窗格中，選擇 Events (事件)。
3. 選擇 Actions (動作)、Manage event windows (管理事件時段)。
4. 選取要刪除的事件時段，然後選擇 Actions (動作)、Delete instance event window (刪除執行個體事件時段)。
5. 當出現提示時，輸入 **`delete`**，然後選擇 Delete (刪除)。

AWS CLI

刪除事件時段

使用指delete-instance-event-window令並指定要刪除的事件視窗。

```
aws ec2 delete-instance-event-window \
    --region us-east-1 \
    --instance-event-window-id iew-0abcdef1234567890
```

強制刪除事件時段

使用 `--force-delete` 參數 (如果事件時段目前與目標相關聯)。

```
aws ec2 delete-instance-event-window \
    --region us-east-1 \
    --instance-event-window-id iew-0abcdef1234567890 \
    --force-delete
```

預期的輸出結果

```
{
    "InstanceEventWindowState": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "State": "deleting"
    }
}
```

標記事件時段

您可以在建立事件時段時或之後對其進行標記。

若要在建立事件時段時進行標記，請參閱建立事件時段。

使用下列其中一種方法來標記事件時段。

Console

標記現有事件時段

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Events (事件)。
3. 選擇 Actions (動作)、Manage event windows (管理事件時段)。
4. 選取要標記的事件時段，然後選擇 Actions (動作)、Manage instance event window tags (管理執行個體事件時段標籤)。
5. 選擇 Add tag (新增標籤) 以新增標籤。針對每個標籤，重複執行上述動作。
6. 選擇 Save (儲存)。

AWS CLI

標記現有事件時段

使用建立標籤命令來標記現有資源。在下面的例子中，現有事件時段標記為 Key=purpose 和 Value=test。

```
aws ec2 create-tags \
    --resources iew-0abcdef1234567890 \
    --tags Key=purpose,Value=test
```

使用監控執行個體 CloudWatch

您可以使用 Amazon 監控執行個體 CloudWatch，Amazon 將來自 Amazon 的原始資料收集並處理 EC2為可讀且接近即時的指標。這些統計資料會記錄 15 個月的時間，以便您存取歷史資訊，並更清楚 Web 應用程式或服務的執行效能。

根據預設，Amazon EC2 會 CloudWatch 在 5 分鐘內將指標資料傳送到。若要 CloudWatch 在 1 分鐘內將執行個體的指標資料傳送到，您可以啟用執行個體的詳細監控。如需詳細資訊，請參閱管理EC2執行個體的詳細監控。

Amazon EC2 控制台根據來自 Amazon 的原始數據顯示一系列圖形 CloudWatch。根據您的需求，您可能希望從 Amazon 取得執行個體的資料， CloudWatch 而不是主控台中的圖形。

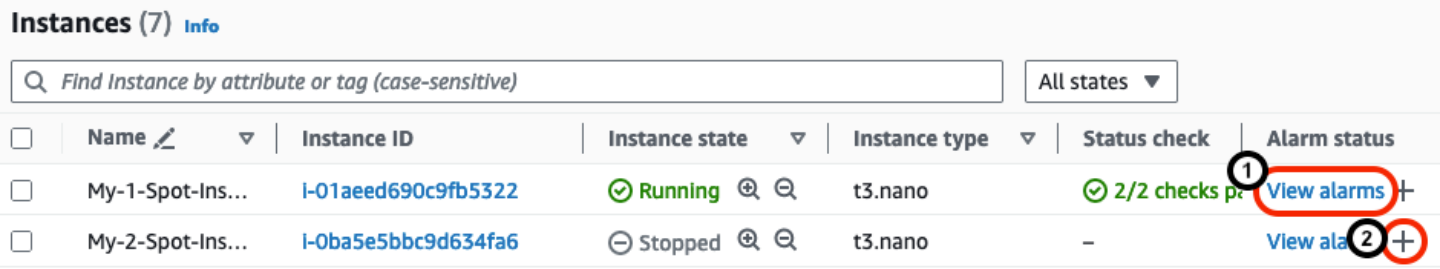
如需 Amazon CloudWatch 帳單和費用資訊，請參閱 Amazon CloudWatch 使用者指南中的 CloudWatch 帳單和費用。

目錄

- 在 Amazon EC2 主控台中管理EC2執行個體的 CloudWatch 警示
- 管理EC2執行個體的詳細監控
- CloudWatch 您執行個體可用的指標
- 使用 Amazon EC2 主控台安裝和設定 CloudWatch代理程式，以新增其他指標
- 執行個體的 CloudWatch 指標統計資料
- 檢視執行個體的監控圖表
- 建立執行個體的 CloudWatch 警示
- 建立警示以停止、終止、重新啟動或復原執行個體

在 Amazon EC2 主控台中管理EC2執行個體的 CloudWatch 警示

您可以從 Amazon EC2 主控台的執行個體畫面管理執行個體的 Amazon CloudWatch 警示。在「執行個體」表格中，「警示狀態」欄提供兩個主控台控制項：用於檢視警示的控制項，另一個用於建立或編輯警示的控制項。下列螢幕擷取畫面指出這些控制台控制項，編號為 1 (檢視警示) 和 2 (用於建立或編輯警示的 + 號)。



從執行個體畫面檢視警示

您可以從「執行個體」畫面檢視每個執行個體的警報。

從執行個體畫面檢視執行個體的警示

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 在「執行個體」表格中，針對您選擇的執行個體選擇 [檢視鬧鐘] (在前面螢幕擷取畫面中編號為 1)

4. 在「警報」詳細資訊中 ***i-0123456789example***視窗中，選擇警示名稱，以便在 CloudWatch 控制台中檢視鬧鐘。

從執行個體畫面建立警示

您可以從「執行個體」畫面為每個執行個體建立警示。

從執行個體畫面建立執行個體的警示

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 在「執行個體」表格中，為您選擇的執行個體選擇加號 (前面螢幕擷取畫面中的編號為 2)。
4. 在「管理 CloudWatch 鬧鐘」畫面中，建立鬧鐘。如需詳細資訊，請參閱建立執行個體的 CloudWatch 警示。

從執行個體畫面編輯警示

您可以從「執行個體」畫面編輯執行個體的警示。

從執行個體畫面編輯執行個體的警示

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 在「執行個體」表格中，為您選擇的執行個體選擇加號 (前面螢幕擷取畫面中的編號為 2)。
4. 在「管理 CloudWatch 鬧鐘」畫面中，編輯鬧鐘。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的編輯或刪除 CloudWatch 警示。

管理EC2執行個體的詳細監控

Amazon CloudWatch 提供兩種監控類別：基本監控和詳細監控。根據預設，您的執行個體設定為基本監控。您可以選擇性地啟用詳細監控，以協助您更快地識別作業問題並採取行動。您可以在啟動或執行個體執行或停止時啟用或關閉詳細監控。

對執行個體啟用詳細監控並不會影響其連接EBS磁碟區的監視。如需詳細資訊，請參閱 Amazon 的 Amazon CloudWatch 指標EBS。

下表重點說明基本監視與執行個體的詳細監視之間的差異。

監控類型	描述	費用
基本監控	狀態檢查量度可在 1 分鐘內使用。所有其他指標均在 5 分鐘內可用。	免費。
詳細監控	所有指標 (包括狀態檢查指標) 在 1 分鐘內可用。若要取得此層級的資料，您必須特別啟用執行個體的這項功能。對於已啟用詳細監控的執行個體，您亦可取得類似執行個體的跨群組彙總資料。	您需要按 Amazon EC2 傳送的指標向您收費 CloudWatch。您不需為資料儲存付費。如需詳細資訊，請參閱 Amazon CloudWatch 定價頁面上的付費方案和範例 1-EC2 詳細監控。

目錄

- 必要許可
- 啟動時啟用詳細監控
- 管理詳細監控

必要許可

若要啟用執行個體的詳細監控，您的使用者必須擁有使用該MonitorInstancesAPI動作的權限。若要關閉執行個體的詳細監控，您的使用者必須擁有使用該UnmonitorInstancesAPI動作的權限。

啟動時啟用詳細監控

請使用下列程序在啟動時啟用詳細監視。根據預設，您的執行個體使用基本監控。

Console

在執行個體啟動時啟用詳細監控

使用 Amazon EC2 主控台啟動執行個體時，在 [進階詳細資料] 下，選取 [詳細 CloudWatch 監控] 核取方塊。

AWS CLI

在執行個體啟動時啟用詳細監控

使用 run-instances 命令搭配 `--monitoring` 旗標來啟用特定執行個體的詳細監控。

```
aws ec2 run-instances --image-id ami-09092360 --monitoring Enabled=true...
```

管理詳細監控

使用下列程序來管理執行中或已停止之執行處理的詳細監督。

Console

管理執行個體的詳細監控

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體。
4. 選擇操作，監視和故障排除，管理詳細監視。
5. 在 [詳細監視] 頁面上，對於 [詳細監視]，執行下列其中一個動作：
   - 詳細監控 — 選取 [啟用]。
   - 基本監控 — 清除啟用。
6. 選擇確認。

AWS CLI

啟用執行個體的詳細監控

使用下列 monitor-instances 命令來啟用特定執行個體的詳細監控。

```
aws ec2 monitor-instances --instance-ids i-1234567890abcdef0
```

關閉執行個體的詳細監控

使用下列 unmonitor-instances 命令來關閉特定執行個體的詳細監控。

```
aws ec2 unmonitor-instances --instance-ids i-1234567890abcdef0
```

CloudWatch 您執行個體可用的指標

Amazon EC2 將指標發送到 Amazon CloudWatch。您可以使用 AWS Management Console AWS CLI、或列出 Amazon EC2 傳送到的指標 CloudWatch。API每個資料點預設會涵蓋執行個體從啟動時間後 5 分鐘內的活動。若您已啟用詳細監控功能，則每個資料點會涵蓋從啟動時間後下一分鐘的活動。請注意，對於「最小值」、「最大值」和「平均」統計資料，EC2提供的測量結果的最小粒度為 1 分鐘。

有關如何使用 AWS Management Console 或檢視可用指標的詳細資訊 AWS CLI，請參閱 Amazon 使用者指南中的檢視可 CloudWatch 用指標。

如需取得這些指標統計資料的資訊，請參閱執行個體的 CloudWatch 指標統計資料。

內容

- 執行個體指標
- CPU信貸指標
- 專用執行個體指標
- 硝基型執行個體的 Amazon EBS 指標
- 狀態檢查指標
- 流量鏡像指標
- Auto Scaling 群組指標
- Amazon 度EC2量維度
- Amazon EC2 用量指標

執行個體指標

`AWS/EC2` 命名空間包含下列執行個體指標。

指標	描述	單位	有意義的統計資料
`CPUUtilization`	Amazon EC2 用於執行執行EC2個體的實體CPU時間百分比，包括執行使用者程式碼和 Amazon EC2 程式碼所花費的時間。	百分比	• 平均數 • 下限 • 最大

指標	描述	單位	有意義的統計資料
	整體而言，CPUUtilization 是訪客 CPUUtilization 和 Hypervisor CPUUtilization 的總和。 CloudWatch 由於傳統裝置模擬、非舊式裝置的組態、大量中斷的工作負載、即時移轉和即時更新等因素，作業系統中的工具顯示的百分比可能會有所不同。		
DiskReadOps	在指定期間，執行個體可用之所有執行個體存放區磁碟區的已完成讀取操作。 若要計算期間的平均每秒 I/O 作業 (IOPS)，請將該期間的總作業除以該期間的秒數。 如果沒有執行個體存放區磁碟區，數值可能為 0 或指標未回報。	計數	• 總和 • 平均數 • 下限 • 最大
DiskWriteOps	在指定期間，執行個體可用之所有執行個體存放區磁碟區的已完成寫入操作。 若要計算期間的平均每秒 I/O 作業 (IOPS)，請將該期間的總作業除以該期間的秒數。 如果沒有執行個體存放區磁碟區，數值可能為 0 或指標未回報。	計數	• 總和 • 平均數 • 下限 • 最大

指標	描述	單位	有意義的統計資料
`DiskReadBytes`	從執行個體可用之所有執行個體存放區磁碟區讀取的位元組。 此指標用來判斷應用程式從執行個體硬碟中讀取的資料磁碟區。這可以用來判斷應用程式的速度。 所報告的數目是在期間內收到的位元組總數。如果您要使用基本 (5 分鐘) 監控，則可以將此數字除以 300，以找到位元組數/秒。如果您具有詳細 (1 分鐘) 監控，請將它除以 60。您也可以使用度 CloudWatch 量數學函數`DIFF_TIME` 來尋找每秒的位元組數。例如，如果您已繪製 CloudWatch 為圖形`m1`，度量數學公式會`m1/(DIFF_TIME(m1))` 傳回以`DiskReadBytes` 位元組/秒為單位的量度。如需有關以`DIFF_TIME` 及其他度量數學函數的詳細資訊，請參閱 Amazon 使用 CloudWatch 者指南中的使用指標數學運算。 如果沒有執行個體存放區磁碟區，數值可能為 0 或指標未回報。	位元組	• 總和 • 平均數 • 下限 • 最大

指標	描述	單位	有意義的統計資料
DiskWriteBytes	寫入至執行個體可用之所有執行個體存放區磁碟區的位元組。 此指標用來判斷應用程式寫入至執行個體硬碟中的資料磁碟區。這可以用來判斷應用程式的速度。 所報告的數目是在期間內收到的位元組總數。如果您要使用基本 (5 分鐘) 監控，則可以將此數字除以 300，以找到位元組數/秒。如果您具有詳細 (1 分鐘) 監控，請將它除以 60。您也可以使用度 CloudWatch 量數學函數DIFF_TIME 來尋找每秒的位元組數。例如，如果您已繪製 CloudWatch為圖形m1，度量數學公式會m1/(DIFF_TIME(m1)) 傳回以DiskWriteBytes 位元組/秒為單位的量度。如需有關以DIFF_TIME 及其他度量數學函數的詳細資訊，請參閱 Amazon 使用 CloudWatch 者指南中的使用指標數學運算。 如果沒有執行個體存放區磁碟區，數值可能為 0 或指標未回報。	位元組	• 總和 • 平均數 • 下限 • 最大
MetadataNoToken	使用不使用 Token 的方法成功存取「執行個體中繼資料服務」(IMDS) 的次數。 此測量結果可用來判斷是否有任何程序存取執行個體中繼資料 (使用「執行處理中繼資料服務版本 1」(IMDSv1)，而這些程序不使用 Token。如果所有要求都使用憑證支援的工作階段，也就是執行個體中繼資料服務版本 2 (IMDSv2)，則值為 0。如需詳細資訊，請參閱轉換為使用 執行個體中繼資料服務第 2 版。	計數	• 總和 • 百分位數

指標	描述	單位	有意義的統計資料
MetadataNoTokenRejected	停用後IMDSv1嘗試IMDSv1通話的次數。 如果出現此指標，則表示已嘗試IMDSv1呼叫並被拒絕。您可以重新啟用IMDSv1或確保所有通話都使用IMDSv2。如需詳細資訊，請參閱轉換為使用 執行個體中繼資料服務第 2 版。	計數	• 總和 • 百分位數
NetworkIn	執行個體在所有網路介面上收到的位元組數目。此指標識別流向單一執行個體之傳入網路流量的磁碟區。 所報告的數目是在期間內收到的位元組總數。如果您要使用基本 (5 分鐘) 監控並且統計數字為總和，則可以將此數字除以 300，以找出每秒的位元組數。如果您具有詳細 (1 分鐘) 監控並且統計數字為總和，請將它除以 60。您也可以使用度 CloudWatch量數學函數DIFF_TIME 來尋找每秒的位元組數。例如，如果您已繪製 CloudWatch 為圖形m1，度量數學公式會m1/(DIFF_TIME(m1)) 傳回以NetworkIn 位元組/秒為單位的量度。如需有關以DIFF_TIME 及其他度量數學函數的詳細資訊，請參閱 Amazon 使用 CloudWatch 者指南中的使用指標數學運算。	位元組	• 總和 • 平均數 • 下限 • 最大

指標	描述	單位	有意義的統計資料
NetworkOut	執行個體在所有網路介面上送出的位元組數目。此指標識別來自單一執行個體之傳出網路流量的磁碟區。 所報告的數目是在期間內送出的位元組總數。如果您要使用基本 (5 分鐘) 監控並且統計數字為總和，則可以將此數字除以 300，以找出每秒的位元組數。如果您具有詳細 (1 分鐘) 監控並且統計數字為總和，請將它除以 60。您也可以使用度 CloudWatch 量數學函數DIFF_TIME 來尋找每秒的位元組數。例如，如果您已繪製 CloudWatch 為圖形m1，度量數學公式會m1/(DIFF_TIME(m1)) 傳回以NetworkOut 位元組/秒為單位的量度。如需有關以DIFF_TIME 及其他度量數學函數的詳細資訊，請參閱 Amazon 使用 CloudWatch 者指南中的使用指標數學運算。	位元組	• 總和 • 平均數 • 下限 • 最大
NetworkPacketsIn	執行個體在所有網路介面上收到的封包數目。此指標識別單一執行個體上的傳入流量 (封包數目)。 此指標僅適用於基本監控 (每隔 5 分鐘)。若要計算執行個體在 5 分鐘內每秒收到的封包數 (PPS)，請將總和統計值除以 300。您也可以使用 CloudWatch 公制數學函數DIFF_TIME 來尋找每秒封包數。例如，如果您已繪製 CloudWatch 為圖形 NetworkPacketsIn m1，則度量數學公式會m1/(DIFF_TIME(m1)) 傳回以封包/秒為單位的量度。如需有關以DIFF_TIME 及其他度量數學函數的詳細資訊，請參閱 Amazon 使用 CloudWatch 者指南中的使用指標數學運算。	計數	• 總和 • 平均數 • 下限 • 最大

指標	描述	單位	有意義的統計資料
NetworkPacketsOut	執行個體在所有網路介面上送出的封包數目。此指標識別單一執行個體上的傳出流量 (封包數目)。 此指標僅適用於基本監控 (每隔 5 分鐘)。若要計算執行個體在 5 分鐘內每秒傳送的封包數 (PPS)，請將總和統計值除以 300。您也可以使用 CloudWatch 公制數學函數DIFF_TIME 來尋找每秒封包數。例如，如果您已繪製 CloudWatch 為圖形 NetworkPacketsOut m1，則度量數學公式會m1/(DIFF_TIME(m1)) 傳回以封包/秒為單位的量度。如需有關以DIFF_TIME 及其他度量數學函數的詳細資訊，請參閱 Amazon 使用 CloudWatch 者指南中的使用指標數學運算。	計數	• 總和 • 平均數 • 下限 • 最大

CPU信貸指標

AWS/EC2命名空間包含下列適用於高載效能執行個體的CPU信用指標。

指標	描述	單位	有意義的統計資料
CPUCreditUsage	執行處理CPU用於使用的CPU點數。一個點CPU數等於以 100% 使用率CPU執行一分鐘的一 vvCPUs，或同等的使用率與時間組合 (例如，一個 v 以 50% 使用率CPU執行兩分鐘，或以 25% 使用率 vCPUs 執行兩分鐘兩分鐘)。 CPU信用指標的頻率僅為 5 分鐘。如果您要指定大於 5 分鐘的期間，請使用 Sum 統計資訊代替 Average 統計資訊。	學分 (v CPU-分鐘)	• 總和 • 平均數 • 下限 • 最大

指標	描述	單位	有意義的統計資料
CPUCredit Balance	執行個體啟動或啟動後累CPU積的已獲得積分數。如果是 T2 Standard，CPUCredit Balance 也包含已產生的啟動額度。 獲得額度後，額度會在額度餘額中累積，並在支付額度時，從額度餘額中移出。額度餘額有最大值限制，它取決於執行個體大小。到達限制之後，任何獲得的新額度都會遭到捨棄。如果是 T2 Standard，啟動額度不會計入此限制。 執行個體中CPUCreditBalance 的積分可用於超出其基準CPU使用率的範圍。 當執行個體執行時，CPUCreditBalance 中的額度不會過期。當 T3 或 T3a 執行個體停止時，CPUCreditBalance 值會持續保存七天。因此，所有累積的額度都會消失。當 T2 執行個體停止時，CPUCreditBalance 值不會持續保存，而且所有累積的額度都將消失。 CPU信用指標的頻率僅為 5 分鐘。	學分 (v CPU-分鐘)	• 總和 • 平均數 • 下限 • 最大
CPUSurplu sCreditBa lance	當 unlimited 執行個體的 CPUCredit Balance 值為 0 時，該執行個體已支出的剩餘額度數量。 該CPUSurplusCreditBalance 值由賺取的CPU積分支付。如果剩餘額度超過執行個體在 24 小時期間可獲得的最大額度數量，超過最大值的支出剩餘額度將必須負擔額外的費用。 CPU信用指標的頻率僅為 5 分鐘。	學分 (v CPU-分鐘)	• 總和 • 平均數 • 下限 • 最大

指標	描述	單位	有意義的統計資料
CPUSurplusCreditsCharged	所賺取的積分未支付的已支付剩餘CPU點數，因此會產生額外費用。 發生以下任何情況時，將收取支出剩餘額度的費用。 • 支出剩餘額度超過執行個體在 24 小時期間可獲得的最大額度數量。在小時結束時，將收取超過最大值的支出剩餘額度的費用。 • 執行個體已停止或終止。 • 執行個體從 unlimited 切換至 standard。 CPU信用指標的頻率僅為 5 分鐘。	學分 (v CPU-分鐘)	• 總和 • 平均數 • 下限 • 最大

專用執行個體指標

AWS/EC2 命名空間包含下列 T3 專用執行個體的指標。

指標	描述	單位	有意義的統計資料
DedicatedHostCPUUtilization	專用執行個體上執行的執行個體目前正在使用的已配置運算容量百分比。	百分比	• 總和 • 平均數 • 下限 • 最大

硝基型執行個體的 Amazon EBS 指標

AWS/EC2命名空間包含附加至非裸機執行個體之 Nitro-based 執行個體的磁碟區的其他 Amazon EBS 指標。

指標	描述	單位	有意義的統計資料
EBSReadOps	在指定時間段內，從連接到執行個體的所有 Amazon EBS 磁碟區完成讀取操作。 若要計算期間內每秒的平均讀取 I/O 作業 (讀取IOPS)，請將該期間的總作業除以該期間的秒數。如果您使用的是基本（5 分鐘）監視，則可以將此數字除以 300 以計算讀取 IOPS。如果您具有詳細 (1 分鐘) 監控，請將它除以 60。您也可以使用 CloudWatch 公制數學函數DIFF_TIME 來尋找每秒運算數。例如，如果您已繪製 CloudWatch 為圖形 EBSReadOps m1，則度量數學公式會m1/(DIFF_TIME(m1)) 傳回運算/秒的量度。如需有關以DIFF_TIME 及其他度量數學函數的詳細資訊，請參閱 Amazon 使用 CloudWatch 者指南中的使用指標數學運算。	計數	• 總和 • 平均數 • 下限 • 最大
EBSWriteOps	在指定時間內，完成對連接至執行個體的所有 EBS磁碟區的寫入作業。 若要計算期間的平均每秒寫入 I/O 作業 (WriteIOPS)，請將該期間的總作業除以該期間的秒數。如果您使用的是基本（5 分鐘）監視，則可以將此數字除以 300 以計算寫入 IOPS。如果您具有詳細 (1 分鐘) 監控，請將它除以 60。您也可以使用 CloudWatch 公制數學函數DIFF_TIME 來尋找每秒運算數。例如，如果您已繪製 CloudWatch 為圖形 EBSWriteOps m1，則度量數學公式會m1/(DIFF_TIME(m1)) 傳回運算/秒的量度。如需有關以DIFF_TIME 及其他度量數學函數的詳細資訊，請參閱 Amazon 使用 CloudWatch 者指南中的使用指標數學運算。	計數	• 總和 • 平均數 • 下限 • 最大

指標	描述	單位	有意義的統計資料
EBSReadBytes	在指定時間內，從連接到執行個體的所有EBS磁碟區讀取位元組。 所報告的數目是在該期間內讀取的位元組總數。如果您正使用基本 (5 分鐘) 監控，則可以將此數字除以 300，以得到所讀取的位元組數/秒。如果您具有詳細 (1 分鐘) 監控，請將它除以 60。您也可以使用度 CloudWatch 量數學函數DIFF_TIME 來尋找每秒的位元組數。例如，如果您已繪製 CloudWatch 為圖形m1，度量數學公式會m1/(DIFF_TIME(m1)) 傳回以EBSReadBytes 位元組/秒為單位的量度。如需有關以DIFF_TIME 及其他度量數學函數的詳細資訊，請參閱 Amazon 使用 CloudWatch 者指南中的使用指標數學運算。	位元組	• 總和 • 平均數 • 下限 • 最大
EBSWriteBytes	在指定時間段內寫入連接至執行個體的所有EBS磁碟區的位元組。 所報告的數目是在該期間內寫入的位元組總數。如果您正使用基本 (5 分鐘) 監控，則可以將此數字除以 300，得到所寫入的位元組數/秒。如果您具有詳細 (1 分鐘) 監控，請將它除以 60。您也可以使用度 CloudWatch 量數學函數DIFF_TIME 來尋找每秒的位元組數。例如，如果您已繪製 CloudWatch 為圖形m1，度量數學公式會m1/(DIFF_TIME(m1)) 傳回以EBSWriteBytes 位元組/秒為單位的量度。如需有關以DIFF_TIME 及其他度量數學函數的詳細資訊，請參閱 Amazon 使用 CloudWatch 者指南中的使用指標數學運算。	位元組	• 總和 • 平均數 • 下限 • 最大

指標	描述	單位	有意義的統計資料
EBSIOBalance%	提供叢發儲存貯體中剩餘 I/O 額度百分比資訊。只有基本監控才提供此指標。 此指標僅適用於一些大小為 *.4xlarge 及更小的執行個體，至少每 24 小時維持最佳效能 30 分鐘。 Sum 統計資料不適用於此指標。	百分比	• 下限 • 最大
EBSByteBalance%	提供叢發儲存貯體中剩餘傳輸量額度百分比的資訊。只有基本監控才提供此指標。 此指標僅適用於一些大小為 *.4xlarge 及更小的執行個體，至少每 24 小時維持最佳效能 30 分鐘。 Sum 統計資料不適用於此指標。	百分比	• 下限 • 最大

如需為EBS磁碟區提供的指標的相關資訊，請參閱 Amazon EBS使用者指南中的 Amazon EBS 磁碟區指標。如需為您的EC2叢集和競價型叢集提供的指標的相關資訊，請參閱。使用以下方式監控您的車EC2隊或 Spot CloudWatch

狀態檢查指標

根據預設，狀態檢查指標的頻率為 1 分鐘，而且免費。對於新啟動的執行個體，只有在執行個體完成初始化狀態 (在執行個體進入 running 狀態的幾分鐘內) 之後，才能使用狀態檢查指標資料。如需EC2狀態檢查的詳細資訊，請參閱Amazon EC2 執行個體的狀態檢查。

AWS/EC2 命名空間包含下列狀態檢查指標。

指標	描述	單位	有意義的統計資料
StatusCheckFailed	報告執行個體在過去一分鐘是否已通過所有狀態檢查。	計數	• 總和 • 平均數

指標	描述	單位	有意義的統計資料
	此指標可以是 0 (通過) 或 1 (失敗)。 根據預設，此指標的頻率為 1 分鐘，而且免費。		
StatusCheckFailed_Instance	報告執行個體是否在最後一分鐘通過執行個體狀態檢查。 此指標可以是 0 (通過) 或 1 (失敗)。 根據預設，此指標的頻率為 1 分鐘，而且免費。	計數	• 總和 • 平均數
StatusCheckFailed_System	報告執行個體是否在最後一分鐘通過系統狀態檢查。 此指標可以是 0 (通過) 或 1 (失敗)。 根據預設，此指標的頻率為 1 分鐘，而且免費。	計數	• 總和 • 平均數
StatusCheckFailed_AttachedEBS	報告執行個體在過去一分鐘是否已通過附加EBS狀態檢查。 此指標可以是 0 (通過) 或 1 (失敗)。 根據預設，此指標的頻率為 1 分鐘，而且免費。	計數	• 總和 • 平均數

AWS/EBS命名空間包含下列狀態檢查測量結果。

指標	描述	單位	有意義的統計資料
VolumeStalledIOCheck	注意：僅適用於 Nitro 執行個體。未針對連接到 Amazon ECS 和 AWS Fargate 任務的磁碟區發佈。 在最後一分鐘報告磁碟區停止的 IO 檢查是通過還是失敗。此指標可以是 0 (通過) 或 1 (失敗)。	無	• 總和 • 平均數 • 下限 • 最大

流量鏡像指標

AWS/EC2 命名空間包含鏡像流量的指標。如需詳細資訊，請參閱 Amazon 流量鏡像指南 CloudWatch 中的使用 Amazon 監控鏡像VPC流量。

Auto Scaling 群組指標

AWS/AutoScaling 命名空間包含 Auto Scaling 組的指標。如需詳細資訊，請參閱 Amazon Auto Scaling 使用者指南中的監控 EC2 Auto Scaling 群組和執行個體的指 CloudWatch 標。

Amazon 度EC2量維度

您可以使用下列維度來強化上表中列出的指標。

維度	描述
AutoScalingGroupName	此維度可篩選您為指定容量群組中所有執行個體請求的資料。如果您使用 Auto Scaling，則「Auto Scaling 群組」是您定義的執行個體集合。只有當執行個體位於此類 Auto Scaling 群組時，此維度才適用於 Amazon EC2 指標。適用於已啟用詳細或基本監控的執行個體。
ImageId	此維度會針對執行此 EC2 Amazon Machine 映像檔 (AMI) 的所有執行個體篩選您請求的資料。適用於已啟用詳細監控的執行個體。
InstanceId	此維度會篩選僅針對已識別執行個體所請求的資料。如此可協助您找出要監控資料的確切執行個體。
InstanceType	此維度可篩選您為以此指定執行個體類型執行的所有執行個體請求的資料。如此可協助您依據執行的執行個體類型將您的資料進行分類。例如，您可以比較來自 m1.small 執行個體與 m1.large 執行個體的資料，判斷哪個執行個體對您的應用程式而言具有較高的商業價值。適用於已啟用詳細監控的執行個體。

Amazon EC2 用量指標

您可以使用 CloudWatch 使用量度來提供您帳戶資源使用情況的可見度。使用這些指標，在 CloudWatch 圖形和儀表板上視覺化您目前的服務使用情況。

Amazon EC2 使用量指標對應於 AWS 服務配額。您可以設定警示，在您的用量接近服務配額時發出警示。如需與服務配額 CloudWatch整合的詳細資訊，請參閱 Amazon AWS 使用 CloudWatch 者指南中的使用量指標。

Amazon EC2 會在AWS/Usage命名空間中發佈下列指標。

指標	描述
ResourceCount	您的帳戶中正在執行的特定資源數量。資源由與指標相關聯的維度定義。 此指標最有用的統計資料是 MAXIMUM，它表示 1 分鐘內使用的最大資源數量。

下列維度用於精簡 Amazon 發佈的使用量指標EC2。

維度	描述
Service	包含資源的 AWS 服務名稱。對於 Amazon EC2 使用量指標，此維度的值為EC2。
Type	正在報告的實體類型。目前，Amazon EC2 使用量指標的唯一有效值是Resource。
Resource	正在執行的資源類型。目前 Amazon EC2 使用量指標的唯一有效值是vCPU，此值會傳回正在執行的執行個體的相關資訊。
Class	正在追蹤的資源類別。對於維度值vCPU作為 Amazon EC2 使用量指Resource標，有效值為Standard/OnDemand F/OnDemand G/OnDemand 、Inf/OnDemand 、P/OnDemand 、和X/OnDemand 。 該維度的值定義指標報告之執行個體類型的第一個字母。例如，Standard/OnDemand 傳回開頭為 A、C、D、H、I、M、R、T 和 Z 之所有執行中執行個體的相關資訊，,而 G/OnDemand 傳回開頭為 G 之所有執行中執行個體的相關資訊。

使用 Amazon EC2 主控台安裝和設定 CloudWatch代理程式，以新增其他指標

使用 Amazon EC2 主控台安裝和設定 CloudWatch 代理程式已處於測試階段，EC2且可能會有所變更。

根據預設，Amazon 會 CloudWatch 提供基本指標，例如`CPUUtilization`和`NetworkIn`，用於監控您的 Amazon EC2 執行個體。若要收集其他指標，您可以在EC2執行個體上安裝 CloudWatch 代理程式，然後將代理程式設定為發出選取的指標。您可以使用 Amazon EC2 主控台為您執行此操作，而不是在每個EC2執行個體上手動安裝和設定 CloudWatch 代理程式。

本主題說明如何使用 Amazon 主EC2控台在執行個體上安裝 CloudWatch 代理程式，並將代理程式設定為發出選取的指標。

如需此程序的手動步驟，請參閱 Amazon 使用 CloudWatch 者指南 AWS Systems Manager中的使用安裝 CloudWatch 代理程式。如需 CloudWatch 代理程式的詳細資訊，請參閱使用代理程 CloudWatch 式收集指標、記錄檔和追蹤。

主題

- 必要條件
- 運作方式
- 成本
- 安裝和設定 CloudWatch 代理程式

必要條件

若要使用 Amazon EC2 安裝和設定 CloudWatch 代理程式，您必須符合本節所述的使用者和執行個體先決條件。

使用者必要

若要使用此功能，您的IAM主控台使用者或角色必須具有使用 Amazon 所需的許可EC2和下列IAM許可：

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

```
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetParameter",
                "ssm:PutParameter"
            ],
            "Resource": "arn:aws:ssm:*:*:parameter/EC2-Custom-Metrics-*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:SendCommand",
                "ssm:ListCommandInvocations",
                "ssm:DescribeInstanceInformation"
            ],
            "Resource": [
                "*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:GetInstanceProfile",
                "iam:SimulatePrincipalPolicy"
            ],
            "Resource": "*"
        }
    ]
}
```

實例先決條

- 執行個體狀態：`running`
- 支援的作業系統:
- AWS Systems Manager 代理程式 (SSM代理程式)：已安裝。SSM代理程式的兩個注意事項
  - SSM代理程式已預先安裝在由 AWS 受信任的協力廠商提供的某些 Amazon 機器映像 (AMIs) 上。如需有關支援的資訊以AMIs及安裝SSM代理程式的說明，請參閱AWS Systems Manager 使用指南中預先安裝SSM代理程式的 Amazon Machine Image (AMIs)。
  - 如果您遇到SSM代理程式的問題，請參閱AWS Systems Manager 使用指南中的SSM代理程式疑難排解。

- IAM執行個體的權限：必須將下列 AWS 受管理的政策新增至連結至執行個體的IAM角色：
  - 答 mazonSSMManaged InstanceCore：可讓執行個體使用 Systems Manager 來安裝和設定 CloudWatch 代理程式。
  - CloudWatchAgentServerPolicy— 可讓執行個體使用 CloudWatch代理程式將資料寫入 CloudWatch。

如需有關如何新增執行個體IAM權限的詳細資訊，請參閱使用指南中的使用執行個體設定檔。IAM

運作方式

在使用 Amazon EC2 主控台安裝和設定 CloudWatch 代理程式之前，必須確保您的使用IAM者或角色以及要新增指標的執行個體符合特定先決條件。然後，您可以使用 Amazon 主EC2控台在選取的執行個體上安裝和設定 CloudWatch 代理程式。

首先符合先決條件

- 您需要必要的IAM權限 — 開始之前，請確定您的主控台使用者或角色具有使用此功能的必要IAM權限。
- 執行個體 — 若要使用此功能，您的EC2執行個體必須是 Linux 執行個體、已安裝SSM代理程式、擁有必要的IAM權限，並且正在執行。

然後，您可以使用該功能

1. 選取執行個體 — 在 Amazon EC2 主控台中，選取要在其上安裝和設定 CloudWatch 代理程式的執行個體。然後，您可以選擇設定 CloudWatch 代理程式來啟動程序。
2. 驗證SSM代理程式 — Amazon EC2 會檢查SSM代理程式是否已在每個執行個體上安裝並啟動。任何未通過此檢查的執行個體都會從程序中排除。SSM代理程式用於在此程序期間對執行個體執行動作。
3. 驗證IAM許可 — Amazon EC2 會檢查每個執行個體是否具有此程序所需的IAM許可。任何未通過此檢查的執行個體都會從程序中排除。這些IAM權限可讓 CloudWatch 代理程式從執行個體收集指標，並與之整合 AWS Systems Manager 以使用SSM代理程式。
4. 驗證 CloudWatch 代理程式 — Amazon EC2 會檢查 CloudWatch 代理程式是否已在每個執行個體上安裝並執行。如果有任何執行個體未通過此檢查，Amazon EC2 會為您安裝和啟動 CloudWatch 代理程式。此程序完成後， CloudWatch 代理程式會在每個執行個體上收集選取的指標。
5. 選取測量結果組態 — 您可以選取 CloudWatch 代理程式要從執行個體發出的指標。選取之後，Amazon 會將組態檔EC2存放在參數存放區中，該檔案會一直保留到程序完成為止。Amazon

EC2 將從參數存放區刪除組態檔案，除非程序中斷。請注意，如果您未選取指標，但先前已將其新增至執行個體，則在完成此程序後，系統會將該指標從您的執行個體中移除。

6. 更新 CloudWatch 代理程式組態 — Amazon EC2 會將指標組態傳送給 CloudWatch 代理程式。這是程序中的最後一個步驟。如果成功，您的執行個體可以發出所選指標的資料，而 Amazon EC2 會從參數存放區刪除組態檔案。

成本

您在此程序期間新增的其他指標會以自訂指標計費。如需有關 CloudWatch 指標定價的詳細資訊，請參閱 Amazon CloudWatch 定價。

安裝和設定 CloudWatch 代理程式

您可以使用 Amazon 主EC2控台安裝和設定 CloudWatch 代理程式，以新增其他指標。

> **Note**
>
> 每次執行此程序時，都會覆寫現有的 CloudWatch 代理程式組態。如果您沒有選取先前選取的指標，它會從執行個體中移除。

使用 Amazon EC2 主控台安裝和設定 CloudWatch 代理程式

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取要安裝的執行個體並設定 CloudWatch 代理程式。
4. 選擇 [動作]、[監視和疑難排解]、[設定 CloudWatch 代

> **Tip**
>
> 並非所有功能均可使用此功能 AWS 區域。如果無法使用 [設定 CloudWatch代理程式]，請嘗試其他區域。

5. 對於程序中的每個步驟，請閱讀主控台文字，然後選擇 [下一步]。
6. 若要完成此程序，請在最後一個步驟中選擇 [完成]。

執行個體的 CloudWatch 指標統計資料

您可以取得執行個體 CloudWatch 指標的統計資料。統計資料是指定期間內的測量結果資料彙總。CloudWatch 根據您的自訂資料提供或在中由其他服務提供的指標資料點 AWS 來提供統計資料 CloudWatch。彙總會使用在您指定期間的命名空間、公制名稱、維度和量測資料點單位來進行。下表說明可用的統計資料。

統計數字	描述
Minimum	在指定期間內觀察到的最低值。您可以使用這個值來判斷適用於您應用程式的低活動磁碟區。
Maximum	在指定期間內觀察到的最高值。您可以使用這個值來判斷適用於您應用程式的高活動磁碟區。
Sum	為符合指標總和提交的所有值。這個統計資料在決定指標的總磁碟區時非常有用。
Average	指定期間內的 Sum/SampleCount 值。透過比較這個統計資料與 Minimum 和 Maximum，您可以確定指標的完整範圍，以及 Minimum 和 Maximum 與平均使用的接近程度。這個比較可協助您知道何時需要增加或減少您的資源。
SampleCount	用於統計資料計算的資料點計數 (數值)。
pNN.NN	指定百分位數的值。您可以指定任何百分位數，最多使用兩位小數 (例如，p95.45)。

目錄

- 取得特定執行個體的統計資料
- 彙總所有執行個體的統計資料
- 依據 Auto Scaling 群組彙總統計資料
- 彙總統計資料 AMI

取得特定執行個體的統計資料

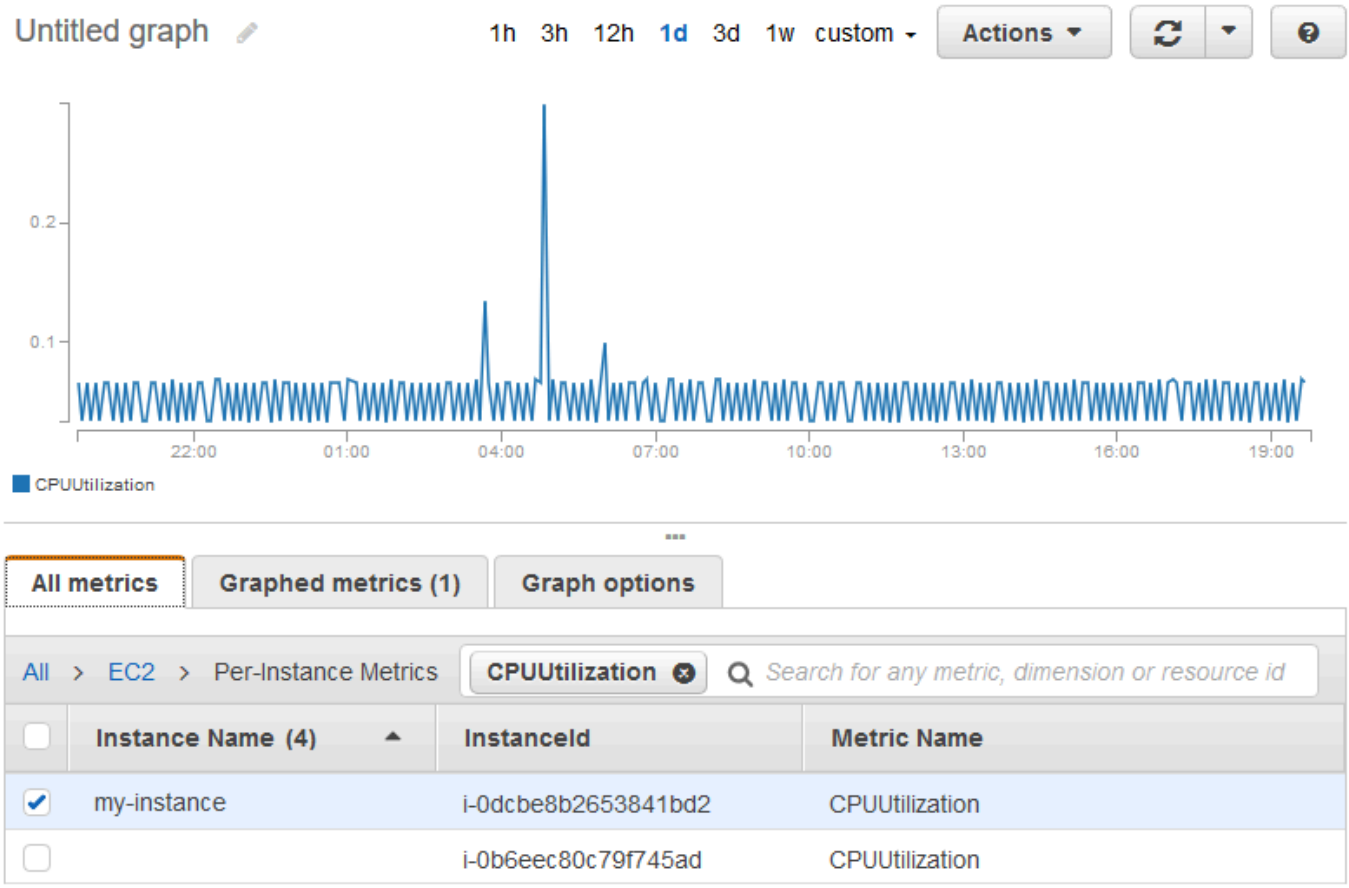
您可以使用 AWS Management Console 或取得特 AWS CLI 定執行個體的統計資料。下列範例說明如何使用 AWS Management Console 或來判斷特定EC2執行個體的最大使用CPU率。 AWS CLI

要求

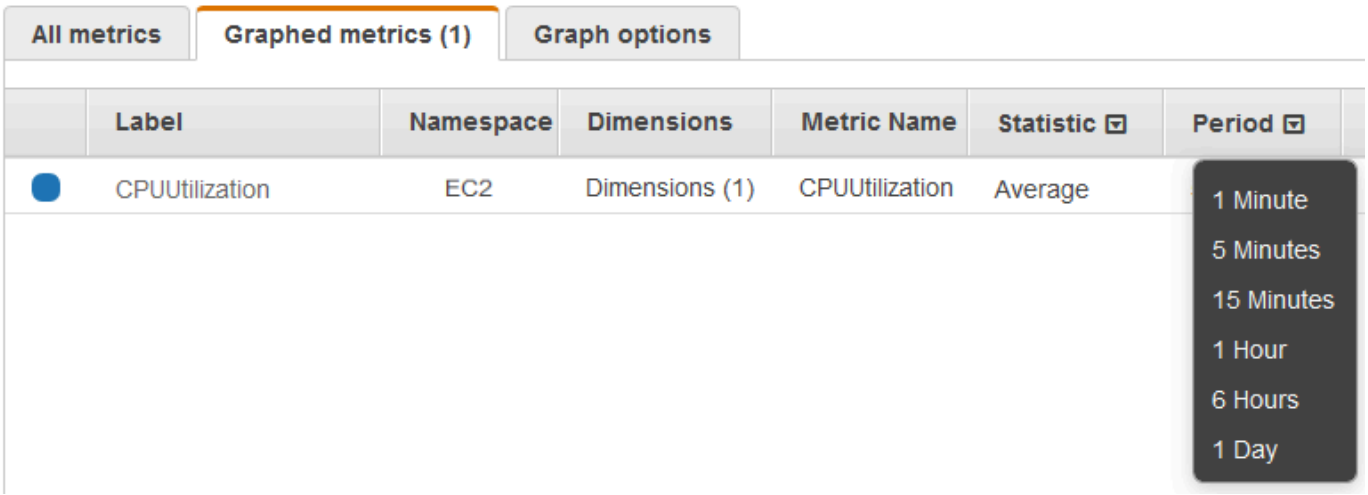
- 您必須擁有該執行個體的 ID。您可以使用 AWS Management Console 或 describe-instances 命令以取得執行個體 ID。
- 預設會啟用基本監控，但您可以啟用詳細監控。如需詳細資訊，請參閱管理EC2執行個體的詳細監控。

顯示特定執行個體 (主控台) 的CPU使用率

1. 在開啟 CloudWatch 主控台https://console.aws.amazon.com/cloudwatch/。
2. 在導覽窗格中，選擇 指標。
3. 選擇EC2測量結果命名空間。
4. 選擇 Per-Instance Metrics (每個執行個體指標) 維度。
5. 在搜尋欄位中，輸入 **`CPUUtilization`**，然後按 Enter 鍵。選擇特定執行處理的資料列，此資料列會顯示執行處理的CPUUtilization測量結果圖表。若要為圖形命名，請選擇鉛筆圖示。若要變更時間範圍，請選取一個預先定義的值，或選擇 custom (自訂)。



6. 若要變更指標的統計資料或期間，請選擇 Graphed metrics (圖表化指標) 索引標籤。選擇欄位標題或個別的值，然後選擇不同的值。



取得特定執行個體的CPU使用率 (AWS CLI)

您可以使用下列get-metric-statistics命令，取得指定之執行處理的CPUUtilization測量結果 (使用指定的期間和時間間隔)：

```
aws cloudwatch get-metric-statistics --namespace AWS/EC2 --metric-name CPUUtilization
 --period 3600 \
--statistics Maximum --dimensions Name=InstanceId,Value=i-1234567890abcdef0 \
--start-time 2022-10-18T23:18:00 --end-time 2022-10-19T23:18:00
```

下列為範例輸出。每個值代表單一執行個體的最大CPU使用率百分EC2比。

```
{
    "Datapoints": [
        {
            "Timestamp": "2022-10-19T00:18:00Z",
            "Maximum": 0.33000000000000002,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2022-10-19T03:18:00Z",
            "Maximum": 99.670000000000002,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2022-10-19T07:18:00Z",
```

```
            "Maximum": 0.34000000000000002,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2022-10-19T12:18:00Z",
            "Maximum": 0.34000000000000002,
            "Unit": "Percent"
        },
        ...
    ],
    "Label": "CPUUtilization"
}
```

彙總所有執行個體的統計資料

已啟用詳細監控的執行個體可使用彙總統計資料。彙總資料不含使用基本監控的執行個體。在取得所有執行個體的彙總統計資料前，您必須啟用詳細監控 (額外付費)，它以 1 分鐘的間隔時間提供資料。

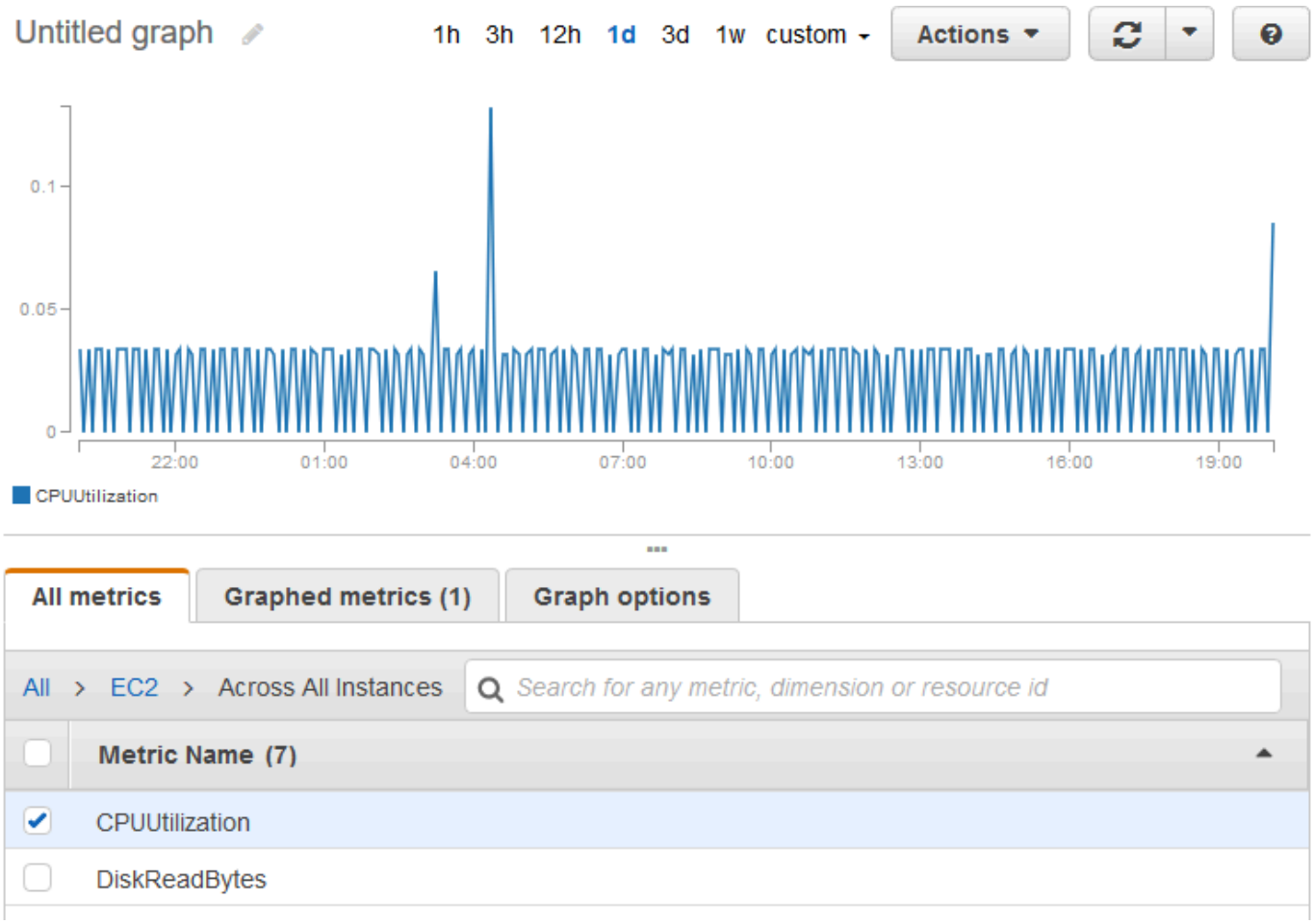
請注意，Amazon CloudWatch 無法跨 AWS 區域彙總資料。區域之間的指標是完全獨立的。

此範例說明如何使用詳細監控來取得EC2執行個體的平均CPU使用量。因為未指定維度，所以會 CloudWatch 傳回`AWS/EC2`命名空間中所有維度的統計資料。

> **Important**
>
> 這種跨 AWS 命名空間擷取所有維度的技術不適用於您發佈到 Amazon CloudWatch 的自訂命名空間。使用自訂命名空間，您必須指定與任何特定資料點建立關聯的一組完整維度，以擷取包含該資料點的統計資料。

顯示執行個體的平均CPU使用率 (主控台)

1. 在開啟 CloudWatch 主控台https://console.aws.amazon.com/cloudwatch/。
2. 在導覽窗格中，選擇 指標。
3. 選擇EC2命名空間，然後選擇「跨所有執行個體」。
4. 選擇包含的資料列 CPUUtilization，此資料列會顯示所有EC2執行處理的量度圖表。若要為圖形命名，請選擇鉛筆圖示。若要變更時間範圍，請選取一個預先定義的值，或選擇 custom (自訂)。



5. 若要變更指標的統計資料或期間，請選擇 Graphed metrics (圖表化指標) 索引標籤。選擇欄位標題或個別的值，然後選擇不同的值。

取得執行個體的平均CPU使用率 (AWS CLI)

如下所示使用get-metric-statistics命令，即可取得執行個體中指CPUUtilization標的平均值。

```
aws cloudwatch get-metric-statistics \
    --namespace AWS/EC2 \
    --metric-name CPUUtilization \
    --period 3600  --statistics "Average" "SampleCount" \
    --start-time 2022-10-11T23:18:00 \
    --end-time 2022-10-12T23:18:00
```

下列為範例輸出：

```
{
    "Datapoints": [
```

```
        {
            "SampleCount": 238.0,
            "Timestamp": "2022-10-12T07:18:00Z",
            "Average": 0.038235294117647062,
            "Unit": "Percent"
        },
        {
            "SampleCount": 240.0,
            "Timestamp": "2022-10-12T09:18:00Z",
            "Average": 0.16670833333333332,
            "Unit": "Percent"
        },
        {
            "SampleCount": 238.0,
            "Timestamp": "2022-10-11T23:18:00Z",
            "Average": 0.041596638655462197,
            "Unit": "Percent"
        },
        ...
    ],
    "Label": "CPUUtilization"
}
```

依據 Auto Scaling 群組彙總統計資料

您可以彙總「Auto Scaling」群組中EC2執行個體的統計資料。請注意，Amazon CloudWatch 無法跨 AWS 區域彙總資料。區域之間的指標是完全獨立的。

此範例顯示如何擷取一個 Auto Scaling 群組寫入磁碟的總位元組。系統會針對指定 Auto Scaling 群組中的所有EC2執行個體，以 24 小時間隔計算總計的 1 分鐘期間。

若要顯示 DiskWriteBytes Auto Scaling 群組中的執行個體 (主控台)

1. 在開啟 CloudWatch 主控台https://console.aws.amazon.com/cloudwatch/。
2. 在導覽窗格中，選擇 指標。
3. 選擇EC2命名空間，然後選擇「依 Auto Scaling 群組」。
4. 選擇DiskWriteBytes量度和特定「Auto Scaling 例」群組的列，此群組會顯示「Auto Scaling 比例」群組中執行個體的量度圖表。若要為圖形命名，請選擇鉛筆圖示。若要變更時間範圍，請選取一個預先定義的值，或選擇 custom (自訂)。
5. 若要變更指標的統計資料或期間，請選擇 Graphed metrics (圖表化指標) 索引標籤。選擇欄位標題或個別的值，然後選擇不同的值。

顯示 DiskWriteBytes Auto Scaling 群組中的例證 (AWS CLI)

使用 get-metric-statistics 命令，如下所示。

```
aws cloudwatch get-metric-statistics --namespace AWS/EC2 --metric-name DiskWriteBytes
 --period 360 \
--statistics "Sum" "SampleCount" --dimensions Name=AutoScalingGroupName,Value=my-asg --
start-time 2022-10-16T23:18:00 --end-time 2022-10-18T23:18:00
```

下列為範例輸出：

```
{
    "Datapoints": [
        {
            "SampleCount": 18.0,
            "Timestamp": "2022-10-19T21:36:00Z",
            "Sum": 0.0,
            "Unit": "Bytes"
        },
        {
            "SampleCount": 5.0,
            "Timestamp": "2022-10-19T21:42:00Z",
            "Sum": 0.0,
            "Unit": "Bytes"
        }
    ],
    "Label": "DiskWriteBytes"
}
```

彙總統計資料 AMI

您可以AMI針對已啟用詳細監控的執行個體彙總統計資料。彙總資料不含使用基本監控的執行個體。在取得所有執行個體的彙總統計資料前，您必須啟用詳細監控 (額外付費)，它以 1 分鐘的間隔時間提供資料。

請注意，Amazon CloudWatch 無法跨 AWS 區域彙總資料。區域之間的指標是完全獨立的。

此範例說明如何判斷使用特定 Amazon 機器映像 (AMI) 的所有執行個體的平均使用CPU率。此平均是以一天期間內每 60 秒的時間間隔計算。

若要依據 AMI (主控台) 顯示平均CPU使用率

1. 在開啟 CloudWatch 主控台https://console.aws.amazon.com/cloudwatch/。

2. 在導覽窗格中，選擇 指標。
3. 選擇命EC2名空間，然後選擇「依影像」(AMI) ID。
4. 選擇CPUUtilization測量結果和特定資料列AMI，此資料列會顯示指定之測量結果的圖表AMI。若要為圖形命名，請選擇鉛筆圖示。若要變更時間範圍，請選取一個預先定義的值，或選擇 custom (自訂)。
5. 若要變更指標的統計資料或期間，請選擇 Graphed metrics (圖表化指標) 索引標籤。選擇欄位標題或個別的值，然後選擇不同的值。

取得影像 ID 的平均CPU使用率 (AWS CLI)

使用 get-metric-statistics 命令，如下所示。

```
aws cloudwatch get-metric-statistics --namespace AWS/EC2 --metric-name CPUUtilization
 --period 3600 \
--statistics Average --dimensions Name=ImageId,Value=ami-3c47a355 --start-
time 2022-10-10T00:00:00 --end-time 2022-10-11T00:00:00
```

下列為範例輸出。每個值代表執行指定的EC2執行處理的平均CPU使用率百分比AMI。

```
{
    "Datapoints": [
        {
            "Timestamp": "2022-10-10T07:00:00Z",
            "Average": 0.041000000000000009,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2022-10-10T14:00:00Z",
            "Average": 0.079579831932773085,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2022-10-10T06:00:00Z",
            "Average": 0.036000000000000011,
            "Unit": "Percent"
        },
        ...
    ],
    "Label": "CPUUtilization"
}
```

檢視執行個體的監控圖表

啟動執行個體後，您可以開啟 Amazon EC2 主控台，並在監控索引標籤上檢視該執行個體的監控圖表。每個圖表都是以其中一個可用的 Amazon EC2 指標為基礎。

下列圖表可供使用：

- 平均CPU使用率 (百分比)
- 平均磁碟讀取 (位元組)
- 平均磁碟寫入 (位元組)
- 最大網路輸入 (位元組)
- 最大網路輸出 (位元組)
- 磁碟讀取操作摘要 (計數)
- 磁碟寫入操作摘要 (計數)
- 狀態摘要 (任意)
- 執行個體狀態摘要 (計數)
- 系統狀態摘要 (計數)

如需指標及其向圖表提供資料的詳細資訊，請參閱 CloudWatch 您執行個體可用的指標。

使用 CloudWatch 主控台繪製指標圖形

您也可以使用 CloudWatch 主控台繪製 Amazon EC2 和其他 AWS 服務產生的指標資料圖形。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的繪製指標圖形。

建立執行個體的 CloudWatch 警示

您可以建立 CloudWatch 警示來監控其中一個執行個體的 CloudWatch 指標。 CloudWatch當測量結果達到您指定的臨界值時，會自動傳送通知給您。您可以使用 Amazon EC2 主控台或使用主控台提供的更進階選項建立 CloudWatch 警示。 CloudWatch

使用 CloudWatch 主控台建立鬧鐘

有關示例，請參閱 Amazon CloudWatch 用戶指南中的創建 Amazon CloudWatch 警報。

若要使用 Amazon EC2 主控台建立警示

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。

2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後選擇 [動作]、[監控及疑難排解]、 CloudWatch [管理
4. 在 [管理 CloudWatch 鬧鐘詳細資料] 頁面的 [新增或編輯鬧鐘] 底下，選取 [建立鬧鐘]。
5. 對於警示通知，請選擇是否設定 Amazon 簡單通知服務 (AmazonSNS) 通知。輸入現有的 Amazon SNS 主題或輸入名稱以建立新主題。
6. 在警示動作選擇是否要指定觸發警示時採取的動作。從清單中選擇一個動作。
7. 針對 Alarm thresholds (警示閾值)，選取警示的指標和準則。例如，若要建立在 5 分鐘內CPU使用率達到 80% 時觸發的警示，請執行下列動作：
   a. 保留「依據 (平均)」和「要取樣的資料類型」(CPU使用率)「群組範例」的預設設定。
   b. 在警示時機選擇 >= 並在百分比輸入 **0.80**。
   c. 在連續期間輸入 **1**，並在期間選取 5 分鐘。
8. (選用) 針對 Sample metric data (範例指標資料)，選擇 Add to dashboard (新增至儀表板)。
9. 選擇 Create (建立)。

您可以從 Amazon EC2 主控台或主控台編輯 CloudWatch 鬧鐘設定。 CloudWatch 如果要刪除鬧鐘，可以從 CloudWatch 主控台刪除鬧鐘。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的編輯或刪除 CloudWatch 警示。

建立警示以停止、終止、重新啟動或復原執行個體

使用 Amazon CloudWatch 警示動作，您可以建立自動停止、終止、重新開機或復原執行個體的警示。當執行個體不再需要執行，您可以使用停止或終止動作以協助您節省成本。如果發生系統受損，您可以使用重新啟動和復原動作，自動重新啟動這些執行個體或將它們復原到新的硬體。

> **Note**
> 如需 Amazon CloudWatch 警示帳單和定價資訊，請參閱 Amazon CloudWatch 使用者指南中的 CloudWatch 帳單和費用。

`AWSServiceRoleForCloudWatchEvents`服務連結角色可 AWS 讓您代表執行警示動作。當您第一次在 AWS Management Console、或中建 CloudWatch 立警示時 AWS CLI，會為您建立服務連結角色。IAM API

在許多情況下，您可能想要自動停止或終止您的執行個體。例如，您可能有專門批次處理薪資作業或科學運算任務的執行個體，它們在執行一段時間後完成工作。您不必讓這些執行個體閒置 (及累積費用)，您可以停止或終止它們以協助您節省成本。使用停止和終止警示動作的主要差別是，如果執行個體之後需要重新執行，您將可以輕鬆地啟動已停止的執行個體，而且您可以保持相同的執行個體 ID 和根磁碟區。不過，您無法啟動已終止的執行個體。相反地，您必須啟動新的執行個體。在停止或終止執行個體時，執行個體儲存體磁碟區上的資料會遺失。

您可以將停止、終止、重新啟動或復原動作新增至 Amazon EC2 每個執行個體指標上設定的任何警示，包括 Amazon CloudWatch (在`AWS/EC2`命名空間中) 提供的基本和詳細監控指標，以及包含`InstanceId`維度的任何自訂指標 (只要其值參考有效的執行中 Amazon EC2 執行個體)。

> **Important**
>
> 如果缺少指標資料點，狀態檢查警示可以暫時進入狀態。`INSUFFICIENT_DATA`雖然很少見，但是當指標報告系統發生中斷時，即使執行個體健康狀態良好，也可能發生這種情況。我們建議您將`INSUFFICIENT_DATA`狀態視為遺失資料，而非警示洩漏，尤其是將警示設定為停止、終止、重新啟動或復原執行個體時。

主控台支援

您可以使用 Amazon EC2 主控台或主控台建立警示。 CloudWatch 本文件中的程序使用 Amazon EC2 主控台。如需使用 CloudWatch 主控台的程序，請參閱 Amazon 使用 CloudWatch 者指南中的建立停止、終止、重新開機或復原執行個體的警示。

許可

您必須具有才能建立或修改執行警示動作的EC2警示。`iam:CreateServiceLinkedRole`服務角色是服務假定代表您執行動作的IAM角色。IAM管理員可以從中建立、修改和刪除服務角色IAM。如需詳細資訊，請參閱《IAM使用指南》 AWS 服務中的建立角色以將權限委派給

目錄

- 向 Amazon CloudWatch 警報添加停止操作
- 將終止動作新增至 Amazon CloudWatch 警示
- 將重新開機動作新增至 Amazon CloudWatch 警示
- 將恢復動作添加到 Amazon CloudWatch 警報
- Amazon CloudWatch 警報動作案例

向 Amazon CloudWatch 警報添加停止操作

您可以建立警示，在達到特定閾值時停止 Amazon EC2 執行個體。例如，您可以執行開發或測試執行個體，並偶爾忘記關閉它們。您可以建立在 24 小時內的平均使用CPU率百分比低於 10% 時觸發的警示，表示該警示處於閒置狀態且不再使用。您可以根據需要調整閾值、持續時間和期間，此外還可以新增 Amazon 簡單通知服務 (AmazonSNS) 通知，以便在觸發警示時收到電子郵件。

使用 Amazon EBS 磁碟區做為根裝置的執行個體可以停止或終止，而使用執行個體存放區做為根裝置的執行個體只能終止。在終止或停止執行個體時，執行個體儲存體磁碟區上的資料會遺失。

若要建立警示以停止閒置執行個體 (Amazon EC2 主控台)

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後選擇 [動作]、[監控及疑難排解]、 CloudWatch [管理

   或者，您可以在 Alarm status (警示狀態) 資料欄中選擇加號 (+)。
4. 在「管理 CloudWatch 警示」頁面上，執行下列動作：

   a. 選擇 Create an alarm (建立警示)。

   b. 若要在觸發警示時接收電子郵件，對於警示通知，請選擇現有的 Amazon SNS 主題。您首先需要使用 Amazon 控制台創建一個 Amazon SNS SNS 主題。如需詳細資訊，請參閱 Amazon 簡單通知服務開發人員指南中的使用 Amazon SNS 進行 application-to-person (A2P) 簡訊。

   c. 切換 Alarm action (警示動作)，然後選擇 Stop (停止)。

   d. 若要將 Group samples by (分組範例條件) 和 Type of data to sample (要作為範例的資料類型)，請選擇統計資料和指標。在此範例中，選擇「平均」與「CPU使用率」。

   e. 針對 Alarm When (警示時間) 和 Percent (百分比)，指定指標閾值。在此範例中，指定 <= 和 10%。

   f. 針對 Consecutive period (連續期間) 和Period (期間)，指定警示的評估期間。在此範例中，請指定 1 個每隔 5 Minute (5 分鐘) 的連續期間。

   g. Amazon CloudWatch 會自動為您創建一個警報名稱。若要變更名稱，請在 Alarm name (警示名稱) 中輸入新名稱。警示名稱只能包含ASCII字元。

> **Note**
>
> 您可以在建立警示之前根據自己的需求調整警示組態，也可以稍後再進行編輯。這包括指標、閾值、持續時間、動作和通知設定。不過，在您建立警示之後，以後就無法編輯其名稱。

h. 選擇 Create (建立)。

將終止動作新增至 Amazon CloudWatch 警示

您可以建立警示，在達到特定臨界值時自動終止EC2執行個體 (只要執行個體沒有啟用終止保護)。例如，您可能想要在執行個體完成作業時予以終止，而且不再需要該執行個體。如果您之後還要使用該執行個體，您應該停止而非終止執行個體。在終止執行個體時，執行個體儲存體磁碟區上的資料會遺失。如需為執行個體啟用和停用終止保護的詳細資訊，請參閱 啟用終止保護。

若要建立警示以終止閒置執行個體 (Amazon EC2 主控台)

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後選擇 [動作]、[監控及疑難排解]、 CloudWatch [管理

   或者，您可以在 Alarm status (警示狀態) 資料欄中選擇加號 (+)。
4. 在「管理 CloudWatch 警示」頁面上，執行下列動作：

   a. 選擇 Create an alarm (建立警示)。

   b. 若要在觸發警示時接收電子郵件，對於警示通知，請選擇現有的 Amazon SNS 主題。您首先需要使用 Amazon 控制台創建一個 Amazon SNS SNS 主題。如需詳細資訊，請參閱 Amazon 簡單通知服務開發人員指南中的使用 Amazon SNS 進行 application-to-person (A2P) 簡訊。

   c. 切換 Alarm action (警示動作)，然後選擇 Terminate (終止)。

   d. 若要將 Group samples by (分組範例條件) 和 Type of data to sample (要作為範例的資料類型)，請選擇統計資料和指標。在此範例中，選擇「平均」與「CPU使用率」。

   e. 針對 Alarm When (警示時間) 和 Percent (百分比)，指定指標閾值。在此範例中，指定 >= 和 10%。

f. 針對 Consecutive period (連續期間) 和Period (期間)，指定警示的評估期間。在此範例中，請指定 24 個每隔 1 Hour (1 小時) 的連續期間。

g. Amazon CloudWatch 會自動為您創建一個警報名稱。若要變更名稱，請在 Alarm name (警示名稱) 中輸入新名稱。警示名稱只能包含ASCII字元。

> **Note**
>
> 您可以在建立警示之前根據自己的需求調整警示組態，也可以稍後再進行編輯。這包括指標、閾值、持續時間、動作和通知設定。不過，在您建立警示之後，以後就無法編輯其名稱。

h. 選擇 Create (建立)。

將重新開機動作新增至 Amazon CloudWatch 警示

您可以建立 Amazon CloudWatch 警示來監控 Amazon EC2 執行個體並自動重新啟動執行個體。重新啟動警示動作建議用於執行個體運作狀態檢查失敗 (相對的，復原警示動作則適用於系統運作狀態檢查失敗)。重新啟動執行個體等同於重新啟動作業系統。在大多數情況下，將執行個體重新開機只需要幾分鐘的時間。重新啟動執行個體時，它會保留在相同的實體主機上，因此您的執行個體會保留其公開DNS名稱、私有 IP 位址，以及執行個體儲存磁碟區上的任何資料。

有別於停用和重新開始執行個體，重新啟動執行個體不會啟動新的執行個體計費時段 (最少收費一分鐘)。在重新啟動執行個體時，會保留執行個體儲存體磁碟區上的資料。執行個體儲存體磁碟區必須在重新啟動後重新掛載到檔案系統中。如需詳細資訊，請參閱重新啟動您的執行個體。

> **Important**
>
> 為了避免重新開機和復原動作之間的競爭情況，請避免為重新開機警示和復原警示設定相同的評估期間數。我們建議您將重新開機警示設定為三個各一分鐘的評估期間。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的評估警示。

若要建立警示以重新啟動執行個體 (Amazon EC2 主控台)

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後選擇 [動作]、[監控及疑難排解]、 CloudWatch [管理

或者，您可以在 Alarm status (警示狀態) 資料欄中選擇加號

(+)。

4. 在「管理 CloudWatch 警示」頁面上，執行下列動作：

   a. 選擇 Create an alarm (建立警示)。

   b. 若要在觸發警示時接收電子郵件，對於警示通知，請選擇現有的 Amazon SNS 主題。您首先需要使用 Amazon 控制台創建一個 Amazon SNS SNS 主題。如需詳細資訊，請參閱 Amazon 簡單通知服務開發人員指南中的使用 Amazon SNS 進行 application-to-person (A2P) 簡訊。

   c. 切換 Alarm action (警示動作)，然後選擇 Reboot (重新開機)。

   d. 若要將 Group samples by (分組範例條件) 和 Type of data to sample (要作為範例的資料類型)，請選擇統計資料和指標。在此範例中，選擇 Average (平均) 和 Status check failed: instance (狀態檢查失敗：執行個體)。

   e. 針對 Consecutive period (連續期間) 和Period (期間)，指定警示的評估期間。在此範例中，輸入 3 個連續 1 分鐘的期間。如果停用 1 分鐘，您必須啟用詳細監視，或者您可以改為選擇 5 分鐘。

   f. Amazon CloudWatch 會自動為您創建一個警報名稱。若要變更名稱，請在 Alarm name (警示名稱) 中輸入新名稱。警示名稱只能包含ASCII字元。

   g. 選擇 Create (建立)。

將恢復動作添加到 Amazon CloudWatch 警報

您可以創建一個 Amazon CloudWatch 警報來監控 Amazon EC2 實例。如果執行個體因為基礎硬體故障或需要 AWS 參與修復的問題而受損，您可以自動復原執行個體。已終止的執行個體無法復原。復原後的執行個體與原始執行個體相同，包括執行個體 ID、私有 IP 地址、彈性 IP 地址及所有執行個體中繼資料。

CloudWatch 防止您將復原動作新增至不支援復原動作的執行個體上的警示。

觸發`StatusCheckFailed_System`警示並啟動復原動作時，Amazon SNS 主題會在您建立警示並關聯復原動作時收到您所選擇的通知。在執行個體復原期間，執行個體會在重新啟動期間遷移，記憶體內的任何資料都將遺失。程序完成後，資訊會發佈至您針對警示設定的SNS主題。訂閱此SNS主題的任何人都會收到電子郵件通知，其中包含復原嘗試的狀態以及任何進一步的指示。您會發現執行個體在已復原的執行個體上重新啟動。

> **Note**
>
> 復原動作只能用於 `StatusCheckFailed_System`，而非 `StatusCheckFailed_Instance`。

下列為可能導致系統狀態檢查失敗的問題：

- 網路連線中斷
- 系統電力中斷
- 實體主機的軟體問題
- 實體主機上會影響網路連線的硬體問題

復原動作僅在符合某些特性的執行個體上受到支援：如需詳細資訊，請參閱執行個體恢復。

如果您的執行個體具有公有 IP 地址，它將在復原後保留公有 IP 地址。

> **Important**
>
> 為了避免重新開機和復原動作之間的競爭情況，請避免為重新開機警示和復原警示設定相同的評估期間數。我們建議您將復原警示設定為兩個各一分鐘的評估期間。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的評估警示。

若要建立警示以復原執行個體 (Amazon EC2 主控台)

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後選擇 [動作]、[監控及疑難排解]、 CloudWatch [管理

   或者，您可以在 Alarm status (警示狀態) 資料欄中選擇加號 (+)。
4. 在「管理 CloudWatch 警示」頁面上，執行下列動作：

   a. 選擇 Create an alarm (建立警示)。

   b. 若要在觸發警示時接收電子郵件，對於警示通知，請選擇現有的 Amazon SNS 主題。您首先需要使用 Amazon 控制台創建一個 Amazon SNS SNS 主題。如需詳細資訊，請參閱

Amazon 簡單通知服務開發人員指南中的使用 Amazon SNS 進行 application-to-person (A2P) 簡訊。

> **Note**
>
> 使用者必須訂閱指定的SNS主題，才能在警示觸發時接收電子郵件通知。一 AWS 帳戶根使用者 律會在執行個體復原動作發生時收到電子郵件通知，即使未指定SNS主題或 root 使用者未訂閱指定的SNS主題也一樣。

c. 切換 Alarm action (警示動作)，然後選擇 Recover (復原)。

d. 若要將 Group samples by (分組範例條件) 和 Type of data to sample (要作為範例的資料類型)，請選擇統計資料和指標。在此範例中，選擇 Average (平均) 和 Status check failed: system (狀態檢查失敗：系統)。

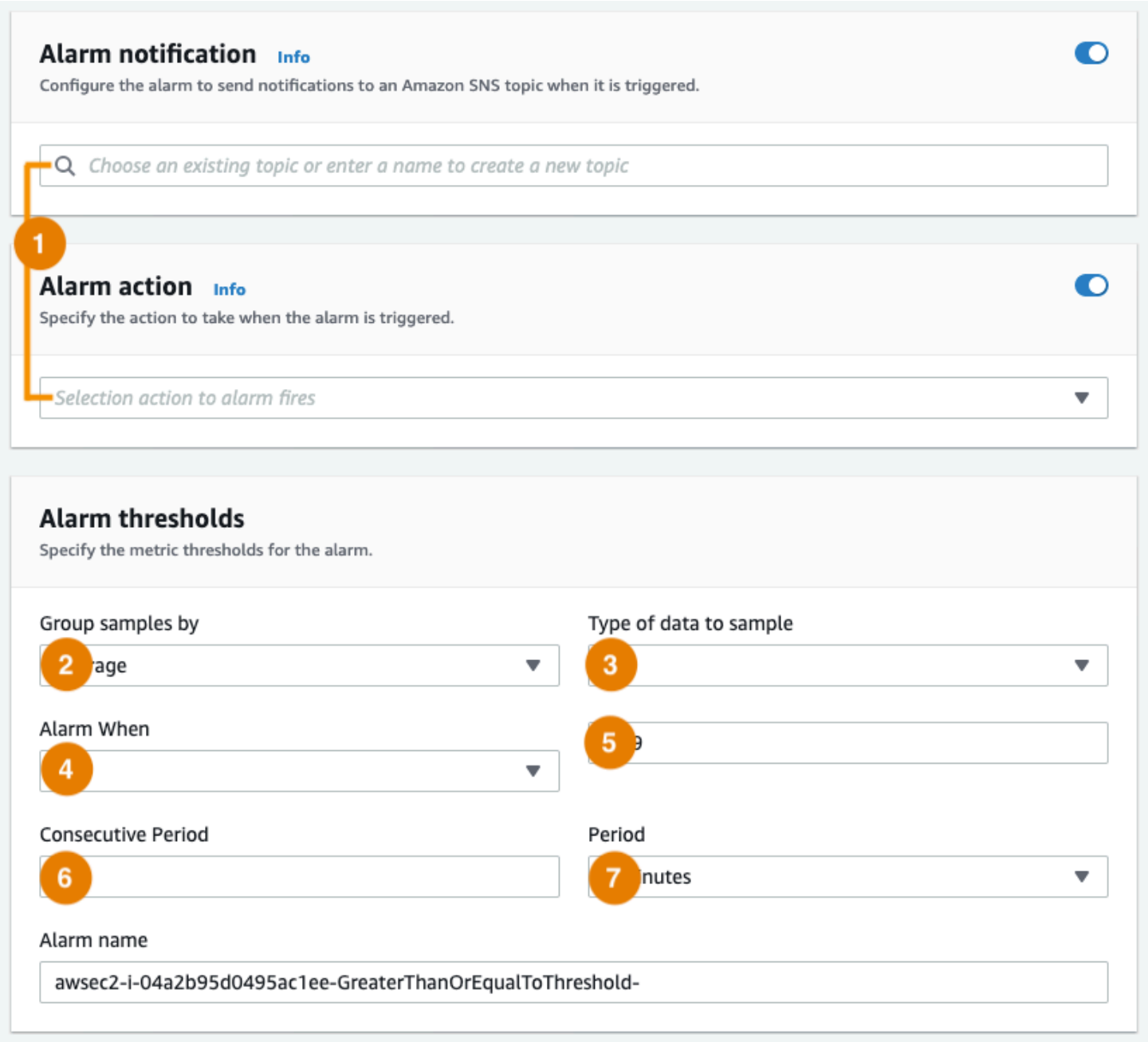
e. 針對 Consecutive period (連續期間) 和Period (期間)，指定警示的評估期間。在此範例中，輸入 2 個連續 1 分鐘的期間。如果停用 1 分鐘，您必須啟用詳細監視，或者您可以改為選擇 5 分鐘。

f. Amazon CloudWatch 會自動為您創建一個警報名稱。若要變更名稱，請在 Alarm name (警示名稱) 中輸入新名稱。警示名稱只能包含ASCII字元。

g. 選擇 Create (建立)。

Amazon CloudWatch 警報動作案例

您可以使用 Amazon EC2 主控台建立警示動作，在符合特定條件時停止或終止 Amazon EC2 執行個體。在下列用來設定警示動作的主控台頁面螢幕截圖中，我們為設定加上編號。我們也在隨後的案例中為設定加上編號，協助您建立適當的動作。



案例 1：停止閒置的開發和測試執行個體

建立警示，在用於軟體開發或測試的執行個體已閒置至少一小時後，將其停止。

設定	Value
1	停止
2	最大

設定	Value
3	CPU利用率
4	<=
5	10%
6	1
7	1 小時

案例 2：停止閒置的執行個體

建立警示，在執行個體已閒置 24 小時後，停止執行個體並傳送電子郵件。

設定	Value
1	停止和寄送電子郵件
2	平均數
3	CPU利用率
4	<=
5	5%
6	24
7	1 小時

案例 3：針對發生異常高流量的 Web 伺服器傳送電子郵件

建立警示，在執行個體每天超過 10 GB 傳出網路流量後，傳送電子郵件。

設定	Value
1	電子郵件

設定	Value
2	總和
3	網路輸出
4	>
5	10 GB
6	24
7	1 小時

案例 4：停止發生異常高流量的 Web 伺服器

建立停止執行個體的警示，並在輸出流量超過每小時 1 GB 時傳送文字訊息 (SMS)。

設定	Value
1	停止並傳送 SMS
2	總和
3	網路輸出
4	>
5	1 GB
6	1
7	1 小時

案例 5：停止受損的執行個體

建立警示，停止未通過三次連續運作狀態檢查的執行個體 (以 5 分鐘間隔執行)。

設定	Value
1	停止
2	平均數
3	狀態檢查失敗：系統
4	-
5	-
6	1
7	15 分鐘

案例 6：批次處理任務完成時終止執行個體

建立警示，當執行批次任務的執行個體不再傳送結果資料時，終止該執行個體。

設定	Value
1	終止
2	最大
3	網路輸出
4	<=
5	100,000 位元組
6	1
7	5 分鐘

使用自動 EC2 Amazon EventBridge

您可以使用 Amazon EventBridge 自動化您的系統事件 AWS 服務 並自動回應，例如應用程式可用性問題或資源變更。來自 AWS 服務的事件會以近乎即時 EventBridge 的方式傳送到。您可建立規則來指示您在意的事件，以及當事件符合規則時執行的動作。可以自動觸發的動作如下：

- 調用一個 AWS Lambda 函數
- 調用 Amazon EC2 運行命令
- 將事件轉送至 Amazon Kinesis Data Streams
- 啟動 AWS Step Functions 狀態機
- 通知 Amazon SNS 主題
- 通知 Amazon SQS 隊列

以下是如何 EventBridge 與 Amazon 一起使用的示例EC2：

- 每當執行個體進入執行狀態時，啟用 Lambda 函數。
- 建立或修改 Amazon EBS 磁碟區時，請通知 Amazon SNS 主題。
- 每當其他 AWS 服務中發生特定事件時，請使用 EC2 Amazon EC2 Run 命令將命令傳送至一或多個 Amazon 執行個體。

如需詳細資訊，請參閱 Amazon EventBridge 使用者指南。

Amazon EC2 事件類型

Amazon EC2 支持以下事件類型：

- EC2AMI狀態變更
- EC2快速啟動狀態變更通知
- EC2艦隊錯誤
- EC2車隊資訊
- EC2機隊實例變更
- EC2叢集競價型執行個體請求
- EC2車隊狀態變更

- EC2執行處理重新平衡建
- EC2執行個體狀態變更通知
- EC2發現艦隊錯誤
- EC2現貨車隊資訊
- EC2競價型叢集實例變更
- EC2競價型叢集 Spot 執行個體請求
- EC2發現艦隊狀態變更
- EC2競價型執行個體中斷
- EC2競價型執行個體請求
- EC2ODCR使用率不足通知

有關 Amazon 支持的事件類型的信息EBS，請參 EventBridge 閱 Amazon Amazon EBS。

使用記錄 Amazon EC2 API 呼叫 AWS CloudTrail

Amazon EC2 API 與服務集成在一起 AWS CloudTrail，該服務可提供用戶，角色或 AWS 服務. CloudTrail 將所有 Amazon EC2 API 呼叫作為事件捕獲。捕獲的呼叫包括控制台發出的呼叫。使用收集的資訊 CloudTrail，您可以判斷向 Amazon 發出的請求EC2API、提出請求的 IP 地址以及提出請求的時間。

每一筆事件或日誌專案都會包含產生請求者的資訊。身分資訊可協助您判斷下列事項：

- 該請求是使用根使用者還是使用者憑證提出。
- 是否代表IAM身分識別中心使用者提出要求。
- 提出該請求時，是否使用了特定角色或聯合身分使用者的暫時安全憑證。
- 該請求是否由另一項 AWS 服務服務提出。

CloudTrail 在您創建帳戶 AWS 帳戶 時處於活動狀態，並且您自動可以訪問 CloudTrail 事件歷史記錄。 CloudTrail 事件歷史記錄提供了過去 90 天中記錄的管理事件的可查看，可搜索，可下載和不可變的記錄。 AWS 區域若要取得更多資訊，請參閱《使用指南》中的〈AWS CloudTrail 使用 CloudTrail 事件歷程〉。查看活動歷史記錄不 CloudTrail收取任何費用。

如需過 AWS 帳戶 去 90 天內持續的事件記錄，請建立追蹤或 CloudTrailLake 事件資料存放區。

CloudTrail 小徑

追蹤可 CloudTrail 將日誌檔交付到 Amazon S3 儲存貯體。使用建立的所有系統線 AWS Management Console 都是多區域。您可以使用建立單一區域或多區域系統線。 AWS CLI建議您建立多區域追蹤，因為您會擷取帳戶 AWS 區域 中的所有活動。如果您建立單一區域追蹤，則只能檢視追蹤記錄中的 AWS 區域事件。如需有關追蹤的詳細資訊，請參閱《AWS CloudTrail 使用指南》中的「為您的建立追蹤」 AWS 帳戶和「為組織建立追蹤」。

您可以透 CloudTrail 過建立追蹤，免費將一份正在進行的管理事件副本傳遞到 Amazon S3 儲存貯體，但是需要支付 Amazon S3 儲存費用。如需有關 CloudTrail 定價的詳細資訊，請參閱AWS CloudTrail 定價。如需 Amazon S3 定價的相關資訊，請參閱 Amazon S3 定價。

CloudTrail 湖泊事件資料存放區

CloudTrail Lake 可讓您針對事件執行SQL基於查詢。 CloudTrail 湖泊將基於行的JSON格式現有的事件轉換為 Apache ORC 格式。ORC是一種針對快速擷取資料進行最佳化的單欄式儲存格式。系統會將事件彙總到事件資料存放區中，事件資料存放區是事件的不可變集合，其依據為您透過套用進階事件選取器選取的條件。套用於事件資料存放區的選取器控制哪些事件持續存在並可供您查詢。若要取得有關 CloudTrail Lake 的更多資訊，請參閱使用指南中的〈AWS CloudTrail 使用 AWS CloudTrail Lake 〉。

CloudTrail Lake 事件資料存放區和查詢會產生費用。建立事件資料存放區時，您可以選擇要用於事件資料存放區的定價選項。此定價選項將決定擷取和儲存事件的成本，以及事件資料存放區的預設和最長保留期。如需有關 CloudTrail 定價的詳細資訊，請參閱AWS CloudTrail 定價。

Amazon EC2 API 管理事件 CloudTrail

管理事件提供有關在您的資源上執行的管理作業的資訊 AWS 帳戶。這些也稱為控制平面操作。依預設，會 CloudTrail 記錄管理事件。

所有 Amazon EC2 API 動作都會記錄為管理事件。如需記錄到的API動作清單 CloudTrail，請參閱 Amazon EC2 API 參考。例如，對RunInstancesDescribeInstances、和StopInstances動作的呼叫會記錄為管理事件。

Amazon EC2 API 活動示例

事件代表來自任何來源的單一請求，包括有關請求的API操作，操作的日期和時間，請求參數等信息。CloudTrail 日誌文件不是公共API調用的有序堆棧跟踪，因此事件不會以任何特定順序出現。

以下日誌檔記錄顯示某位使用者終止了執行個體。

```
{
   "Records":[
      {
         "eventVersion":"1.03",
         "userIdentity":{
            "type":"Root",
            "principalId":"123456789012",
            "arn":"arn:aws:iam::123456789012:root",
            "accountId":"123456789012",
            "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
            "userName":"user"
         },
         "eventTime":"2016-05-20T08:27:45Z",
         "eventSource":"ec2.amazonaws.com",
         "eventName":"TerminateInstances",
         "awsRegion":"us-west-2",
         "sourceIPAddress":"198.51.100.1",
         "userAgent":"aws-cli/1.10.10 Python/2.7.9 Windows/7botocore/1.4.1",
         "requestParameters":{
            "instancesSet":{
               "items":[{
                  "instanceId":"i-1a2b3c4d"
               }]
            }
         },
         "responseElements":{
            "instancesSet":{
               "items":[{
                  "instanceId":"i-1a2b3c4d",
                  "currentState":{
                     "code":32,
                     "name":"shutting-down"
                  },
                  "previousState":{
                     "code":16,
                     "name":"running"
                  }
               }]
            }
         },
         "requestID":"be112233-1ba5-4ae0-8e2b-1c302EXAMPLE",
         "eventID":"6e12345-2a4e-417c-aa78-7594fEXAMPLE",
         "eventType":"AwsApiCall",
```

```
            "recipientAccountId":"123456789012"
        }
    ]
}
```

若要取得有關 CloudTrail 記錄內容的資訊，請參閱AWS CloudTrail 使用指南中的CloudTrail記錄內容。

使用EC2執行個體 Connect 線建立的稽核

您可以使用執 AWS CloudTrail 行個體連線來稽核 Connect 至執行個體的使用EC2者。

使用 AWS CloudTrail 主控台透過EC2執行個體 Connect 稽核SSH活動

1. 在開啟 CloudTrail 主控台https://console.aws.amazon.com/cloudtrail/。
2. 確認您位於正確的區域。
3. 在導覽窗格中，選擇 Event history (事件歷史記錄)。
4. 對於 Filter (篩選條件)，請選擇 Event source (事件來源)、ec2-instance-connect.amazonaws.com。
5. (選用) 針對 Time range (時間範圍)，選取時間範圍。
6. 選擇 Refresh events (重新整理事件) 圖示。
7. 此頁面會顯示對應於SendSSHPublicKeyAPI呼叫的事件。使用箭頭展開事件以檢視其他詳細資料，例如用於建立SSH連線的使用者名稱和 AWS 存取金鑰，以及來源 IP 位址。
8. 若要以JSON格式顯示完整的事件資訊，請選擇 [檢視事件]。此requestParameters欄位包含用來建立SSH連線的目的地執行個體 ID、作業系統使用者名稱和公開金鑰。

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "ABCDEFGONGNOMOOCB6XYTQEXAMPLE",
        "arn": "arn:aws:iam::1234567890120:user/IAM-friendly-name",
        "accountId": "123456789012",
        "accessKeyId": "ABCDEFGUKZHNAW4OSN2AEXAMPLE",
        "userName": "IAM-friendly-name",
        "sessionContext": {
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2018-09-21T21:37:58Z"}
```

```
        }
    },
    "eventTime": "2018-09-21T21:38:00Z",
    "eventSource": "ec2-instance-connect.amazonaws.com",
    "eventName": "SendSSHPublicKey ",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "123.456.789.012",
    "userAgent": "aws-cli/1.15.61 Python/2.7.10 Darwin/16.7.0 botocore/1.10.60",
    "requestParameters": {
        "instanceId": "i-0123456789EXAMPLE",
        "osUser": "ec2-user",
        "SSHKey": {
            "publicKey": "ssh-rsa ABCDEFGHIJKLMNO01234567890EXAMPLE"
        }
     },
    "responseElements": null,
    "requestID": "1a2s3d4f-bde6-11e8-a892-f7ec64543add",
    "eventID": "1a2w3d4r5-a88f-4e28-b3bf-30161f75be34",
    "eventType": "AwsApiCall",
    "recipientAccountId": "0987654321"
}
```

如果您已將 AWS 帳戶設定為收集 S3 儲存貯體中的 CloudTrail 事件，則可以透過程式設計方式下載和稽核資訊。若要取得更多資訊，請參閱《AWS CloudTrail 使用指南》中的〈取得和檢視 CloudTrail 記錄檔〉。

監視器。 NET和使用應用 CloudWatch 程式洞察的SQL伺服器

CloudWatch 應用程式深入解析可協助您監控. NET和使用 Amazon EC2 執行個體以及其他應用AWS程式資源的SQL伺服器應用程式。它可識別並設定應用程式資源和技術堆疊 (例如 Microsoft SQL Server 資料庫、Web (IIS) 和應用程式伺服器、作業系統、負載平衡器和佇列中的關鍵指標記錄和警示。並持續監控指標和日誌，以偵測和建立異常及錯誤的關聯。偵測到錯誤和異常時，「應用程式深入解析」會產生CloudWatch 事件，讓您用來設定通知或採取動作。為協助故障診斷，它會建立已偵測到問題的自動化儀表板，包括關聯的指標異常和日誌錯誤以及其他洞見，指出可能的根本原因。自動化儀表板可協助您採用靈活的補救動作，讓應用程式保持良好的運作狀態，防止應用程式的最終使用者受到影響。

若要檢視支援的日誌和指標的完整清單，請參閱 Amazon CloudWatch 應用程式洞察支援的日誌和指標。

有關偵測到的問題資訊：

- 問題的簡短摘要
- 問題的開始時間和日期
- 問題嚴重性：高/中/低
- 偵測到的問題狀態：正在進行/已解決
- 洞見：對所偵測問題及可能根本原因自動產生的洞見
- 洞察意見回饋：您提供的意見反 CloudWatch 應，瞭解應用程式洞察所產生的見解的實用性。 NET 和SQL伺服器
- 相關觀察：詳細檢視與跨各種應用程式元件問題有關的相關日誌指標異常和錯誤程式碼片段

意見回饋

您可以指出針對所偵測問題自動產生的洞見是否有用，提供意見回饋。您對洞見的意見回饋以及您的應用程式診斷 (指標異常和日誌例外狀況)，都會用來提升未來對類似問題的偵測。

如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的應用CloudWatch程式洞察文件。

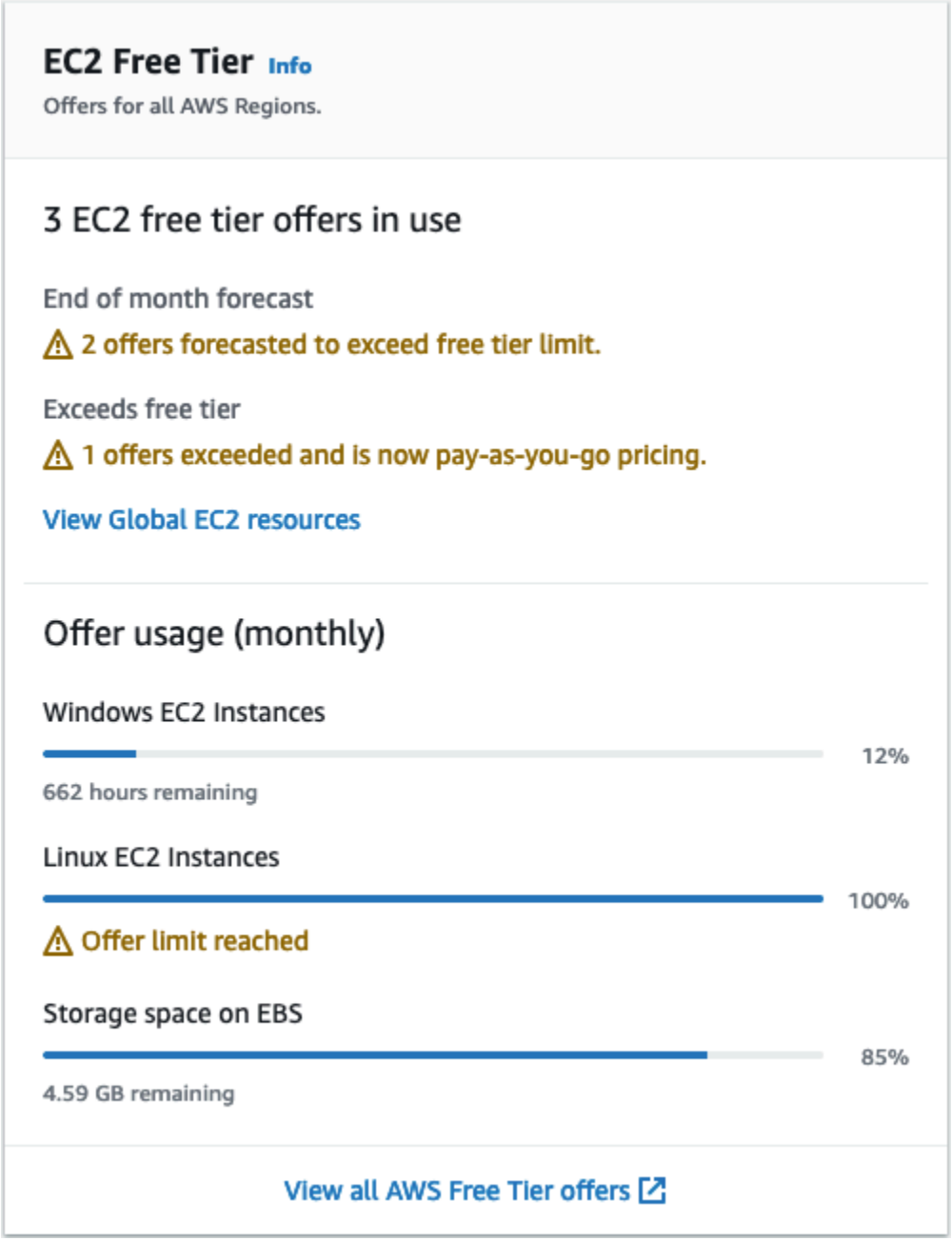
追蹤您的 Amazon 免費方案使用情況 EC2

如果您是 AWS 客戶少於 12 個月並且保持在用 AWS 免費方案 量限制內，則可以使用 Amazon EC2 而不會產生任何費用。請務必要追蹤免費方案的用量，以避免出現意外的帳單費用。如果您超過免費方案限制，就會產生標準 pay-as-go 費用。如需詳細資訊，請參閱AWS 免費方案。

> **Note**
> 如果您成為 AWS 客戶超過 12 個月，就不再符合使用免費方案的資格，也不會看到下列程序中所述的EC2免費方案方塊。

追蹤您的免費方案用量

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 [EC2儀表板]。
3. 找到EC2免費方案方塊（位於右上角）。



4. 在「EC2免費方案」方塊中，檢查您的免費方案用量，如下所示：

   - 在使用中的EC2免費方案優惠下，請注意以下警告：
     - 月底預測 – 這項警告會提醒您，若您繼續目前的用量模式，則本月將會產生費用。
     - 超過免費方案 – 這項警告會提醒您，已超過了免費方案限制且已產生費用。
   - 在 [提供使用量 (每月)] 下方，記下您對 Linux 執行個體、Windows 執行個體和EBS儲存體的使用情況。該百分比表示您本月使用了多少免費方案限制。如果您達到 100%，則進一步使用將會產生費用。

> **Note**
>
> 本資訊僅限於在您建立了執行個體之後才會顯示。不過，用量資訊並非即時更新；其更新頻率為一天三次。

5. 為了避免產生進一步的費用，請刪除任何現在產生費用的資源，否則若超過免費方案限制用量，將會產生費用。
   - 如需刪除執行個體的指示，請參閱終止 Amazon EC2 實例。
   - 若要檢查其他區域中是否有可能產生費用的資源，請在 [EC2免費方案] 方塊中選擇 [檢視全域 EC2資源] 以開啟 [EC2全域檢視]。如需詳細資訊，請參閱使用 Amazon EC2 全球視圖查看跨區域的資源。
6. 若要在 AWS 服務 [EC2免費方案] 方塊底部檢視所有人的資源使用量，請選擇 [檢視所 AWS 免費方案 有選件]。 AWS 免費方案如需詳細資訊，請參閱《AWS 帳單使用者指南》中的使用 AWS 免費方案。

疑難排解 Amazon EC2 執行個體問題

下列程序和秘訣可協助您疑難排解 Amazon EC2 執行個體的問題。

問題

- 解決 Amazon EC2 執行個體啟動問題
- 解決 Amazon EC2 執行個體停止問題
- 解決 Amazon EC2 執行個體終止問題
- 疑難排解無法存取的 Amazon EC2 執行個
- 解決連線到 Amazon EC2 Linux 執行個體的問題
- 使用失敗的狀態檢查進行 Amazon EC2 Linux 執行個體
- 疑難排解從錯誤的磁碟區開機的 Amazon EC2 Linux 執行個
- 疑難排解連線到 Amazon 視EC2窗執行個體的
- Amazon EC2 視窗執行個體啟動問題
- 解決 Amazon EC2 視窗執行個體問題
- 重設 Amazon EC2 視窗執行個體的管理員密碼
- 疑難排解 Amazon EC2 視窗執行個體的系統問題
- 使用疑難排解受損的 Amazon EC2 Linux EC2Rescue
- 使用疑難排解受損的 Amazon EC2 Windows EC2Rescue
- EC2實例的序列主控台
- 傳送診斷中斷以偵錯無法連線的 Amazon EC2 執行個體

解決 Amazon EC2 執行個體啟動問題

以下是可協助您解決啟動 Amazon EC2 執行個體時問題的疑難排解秘訣。

啟動問題

- 無效裝置名稱
- 超過執行個體限制
- 執行個體容量不足
- 目前並不支援要求的組態。請查看文件以了解支援的組態。
- 立刻終止執行個體

- 許可不足
- Windows 啟動後不久CPU使用率高 (僅適用於 Windows 執行個體)

無效裝置名稱

描述

當您嘗試啟動新的執行個體時，發生 `Invalid device name` *`device_name`* 錯誤。

原因

如果您在嘗試啟動執行個體時收到此錯誤，表示在請求中為一個或多個磁碟區指定的裝置名稱具有無效的裝置名稱。可能的原因包括：

- 選取的裝置可能正在使用該裝置名稱AMI。
- 裝置名稱可能會保留給根磁碟區。
- 裝置名稱可能會用於請求中的其他磁碟區。
- 裝置名稱可能對作業系統無效。

解決方案

要解決問題：

- 請確定您選取的裝置名稱AMI未使用。執行下列命令以檢視使用的裝置名稱AMI。

```
aws ec2 describe-images --image-id ami_id --query
 'Images[*].BlockDeviceMappings[].DeviceName'
```

- 請確保您沒有使用為根磁碟區保留的裝置名稱。如需詳細資訊，請參閱可用裝置名稱。
- 請確保請求中指定的每個磁碟區都有唯一的裝置名稱。
- 請確保您指定的裝置名稱格式正確。如需詳細資訊，請參閱可用裝置名稱。

超過執行個體限制

描述

當您嘗試啟動新的執行個體或重新啟動已停止的執行個體時，發生 `InstanceLimitExceeded` 錯誤。

原因

當您嘗試啓動新的執行個體或重新啟動已停止的執行個體時，如果發生 `InstanceLimitExceeded` 錯誤，就表示已達到您在區域中可啟動的執行個體數目限制。建立 AWS 帳戶時，我們會針對每個區域可執行的執行個體數量設定預設限制。

解決方案

您可以要求提高每一區域的執行個體限制。如需詳細資訊，請參閱 Amazon EC2 服務配額。

執行個體容量不足

描述

當您嘗試啟動新的執行個體或重新啟動已停止的執行個體時，發生 `InsufficientInstanceCapacity` 錯誤。

原因

當您嘗試啓動執行個體或重新啟動已停止的執行個體時，如果發生此錯誤，就表示 AWS 目前可用的隨需容量不足，無法滿足您的請求。

解決方案

若要解決問題，請嘗試下列方法：

- 等候幾分鐘，然後再次提交您的請求；容量會頻繁轉移。
- 以降低的執行個體數提交新請求。例如，如果您要提出一次啟動 15 個執行個體的請求，請改嘗試提出 3 個啟動 5 個執行個體的請求，或 15 個啟動 1 個執行個體的請求。
- 如果啟動執行個體，請提交新的請求，而不要指定可用區域。
- 如果啟動執行個體，請使用不同的執行個體類型 (您可以在後面的階段調整大小) 來提交新的請求。如需詳細資訊，請參閱 Amazon EC2 實例類型更改。
- 如果透過叢集置放群組來啟動執行個體，可能會出現容量不足的錯誤。

目前並不支援要求的組態。請查看文件以了解支援的組態。

描述

當您嘗試啟動新的執行個體時，由於不支援執行個體組態，因此發生 `Unsupported` 錯誤。

原因

錯誤訊息會提供其他詳細資訊。例如，指定的區域或可用區域可能不支援執行個體類型或執行個體購買選項。

解決方案

請嘗試不同的執行個體組態。若要搜尋符合您需求的執行個體類型，請參閱 尋找 Amazon 執行個EC2體類型。

立刻終止執行個體

描述

您的執行個體會從 `pending` 狀態進入 `terminated` 狀態。

原因

下列是執行個體為什麼可能會立即終止的幾個原因：

- 你已超出音EBS量限制。如需詳細資訊，請參閱Amazon EC2 實例的EBS數量限制。
- EBS快照已損毀。
- 根EBS磁碟區已加密，您沒有存取要解密的KMS金鑰的權限。
- 在為的區塊裝置對應中指定的快照AMI已加密，您沒有存取要解密的KMS金鑰的權限，或者您無權存取加密還原磁碟區的KMS金鑰。
- 您用來啟動執行個體AMI的執行個體存放區支援缺少必要的部分 (image.part. xx 檔案）。

如需詳細資訊，請使用下列其中一種方法來查明終止原因。

使用 Amazon EC2 控制台獲取終止原因

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)，然後選取該執行個體。
3. 在第一個索引標籤上，找出狀態轉換原因旁的原因。

若要取得終止原因，請使用 AWS Command Line Interface

1. 使用 describe-instances 命令並指定執行個體 ID。

```
aws ec2 describe-instances --instance-id instance_id
```

2. 檢閱指令傳JSON回的回應，並記下回StateReason應元素中的值。

   下列程式碼區塊顯示 StateReason 回應元素的範例。

```
"StateReason": {
  "Message": "Client.VolumeLimitExceeded: Volume limit exceeded",
  "Code": "Server.InternalError"
},
```

要獲取終止原因 AWS CloudTrail

如需詳細資訊，請參閱AWS CloudTrail 使用指南中的檢視具有 CloudTrail 事件歷程記錄的事件。

解決方案

根據終止原因，採取下列其中一項動作：

- **`Client.VolumeLimitExceeded: Volume limit exceeded`** — 刪除未使用的磁碟區。您可以提交申請來增加磁碟區容量上限。
- **`Client.InternalError: Client error on launch`**— 確定您具有存取 AWS KMS keys 用於解密和加密磁碟區的所需權限。如需詳細資訊，請參閱 AWS Key Management Service 開發人員指南中的 在 AWS KMS中使用金鑰政策。

許可不足

描述

當您嘗試啟動新的執行個體時發生 `"errorMessage": "You are not authorized to perform this operation."` 錯誤，並且啟動失敗。

原因

如果您在嘗試啟動執行個體時看到這個錯誤，表示您沒有啟動執行個體的必要IAM權限。

可能缺少的許可包括：

- `ec2:RunInstances`

- `iam:PassRole`

可能還缺少其他許可。如需啟動執行個體所需的權限清單，請參閱範例：使用EC2啟動執行個體精靈和下方的範例IAM政策啟動執行個體 (RunInstances)。

解決方案

要解決問題：

- 如果您以IAM使用者身分提出要求，請確認您具有下列權限：
  - `ec2:RunInstances` 使用萬用字元資源 (「*」)
  - `iam:PassRole`使用符合角色的資源 ARN (例如，`arn:aws:iam::999999999999:role/ExampleRoleName`)
- 如果您沒有先前的權限，請編輯與IAM角色或使用者相關聯的IAM原則，以新增缺少的必要權限。

如果您的問題尚未解決，且繼續收到啟動失敗錯誤，可以解碼錯誤中包含的授權失敗訊息。解碼的訊息包含原則中遺失的IAM權限。如需詳細資訊，請參閱如何在EC2執行個體啟動期間收到「UnauthorizedOperation」錯誤後解碼授權失敗訊息？

Windows 啟動後不久CPU使用率高 (僅適用於 Windows 執行個體)

> **Note**
>
> 此疑難排解提示僅適用於 Windows 執行個體。

如果 Windows Update 設置為檢查更新，但讓我選擇是否下載並安裝它們（默認實例設置），則此檢查可以消耗實例CPU上 50-99% 的任何位置。如果此CPU消耗導致應用程式發生問題，您可以在控制台中手動變更 Windows Update 設定，或者您可以在 Amazon 使用EC2者資料欄位中使用下列指令碼：

```
reg add "HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\WindowsUpdate\Auto Update" /v
 AUOptions /t REG_DWORD /d 3 /f net stop wuauserv net start wuauserv
```

當您執行此指令碼時，請指定 /d 的值。預設值為 3。可能的值包括以下：

1. 絕不檢查更新

2. 檢查是否有更新，但讓我選擇是否下載並安裝
3. 下載更新，但讓我選擇是否安裝
4. 自動安裝更新

您修改執行個體的使用者資料之後，就可以執行它。如需詳細資訊，請參閱啟動時在 Windows 執行個體上執行命令。

解決 Amazon EC2 執行個體停止問題

如果您已停止 Amazon EBS 支援的執行個體，且該執行個體出現停滯在stopping狀態中，則基礎主機電腦可能存在問題。首先，嘗試強制停止實例。如果執行個體沒有停止，您可以要求解決此問題的協助。

當執行個體處於 stopping 狀態或 running 以外的任何狀態時，將不會收取執行個體使用費。當執行個體處於 running 狀態時，您只需支付執行個體使用費。

目錄

- 強制停止執行個體
- (選擇性) 建立取代執行個體

強制停止執行個體

請使用主控台或 AWS CLI來強制停止執行個體。

> **Note**
>
> 只有在執行個體為 stopping 狀態時，您才可以使用主控台強制停止執行個體。不論執行個體狀態為何 (shutting-down 和 terminated 除外)，您都可使用 AWS CLI 強制停用執行個體。

Console

使用主控台強制停止執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。

2. 在導覽窗格中，選擇 Instances (執行個體)，然後選取凍結的執行個體。
3. 選擇「Instance state (執行個體狀態)」，「Force stop instance (強制停止執行個體)」，「Stop (停止)」。

   請注意，唯有在您的執行個體處於 `stopping` 狀態的情況下，才能在主控台使用 Force stop instance (強制停止執行個體)。如果您的執行個體處於另一個狀態 (除了`shutting-down`和`terminated`)，您可以使用 AWS CLI 來強制停止執行個體。

AWS CLI

若要使用強制停止執行個體 AWS CLI

如下所示使用 stop-instances 命令和 `--force` 選項：

```
aws ec2 stop-instances --instance-ids i-0123ab456c789d01e --force
```

如果經過 10 分鐘後，執行個體仍未停止，請在 AWS re:Post 上發布請求協助的訊息。請在訊息中註明執行個體 ID，並說明您已採取的步驟，以利加速解決問題。或者，如果您已購買支援方案，可到 Support 中心建立技術支援案例。

(選擇性) 建立取代執行個體

在等待支援中心AWS re:Post或 Sup port 中心的協助時，您可以視需要建立替代執行個體。AMI從卡住的實例創建一個，然後使用 new 啟動一個新的實例AMI。

> **Important**
>
> 如果卡住的執行個體只會產生系統狀態檢查，您可以建立取代執行個體，因為執行個體狀態檢查會導致AMI複製損毀作業系統的精確複本。確認狀態訊息後，請使用新的執行個體建立AMI並啟動新的執行個體AMI。

Console

使用主控台來建立替換執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)，然後選取凍結的執行個體。

3. 選擇 Actions (動作)、Image and templates (映像和範本)、Create image (建立映像)。
4. 在「Create image (建立映像)」頁面，執行下列動作：

   a. 輸入的名稱和說明AMI。

   b. 清除重新啟動實例。

   c. 選擇建立映像。

   如需詳細資訊，請參閱the section called “從執行AMI個體建立”。
5. 從啟動新執行個體，AMI並確認新執行個體是否正常運作。
6. 選取凍結的執行個體，然後依序選擇「Actions (動作)」、「Instance state (執行個體狀態)」、「Terminate instance (終止執行個體)」。如果執行個體也卡住終止，Amazon EC2 會自動強制執行個體在幾個小時內終止。

AWS CLI

若要使用建立取代例證 CLI

1. 使用 create-image (AWS CLI) 命令和`--no-reboot`選項，AMI從卡住的實例創建一個如下：。

```
aws ec2 create-image --instance-id i-0123ab456c789d01e --name "AMI" --
description "AMI for replacement instance" --no-reboot
```

2. 從AMI使用 run-instance (AWS CLI) 命令啟動新的執行個體，如下所示：

```
aws ec2 run-instances --image-id ami-1a2b3c4d --count 1 --instance-type c3.large
 --key-name MyKeyPair --security-groups MySecurityGroup
```

3. 確認新的執行個體可正常運作。
4. 利用 terminate-instances (AWS CLI) 命令來終止凍結的執行個體，如下所示：

```
aws ec2 terminate-instances --instance-ids i-1234567890abcdef0
```

如果您無法依照先前程序所述AMI從執行個體建立執行個體，您可以依照下列方式設定取代執行個體：

(替代選項) 使用主控台來建立替換執行個體

1. 選取執行個體，然後依序選擇 Description (說明)、Block devices (區塊型儲存設備)。選取每個磁碟區，並記下其磁碟區 ID。請務必註明哪一個磁碟區是根磁碟區。
2. 在導覽窗格中，選擇 Volumes (磁碟區)。針對執行個體選取每個磁碟區，然後依序選擇 Actions (動作)、Create Snapshot (建立快照)。
3. 在導覽窗格中，選擇 Snapshots (快照)。選擇您剛建立的快照，然後依序選擇 Actions (動作)、Create Volume (建立磁碟區)。
4. 使用與凍結執行個體相同的作業系統來啟動執行個體。請寫下其根磁碟區的磁碟區 ID 和裝置名稱。
5. 在導覽窗格中，選擇「Instances (執行個體)」，選取剛才啟動的執行個體，然後依序選擇「Instance state (執行個體狀態)」、「Stop instance (停止執行個體)」。
6. 在導覽窗格中選擇 Volumes (磁碟區)，選取已停止執行個體的根磁碟區，然後依序選擇 Actions (動作)、Detach Volume (分離磁碟區)。
7. 選取從凍結執行個體建立的根磁碟區，然後依序選擇「Actions (動作)」、「Attach Volume (連結磁碟區)」，接著將該磁碟區連結到新的執行個體，並做為其根磁碟區 (使用您之前記下的裝置名稱)。將其他所有非根磁碟區的磁碟區，連結到執行個體。
8. 在導覽窗格中，選擇 Instances (執行個體)，然後選取替換個體。選擇 Instance state (執行個體狀態)、Start instance (啟動執行個體)。確認執行個體可正常運作。
9. 選取凍結的執行個體，然後依序選擇 Instance state (執行個體狀態)、Terminate instance (終止執行個體)。如果執行個體也卡住終止，Amazon EC2 會自動強制執行個體在幾個小時內終止。

解決 Amazon EC2 執行個體終止問題

關閉或刪除執行個體稱為執行個體終止。下列資訊可協助您疑難排解終止執行個體時的問題。

當執行個體未處於 `running` 狀態時，將不會向您收取任何執行個體使用費。換句話說，當您終止執行個體時，只要其狀態變更為 `shutting-down`，即刻起就會停止收取該執行個體所產生的費用。

立刻終止執行個體

若干問題可能會導致執行個體在啟動時立即終止。如需詳細資訊，請參閱「立刻終止執行個體」。

延遲的執行個體終止

如果執行個體處於 `shutting-down` 狀態超過幾分鐘，可能是該執行個體正在執行的關閉指令碼造成了延遲。

另一個可能的原因是底層主機電腦的問題。如果您的執行個體維持在`shutting-down`狀態數小時，Amazon EC2 會將其視為卡住的執行個體，並強制終止執行個體。

如果執行個體似乎在終止時凍結，而且已經持續超過好幾個小時，請在 AWS re:Post 發佈求助訊息。請在訊息中註明執行個體 ID，並說明您已採取的步驟，以利加速解決問題。或者，如果您已購買支援方案，可到 Support 中心建立技術支援案例。

仍顯示已終止的執行個體

在執行個體終止後，仍會短暫顯示，然後才會遭到刪除。狀態會顯示為 `terminated`。如果經過好幾個小時仍未刪除此項目，請聯絡 Support 中心。

錯誤：執行個體可能無法終止。修改其 'disableApiTermination' 實例屬性

如果您嘗試終止執行個體並收到 `The instance instance_id may not be terminated. Modify its 'disableApiTermination' instance attribute` 錯誤訊息，則表示執行個體已啟用終止保護。終止保護可防止執行個體被意外終止。如需詳細資訊，請參閱啟用終止保護。

您必須先停用終止保護，才能終止執行個體。

若要使用 Amazon EC2 主控台停用終止保護，請選取執行個體，然後選擇 [動作]、[執行個體設定]、[變更終止保護]

若要使用停用終止保護 AWS CLI，請使用下列命令。

```
aws ec2 modify-instance-attribute --instance-id instance_id --no-disable-api-
termination
```

執行個體自動啟動或終止

一般而言，下列行為表示您已使用 Amazon EC2 Auto Scaling、EC2叢集或 Spot 叢集，根據您定義的準則自動擴展運算資源：

- 您終止某個執行個體，且新執行個體自動啟動。

- 您啟動某個執行個體，且其中一個執行個體自動終止。
- 您停止某個執行個體，該執行個體終止，且新執行個體自動啟動。

若要停止自動調整規模，請尋找 Auto Scaling 群組或正在啟動執行個體的叢集，並將其容量設為 0 或刪除它。

疑難排解無法存取的 Amazon EC2 執行個

下列資訊可協助您疑難排解無法存取的 Amazon EC2 執行個體。您可以擷取螢幕擷取畫面或存取主控台輸出，以協助診斷問題，並判斷是否應重新啟動執行個體。對於無法存取的 Windows 執行個體，請檢閱服務傳回的螢幕擷取畫面進行疑難

目錄

- 重新啟動執行個體
- 執行個體主控台輸出
- 擷取無法連線執行個體的螢幕擷取畫面
- 疑難排解無法存取的 Windows 執行個體的常見
- 當主機電腦故障時的執行個體復原

重新啟動執行個體

對於故障診斷和一般的執行個體管理，能夠重新啟動無法連線的執行個體很重要。

就像您可以通過按重置按鈕重置計算機一樣，您可以使用 Amazon EC2 控制台CLI，或重置EC2實例API。如需詳細資訊，請參閱重新啟動您的執行個體。

執行個體主控台輸出

主控台輸出是診斷問題的一項寶貴工具。對於疑難排解可能導致執行個體終止或無法連線的核心問題和服務組態問題，這項功SSH能特別有用。

- Linux 執行個體 — 執行個體主控台輸出會顯示通常會在連接至電腦的實體監視器上顯示的確切主控台輸出。主控台輸出會傳回經過緩衝的資訊，這些資訊會在執行個體轉換狀態 (啟動、停止、重新啟動和終止) 之後不久發布。已發布的輸出不會連續更新，只有在可能具有最高價值時才會更新。
- Windows 執行個體 — 執行個體主控台輸出包含最後三個系統事件記錄錯誤。

只有執行個體才能存取主控台輸出資料。

您可以在執行個體生命週期中擷取最新的序列主控台輸出。只有在 AWS Nitro 系統上建置的執行個體上才支援此選項。它不是通過 Amazon EC2 控制台支持。

Console

無法取得主控台輸出

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側導覽窗格中，選擇執行個體。
3. 選取執行個體，然後再依序選擇動作、監控和故障診斷和取得系統記錄檔。

Command line

無法取得主控台輸出

您可以使用下列其中一個命令。如需有關這些命令列介面的詳細資訊，請參閱訪問 Amazon EC2。

- get-console-output (AWS CLI)
- Get-EC2ConsoleOutput (AWS Tools for Windows PowerShell)

擷取無法連線執行個體的螢幕擷取畫面

如果您無法連線至執行個體，您可以擷取執行個體的螢幕擷取畫面，並以映像檔的形式檢視。此影像可讓您看到執行個體的狀態，有利於更快進行故障診斷。

您可以在執行個體執行中或執行個體當機後產生螢幕擷取畫面。影像會以JPG格式產生，且不大於 100 kb。螢幕擷取畫面不收取資料傳輸費用。

限制

下列項目不支援此功能：

- 裸機執行個體 (`*.metal` 類型的執行個體)
- 實例正在使用NVIDIAGRID驅動程序
- 以 ARM 為基礎的重力子處理器提供支援的執行個體
- 視窗執行個體 AWS Outposts

- L AWS ocal Zones 上的 Windows 執行個體

支援的區域

此功能在以下 區域可用：

- US East (N. Virginia) Region
- 美國東部 (俄亥俄) 區域
- 美國西部 (加利佛尼亞北部) 區域
- 美國西部 (奧勒岡) 區域
- 非洲 (開普敦) 區域
- 亞太區域 (香港) 區域
- 亞太區域 (海德拉巴)
- 亞太區域 (雅加達)
- 亞太區域 (墨爾本) 區域
- 亞太 (孟買) 區域
- 亞太 (大阪) 區域
- 亞太 (首爾) 區域
- 亞太區域 (新加坡) 區域
- 亞太 (雪梨) 區域
- 亞太 (東京) 區域
- 加拿大 (中部) 區域
- 加拿大西部 (卡加利) 區域
- 中國 (北京) 區域
- 中國 (寧夏) 區域
- 歐洲 (法蘭克福) 區域
- 歐洲 (愛爾蘭) 區域
- 歐洲 (倫敦) 區域
- Europe (Milan) Region
- 歐洲 (巴黎) 區域
- 歐洲 (西班牙) 區域

- 歐洲 (斯德哥爾摩) 區域
- 歐洲 (蘇黎世) 區域
- 以色列 (特拉維夫) 區域
- 南美洲 (聖保羅) 區域
- Middle East (Bahrain) Region
- 中東 (UAE) 區域

Console

取得執行個體的螢幕擷取畫面

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側導覽窗格中選擇 (執行個體)。
3. 選取要擷取的執行個體。
4. 選擇 動作、監視和故障診斷、取得執行個體螢幕截圖。
5. 選擇 下載，或在要下載的影像上按一下滑鼠右鍵並儲存。

Command line

擷取執行個體的螢幕擷取畫面

您可以使用下列其中一個命令。傳回的內容為 base64 編碼。如需有關這些命令列介面的詳細資訊，請參閱訪問 Amazon EC2。

- get-console-screenshot (AWS CLI)
- GetConsoleScreenshot（Amazon EC2 查詢API）

疑難排解無法存取的 Windows 執行個體的常見

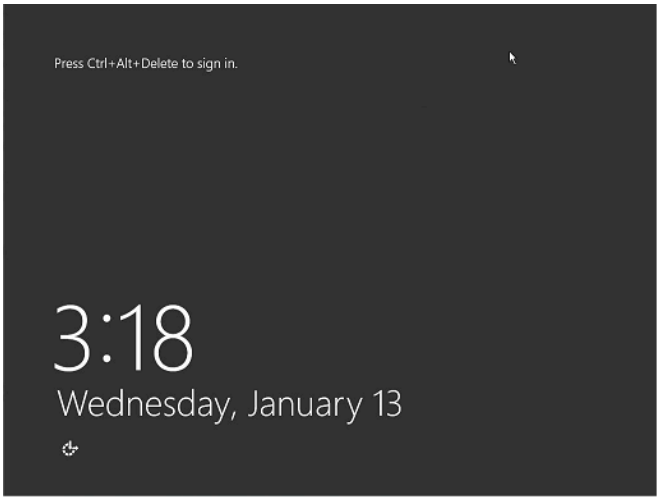
您可根據服務傳回的螢幕截取畫面，使用下列資訊協助您故障診斷無法連線的 Windows 執行個體。

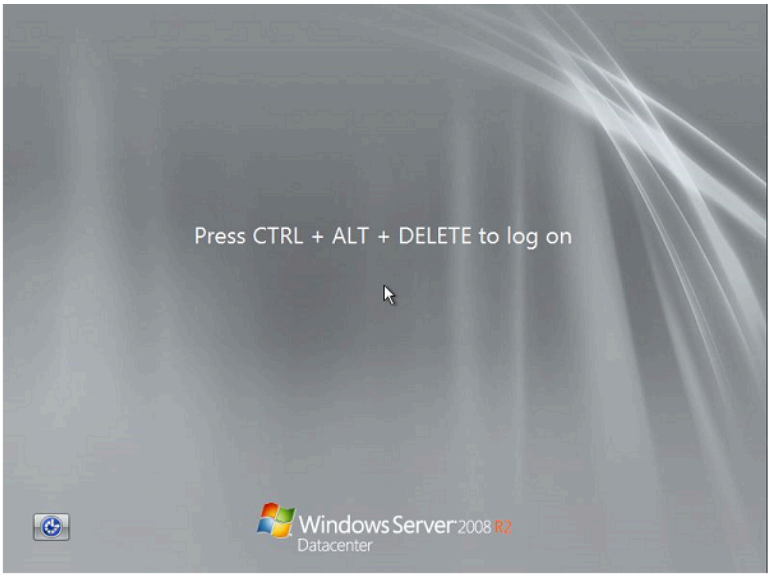
- 登入畫面 (Ctrl+Alt+Delete)
- 復原主控台螢幕
- Windows 開機管理程式畫面
- Sysprep 畫面

- 就緒畫面
- Windows Update 畫面
- Chkdsk

登入畫面 (Ctrl+Alt+Delete)

主控台螢幕截取畫面服務傳回下列內容。





如果在登入期間無法連線某個執行個體，可能是您的網路組態或 Windows 遠端桌面服務發生問題。如果程序使用大量的CPU執行個體，則執行個體也可能沒有回應。

網路組態

請使用下列資訊來確認您 AWS的 Microsoft Windows 和本機 (或內部部署) 網路組態並未封鎖對執行個體的存取。

AWS 網路組態

組態	確認
安全群組組態	確認連接埠 3389 已開放給您的安全群組。確認您連線到正確的公有 IP 地址。如果執行個體與彈性 IP 地址無關聯，則公有 IP 地址會在執行個體停止/啟動後變更。如需詳細資訊，請參閱遠端桌面無法連線到遠端電腦。

組態	確認
VPC配置（網絡ACLs）	確認 Amazon 的存取控制清單 (ACL) VPC 未封鎖存取。如需詳細資訊，請參閱 Amazon VPC 使用者指南ACLs中的網路。
VPN配置	如果您VPC使用虛擬私人網路 (VPN) 連線到您的，請驗證VPN通道連線。如需詳細資訊，請參閱如何疑難排解與 Amazon 的VPN隧道連線問題VPC？

Windows 網路組態

組態	確認
Windows 防火牆	確認 Windows 防火牆未封鎖對您執行個體的連線。如遠端桌面故障診斷一節遠端桌面無法連線到遠端電腦第 7 項所述，停用 Windows 防火牆。
進階 TCP /IP 設定 (使用靜態 IP)	因為您設定了靜態 IP 地址，所以執行個體可能不回應。對於 aVPC，請建立網路介面並將其附加至執行個體。

本機或內部部署的網路組態

確認本機網路組態未封鎖存取。嘗試連接到與無法訪問的實VPC例相同的另一個實例。如果您無法存取另一個執行個體，請與您的本機網路管理員合作，判斷本機政策是否限制存取。

遠端桌面服務問題

如果在登入期間無法連線到執行個體，則執行個體上的遠端桌面服務 (RDS) 可能發生問題。

> **Tip**
>
> 您可以使用 `AWSSupport-TroubleshootRDP` runbook 來檢查和修改可能會影響遠端桌面通訊協定 (RDP) 連線的各種設定。如需詳細資訊，請參閱 AWS Systems Manager Automation Runbook 參考 中的 `AWSSupport-TroubleshootRDP`。

遠端桌面服務組態

組態	確認
RDS正在執行	確認執行個體RDS正在執行個體上執行。使用 Microsoft 管理主控台 (MMC) 服務嵌入式管理單元 () Connect 線至執行個體。`services.msc` 在服務清單中，確認 Remote Desktop Services (遠端桌面服務) 為 Running (執行中)。如果不是，請啟動它，然後將啟動類型設為 Automatic (自動)。如果您無法使用 [服務] 嵌入式管理單元連線至執行個體，請從執行個體卸離根磁碟區、建立磁碟區的快照或AMI從該磁碟區建立、將原始磁碟區附加到相同可用區域中的另一個執行個體做為輔助磁碟區，然後修改 Start 登錄機碼。完成後，重新將根磁碟區連接到原始執行個體。
RDS已啟用	即使服務已啟動，仍可能停用。從執行個體中分離根磁碟區、建立磁碟區的快照或建立AMI磁碟區、將原始磁碟區連接到相同可用區域中的另一個執行個體做為輔助磁碟區，以及如中所述修改終端機伺服器登錄機碼以啟用服務在具有遠端登錄的EC2執行個體上啟用遠端桌面。 完成後，重新將根磁碟區連接到原始執行個體。

高CPU使用率

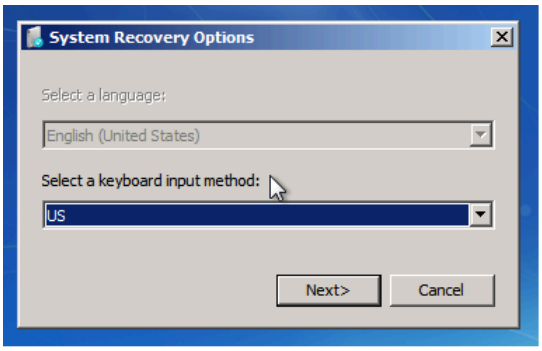
使用 Amazon 檢查執行個體上的 CPUUtilization(最大值) 指標 CloudWatch。如果 CPUUtilization（最大值）是很高的數字，請等待下CPU降，然後嘗試再次連接。高CPU使用率可能是由以下原因造成：

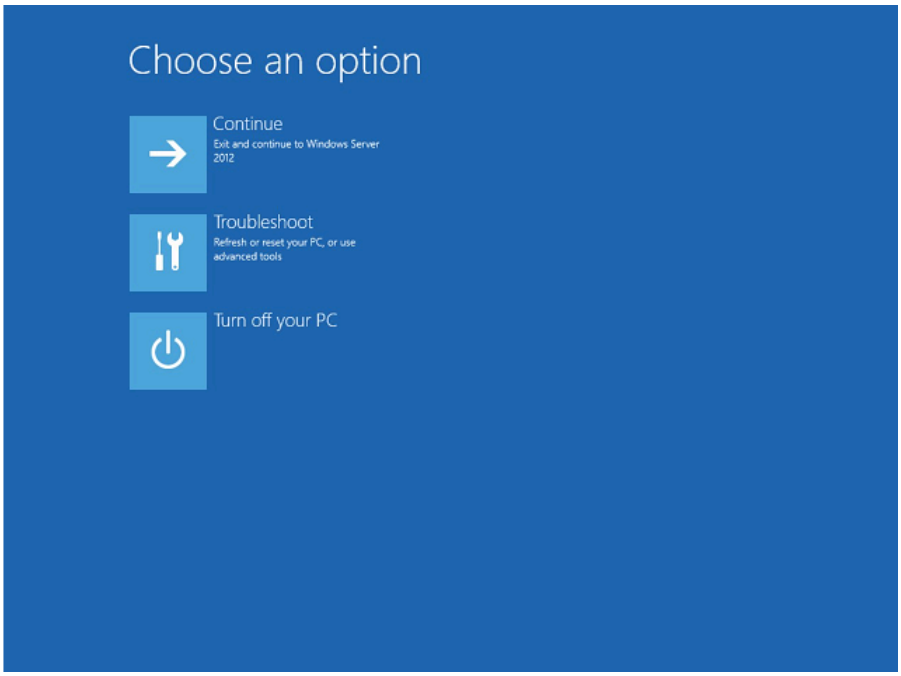
- Windows 更新
- 安全軟體掃描
- 自訂啟動指令碼
- 任務排程

如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的取得特定資源的統計資料。如需其他的故障診斷秘訣，請參閱 Windows 啟動後不久CPU使用率高 (僅適用於 Windows 執行個體)。

復原主控台螢幕

主控台螢幕截取畫面服務傳回下列內容。





作業系統可能在復原主控台中開機，然後卡在這個狀態，如果 `bootstatuspolicy` 不是設為 `ignoreallfailures`。請使用下列程序將 `bootstatuspolicy` 組態變更為 `ignoreallfailures`。

依預設，所AMIs提供之公用視窗的原則組態設定為`ignoreallfailures`。 AWS

1. 停止無法連線的執行個體。
2. 建立根磁碟區的快照。根磁碟區連接至此執行個體做為 `/dev/sda1`。

   將根磁碟區從無法連接的執行個體中分離、建立磁碟區的快照或AMI從該磁碟區建立快照，然後將其作為輔助磁碟區附加至相同可用區域中的另一個執行個體。

   > **Warning**
   >
   > 如果您的暫存執行個體和原始執行個體是使用相同的執行個體啟動AMI，您必須完成其他步驟，否則您將無法在還原其根磁碟區後，因為磁碟簽章衝突而無法啟動原始執行個體。如果您必須使用相同的執行個體建立暫存執行個體AMI，為避免磁碟簽章衝突，請完成中的步驟磁碟簽章衝突。

> 或者，為暫存例證選AMI取不同的執行個體。例如，如果原始執行個體使用適用AMI於視窗伺服器 2016，請使用適用AMI於 Windows 伺服器 2019 的啟動暫存執行個體。

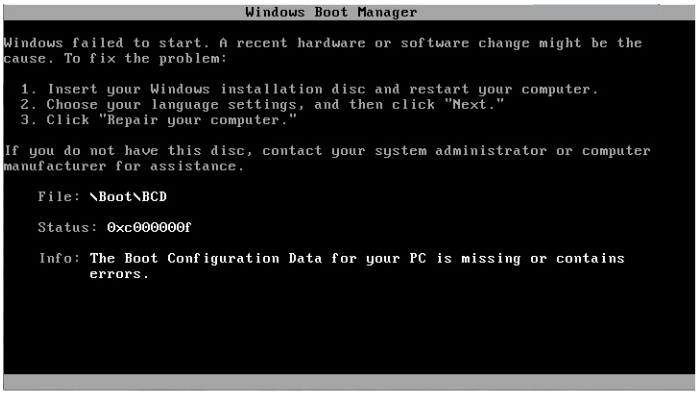
3. 登入執行個體，從命令提示執行下列命令以將 `bootstatuspolicy` 組態變更為 `ignoreallfailures`。

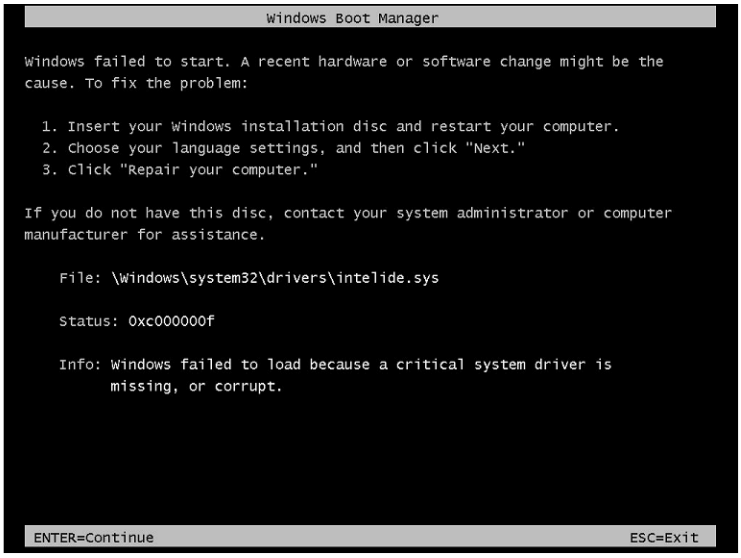
```
bcdedit /store Drive Letter:\boot\bcd /set {default} bootstatuspolicy
 ignoreallfailures
```

4. 將磁碟區重新連接至無法連線的執行個體，然後再次啟動執行個體。

Windows 開機管理程式畫面

主控台螢幕截取畫面服務傳回下列內容。

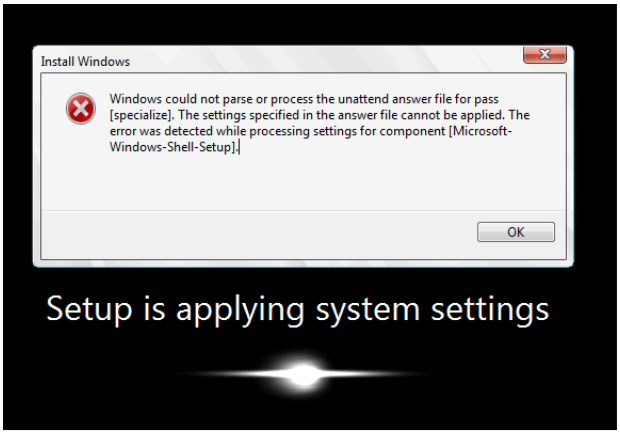


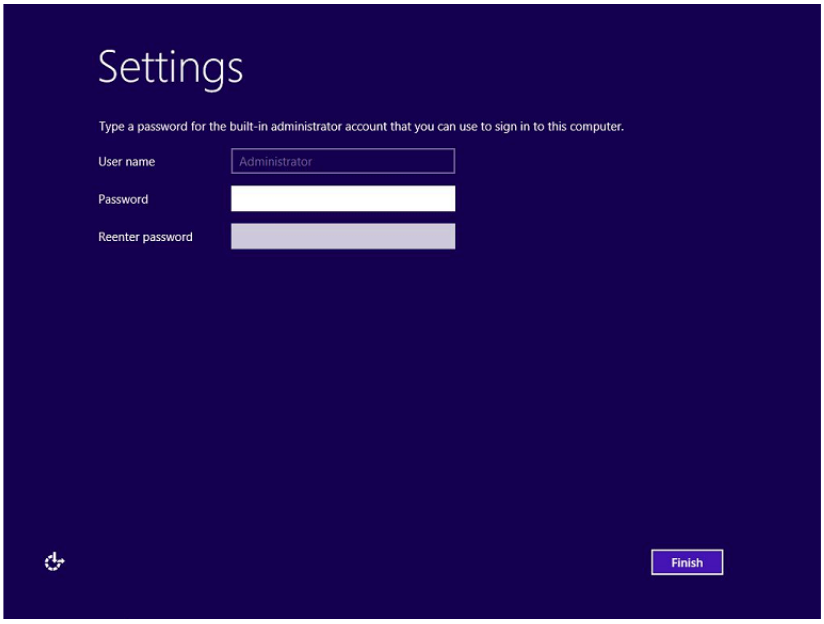


作業系統的系統檔案及/或登錄發生重大損毀。執行個體停留在此狀態時，您應該從最近的備份復原執行個體，AMI或啟動替代執行個體。如果您需要存取執行個體上的資料，請從無法連接的執行個體中分離任何根磁碟區、建立這些磁碟區的快照，或AMI從這些磁碟區建立一個，然後將它們連接到相同可用區域中的另一個執行個體做為輔助磁碟區。

Sysprep 畫面

主控台螢幕截取畫面服務傳回下列內容。

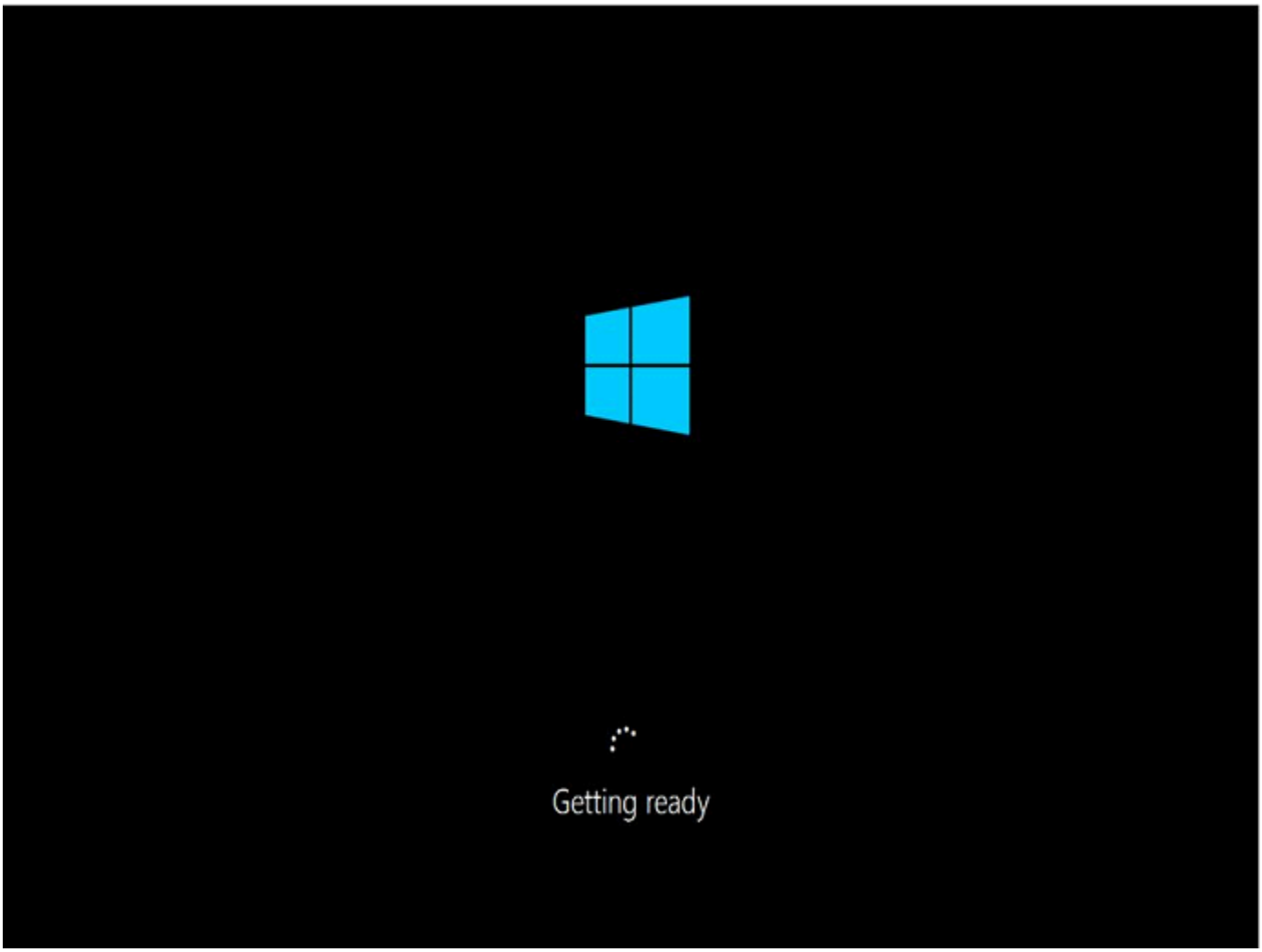




如果您未使用EC2Config服務呼叫 Sysprep，或者作業系統在執行 Sysprep 時發生故障，您可能會看到此畫面。您可以使用重設密碼EC2Rescue。否則，請參閱 EC2AMI使用視窗系統創建一個 Amazon。

就緒畫面

主控台螢幕截取畫面服務傳回下列內容。



重複更新執行個體主控台螢幕擷取畫面服務的畫面，以確認進度環是否旋轉。如果進度環不停旋轉，請等候作業系統啟動。您也可以使用 Amazon 查看執行個體上的 CPUUtilization(最大值) 指標，CloudWatch 查看作業系統是否處於作用中狀態。如果進度環不旋轉，則執行個體可能卡在開機程序。重新啟動執行個體。如果重新啟動無法解決問題，請從最近的備份中復原執行個體，AMI或啟動替代執行個體。如果您需要存取執行個體上的資料，請從無法連接的執行個體中分離根磁碟區、建立磁碟區的快照，或AMI從該磁碟區建立快照。然後將它連接到同一可用區域的另一個執行個體做為輔助磁碟區。

Windows Update 畫面

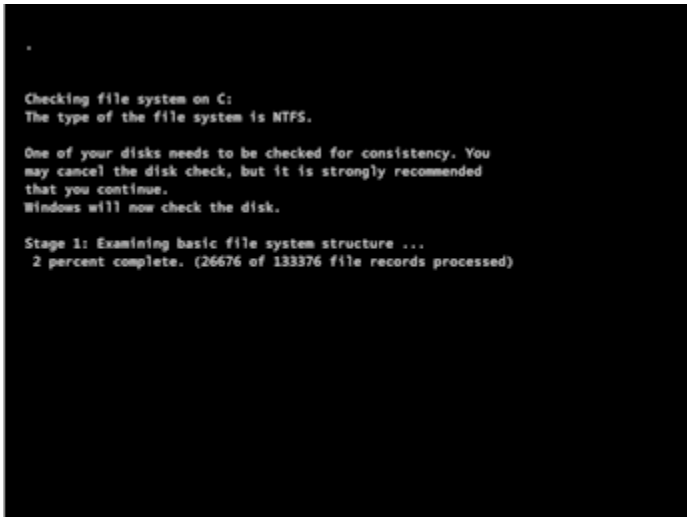
主控台螢幕截取畫面服務傳回下列內容。

Windows Update 程序正在更新登錄。等候更新完成。不要重新開機或停止執行個體，因為這會在更新期間造成資料損毀。

> **Note**
>
> Windows Update 程序在更新期間會使用伺服器的資源。如果您經常遇到此問題，請考慮使用更快的執行個體類型和更快的EBS磁碟區。

Chkdsk

主控台螢幕截取畫面服務傳回下列內容。



Windows 對磁碟機使用 chkdsk 系統工具，以確認檔案系統完整性並修復邏輯檔案系統錯誤。請等候程序完成。

當主機電腦故障時的執行個體復原

如果底層主機電腦的硬體具有無法修復的問題， AWS 可能會排程執行執行個體停止事件。您會事先收到此類事件的電子郵件通知。

復原在主機電腦上執行失敗的 Amazon EBS 支援執行個體

1. 將執行個體存放磁碟區上的任何重要資料備份到亞馬遜EBS或 Amazon S3。
2. 停止執行個體。
3. 啟動實例。
4. 復原所有重要的資料。

如需詳細資訊，請參閱停止並啟動 Amazon EC2 執行個體。

在主機電腦故障時復原其上執行的執行個體存放區後端執行個體

1. AMI從執行個體建立。
2. 將映像上傳到 Amazon S3。
3. 將重要數據備份到 Amazon EBS 或 Amazon S3。
4. 終止執行個體。
5. 從啟動新執行個體AMI。
6. 將所有重要的資料還原到新的執行個體上。

解決連線到 Amazon EC2 Linux 執行個體的問題

下列資訊和常見錯誤可協助您疑難排解連線到執行個體的問題。

連線問題

- 連線問題的常見原因
- 連線到執行個體時發生錯誤：連線時間逾時
- 錯誤：無法載入金鑰 ... 期待：ANYPRIVATEKEY
- 錯誤：伺服器無法辨識使用者金鑰
- 錯誤：許可遭拒或 [執行個體] 連接埠 22 已關閉連線
- 錯誤：未受保護的私有金鑰檔案

- 錯誤:私密金鑰必須以「-------」開頭，並以「BEGINRSAPRIVATEKEY----------」結尾 END RSA PRIVATE KEY
- 錯誤：伺服器拒絕了我們的金鑰，或是沒有可用的支援驗證方法
- 無法對執行個體執行 Ping 動作
- 錯誤：伺服器意外關閉網路連線
- 錯誤：EC2執行個體 Connect 的主機金鑰驗證失敗
- 無法使用執行個體 Connect 至 Ubuntu EC2 執行個體
- 我的私有金鑰遺失。如何連線到我的執行個體？

連線問題的常見原因

建議您透過確認已正確執行下列任務來開始疑難排解執行個體的連線問題。

確認執行個體的使用者名稱

您可以使用使用者帳戶的使用者名稱或用來啟動執行個體的預設使AMI用者名稱來連線至執行個體。

- 取得使用者帳戶的使用者名稱。

  如需如何建立使用者帳戶的詳細資訊，請參閱管理 Amazon EC2 Linux 執行個體上的系統使用者。

- 取得您用來啟動執行AMI個體的預設使用者名稱。

AMI用於啟動實例	預設使用者名稱
Amazon Linux	`ec2-user`
CentOS	`centos` 或 `ec2-user`
Debian	`admin`
Fedora	`fedora` 或 `ec2-user`
RHEL	`ec2-user` 或 `root`
SUSE	`ec2-user` 或 `root`

AMI用於啟動實例	預設使用者名稱
Ubuntu	`ubuntu`
Oracle	`ec2-user`
Bitnami	`bitnami`
Rocky Linux	`rocky`
其他	請洽詢AMI供應商

確認您的安全性群組規則允許流量

確保與執行個體相關聯的安全群組允許來自 IP 位址的傳入SSH流量。預設的預設安全性群組預設VPC不允許傳入SSH流量。啟動執行個體精靈所建立的安全性群組預設會啟用SSH流量。如需將輸入SSH流量規則新增至 Linux 執行個體的步驟，請參閱從您電腦連線到執行個體的規則。如需驗證步驟，請參閱 連線到執行個體時發生錯誤：連線時間逾時。

確認您的執行個體已就緒

啟動執行個體之後，執行個體可能需要幾分鐘的時間才能準備好接受連線要求。檢查您的執行個體，以確保它正在執行並通過其狀態檢查。

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選取 Instances (執行個體)，然後選取您的執行個體。
3. 請確認下列內容：
   a. 在 Instance state (執行個體狀態) 欄中，驗證您的執行個體處於 `running` 狀態。
   b. 在 Status check (狀態檢查) 欄中，驗證您的執行個體已通過兩項狀態檢查。

確認已滿足連線的所有先決條件

確保您擁有連接所需的所有信息。如需詳細資訊，請參閱使用 Connect 到您的 Linux 執行個體 SSH。

從 X 或者 X Connect

如果您的本機電腦作業系統是 Linux 或 macOS X，請檢查下列各項以取得連線至 Linux 執行個體的特定先決條件：

- SSH用戶端
- EC2實例 Connect
- AWS Systems Manager 會話管理器

從 Windows 進行連線

如果您的本機電腦作業系統是 Windows，請檢查下列項目以取得連線至 Linux 執行個體的特定先決條件：

- 打開 SSH
- 聚氨酯 TTY
- AWS Systems Manager 會話管理器
- 適用於 Linux 的 Windows 子系統

連線到執行個體時發生錯誤：連線時間逾時

如果您試著連線至執行個體，並收到錯誤訊息 `Network error: Connection timed out` 或 `Error connecting to [instance], reason: -> Connection timed out: connect`，則請嘗試下列動作：

檢查安全群組規則。

您需要一個安全性群組規則，允許從適當連接埠上的本機電腦公用IPv4位址傳入流量。

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選取 Instances (執行個體)，然後選取您的執行個體。
3. 在主控台頁面底部的 Security (安全) 標籤上，檢查 Inbound rules (傳入規則) 下方所選執行個體已生效的規則清單。確認有規則可允許從本機電腦傳輸到通訊埠 22 (SSH) 的流量。

   如果安全群組不具有規則來允許從本機電腦傳入流量，請在安全群組中新增規則。如需詳細資訊，請參閱從您電腦連線到執行個體的規則。
4. 針對允許傳入流量的規則，請檢查 Source (來源) 欄位。如果該值是單一 IP 地址，而且 IP 地址不是靜態的，則每次重新啟動電腦時都會指派新的 IP 地址。這將導致規則不會包括您電腦的 IP 地址流量。如果您的電腦位於公司網路上，或者您透過網際網路服務供應商 (ISP) 進行連線，或者您的電腦 IP 位址是動態的，且每次重新啟動電腦時都會變更，IP 位址可能不會是靜態的。若要確保您的安全群組規則允許從本機電腦傳入流量，請指定用戶端電腦使用的 IP 地址範圍，而不是為 Source (來源) 指定單一 IP 地址。

如需有關安全群組規則的詳細資訊，請參閱 Amazon VPC 使用者指南中的安全群組規則。

查看子網的路由表。

您需要一個路由，將所有流量發送VPC到網際網路閘道之VPC外。

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選取 Instances (執行個體)，然後選取您的執行個體。
3. 在 [網路] 索引標籤上，記下 VPCID 和子網路 ID 的值。
4. 在打開 Amazon VPC 控制台https://console.aws.amazon.com/vpc/。
5. 在導覽窗格中，選擇 Internet Gateways (網際網路閘道)。請確認您的VPC. 否則請選擇 Create internet gateway (建立網際網路閘道)、輸入網際網路閘道的名稱，然後選擇 Create internet gateway (建立網際網路閘道)。然後，針對您建立的網際網路閘道，選擇 [動作]、[附加至] VPC、選取您的VPC，然後選擇 [連接網際網路閘道] 將其附加至您的VPC。
6. 在導覽窗格中，選取 Subnets (子網)，然後選取子網。
7. 在「路由」(Route) 表格索引標籤上，確認有路由`0.0.0.0/0`作為目的地，以及VPC作為目標的網際網路閘道。如果您使用執行個體的IPv6位址連線至執行個體，請確認所有IPv6流量 (`::/0`) 都有指向網際網路閘道的路由。若否，請執行下列操作：
   a. 選擇路由表 ID (rtb-xxxxxxxx) 以導覽至路由表。
   b. 在 Routes (路由) 標籤中，選擇 Edit routes (編輯路由)。選擇 Add route (新增路由)，使用 `0.0.0.0/0` 做為目的地，並以網際網路閘道為目標。對於IPv6，選擇 [新增路由]，`::/0`做為目的地，使用網際網路閘道做為目標。
   c. 選擇 Save routes (儲存路由)。

檢查子網路的網路存取控制清單 (ACL)。

網路ACLs必須允許來自連接埠 22 上本機 IP 位址的輸入SSH流量。它也必須允許傳送到暫時連接埠 (1024-65535) 的傳出流量。

1. 在打開 Amazon VPC 控制台https://console.aws.amazon.com/vpc/。
2. 在導覽窗格中，選擇 Subnets (子網)。
3. 選取您的子網。
4. 在 ACL [網路] 索引標籤上，對於輸入規則，請確認規則允許來自電腦所需連接埠的輸入流量。否則，請刪除或修改會阻止流量的規則。

5. 對於 Outbound Rules (傳出規則)，確認這些規則允許透過暫時連接埠將流量傳出到電腦。否則，請刪除或修改會阻止流量的規則。

如果您的電腦位於企業網路中

詢問您的網路管理員內部防火牆是否允許從連接埠 22 上的電腦輸入和輸出流量。

如果您的電腦上有防火牆，請確認防火牆允許來自通訊埠 22 的電腦輸入和輸出流量。

檢查您的執行個體是否有公開IPv4位址。

如果沒有，您可建立彈性 IP 地址與您執行個體的關聯。如需詳細資訊，請參閱彈性 IP 地址。

檢查執行個體的CPU負載；伺服器可能超載。

AWS 自動提供 Amazon CloudWatch 指標和執行個體狀態等資料，您可以使用這些資料來查看執行個體有多少負CPU載，並在必要時調整負載的處理方式。如需詳細資訊，請參閱使用監控執行個體 CloudWatch。

- 如果負載會變動不定，您可以利用 Auto Scaling 和 Elastic Load Balancing 功能，來自動擴展或縮減執行個體的規模。
- 如果負載穩定增加，您可以改用較大容量的執行個體類型。如需詳細資訊，請參閱Amazon EC2 實例類型更改。

若要使用IPv6位址連線至執行個體，請檢查下列項目：

- 您的子網路必須與路由表相關聯，該路由表具有IPv6通往網際網路閘道的流量 (`::/0`)。
- 您的安全性群組規則必須允許來自連接埠 22 上本機IPv6位址的輸入流量。
- 您的網路ACL規則必須允許輸入和輸出IPv6流量。
- 如果您從舊版執行個體啟動執行個體AMI，則可能不會設定執行個體 DHCPv6 (網路介面上無法自動辨識IPv6位址)。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的在執行個體IPv6上進行設定。
- 您的本機電腦必須具有IPv6位址，且必須設定才能使用IPv6。

錯誤：無法載入金鑰 ... 期待：ANYPRIVATEKEY

如果您嘗試連接到您的執行個體並收到錯誤訊息 `unable to load key ... Expecting: ANY PRIVATE KEY`，則表示存放私有金鑰的檔案設定不正確。如果私有金鑰檔案以 `.pem` 結尾，則表示可能仍設定不正確。私有金鑰檔案設定錯誤的可能原因是憑證遺失。

如果私有金鑰檔案設定不正確，請依照以下步驟解決錯誤

1. 建立新的金鑰對。如需詳細資訊，請參閱 使用 Amazon 創建一個 key pair EC2。

> **Note**
>
> 或者，您也可以使用第三方工具來建立新的金鑰對。如需詳細資訊，請參閱 使用第三方工具建立 key pair，並將公開金鑰匯入 Amazon EC2。

2. 將新的金鑰對新增到執行個體。如需詳細資訊，請參閱我的私有金鑰遺失。如何連線到我的執行個體？。
3. 使用新的金鑰對連線至執行個體。

錯誤：伺服器無法辨識使用者金鑰

如果您使用連線SSH至執行個體

- 使用 `ssh -vvv`，在連線時取得三重 verbose (詳細模式) 的除錯資訊：

```
ssh -vvv -i path/key-pair-name.pem instance-user-
name@ec2-203-0-113-25.compute-1.amazonaws.com
```

下列的輸出範例，示範了在嘗試使用伺服器無法辨識的金鑰來連線到執行個體時，可能會顯示的訊息：

```
open/ANT/myusername/.ssh/known_hosts).
debug2: bits set: 504/1024
debug1: ssh_rsa_verify: signature correct
debug2: kex_derive_keys
debug2: set_newkeys: mode 1
debug1: SSH2_MSG_NEWKEYS sent
debug1: expecting SSH2_MSG_NEWKEYS
debug2: set_newkeys: mode 0
debug1: SSH2_MSG_NEWKEYS received
debug1: Roaming not allowed by server
debug1: SSH2_MSG_SERVICE_REQUEST sent
debug2: service_accept: ssh-userauth
debug1: SSH2_MSG_SERVICE_ACCEPT received
debug2: key: boguspem.pem ((nil))
debug1: Authentications that can continue: publickey
```

```
debug3: start over, passed a different list publickey
debug3: preferred gssapi-keyex,gssapi-with-mic,publickey,keyboard-
interactive,password
debug3: authmethod_lookup publickey
debug3: remaining preferred: keyboard-interactive,password
debug3: authmethod_is_enabled publickey
debug1: Next authentication method: publickey
debug1: Trying private key: boguspem.pem
debug1: read PEM private key done: type RSA
debug3: sign_and_send_pubkey: RSA 9c:4c:bc:0c:d0:5c:c7:92:6c:8e:9b:16:e4:43:d8:b2
debug2: we sent a publickey packet, wait for reply
debug1: Authentications that can continue: publickey
debug2: we did not send a packet, disable method
debug1: No more authentication methods to try.
Permission denied (publickey).
```

如果您使用 Pu 連線TTY至執行個體

- 確認您的私密金鑰 (.pem) 檔案已轉換為 Pu TTY (.ppk) 所辨識的格式。如需有關轉換私有金鑰的詳細資訊，請參閱使用 Pu Connect 到您的 Linux 執行個體 TTY。

> **Note**
>
> 在 P 中uTTYgen，載入您的私密金鑰檔案，然後選取儲存私密金鑰，而不是產生。

- 請確認您使用的是適當的使用者名稱連線AMI。在「PU TTY 組態」視窗的「主機名稱」方塊中輸入使用者名稱。

AMI用於啟動實例	預設使用者名稱
Amazon Linux	`ec2-user`
CentOS	`centos` 或 `ec2-user`
Debian	`admin`
Fedora	`fedora` 或 `ec2-user`
RHEL	`ec2-user` 或 `root`

AMI用於啟動實例	預設使用者名稱
SUSE	`ec2-user` 或 `root`
Ubuntu	`ubuntu`
Oracle	`ec2-user`
Bitnami	`bitnami`
Rocky Linux	`rocky`
其他	請洽詢AMI供應商

- 確認已建立傳入安全群組的規則，以允許從適當的連接埠傳入流量。如需詳細資訊，請參閱從您電腦連線到執行個體的規則。

錯誤：許可遭拒或 [執行個體] 連接埠 22 已關閉連線

如果您使用連線至執行個體，SSH並且出現下列任何錯誤`Host key not found in [directory]`、`Permission denied (publickey)`、`Authentication failed, permission denied`、或`Connection closed by [instance] port 22`，請確認您使用的是適當的使用者名稱連線，AMI且您已指定適當的私密金鑰 (執行個體的`.pem`)檔案)。

適當的使用者名稱如下：

AMI用於啟動實例	預設使用者名稱
Amazon Linux	`ec2-user`
CentOS	`centos` 或 `ec2-user`
Debian	`admin`
Fedora	`fedora` 或 `ec2-user`
RHEL	`ec2-user` 或 `root`
SUSE	`ec2-user` 或 `root`

AMI用於啟動實例	預設使用者名稱
Ubuntu	`ubuntu`
Oracle	`ec2-user`
Bitnami	`bitnami`
Rocky Linux	`rocky`
其他	請洽詢AMI供應商

例如，若要使用用SSH戶端連線至 Amazon Linux 執行個體，請使用下列命令：

```
ssh -i /path/key-pair-name.pem instance-user-
name@ec2-203-0-113-25.compute-1.amazonaws.com
```

請確認在啟動執行個體時，所使用的私有金鑰檔案可對應到您所選取的金鑰對。

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選取 Instances (執行個體)，然後選取您的執行個體。
3. 在 Details (詳細資訊) 標籤上，驗證 Instance details (執行個體詳細資訊) 下方中 Key pair name (金鑰對名稱) 的值。
4. 如果在啟動執行個體時，您並未指定金鑰對，可以終止該執行個體並啟動新的執行個體，來確保指定金鑰對。如果這是一直在使用的執行個體，但是您已經沒有金鑰對的 `.pem` 檔案，可以用新的金鑰對來取代舊的。如需詳細資訊，請參閱我的私有金鑰遺失。如何連線到我的執行個體？。

如果您產生了自己的金 key pair，請確保您的金鑰產生器已設置為建立RSA金鑰。DSA不接受金鑰。

如果收到 `Permission denied (publickey)` 錯誤，而上述的所有情況皆不適用 (例如，先前可以連線)，可能是對執行個體家目錄的權限已變更。對 `/home/`*`instance-user-name`*`/.ssh/authorized_keys` 的權限必須限定為只指派給擁有者。

在執行個體上確認權限

1. 停止執行個體並分離根磁碟區。如需詳細資訊，請參閱停止並啟動 Amazon EC2 執行個體。
2. 在與目前執行個體相同的可用區域中啟動暫存執行個體 (使用與目前執行個體使用的類似或AMI相同的執行個體)，然後將根磁碟區附加至暫存執行個體。

3. 連線到暫時執行個體、建立掛載點，然後掛載已連結的磁碟區。
4. 從暫時執行個體，檢查已連結磁碟區 /home/*instance-user-name*/ 目錄的權限。必要時，請依照下列方式來調整權限：

```
[ec2-user ~]$ chmod 600 mount_point/home/instance-user-name/.ssh/authorized_keys
```

```
[ec2-user ~]$ chmod 700 mount_point/home/instance-user-name/.ssh
```

```
[ec2-user ~]$ chmod 700 mount_point/home/instance-user-name
```

5. 取消掛載磁碟區、將該磁碟區從暫時執行個體分離，然後再重新連結到原始執行個體。請務必為根磁碟區指定正確的裝置名稱，例如 /dev/xvda。
6. 啟動執行個體。如果不再需要暫時執行個體，您可以將其終止。

錯誤：未受保護的私有金鑰檔案

私有金鑰檔案必須受到保護，不讓其他任何使用者讀取或寫入。如果除了您以外的任何人都可以讀取或寫入您的私鑰，則SSH忽略您的密鑰，並在下面看到以下警告消息。

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@         WARNING: UNPROTECTED PRIVATE KEY FILE!          @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
Permissions 0777 for '.ssh/my_private_key.pem' are too open.
It is required that your private key files are NOT accessible by others.
This private key will be ignored.
bad permissions: ignore key: .ssh/my_private_key.pem
Permission denied (publickey).
```

如果在您試著登入執行個體時，出現類似的訊息，請檢驗該錯誤訊息的第一行，來確認是否針對執行個體使用了正確的公有金鑰。上列的範例使用私有金鑰 .ssh/my_private_key.pem，此金鑰具有 0777 的檔案權限，可讓所有人讀取或寫入此檔案。此權限層級非常不安全，因此SSH忽略此金鑰。

如果從 macOS 或 Linux 連線，請執行下列命令來修正此錯誤，替換私有金鑰檔案的路徑。

```
[ec2-user ~]$ chmod 0400 .ssh/my_private_key.pem
```

如果您要從 Windows 連線到 Linux 執行個體，請在本機電腦上執行下列步驟。

1. 導覽至 .pem 檔案。
2. 在 .pem 檔案上按一下右鍵，然後選取 Properties (屬性)。
3. 選擇 Security (安全) 標籤。
4. 選取 Advanced (進階)。
5. 請確認您是檔案的擁有者。如果不是，請將擁有者變更為您的使用者名稱。
6. 選取 Disable inheritance (停用繼承) 和 Remove all inherited permissions from this object (移除此物件的所有繼承許可)。
7. 選取 Add (新增)、Select a principal (選取委託人)，輸入您的使用者名稱，然後選取 OK (確定)。
8. 從 Permission Entry (許可項目) 視窗中，授予 Read (讀取) 許可，然後選取 OK (確定)。
9. 按一下 Apply (套用)，以確保儲存所有設定。
10. 選取 OK (確定) 關閉 Advanced Security Settings (進階安全設定) 視窗。
11. 選取 OK (確定) 關閉 Properties (屬性) 視窗。
12. 您應該能夠使用連接到您的 Linux 實例SSH。

在 Windows 命令提示中執行下列命令。

1. 在命令提示中，導覽至 .pem 檔案的檔案路徑位置。
2. 執行下列命令，重設並移除明確許可：

```
icacls.exe $path /reset
```

3. 執行下列命令，對目前使用者授予讀取許可：

```
icacls.exe $path /GRANT:R "$($env:USERNAME):(R)"
```

4. 執行下列命令以停用繼承並移除繼承的許可。

```
icacls.exe $path /inheritance:r
```

5. 您應該能夠使用連接到您的 Linux 實例SSH。

錯誤:私密金鑰必須以「-------」開頭，並以「BEGINRSAPRIVATEKEY----------」結尾 END RSA PRIVATE KEY

如果您使用第三方工具 (例如ssh-keygen，建立 RSA key pair)，它會以 Open 金鑰格式產生私SSH密金鑰。當您連線至執行個體時，如果您使用 Open SSH 格式的私密金鑰來解密密碼，就會收到錯誤訊息`Private key must begin with "-----BEGIN RSA PRIVATE KEY-----" and end with "-----END RSA PRIVATE KEY-----"`。

若要解決錯誤，私密金鑰的PEM格式必須為。使用下列命令建立格PEM式的私密金鑰：

```
ssh-keygen -m PEM
```

錯誤：伺服器拒絕了我們的金鑰，或是沒有可用的支援驗證方法

如果您使用 Pu 連線TTY至執行個體，並得到下列任一錯誤，錯誤：伺服器拒絕我們的金鑰或錯誤：沒有可用的支援驗證方法，請確認您使用的是正確的使用者名稱連線AMI。在 Pu TTY 組態視窗的使用者名稱中輸入使用者名稱。

適當的使用者名稱如下：

AMI用於啟動實例	預設使用者名稱
Amazon Linux	`ec2-user`
CentOS	`centos` 或 `ec2-user`
Debian	`admin`
Fedora	`fedora` 或 `ec2-user`
RHEL	`ec2-user` 或 `root`
SUSE	`ec2-user` 或 `root`
Ubuntu	`ubuntu`
Oracle	`ec2-user`
Bitnami	`bitnami`

AMI用於啟動實例	預設使用者名稱
Rocky Linux	`rocky`
其他	請洽詢AMI供應商

您也應該確認：

- 您正在使用最新版本的 Pu TTY。如需詳細資訊，請參閱 Pu TTY 網頁。
- 您的私鑰（.pem）文件已正確轉換為 PuTTY（.ppk）識別的格式。如需有關轉換私有金鑰的詳細資訊，請參閱使用 Pu Connect 到您的 Linux 執行個體 TTY。

無法對執行個體執行 Ping 動作

命`ping`令是一種ICMP流量類型 — 如果您無法對執行個體進行 Ping 操作，請確保輸入安全性群組規則允許來自所有來源或發出指令的電腦或執行個體的`Echo Request`訊息ICMP流量。

如果您無法從執行個體發出`ping`命令，請確定輸出安全性群組規則允許`Echo Request`郵件傳送至所有目的地或嘗試 Ping 的主機的ICMP流量。

`Ping` 命令也可能會遭到防火牆封鎖，或因為網路延遲或硬體問題而逾時。如需進一步疑難排解的協助，請洽詢您的區域網路或系統管理員。

錯誤：伺服器意外關閉網路連線

如果您使用 Pu 連線至執行個體，TTY且收到「伺服器意外關閉網路連線」錯誤訊息，請確認您已在 Pu TTY 組態的 [連線] 頁面上啟用 keepalives，以避免中斷連線。部分伺服器在指定的期間內未收到任何資料時會中斷連接用戶端。將 Seconds between keepalives (保持連線之間的秒數) 設為 59 秒。

如果啟用 Keepalives 後仍然遇到問題，請嘗試在 Pu 配置的「連接」頁面上禁用 Nagle 的算法。TTY

錯誤：EC2執行個體 Connect 的主機金鑰驗證失敗

如果您輪替執行個體主機金鑰，新的主機金鑰不會自動上傳到 AWS 受信任的主機金鑰資料庫。當您嘗試使用 Instance Connect 瀏覽器型用戶端連線至執行個體時，這會導致主機金鑰驗證失敗，且您無法連線至執行個體。EC2

若要解決錯誤，您必須在執行個體上執行`eic_harvest_hostkeys`指令碼，該指令碼會將新的主機金鑰上傳至EC2執行個體 Connect。指令碼位於 Amazon Linux 2 執行個體上的 `/opt/aws/bin/` 和 Ubuntu 執行個體上的 `/usr/share/ec2-instance-connect/`。

Amazon Linux 2

解決 Amazon Linux 2 執行個體上的主機金鑰驗證失敗錯誤

1. 使用 Connect 至您的執行個體SSH。

   您可以使用EC2執行個體連線，CLI或使用啟動執行個體時指派給執行個體的 SSH key pair，以及您用來啟動執行個體的預設使用者名稱來連線。AMI對於 Amazon Linux 2，默認的用戶名是`ec2-user`.

   例如，如果您的執行個體是使用 Amazon Linux 2 啟動，則執行個體的公共DNS名稱為`ec2-a-b-c-d.us-west-2.compute.amazonaws.com`，key pair為`my_ec2_private_key.pem`，請使用下列命令進SSH入執行個體：

   ```
   $ ssh -i my_ec2_private_key.pem ec2-user@ec2-a-b-c-d.us-
   west-2.compute.amazonaws.com
   ```

   如需連線至執行個體的詳細資訊，請參閱使用用SSH戶端 Connect 至您的 Linux 執行個體。

2. 導覽至以下資料夾。

   ```
   [ec2-user ~]$ cd /opt/aws/bin/
   ```

3. 在執行個體上執行以下命令。

   ```
   [ec2-user ~]$ ./eic_harvest_hostkeys
   ```

   請注意，呼叫成功時不會產生輸出。

   您現在可以使用「EC2執行個體 Connect」瀏覽器用戶端連線至執行個體。

Ubuntu

解決 Ubuntu 執行個體上的主機金鑰驗證失敗錯誤

1. 使用 Connect 至您的執行個體SSH。

您可以使用EC2執行個體連線，CLI或使用啟動執行個體時指派給執行個體的 SSH key pair，以及您用來啟動執行個體的預設使用者名稱來連線。AMI對於 Ubuntu，預設的使用者名稱為`ubuntu`。

例如，如果您的執行個體是使用 Ubuntu 啟動，則執行個體的公開DNS名稱為`ec2-a-b-c-d.us-west-2.compute.amazonaws.com`，key pair 為`my_ec2_private_key.pem`，請使用下列命令進SSH入執行個體：

```
$ ssh -i my_ec2_private_key.pem ubuntu@ec2-a-b-c-d.us-
west-2.compute.amazonaws.com
```

如需連線至執行個體的詳細資訊，請參閱使用用SSH戶端 Connect 至您的 Linux 執行個體。

2. 導覽至以下資料夾。

```
[ec2-user ~]$ cd /usr/share/ec2-instance-connect/
```

3. 在執行個體上執行以下命令。

```
[ec2-user ~]$ ./eic_harvest_hostkeys
```

請注意，呼叫成功時不會產生輸出。

您現在可以使用「EC2執行個體 Connect」瀏覽器用戶端連線至執行個體。

無法使用執行個體 Connect 至 Ubuntu EC2 執行個體

如果您使用「EC2執行個體連線」連線至 Ubuntu 執行個體，但嘗試連線時看到錯誤訊息，您可以使用下列資訊嘗試修正問題。

可能的原因

該執行個體的 `ec2-instance-connect` 套件並非最新版本。

解決方案

如下所示將該執行個體的 `ec2-instance-connect` 套件更新至最新版本：

1. 使用執行個體 Connect 以外的方法 Connect 至EC2執行個體。
2. 在您的執行個體上執行以下命令，將 `ec2-instance-connect` 套件更新至最新版本。

```
apt update && apt upgrade
```

我的私有金鑰遺失。如何連線到我的執行個體？

如果您遺失了EBS備份執行個體的私密金鑰，您可以重新取得執行個體的存取權。您必須停止執行個體、中斷其根磁碟區的連結，並將該磁碟區做為資料磁碟區連接至其他執行個體、使用新的公有金鑰修改 `authorized_keys` 檔案、將磁碟區移回到原始的執行個體，然後重新啟動執行個體。關於啟動、連線到和停用執行個體，詳細資訊請參閱 Amazon EC2 實例狀態更改。

只有具有EBS根磁碟區的執行個體才支援此程序。如果根裝置是執行個體存放區磁碟區，則無法使用此程序來重新獲取執行個體的存取權；您必須具備私有金鑰，才能連線到執行個體。若要判斷執行個體的根裝置類型，請開啟 Amazon EC2 主控台、選擇執行個體、選取執行個體、選擇儲存體索引標籤，然後在根裝置詳細資訊區段中檢查根裝置類型的值。

此值為 `EBS` 或 `INSTANCE-STORE`。

除了下列步驟之外，遺失私有金鑰時還有其他方法可以連線到 Linux 執行個體。如需詳細資訊，請參閱如果我在 Amazon EC2 執行個體初始啟動後遺失 SSH key pair，該如何連線到 Amazon 執行個體？

使用不同 key pair 連線至 EBS-backed 執行個體的步驟

- 步驟 1：建立新的金鑰對
- 步驟 2：取得關於原始執行個體及其根磁碟區的資訊
- 步驟 3：停止原始執行個體
- 步驟 4：啟動暫時執行個體
- 步驟 5：將根磁碟區與原始執行個體分離，並將其連接到暫時執行個體
- 步驟 6：在掛載到暫時執行個體的原始磁碟區上，將新的公有金鑰新增至的 authorized_keys
- 步驟 7：取消掛載原始磁碟區，並將該磁碟區從暫時執行個體分離，然後再重新連接到原始執行個體
- 步驟 8：使用新的金鑰對連線到原始執行個體
- 步驟 9：清理

步驟 1：建立新的金鑰對

使用 Amazon EC2 主控台或第三方工具建立新的 key pair。如果希望新金鑰對的名稱，和已遺失私有金鑰的相同，您必須先刪除現有的金鑰對。如需建立新金鑰對的詳細資訊，請參閱使用 Amazon 創建一個 key pair EC2或使用第三方工具建立 key pair，並將公開金鑰匯入 Amazon EC2。

步驟 2：取得關於原始執行個體及其根磁碟區的資訊

請儲存下列資訊，完成此程序將會需要用到這些資訊。

取得原始執行個體的相關資訊

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中選取 Instances (執行個體)，然後選取您要連線的執行個體。(我們會將此執行個體稱為「原始」執行個體。)
3. 在 [詳細資料] 索引標籤上，記下執行個體 ID 和 AMI ID。
4. 在 網路 索引標籤上記下 「可用區域」。
5. 在 Storage (儲存) 索引標籤下的 根裝置名稱，記下根磁碟區的裝置名稱 (例如：`/dev/xvda`)。接著，在 區塊型儲存裝置 下找出這個裝置名稱，並記下磁碟區 ID (例如：vol-0a1234b5678c910de)。

步驟 3：停止原始執行個體

選擇 Instance state (執行個體狀態)、Stop instance (停止執行個體)。如果此選項已停用，則執行個體已停止或其根設備是執行個體存放磁碟區。

> **Warning**
> 停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。若要保留執行個體存放區磁碟區的資料，請將資料備份至持久性儲存。

步驟 4：啟動暫時執行個體

啟動暫時執行個體

1. 在導覽窗格中，選擇 Instances (執行個體)，然後選擇 Launch instances (啟動執行個體)。
2. 在 Name and tags (名稱和標籤) 區段，針對 Name (名稱)，輸入 Temporary (暫時)。
3. 在 [應用程式和作業系統映像] 區段中，選取您AMI用來啟動原始執行個體的相同項目。如果無法使AMI用，您可以建立可從停止AMI的執行個體使用的執行個體。如需詳細資訊，請參閱創建一個 Amazon 支EBS持 AMI。
4. 在 Instance type (執行個體類型) 區段，保留預設執行個體類型。

5. 在 Key pair (金鑰對) 區段，針對 Key pair name (金鑰對名稱)，選取要使用的現有金鑰對或建立新的金鑰對。
6. 在 Network settings (網路設定) 區段，選取 Edit (編輯)，然後針對 Subnet (子網)，選取與原始執行個體相同可用區域中的子網。
7. 在 Summary (摘要) 面板中，選擇 Launch (啟動)。

步驟 5：將根磁碟區與原始執行個體分離，並將其連接到暫時執行個體

1. 在導覽窗格中選取 Volumes (磁碟區)，接著再選取原始執行個體的根裝置磁碟區 (您已在先前的步驟中記下其磁碟區 ID)。選取 Actions (動作)、Detach Volume (分離磁碟區)，然後選取 Detach (分離)。等待磁碟區的狀態變成 `available`。(您可能需要選取 Refresh (重新整理) 圖示。)
2. 在已選取磁碟區的狀態下，選取 Actions (動作)，然後再選取 Attach volume (連接磁碟區)。選取暫時執行個體的執行個體 ID、記下 Device name (裝置名稱) 中的指定裝置名稱 (例如：`/dev/sdf`)，接著再選取 Attach volume (連接磁碟區)。

> **Note**
>
> 如果您從啟動原始執行個體， AWS Marketplace AMI且磁碟區包含 AWS Marketplace 代碼，則必須先停止暫存執行個體，然後才能連接磁碟區。

步驟 6：在掛載到暫時執行個體的原始磁碟區上，將新的公有金鑰新增至的 `authorized_keys`

1. 連線到暫時執行個體。
2. 從暫時執行個體掛載已連結到執行個體的磁碟區，如此您就能存取其檔案系統。例如，如果裝置名稱為 `/dev/sdf`，請使用下列的指令來將磁碟區掛載為 `/mnt/tempvol`。

> **Note**
>
> 在執行個體上所顯示的裝置名稱可能會有不同。例如，掛載為 `/dev/sdf` 的裝置，在執行個體上可能會顯示為 `/dev/xvdf`。Red Hat 的某些版本 (或是其變體版本，例如 CentOS) 甚至可能會將結尾的字母增加 4 個字元，其中 `/dev/sd`*`f`* 會變成 `/dev/xvd`*`k`*。

   a. 使用 lsblk 命令來查看磁碟區是否已分割。

```
[ec2-user ~]$ lsblk
NAME    MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
xvda    202:0    0    8G  0 disk
##xvda1 202:1    0    8G  0 part /
xvdf    202:80   0  101G  0 disk
##xvdf1 202:81   0  101G  0 part
xvdg    202:96   0   30G  0 disk
```

在上述的範例中，/dev/xvda 和 /dev/xvdf 為分割的磁碟區，而 /dev/xvdg 則不是。如果磁碟區已分割，則在接下來的步驟中，您會掛載 (/dev/xvdf1) 而非原始裝置 (/dev/xvdf)。

b. 建立暫時的目錄來掛載磁碟區。

```
[ec2-user ~]$ sudo mkdir /mnt/tempvol
```

c. 使用您先前找出的磁碟區名稱或裝置名稱，將磁碟區 (或分割磁碟區) 掛載至暫時掛載點。必要命令取決於作業系統的檔案系統。請注意，在執行個體上所顯示的設備名稱可能會有不同。如需詳細資訊，請參閱步驟 6 中的 note。

- Amazon Linux、Ubuntu 和 Debian

```
[ec2-user ~]$ sudo mount /dev/xvdf1 /mnt/tempvol
```

- Amazon 伺服器 2、Cent SUSE OS 12 及 7.x RHEL

```
[ec2-user ~]$ sudo mount -o nouuid /dev/xvdf1 /mnt/tempvol
```

> **Note**
>
> 如果您收到檔案系統毀損的錯誤，請執行下列命令，使用 fsck 公用程式來檢查檔案系統並修復任何問題：
>
> ```
> [ec2-user ~]$ sudo fsck /dev/xvdf1
> ```

3. 從暫時執行個體使用下列的指令，用暫時執行個體的 authorized_keys 所產生的新公有金鑰，來更新已掛載磁碟區上的 authorized_keys。

> **Important**
>
> 下列範例使用 Amazon Linux 使用者名稱`ec2-user`。您可能需要替換不同的使用者名稱，`ubuntu`例如 Ubuntu 執行個體。

```
[ec2-user ~]$ cp .ssh/authorized_keys /mnt/tempvol/home/ec2-user/.ssh/
authorized_keys
```

如果此項複製動作成功，您就可以進行下一個步驟。

(選用) 否則，如果您沒有編輯 `/mnt/tempvol` 中的檔案的許可，必須使用 sudo 來更新檔案，然後檢查對於該檔案的許可，以確認您是否能夠登入原始執行個體。使用下列命令以檢查對檔案的許可。

```
[ec2-user ~]$ sudo ls -l /mnt/tempvol/home/ec2-user/.ssh
total 4
-rw------- 1 222 500 398 Sep 13 22:54 authorized_keys
```

在此示例輸出中，*222* 是使用者識別碼，*500* 是群組識別碼。接下來，請使用 sudo 來重新執行先前失敗的複製命令。

```
[ec2-user ~]$ sudo cp .ssh/authorized_keys /mnt/tempvol/home/ec2-user/.ssh/
authorized_keys
```

再次執行下列命令，以確定許可是否已變更。

```
[ec2-user ~]$ sudo ls -l /mnt/tempvol/home/ec2-user/.ssh
```

如果使用者 ID 和群組 ID 已變更，請使用下列命令來還原這些 ID。

```
[ec2-user ~]$ sudo chown 222:500 /mnt/tempvol/home/ec2-user/.ssh/authorized_keys
```

步驟 7：取消掛載原始磁碟區，並將該磁碟區從暫時執行個體分離，然後再重新連接到原始執行個體

1. 從暫時執行個體取消掛載已連結的磁碟區，如此您就能將該磁碟區重新連結到原始執行個體。例如，使用以下命令來取消掛載位於 `/mnt/tempvol` 的磁碟區。

```
[ec2-user ~]$ sudo umount /mnt/tempvol
```

2. 從暫存執行個體中分離磁碟區 (您在上一個步驟中將其卸載)：在 Amazon EC2 主控台中，選擇導覽窗格中的 Volumes，選取原始執行個體的根裝置磁碟區 (您在上一個步驟中記下磁碟區 ID)，選擇 [動作]、[卸離磁碟區]，然後選擇 [分離]。等待磁碟區的狀態變成 `available`。(您可能需要選取 Refresh (重新整理) 圖示。)
3. 將磁碟區重新連接到原始執行個體：在已選取磁碟區的狀態下，選取 Actions (動作)、Attach volume (連接磁碟區)。選取原始執行個體的執行個體 ID、指定您先前在步驟 2 中為原始根裝置連接 (`/dev/sda1` 或 `/dev/xvda`) 記錄的裝置名稱，然後選取 Attach volume (連接磁碟區)。

> **Important**
>
> 如果您並未指定與原始連結相同的裝置名稱，則無法啟動原始執行個體。Amazon EC2 預計根設備卷在`sda1`或`/dev/xvda`。

步驟 8：使用新的金鑰對連線到原始執行個體

選取原始執行個體，然後依序選取 Instance state (執行個體狀態)、Start instance (啟動執行個體)。在執行個體進入 `running` 狀態之後，您可以使用新金鑰對的私有金鑰檔案，來連線到該執行個體。

> **Note**
>
> 如果新金鑰對的名稱和對應的私有金鑰檔案，與原始金鑰對的名稱不同，請務必在連線到執行個體時，指定新私有金鑰檔案的名稱。

步驟 9：清理

(選用) 如果不需再使用時，您可以終止暫時執行個體。選取暫時執行個體，再依序選取 Instance state (執行個體狀態)、Terminate instance (終止執行個體)。

使用失敗的狀態檢查進行 Amazon EC2 Linux 執行個體

下列資訊可協助您排解 Linux 執行個體無法進行狀態檢查時的問題。首先判斷應用程式是否出現任何問題。如果確認執行個體並未如預期執行應用程式，請檢閱狀態檢查資訊和系統日誌。

關於導致狀態檢查失敗的問題範例，請參閱 Amazon EC2 執行個體的狀態檢查。

內容

- 檢閱狀態檢查資訊
- 擷取系統日誌
- 解決 Linux 執行個體的系統記錄錯誤
- 記憶體不足：終止程序
- ERROR: mmu_update 失敗 (記憶體管理更新失敗)
- I/O 錯誤 (區塊型儲存設備故障)
- I/OERROR：既不是本地磁盤也不是遠程磁盤（破碎的分佈式塊設備）
- request_module：失控迴圈 modprobe (在舊版 Linux 上建立舊核心 modprobe 的迴圈)
- 「FATAL: 內核太舊了」和「fsck：嘗試打開 /dev 時沒有這樣的文件或目錄」（內核和不匹配）AMI
- 「FATAL: 無法載入 /lib/ 模組" 或 "BusyBox" (遺失核心模組)
- ERROR無效的內核（EC2不兼容的內核）
- fsck：嘗試開啟時找不到此等檔案或目錄… (找不到檔案系統)
- 掛載檔案作業系統的一般錯誤 (掛載失敗)
- VFS: 無法在未知的區塊上掛載根 fs (根檔案系統不符)
- 錯誤：無法判定根設備的主/次編號… (根檔案系統/裝置不相符)
- XENBUS：沒有驅動程序的設備...
- …未進行檢查的天數，強制進行檢查 (需要進行檔案系統檢查)
- fsck 凍結於結束狀態… (缺少裝置)
- GRUB提示符（灰塵 >）
- 啟動介面 eth0：裝置 eth0 具有與預期不同的MAC位址，而忽略。(硬式編碼MAC位址)
- 無法載入SELinux策略。機器正處於強制執行模式。現在停止中。（SELinux配置錯誤）
- XENBUS：連接到設備的超時（Xenbus 超時）

檢閱狀態檢查資訊

使用 Amazon EC2 主控台調查受損的執行個體

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選取 Instances (執行個體)，然後選取您的執行個體。
3. 在詳細資訊窗格中，請選擇狀態和警示，以檢視所有系統狀態檢查和執行個體狀態檢查的個別結果。

如果未通過系統狀態檢查，您可以嘗試下列其中一個選項：

- 建立執行個體復原警示。如需詳細資訊，請參閱建立警示以停止、終止、重新啟動或復原執行個體。
- 如果您將執行個體類型變更為 AWS Nitro System 上建置的執行個體，如果您從沒有必要ENA和NVMe驅動程式的執行個體移轉，狀態檢查會失敗。如需詳細資訊，請參閱變更執行個體類型的相容性。
- 對於使用 Amazon EBS 支援的執行個體AMI，請停止並重新啟動執行個體。
- 對於使用支援執行個體存放區的執行個體AMI，請終止執行個體並啟動取代作業。
- 等待 Amazon EC2 解決問題。
- 將您的問題發佈至 AWS re:Post。
- 如果您的執行個體位於 Auto Scaling 群組中，Amazon EC2 Auto Scaling 服務會自動啟動替換執行個體。如需詳細資訊，請參閱 Amazon Auto Scaling 使用者指南中的自 Au EC2 to Scaling 執行個體運作 Health 態檢查。
- 擷取系統日誌並尋找錯誤。

擷取系統日誌

如果執行個體狀態檢查未通過，您可以重新啟動執行個體和擷取系統日誌。日誌可能會顯示錯誤，有助於對發生的問題進行故障診斷。重新啟動作業會清除日誌中不必要的資訊。

重新啟動執行個體和擷取系統日誌

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)，然後選取您的執行個體。
3. 選擇 Instance state (執行個體狀態)、Reboot intance (重新啟動執行個體)。執行個體重新啟動可能需要幾分鐘的時間。

4. 確認問題是否仍持續存在；在某些情況中，重新啟動也許可以解決問題。
5. 當執行個體處於 `running` 狀態時，選擇 Actions (動作)、Monitor and troubleshoot (監視和疑難排解)、Get system log (取得系統日誌檔)。
6. 檢閱畫面上所顯示的日誌，然後使用下列的已知系統日誌錯誤陳述清單，來進行問題的故障診斷。
7. 如果問題無法解決，您可以在 AWS re:Post 中發佈您的問題。

解決 Linux 執行個體的系統記錄錯誤

對於執行個體狀態檢查失敗的 Linux 執行個體 (例如執行個體可連接性檢查)，請確認您是否遵循上述步驟擷取系統記錄。下列清單包含了一些常見的系統日誌錯誤和建議的動作，您可以針對每項錯誤採取這些動作，來解決問題。

Memory Errors (記憶體錯誤)

- 記憶體不足：終止程序
- ERROR: mmu_update 失敗 (記憶體管理更新失敗)

Device Errors (裝置錯誤)

- I/O 錯誤 (區塊型儲存設備故障)
- I/OERROR：既不是本地磁盤也不是遠程磁盤（破碎的分佈式塊設備）

Kernel Errors (核心錯誤)

- request_module：失控迴圈 modprobe (在舊版 Linux 上建立舊核心 modprobe 的迴圈)
- 「FATAL: 內核太舊了」和「fsck：嘗試打開 /dev 時沒有這樣的文件或目錄」（內核和不匹配）AMI
- 「FATAL: 無法載入 /lib/ 模組" 或 "BusyBox" (遺失核心模組)
- ERROR無效的內核（EC2不兼容的內核）

File System Errors (檔案系統錯誤)

- fsck：嘗試開啟時找不到此等檔案或目錄… (找不到檔案系統)
- 掛載檔案作業系統的一般錯誤 (掛載失敗)
- VFS: 無法在未知的區塊上掛載根 fs (根檔案系統不符)
- 錯誤：無法判定根設備的主/次編號… (根檔案系統/裝置不相符)

- XENBUS：沒有驅動程序的設備...
- ...未進行檢查的天數，強制進行檢查 (需要進行檔案系統檢查)
- fsck 凍結於結束狀態... (缺少裝置)

Operating System Errors (作業系統錯誤)

- GRUB提示符（灰塵 >）
- 啟動介面 eth0：裝置 eth0 具有與預期不同的MAC位址，而忽略。(硬式編碼MAC位址)
- 無法載入SELinux策略。機器正處於強制執行模式。現在停止中。（SELinux配置錯誤）
- XENBUS：連接到設備的超時（Xenbus 超時）

記憶體不足：終止程序

系統記錄項目會指出 out-of-memory 錯誤，類似如下所示。

```
[115879.769795] Out of memory: kill process 20273 (httpd) score 1285879
or a child
[115879.769795] Killed process 1917 (php-cgi) vsz:467184kB, anon-
rss:101196kB, file-rss:204kB
```

可能的原因

記憶體用盡

建議動作

如果是此種執行個體類型	執行此作業
Amazon 支EBS持	執行以下任意一項： • 請停止執行個體，並修改執行個體以使用不同的執行個體類型，然後再次啟動該執行個體。例如，較大容量或記憶體最佳化的執行個體類型。 • 重新啟動執行個體，以讓執行個體回到健全狀態。除非您變更執行個體類型，否則問題可能會再次發生。

如果是此種執行個體類型	執行此作業
執行個體存放區後端	請執行下列其中一項： • 終止執行個體並啟動新的執行個體，指定不同的執行個體類型。例如，較大容量或記憶體最佳化的執行個體類型。 • 重新啟動執行個體，以讓執行個體回到健全狀態。除非您變更執行個體類型，否則問題可能會再次發生。

ERROR: mmu_update 失敗 (記憶體管理更新失敗)

系統日誌記錄項目會顯示記憶體管理更新作業的失敗，類似於下列所示記錄：

```
...
Press `ESC' to enter the menu... 0   [H[J  Booting 'Amazon Linux 2011.09
 (2.6.35.14-95.38.amzn1.i686)'


root (hd0)

 Filesystem type is ext2fs, using whole disk

kernel /boot/vmlinuz-2.6.35.14-95.38.amzn1.i686 root=LABEL=/ console=hvc0 LANG=

en_US.UTF-8 KEYTABLE=us

initrd /boot/initramfs-2.6.35.14-95.38.amzn1.i686.img

ERROR: mmu_update failed with rc=-22
```

可能的原因

Amazon Linux 的問題

建議動作

請在開發人員論壇上發表您的問題，或是聯絡 AWS Support。

I/O 錯誤 (區塊型儲存設備故障)

系統日誌記錄項目會顯示輸入/輸出錯誤，類似於下列的範例：

```
[9943662.053217] end_request: I/O error, dev sde, sector 52428288
[9943664.191262] end_request: I/O error, dev sde, sector 52428168
[9943664.191285] Buffer I/O error on device md0, logical block 209713024
[9943664.191297] Buffer I/O error on device md0, logical block 209713025
[9943664.191304] Buffer I/O error on device md0, logical block 209713026
[9943664.191310] Buffer I/O error on device md0, logical block 209713027
[9943664.191317] Buffer I/O error on device md0, logical block 209713028
[9943664.191324] Buffer I/O error on device md0, logical block 209713029
[9943664.191332] Buffer I/O error on device md0, logical block 209713030
[9943664.191339] Buffer I/O error on device md0, logical block 209713031
[9943664.191581] end_request: I/O error, dev sde, sector 52428280
[9943664.191590] Buffer I/O error on device md0, logical block 209713136
[9943664.191597] Buffer I/O error on device md0, logical block 209713137
[9943664.191767] end_request: I/O error, dev sde, sector 52428288
[9943664.191970] end_request: I/O error, dev sde, sector 52428288
[9943664.192143] end_request: I/O error, dev sde, sector 52428288
[9943664.192949] end_request: I/O error, dev sde, sector 52428288
[9943664.193112] end_request: I/O error, dev sde, sector 52428288
[9943664.193266] end_request: I/O error, dev sde, sector 52428288
...
```

可能的原因

執行個體類型	可能的原因
Amazon 支EBS持	失敗的 Amazon EBS 卷
執行個體存放區後端	故障的實體磁碟機

建議動作

如果是此種執行個體類型	執行此作業
Amazon 支EBS持	請執行下列步驟： 1. 停止執行個體。

如果是此種執行個體類型	執行此作業
	2. 分離磁碟區。 3. 試著將磁碟區復原。 Note 經常快照您的 Amazon EBS 磁碟區是很好的做法。這可以大幅降低因為故障而造成資料遺失的風險。 4. 將磁碟區重新連結到執行個體。 5. 啟動實例。
執行個體存放區後端	終止執行個體並啟動新的執行個體。 Note 資料無法復原。請從備份資料復原。 Note 使用 Amazon S3 或 Amazon 進行備份是一個很好EBS的做法。執行個體存放磁碟區會與單一主機或單一磁碟的故障直接連動。

I/OERROR：既不是本地磁盤也不是遠程磁盤（破碎的分佈式塊設備）

系統日誌記錄項目會顯示裝置上的輸入/輸出錯誤，類似於下列的範例：

```
...
block drbd1: Local IO failed in request_timer_fn. Detaching...

Aborting journal on device drbd1-8.
```

```
block drbd1: IO ERROR: neither local nor remote disk

Buffer I/O error on device drbd1, logical block 557056

lost page write due to I/O error on drbd1

JBD2: I/O error detected when updating journal superblock for drbd1-8.
```

可能的原因

執行個體類型	可能的原因
Amazon 支EBS持	失敗的 Amazon EBS 卷
執行個體存放區後端	故障的實體磁碟機

建議動作

終止執行個體並啟動新的執行個體。

對於 Amazon EBS 支援的執行個體，您可以從最近的快照建立映像來復原資料。在快照建立後所新增的任何資料，皆無法復原。

request_module：失控迴圈 modprobe (在舊版 Linux 上建立舊核心 modprobe 的迴圈)

系統日誌記錄項目會顯示此狀況，類似於下列所示。使用不穩定的或舊版的 Linux 核心 (例如 2.6.16-xenU)，可能會在啟動時造成無止盡的循環狀況。

```
Linux version 2.6.16-xenU (builder@xenbat.amazonsa) (gcc version 4.0.1
20050727 (Red Hat 4.0.1-5)) #1 SMP Mon May 28 03:41:49 SAST 2007

BIOS-provided physical RAM map:

 Xen: 0000000000000000 - 0000000026700000 (usable)

0MB HIGHMEM available.
...
```

```
request_module: runaway loop modprobe binfmt-464c

request_module: runaway loop modprobe binfmt-464c

request_module: runaway loop modprobe binfmt-464c

request_module: runaway loop modprobe binfmt-464c

request_module: runaway loop modprobe binfmt-464c
```

建議動作

如果是此種執行個體類型	執行此作業
Amazon 支EBS持	使用下列其中一個選項來使用較新的核心，無論是GRUB基於或靜態的核心： 選項 1：終止執行個體並啟動新的執行個體，指定 `-kernel` 和 `-ramdisk` 參數。 選項 2： 1. 停止執行個體。 2. 修改核心和記憶體虛擬磁碟 (ramdisk) 的屬性，以使用較新版的核心。 3. 啟動實例。
執行個體存放區後端	終止執行個體並啟動新的執行個體，指定 `-kernel` 和 `-ramdisk` 參數。

「FATAL: 內核太舊了」和「fsck：嘗試打開 /dev 時沒有這樣的文件或目錄」（內核和不匹配）AMI

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
Linux version 2.6.16.33-xenU (root@dom0-0-50-45-1-a4-ee.z-2.aes0.internal)
(gcc version 4.1.1 20070105 (Red Hat 4.1.1-52)) #2 SMP Wed Aug 15 17:27:36 SAST 2007
...
```

```
FATAL: kernel too old
Kernel panic - not syncing: Attempted to kill init!
```

可能的原因

不相容的核心和使用者空間

建議動作

如果是此種執行個體類型	執行此作業
Amazon 支EBS持	請執行下列步驟： 1. 停止執行個體。 2. 修改組態，以使用較新版的核心。 3. 啟動實例。
執行個體存放區後端	請執行下列步驟： 1. 創建一AMI個使用更新內核的。 2. 終止執行個體。 3. 從AMI您建立的執行個體啟動新執行個體。

「FATAL: 無法載入 /lib/ 模組" 或 "BusyBox" (遺失核心模組)

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
[    0.370415] Freeing unused kernel memory: 1716k freed
Loading, please wait...
WARNING: Couldn't open directory /lib/modules/2.6.34-4-virtual: No such file or
 directory
FATAL: Could not open /lib/modules/2.6.34-4-virtual/modules.dep.temp for writing: No
 such file or directory
FATAL: Could not load /lib/modules/2.6.34-4-virtual/modules.dep: No such file or
 directory
Couldn't get a file descriptor referring to the console
Begin: Loading essential drivers... ...
FATAL: Could not load /lib/modules/2.6.34-4-virtual/modules.dep: No such file or
 directory
```

```
FATAL: Could not load /lib/modules/2.6.34-4-virtual/modules.dep: No such file or
 directory
Done.
Begin: Running /scripts/init-premount ...
Done.
Begin: Mounting root file system... ...
Begin: Running /scripts/local-top ...
Done.
Begin: Waiting for root file system... ...
Done.
Gave up waiting for root device.  Common problems:
 - Boot args (cat /proc/cmdline)
   - Check rootdelay= (did the system wait long enough?)
   - Check root= (did the system wait for the right device?)
 - Missing modules (cat /proc/modules; ls /dev)
FATAL: Could not load /lib/modules/2.6.34-4-virtual/modules.dep: No such file or
 directory
FATAL: Could not load /lib/modules/2.6.34-4-virtual/modules.dep: No such file or
 directory
ALERT! /dev/sda1 does not exist. Dropping to a shell!


BusyBox v1.13.3 (Ubuntu 1:1.13.3-1ubuntu5) built-in shell (ash)
Enter 'help' for a list of built-in commands.

(initramfs)
```

可能的原因

這項問題可能是下列一種或多種狀況所造成：

- 缺少記憶體虛擬磁碟 (ramdisk)
- 記憶體虛擬磁碟 (ramdisk) 缺少正確的模組
- Amazon EBS 根卷未正確連接為 `/dev/sda1`

建議動作

如果是此種執行個體類型	執行此作業
Amazon 支EBS持	請執行下列步驟：

如果是此種執行個體類型	執行此作業
	1. 為 Amazon 磁碟EBS區選取更正的磁碟機磁碟。 2. 停止執行個體。 3. 先分離然後修復磁碟區。 4. 將磁碟區連結到執行個體。 5. 啟動實例。 6. 修改AMI以使用更正的磁碟機磁碟。
執行個體存放區後端	請執行下列步驟： 1. 終止執行個體，並使用正確的記憶體虛擬磁碟 (ramdisk) 來啟動新的執行個體。 2. AMI使用正確的磁盤創建一個新的。

ERROR無效的內核（EC2不兼容的內核）

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
...
root (hd0)

 Filesystem type is ext2fs, using whole disk

kernel /vmlinuz root=/dev/sda1 ro

initrd /initrd.img

ERROR Invalid kernel: elf_xen_note_check: ERROR: Will only load images
built for the generic loader or Linux images
xc_dom_parse_image returned -1

Error 9: Unknown boot failure

  Booting 'Fallback'

root (hd0)
```

```
 Filesystem type is ext2fs, using whole disk

kernel /vmlinuz.old root=/dev/sda1 ro

Error 15: File not found
```

可能的原因

這項問題可能是下列狀況中的其中一種或兩種同時造成：

- 提供的核心不受支援 GRUB
- 備用核心不存在

建議動作

如果是此種執行個體類型	執行此作業
Amazon 支EBS持	請執行下列步驟： 1. 停止執行個體。 2. 換成可正常運作的核心。 3. 安裝備用核心。 4. AMI藉由更正核心來修改。
執行個體存放區後端	請執行下列步驟： 1. 終止執行個體，並使用正確的核心來啟動新的執行個體。 2. 使用正確AMI的內核創建一個。 3. (選用) 透過 AWS Support 來尋求資料復原的技術協助。

fsck：嘗試開啟時找不到此等檔案或目錄... (找不到檔案系統)

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
  Welcome to Fedora
```

```
  Press 'I' to enter interactive startup.
Setting clock : Wed Oct 26 05:52:05 EDT 2011 [  OK  ]

Starting udev: [  OK  ]

Setting hostname localhost:  [  OK  ]

No devices found
Setting up Logical Volume Management: File descriptor 7 left open
  No volume groups found
[  OK  ]

Checking filesystems
Checking all file systems.
[/sbin/fsck.ext3 (1) -- /] fsck.ext3 -a /dev/sda1
/dev/sda1: clean, 82081/1310720 files, 2141116/2621440 blocks
[/sbin/fsck.ext3 (1) -- /mnt/dbbackups] fsck.ext3 -a /dev/sdh
fsck.ext3: No such file or directory while trying to open /dev/sdh

/dev/sdh:
The superblock could not be read or does not describe a correct ext2
filesystem.  If the device is valid and it really contains an ext2
filesystem (and not swap or ufs or something else), then the superblock
is corrupt, and you might try running e2fsck with an alternate superblock:
    e2fsck -b 8193 <device>

[FAILED]


*** An error occurred during the file system check.
*** Dropping you to a shell; the system will reboot
*** when you leave the shell.
Give root password for maintenance
(or type Control-D to continue):
```

可能的原因

- ramdisk 檔案系統定義 /etc/fstab 中存在錯誤
- /etc/fstab 中設定錯誤的檔案系統定義
- 找不到磁碟機/磁碟機故障

建議動作

如果是此種執行個體類型	執行此作業
Amazon 支EBS持	請執行下列步驟： 1. 停止執行個體、分離根磁碟區、修復/修改 /etc/fstab 磁碟區、將磁碟區連結到執行個體，以及啟動執行個體。 2. 修正 ramdisk，以納入修改過的 /etc/fstab (如有適用)。 3. 修改AMI以使用較新的磁碟機磁碟。 fstab 中的第 6 個欄位定義了掛載的可用性要求 – 非 0 值暗示 fsck 將在該磁碟區上完成，而且必須成功。在 Amazon 中使用此欄位可能會有問題，EC2因為失敗通常會導致 Amazon 目前無法使用的互動式主控台提示EC2。使用此功能時請特別注意，並閱讀 fstab 的 Linux 手冊頁。
執行個體存放區後端	請執行下列步驟： 1. 終止執行個體並啟動新的執行個體。 2. 卸離任何錯誤的 Amazon EBS 磁碟區和重新啟動執行個體。 3. (選用) 透過 AWS Support 來尋求資料復原的技術協助。

掛載檔案作業系統的一般錯誤 (掛載失敗)

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
Loading xenblk.ko module
xen-vbd: registered block device major 8

Loading ehci-hcd.ko module
Loading ohci-hcd.ko module
```

```
Loading uhci-hcd.ko module
USB Universal Host Controller Interface driver v3.0

Loading mbcache.ko module
Loading jbd.ko module
Loading ext3.ko module
Creating root device.
Mounting root filesystem.
kjournald starting.  Commit interval 5 seconds

EXT3-fs: mounted filesystem with ordered data mode.

Setting up other filesystems.
Setting up new root fs
no fstab.sys, mounting internal defaults
Switching to new root and running init.
unmounting old /dev
unmounting old /proc
unmounting old /sys
mountall:/proc: unable to mount: Device or resource busy
mountall:/proc/self/mountinfo: No such file or directory
mountall: root filesystem isn't mounted
init: mountall main process (221) terminated with status 1

General error mounting filesystems.
A maintenance shell will now be started.
CONTROL-D will terminate this shell and re-try.
Press enter for maintenance
(or type Control-D to continue):
```

可能的原因

執行個體類型	可能的原因
Amazon 支EBS持	• 分離或失敗的 Amazon EBS 卷。 • 已毀損的檔案系統。 • 不匹配的磁盤和AMI組合（例如 Debian 磁盤與 a）。SUSE AMI
執行個體存放區後端	• 故障的磁碟機。

執行個體類型	可能的原因
	• 已毀損的檔案系統。 • 一個不匹配的磁盤和組合（例如，一個 Debian 磁盤與 a）。SUSE AMI

建議動作

如果是此種執行個體類型	執行此作業
Amazon 支EBS持	請執行下列步驟： 1. 停止執行個體。 2. 分離根 磁碟區。 3. 將根磁碟區連結到已知可正常運作的執行個體。 4. 執行檔案系統檢查 (fsck -a /dev/...)。 5. 修正所有錯誤。 6. 將磁碟區從已知可正常運作的執行個體分離。 7. 將磁碟區連結到已停止的執行個體。 8. 啟動實例。 9. 重新檢查執行個體的狀態。
執行個體存放區後端	請嘗試執行下列其中一項操作： • 啟動新的執行個體。 • (選用) 透過 AWS Support 來尋求資料復原的技術協助。

VFS: 無法在未知的區塊上掛載根 fs (根檔案系統不符)

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
Linux version 2.6.16-xenU (builder@xenbat.amazonsa) (gcc version 4.0.1
 20050727 (Red Hat 4.0.1-5)) #1 SMP Mon May 28 03:41:49 SAST 2007
```

```
...
Kernel command line:  root=/dev/sda1 ro 4
...
Registering block device major 8
...
Kernel panic - not syncing: VFS: Unable to mount root fs on unknown-block(8,1)
```

可能的原因

執行個體類型	可能的原因
Amazon 支EBS持	• 未正確地連結裝置。 • 根設備未連結到正確的裝置點。 • 檔案系統未採用預期的格式。 • 使用舊版的核心 (例如 2.6.16-XenU)。 • 在執行個體上最近更新了核心 (更新作業失敗或出現錯誤)
執行個體存放區後端	硬體裝置故障。

建議動作

如果是此種執行個體類型	執行此作業
Amazon 支EBS持	執行以下任意一項： • 停止並重新啟動執行個體。 • 修改根磁碟區，以掛載於正確的裝置點 (可能是 /dev/sda1 而非 /dev/sda)。 • 停止並進行修改，以使用現代化的核心。 • 請參考您 Linux 版本的文件，以查看已知的更新錯誤。變更或重新安裝核心。
執行個體存放區後端	終止執行個體，並使用現代化的核心來啟動新的執行個體。

錯誤：無法判定根設備的主/次編號... (根檔案系統/裝置不相符)

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
...
XENBUS: Device with no driver: device/vif/0
XENBUS: Device with no driver: device/vbd/2048
drivers/rtc/hctosys.c: unable to open rtc device (rtc0)
Initializing network drop monitor service
Freeing unused kernel memory: 508k freed
:: Starting udevd...
done.
:: Running Hook [udev]
:: Triggering uevents...<30>udevd[65]: starting version 173
done.
Waiting 10 seconds for device /dev/xvda1 ...
Root device '/dev/xvda1' doesn't exist. Attempting to create it.
ERROR: Unable to determine major/minor number of root device '/dev/xvda1'.
You are being dropped to a recovery shell
    Type 'exit' to try and continue booting
sh: can't access tty; job control turned off
[ramfs /]#
```

可能的原因

- 找不到虛擬區塊型儲存設備驅動程式，或是該驅動程式設定不正確
- 裝置列舉衝突 (sda 對 xvda 或 sda 而非 sda1)
- 所選的執行個體核心不正確

建議動作

如果是此種執行個體類型	執行此作業
Amazon 支EBS持	請執行下列步驟： 1. 停止執行個體。 2. 分離磁碟區。 3. 修正裝置映射問題。 4. 啟動實例。

如果是此種執行個體類型	執行此作業
	5. 修改AMI以解決裝置對應問題。
執行個體存放區後端	請執行下列步驟： 1. 使用適當AMI的修復程序創建一個新的（正確映射塊設備）。 2. 終止執行個體並從AMI您建立的執行個體啟動新執行個體。

XENBUS：沒有驅動程序的設備...

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
XENBUS: Device with no driver: device/vbd/2048
drivers/rtc/hctosys.c: unable to open rtc device (rtc0)
Initializing network drop monitor service
Freeing unused kernel memory: 508k freed
:: Starting udevd...
done.
:: Running Hook [udev]
:: Triggering uevents...<30>udevd[65]: starting version 173
done.
Waiting 10 seconds for device /dev/xvda1 ...
Root device '/dev/xvda1' doesn't exist. Attempting to create it.
ERROR: Unable to determine major/minor number of root device '/dev/xvda1'.
You are being dropped to a recovery shell
    Type 'exit' to try and continue booting
sh: can't access tty; job control turned off
[ramfs /]#
```

可能的原因

- 找不到虛擬區塊型儲存設備驅動程式，或是該驅動程式設定不正確
- 裝置列舉衝突 (sda 對 xvda)
- 所選的執行個體核心不正確

建議動作

如果是此種執行個體類型	執行此作業
Amazon 支EBS持	請執行下列步驟： 1. 停止執行個體。 2. 分離磁碟區。 3. 修正裝置映射問題。 4. 啟動實例。 5. 修改AMI以解決裝置對應問題。
執行個體存放區後端	請執行下列步驟： 1. 使用適當AMI的修復程序創建一個（正確地映射塊設備）。 2. 終止執行個體並使用AMI您建立的啟動新執行個體。

...未進行檢查的天數，強制進行檢查 (需要進行檔案系統檢查)

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
...
Checking filesystems
Checking all file systems.
[/sbin/fsck.ext3 (1) -- /] fsck.ext3 -a /dev/sda1
/dev/sda1 has gone 361 days without being checked, check forced
```

可能的原因

已超過檔案系統檢查時間；強制進行檔案系統檢查。

建議動作

- 請等候檔案系統檢查完成。視根檔案系統的大小而定，檔案系統檢查可能需要一段長時間才會完成。
- 使用 tune2fs 或檔案系統適用的工具，來修改檔案系統，以移除強制進行檔案系統檢查 (fsck) 的功能。

fsck 凍結於結束狀態... (缺少裝置)

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
Cleaning up ifupdown....
Loading kernel modules...done.
...
Activating lvm and md swap...done.
Checking file systems...fsck from util-linux-ng 2.16.2
/sbin/fsck.xfs: /dev/sdh does not exist
fsck died with exit status 8
[31mfailed (code 8).[39;49m
```

可能的原因

- Ramdisk 尋找缺少的磁碟機
- 強制進行檔案系統一致性檢查
- 磁碟機故障或已分離

建議動作

如果是此種執行個體類型	執行此作業
Amazon 支EBS持	請嘗試執行下列其中一項或多項動作來解決問題： • 停止執行個體、將磁碟區連結到執行中的現有執行個體。 • 手動執行一致性檢查。 • 修正 ramdisk，以加入相關的公用程式。 • 修改檔案系統調校參數，以移除一致性要求 (不建議)。
執行個體存放區後端	請嘗試執行下列其中一項或多項動作來解決問題： • 將 ramdisk 與正確的工具重新綁定為套件。

如果是此種執行個體類型	執行此作業
	• 修改檔案系統調校參數，以移除一致性要求 (不建議)。 • 終止執行個體並啟動新的執行個體。 • (選用) 透過 AWS Support 來尋求資料復原的技術協助。

GRUB提示符（灰塵 > ）

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
    GNU GRUB  version 0.97  (629760K lower / 0K upper memory)

       [ Minimal BASH-like line editing is supported.   For

         the   first   word,  TAB  lists  possible  command

         completions.  Anywhere else TAB lists the possible

         completions of a device/filename. ]

grubdom>
```

可能的原因

執行個體類型	可能的原因
Amazon 支EBS持	• 缺少GRUB配置文件。 • 使用不正確的GRUB圖像，預期GRUB配置文件位於不同的位置。 • 用來儲存GRUB組態檔案的不支援檔案系統 (例如，將根檔案系統轉換為舊版不支援的類型GRUB)。
執行個體存放區後端	• 缺少GRUB配置文件。

執行個體類型	可能的原因
	• 使用不正確的GRUB圖像，預期GRUB配置文件位於不同的位置。 • 用來儲存GRUB組態檔案的不支援檔案系統(例如，將根檔案系統轉換為舊版不支援的類型GRUB)。

建議動作

如果是此種執行個體類型	執行此作業
Amazon 支EBS持	選項 1：修改AMI並重新啟動實例： 1. 修改源代碼AMI以在標準位置（/boot/grub/menu.lst）創建一個GRUB配置文件。 2. 確認您的版本GRUB支援基礎檔案系統類型，並視需GRUB要進行升級。 3. 選擇適當的GRUB圖像，（HD0-1 驅動器或HD00-第一個驅動器，第一個分區）。 4. 終止執行個體並使用您建立的執行個體啟動新執行AMI個體。 選項 2：修改現有的執行個體： 1. 停止執行個體。 2. 分離根檔案系統。 3. 將根檔案系統連結到已知可正常運作的執行個體。 4. 掛載檔案系統。 5. 建立GRUB組態檔案。 6. 確認您的版本GRUB支援基礎檔案系統類型，並視需GRUB要進行升級。 7. 分離檔案系統。

如果是此種執行個體類型	執行此作業
	8. 連結到原始執行個體。 9. 修改內核屬性以使用適當的GRUB映像（第一個磁盤或第一個磁盤上的第一個分區）。 10啟動實例。
執行個體存放區後端	選項 1：修改AMI並重新啟動實例： 1. 在標準位置（/boot/grub/menu.lst）AMI使用GRUB配置文件創建新的文件。 2. 選擇適當的GRUB圖像，（HD0-1 驅動器或HD00-第一個驅動器，第一個分區）。 3. 確認您的版本GRUB支援基礎檔案系統類型，並視需GRUB要進行升級。 4. 終止執行個體並使用AMI您建立的啟動新執行個體。 選項 2：終止執行個體，並啟動新的執行個體(指定正確的核心)。 Note 若要從現有的執行個體復原資料，請聯絡 AWS Support。

啟動介面 eth0：裝置 eth0 具有與預期不同的MAC位址，而忽略。(硬式編碼MAC位址)

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
...
Bringing up loopback interface:  [  OK  ]

Bringing up interface eth0:  Device eth0 has different MAC address than expected,
 ignoring.
```

```
[FAILED]

Starting auditd: [  OK  ]
```

可能的原因

配置中有一個硬編碼MAC的AMI接口

建議動作

如果是此種執行個體類型	執行此作業
Amazon 支EBS持	執行以下任意一項： • 修改AMI以移除硬式編碼並重新啟動執行個體。 • 修改執行個體以移除硬式編碼MAC位址。 或是 請執行下列步驟： 1. 停止執行個體。 2. 分離根 磁碟區。 3. 將磁碟區連接到另一個執行個體，並修改磁碟區以移除硬式編碼MAC位址。 4. 將磁碟區連結到原始執行個體。 5. 啟動實例。
執行個體存放區後端	執行以下任意一項： • 修改執行個體以移除硬式編碼MAC位址。 • 終止執行個體並啟動新的執行個體。

無法載入SELinux策略。機器正處於強制執行模式。現在停止中。（SELinux配置錯誤）

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
audit(1313445102.626:2): enforcing=1 old_enforcing=0 auid=4294967295
Unable to load SELinux Policy. Machine is in enforcing mode. Halting now.
Kernel panic - not syncing: Attempted to kill init!
```

可能的原因

SELinux已啟用錯誤：

- 提供的核心不受支援 GRUB
- 備用核心不存在

建議動作

如果是此種執行個體類型	執行此作業
Amazon 支EBS持	請執行下列步驟： 1. 停止已故障的執行個體。 2. 分離已故障執行個體的根磁碟區。 3. 將根磁碟區連結到另一個執行中的 Linux 執行個體 (之後稱為復原執行個體)。 4. 連線到復原執行個體，並掛載已故障執行個體的根磁碟區。 5. 在掛接的SELinux根磁碟區上停用。不同 Linux 版本的此項程序也會有所差異，如需詳細資訊，請參閱特定作業系統的文件。 Note 在某些系統上，您可以在/*mount_point* /etc/sysconfig/selinux 檔案SELINUX=d

如果是此種執行個體類型	執行此作業
	isabled 中設定SELinux來停用，其中*mount_point* 是您在復原執行個體上掛接磁碟區的位置。 6. 取消掛載根磁碟區、將該磁碟區從復原執行個體分離，然後再重新連結到原始執行個體。 7. 啟動實例。
執行個體存放區後端	請執行下列步驟： 1. 終止執行個體並啟動新的執行個體。 2. (選用) 透過 AWS Support 來尋求資料復原的技術協助。

XENBUS：連接到設備的超時（Xenbus 超時）

系統日誌記錄項目會顯示此狀況，類似於下列所示。

```
Linux version 2.6.16-xenU (builder@xenbat.amazonsa) (gcc version 4.0.1
20050727 (Red Hat 4.0.1-5)) #1 SMP Mon May 28 03:41:49 SAST 2007
...
XENBUS: Timeout connecting to devices!
...
Kernel panic - not syncing: No init found.  Try passing init= option to kernel.
```

可能的原因

- 區塊型儲存設備未連線到執行個體
- 執行個體使用舊版的執行個體核心

建議動作

如果是此種執行個體類型	執行此作業
Amazon 支EBS持	執行以下任意一項：

如果是此種執行個體類型	執行此作業
	• 修改AMI和執行個體以使用現代核心並重新啟動執行個體。 • 重新啟動執行個體。
執行個體存放區後端	請執行下列其中一項： • 終止執行個體。 • 修改AMI為使用現代核心，然後使用此功能啟動新的執行個體AMI。

疑難排解從錯誤的磁碟區開機的 Amazon EC2 Linux 執行個

在某些情況下，除了連接至 Linux 執行個體的磁碟區/dev/xvda或磁碟區以外的磁碟區/dev/sda會變成根磁碟區。這可能是因為您已經將另一個執行個體的根磁碟區，或是從根磁碟區快照所建立的磁碟區，連結到包含現有根磁碟區的執行個體。

這是 Linux 中初始記憶體虛擬磁碟 (ramdisk) 的運作方式所造成的。ramdisk 會選擇在 / 中定義為 /etc/fstab 的磁碟區，而在某些發行版本中，這取決於附加至磁碟區分割區的標籤。具體來說，您會發現 /etc/fstab 看起來類似於下列的內容：

```
LABEL=/ / ext4 defaults,noatime 1 1
tmpfs /dev/shm tmpfs defaults 0 0
devpts /dev/pts devpts gid=5,mode=620 0 0
sysfs /sys sysfs defaults 0 0
proc /proc proc defaults 0 0
```

如果查看這兩個磁碟區的標籤，會發現皆包含 / 標籤：

```
[ec2-user ~]$ sudo e2label /dev/xvda1
/
[ec2-user ~]$ sudo e2label /dev/xvdf1
/
```

在此範例中，最後可能是由 /dev/xvdf1 成為根設備，您的執行個體會在初始的 ramdisk 執行之後，透過此根設備啟動，而非透過您原本要從中啟動的 /dev/xvda1 磁碟區來啟動。為了解決這項問題，請使用同樣的 e2label 命令，針對您不想從其啟動的已連結磁碟區，來變更其標籤。

在某些情況下，指定 UUID in `/etc/fstab` 可以解決此問題。但是，如果兩個磁碟區都來自同一個快照，或是從主要磁碟區的快照建立次要磁碟區，則它們會共用UUID.

```
[ec2-user ~]$ sudo blkid
/dev/xvda1: LABEL="/" UUID=73947a77-ddbe-4dc7-bd8f-3fe0bc840778 TYPE="ext4"
 PARTLABEL="Linux" PARTUUID=d55925ee-72c8-41e7-b514-7084e28f7334
/dev/xvdf1: LABEL="old/" UUID=73947a77-ddbe-4dc7-bd8f-3fe0bc840778 TYPE="ext4"
 PARTLABEL="Linux" PARTUUID=d55925ee-72c8-41e7-b514-7084e28f7334
```

更改已連結 ext4 磁碟區的標籤

1. 使用 e2label 命令，將磁碟區的標籤變更為除了 / 以外的名稱。

   ```
   [ec2-user ~]$ sudo e2label /dev/xvdf1 old/
   ```

2. 確認磁碟區已具有新的標籤。

   ```
   [ec2-user ~]$ sudo e2label /dev/xvdf1
   old/
   ```

更改已連結 xfs 磁碟區的標籤

- 使用 xfs_admin 命令，將磁碟區的標籤變更為除了 / 以外的名稱。

  ```
  [ec2-user ~]$ sudo xfs_admin -L old/ /dev/xvdf1
  writing all SBs
  new label = "old/"
  ```

在如上述變更磁碟區標籤之後，您應該就能夠重新啟動執行個體，而且讓初始 ramdisk 在執行個體啟動時，選擇正確的磁碟區。

> **Important**
>
> 如果您打算分離具有新標籤的磁碟區，並將該磁碟區傳回給另一個執行個體做為根磁碟區使用，則您必須再次執行上述的程序，以將磁碟區的標籤改回原來的值。否則，其他的執行個體就不會啟動，因為 ramdisk 找不到具有 / 標籤的磁碟區。

疑難排解連線到 Amazon 視EC2窗執行個體的

下列資訊和常見錯誤可協助您疑難排解連線至 Windows 執行個體時的問題。

連線問題

- 遠端桌面無法連線到遠端電腦
- 使用 macOS 用RDP戶端時發生錯誤
- RDP顯示黑屏而不是桌面
- 無法使用非管理員的使用者從遠端登入至執行個體
- 使用的遠程桌面問題故障 AWS Systems Manager
- 在具有遠端登錄的EC2執行個體上啟用遠端桌面
- 我的私有金鑰遺失。如何連接到我的 Windows 執行個體？

遠端桌面無法連線到遠端電腦

嘗試下列方法解決執行個體連線相關問題：

- 確認您使用的是正確的公用DNS主機名稱。在 Amazon EC2 主控台中，選取執行個體，然後勾選詳細資料窗格中的 Pub lic DNS (IPv4)。) 如果您的執行個體位於VPC，但您看不到公開DNS名稱，則必須啟用DNS主機名稱。如需詳細資訊，請參閱 Amazon VPC 使用者指南VPC中的DNS屬性。
- 確認您的執行個體具有公開IPv4位址。如果沒有，您可建立彈性 IP 地址與您執行個體的關聯。如需詳細資訊，請參閱 彈性 IP 地址。
- 若要使用IPv6位址連線至執行個體，請檢查您的本機電腦是否有IPv6位址且已設定為使用IPv6。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的在執行個體IPv6上進行設定。
- 確認您的安全性群組具有允許在通訊埠 3389 上RDP存取的規則。
- 如果您已複製密碼但收到錯誤 `Your credentials did not work`，請嘗試在提示時手動輸入密碼。有可能是您在複製密碼時少了一個字元或多了一個空白字元。
- 請確認執行個體已通過狀態檢查。如需詳細資訊，請參閱 Amazon EC2 執行個體的狀態檢查 及 the section called “Linux 執行個體狀態檢查失敗”。
- 確認子網路的路由表具有路由，可將所有流量傳送VPC到的網際網路閘道之VPC外。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的建立自訂路由表 (網際網路閘道)。
- 確認 Windows 防火牆或其他防火牆軟體未封鎖RDP傳送至執行個體的流量。我們建議您停用 Windows 防火牆，並使用安全群組規則控制對您執行個體的存取。您可以使用 AWSSupport-TroubleshootRDP 以 disable the Windows Firewall profiles using SSM Agent：若要在未設定 AWS

Systems Manager、使AWSSupport-ExecuteEC2Rescue用或使用下列手動步驟的 Windows 執行個體上停用 Windows 防火牆：

手動步驟

1. 停止受影響的執行個體並分離其根磁碟區。
2. 啟動與受影響執行個體同一可用區域中的暫時執行個體。

> **Warning**
> 如果您的暫存執行個體是以原始執行AMI個體所依據的相同為基礎，您必須完成其他步驟，否則您將無法在還原其根磁碟區後，因為磁碟簽章衝突而無法啟動原始執行個體。或者，為暫存例證選AMI取不同的執行個體。例如，如果原始執行個體使AMI用 AWS 視窗伺服器 2016 年版，請使用 AWS 視窗AMI啟動暫存執行個體。

3. 從受影響的執行個體將根磁碟區連接到此暫時執行個體。連線至暫時執行個體、開啟 Disk Management (磁碟管理) 公用程式，然後讓磁碟機上線。
4. 打開註冊表編輯器，然後選擇 HKEY_ LOCAL _ MACHINE。從 File (檔案) 功能表中，選擇 Load Hive (載入 Hive)。選取磁碟機、開啟檔案 `Windows\System32\config\SYSTEM`，然後在提示時指定金鑰名稱 (您可使用任何名稱)。
5. 選擇剛載入的機碼並導覽至 `ControlSet001\Services\SharedAccess\Parameters\FirewallPolicy`。為名稱格式為 xxxxProfile 的每一個金鑰選取金鑰，並將 `EnableFirewall` 從 1 變更為 0。再次選取該金鑰，然後從 File (檔案) 功能表中選擇 Unload Hive (取消載入 Hive)。
6. (選擇性) 如果您的暫存執行個體是以原始執行AMI個體所依據的相同為基礎，您必須完成下列步驟，否則您將無法在還原其根磁碟區後，因為磁碟簽章衝突而無法啟動原始執行個體。

> **Warning**
> 下列程序說明如何使用登錄編輯程式來編輯 Windows 登錄檔。如果您不熟悉 Windows 登錄檔或如何使用登錄編輯程式來安全地進行變更，請參閱 設定登錄檔。

   a. 開啟命令提示，輸入 regedit.exe，然後按 Enter。
   b. 在 [登錄編輯程式] 中，MACHINE從內容功能表中選擇 [HKEYLOCAL_ _] (按一下滑鼠右鍵)，然後選擇 [尋找]。

c. 輸入 Windows Boot Manager，然後選擇 Find Next (尋找下一個)。

d. 選擇名為 `11000001` 的金鑰。此金鑰是您在上一個步驟中找到的金鑰的同級。

e. 在右窗格中，選擇 `Element`，然後從內容功能表中選擇 Modify (修改)(按一下右鍵)。

f. 在資料中的位移為 0x38 處找到四位元組的磁碟簽章。反轉位元組以建立磁碟簽章，然後寫下來。例如，下列資料所代表的磁碟簽章是 `E9EB3AA5`：

```
...
0030  00 00 00 00 01 00 00 00
0038  A5 3A EB E9 00 00 00 00
0040  00 00 00 00 00 00 00 00
...
```

g. 在 [命令提示字元] 視窗中，執行下列命令以啟動 Microsoft DiskPart。

```
diskpart
```

h. 執行下列 DiskPart 命令以選取磁碟區。(可以使用磁碟管理公用程式，確認磁碟編號為 1)。

```
DISKPART> select disk 1

Disk 1 is now the selected disk.
```

i. 運行以下 DiskPart 命令以獲取磁盤簽名。

```
DISKPART>  uniqueid disk

Disk ID: 0C764FA8
```

j. 如果上一步中顯示的磁碟簽章與之前寫下BCD的磁碟簽章不符，請使用下列 DiskPart 命令來變更磁碟簽章，使其符合：

```
DISKPART> uniqueid disk id=E9EB3AA5
```

7. 使用 Disk Management (磁碟管理) 公用程式讓磁碟機離線。

> **Note**
>
> 如果暫時執行個體執行的作業系統與受影響執行個體的相同，磁碟機會自動離線，因此您不需要手動將其離線。

8. 分離磁碟區與暫時執行個體。如果不再需要使用此暫時執行個體，您可終止它。
9. 將受影響執行個體的根磁碟區連接為 `/dev/sda1` 來還原它。
10. 啟動實例。

- 在不屬於 Active Directory 域的執行個體上，確認網路層級身分驗證已停用 (使用 AWSSupport-TroubleshootRDP 以 disable NLA)。
- 確認遠端桌面服務 (TermService) 啟動類型為 [自動]，且服務已啟動 (AWSSupport-TroubleshootRDP使用enable and start the RDP service)。
- 確認您連線至正確的遠端桌面通訊協定連接埠，預設為 3389 (使用 AWSSupport-TroubleshootRDP 以 read the current RDP port 和 change it back to 3389)。
- 確認您的執行個體上允許遠端桌面連線 (使用 AWSSupport-TroubleshootRDP 以 enable Remote Desktop connections)。
- 請確認密碼尚未過期。如果密碼已過期，您可以重設它。如需詳細資訊，請參閱 重設 Amazon EC2 視窗執行個體的管理員密碼。
- 如果您嘗試使用在此執行個體上建立的使用者進行連線，但收到錯誤 `The user cannot connect to the server due to insufficient access privileges`，請確認您已授權使用者可在本機登入。如需詳細資訊，請參閱授予成員登入本機的權限。
- 如果您嘗試的同時工RDP作階段超過允許的最大值，則會以訊息終止您的工作階段。根`Your Remote Desktop Services session has ended. Another user connected to the remote computer, so your connection was lost.`據預設，您可以使用兩個並行RDP工作階段進入您的執行環境。

使用 macOS 用RDP戶端時發生錯誤

如果您從 Microsoft 網站使用遠端桌面連線用戶端連線到 Windows 伺服器執行個體，您可能會收到下列錯誤：

```
Remote Desktop Connection cannot verify the identity of the computer that you want to
 connect to.
```

從 Mac App Store 下載 Microsoft Remote Desktop 應用程式，然後使用該應用程式連線到執行個體。

RDP顯示黑屏而不是桌面

請嘗試下列方法解決此問題：

- 如需其他資訊，請參閱主控台輸出。若要使用 Amazon 主控台取得執行個體的EC2主控台輸出，請選取執行個體，然後選擇 [動作]、[監控和疑難排解]、[取得系統記錄]。
- 確認您執行的是最新版本的RDP用戶端。
- 請嘗試用RDP戶端的預設設定。如需詳細資訊，請參閱 遠端工作階段環境。
- 如果您使用的是遠端桌面連線，請嘗試使用 `/admin` 選項啟動它，如下所示。

```
mstsc /v:instance /admin
```

- 如果伺服器執行全螢幕應用程式，可能已停止回應。請使用 Ctrl+Shift+Esc 啟動 Windows 任務管理員，然後結束應用程式。
- 如果伺服器過度使用，可能已停止回應。若要使用 Amazon EC2 主控台監控執行個體，請選取執行個體，然後選取監控索引標籤。如果您需要將執行個體類型變更為較大的大小，請參閱 Amazon EC2 實例類型更改。

無法使用非管理員的使用者從遠端登入至執行個體

如果您無法使用非管理員帳戶的使用者從遠端登入 Windows 執行個體，請確定您已授予該使用者從本機登入的權限。請參閱 授予使用者或群組權限，以從本機登入網域中的網域控制器。

使用的遠程桌面問題故障 AWS Systems Manager

您可以使用疑 AWS Systems Manager 難排解連線至 Windows 執行個體的問題RDP。

AWSSupport-疑難排解 RDP

AWSSupport-疑難排解RDP自動化文件可讓使用者檢查或修改目標執行個體上可能會影響遠端桌面通訊協定 (RDP) 連線的一般設定，例如連接RDP埠、網路層驗證 (NLA) 和 Windows 防火牆設定檔。根據預設，此文件會讀取和輸出這些設定的值。

AWSSupport-疑難排解RDP自動化文件可用於啟用 EC2 (代管執行個體VMs) 的執行個體、內部部署執行個體和虛擬機器 AWS Systems Manager ()。此外，它也可以與 Windows 伺服器的EC2執行個體搭配使用，這些執行個體並未啟用與 Systems Manager 搭配使用。如需有關啟用執行個體以搭配使用的資訊 AWS Systems Manager，請參閱《使AWS Systems Manager 用指南》中的受管節點。

使用疑難排解文件進 AWSSupport行疑難排解 RDP

1. 登入 Systems Manager 主控台。
2. 確認您位於受損執行個體所在的同一區域。

3. 從左側導覽窗格中選擇 Documents (文件)。
4. 在 Owned by Amazon (由 Amazon 所有) 標籤中，在搜尋欄位中輸入 `AWSSupport-TroubleshootRDP`。當 `AWSSupport-TroubleshootRDP` 文件出現時，請選取它。
5. 選擇 Execute automation (執行自動化)。
6. 對於 Execution Mode (執行模式)，選擇 Simple execution (簡易執行)。
7. 針對輸入參數 InstanceId，啟用顯示互動式執行個體選擇器。
8. 選擇您的 Amazon EC2 實例。
9. 檢閱範例，然後選擇 Execute (執行)。
10. 若要監控執行進度，對於 Execution status (執行狀態)，等待狀態從 Pending (待定) 變成 Success (成功)。展開輸出以檢視結果。若要檢視個別步驟的輸出，請在 Executed Steps (執行的步驟) 中從 Step ID (步驟 ID) 選擇一個項目。

AWSSupport-疑難排解範例 RDP

下列範例說明如何使用 AWSSupport-疑難排解完成常見的疑難排解RDP工作。您可以使用範例 AWS CLI start-automation-execution指令或提供的連結到 AWS Management Console.

Example 範例：檢查目前RDP狀態

AWS CLI:

```
aws ssm start-automation-execution --document-name "AWSSupport-TroubleshootRDP" --
parameters "InstanceId=instance_id, Action=Custom" --region region_code
```

AWS Systems Manager 控制台：

```
https://console.aws.amazon.com/systems-manager/automation/execute/AWSSupport-
TroubleshootRDP?region=region#documentVersion=$LATEST
```

Example 範例：停用 Windows 防火牆

AWS CLI:

```
aws ssm start-automation-execution --document-name "AWSSupport-TroubleshootRDP"
 --parameters "InstanceId=instance_id, Action=Custom, Firewall=Disable" --
region region_code
```

AWS Systems Manager 控制台：

```
https://console.aws.amazon.com/systems-manager/automation/execute/AWSSupport-
TroubleshootRDP?region=region_code#documentVersion=$LATEST&Firewall=Disable
```

Example 範例：停用網路層級身分驗證

AWS CLI:

```
aws ssm start-automation-execution --document-name "AWSSupport-TroubleshootRDP"
 --parameters "InstanceId=instance_id, Action=Custom, NLASettingAction=Disable" --
region region_code
```

AWS Systems Manager 控制台：

```
https://console.aws.amazon.com/systems-manager/automation/execute/AWSSupport-
TroubleshootRDP?region=region_code#documentVersion
```

Example 範例：將「RDP服務啟動類型」設定為「自動」並啟動RDP服務

AWS CLI:

```
aws ssm start-automation-execution --document-name "AWSSupport-TroubleshootRDP"
 --parameters "InstanceId=instance_id, Action=Custom, RDPServiceStartupType=Auto,
 RDPServiceAction=Start" --region region_code
```

AWS Systems Manager 控制台：

```
https://console.aws.amazon.com/systems-manager/automation/execute/
AWSSupport-TroubleshootRDP?region=region_code#documentVersion=
$LATEST&RDPServiceStartupType=Auto&RDPServiceAction=Start
```

Example 範例：還原預設RDP連接埠 (3389)

AWS CLI:

```
aws ssm start-automation-execution --document-name "AWSSupport-TroubleshootRDP"
 --parameters "InstanceId=instance_id, Action=Custom, RDPPortAction=Modify" --
region region_code
```

AWS Systems Manager 控制台：

```
https://console.aws.amazon.com/systems-manager/automation/execute/AWSSupport-
TroubleshootRDP?region=region_code#documentVersion=$LATEST&RDPPortAction=Modify
```

Example 範例：允許遠端連線

AWS CLI:

```
aws ssm start-automation-execution --document-name "AWSSupport-TroubleshootRDP"
 --parameters "InstanceId=instance_id, Action=Custom, RemoteConnections=Enable" --
region region_code
```

AWS Systems Manager 控制台：

```
https://console.aws.amazon.com/systems-manager/automation/execute/AWSSupport-
TroubleshootRDP?region=region_code#documentVersion=$LATEST&RemoteConnections=Enable
```

AWSSupport-執行 EC2Rescue

AWSSupport-執行自動EC2Rescue化文件使EC2Rescue用 Windows Server 自動疑難排解和還原EC2執行個體連線和RDP問題。如需詳細資訊，請參閱在無法存取的執行個體上執行EC2Rescue工具。

AWSSupport-Execute EC2Rescue 自動化文件需要停止並重新啟動執行個體。Systems Manager 自動化會停止執行個體並建立 Amazon 機器映像 (AMI)。存放在執行個體存放磁碟區的資料會遺失。如果您不是使用彈性 IP 地址，則公有 IP 地址會變更。如需詳細資訊，請參閱《AWS Systems Manager 使用指南》中的對無法存取的執行個體執行EC2Rescue工具。

使用 AWSSupport-Execute 文件進行EC2Rescue疑難排解

1. 開啟 Systems Manager 主控台。
2. 確認您與受損的 Amazon EC2 執行個體位於相同的區域。
3. 在導覽面板中，選擇文件。
4. 搜尋並選取 `AWSSupport-ExecuteEC2Rescue` 文件，然後選擇執行自動化。
5. 在 Execution Mode (執行模式) 中，選擇 Simple execution (簡易執行)。
6. 在「輸入參數」區段中，輸入無法存取之EC2執行個體的 Amazon 執行個體 ID。UnreachableInstanceId

7. (選擇性) 若要收集作業系統日誌以進行 Amazon 執行個體疑難排解 LogDestination，請輸入 Amazon 簡單儲存服務 (Amazon S3) 儲存貯EC2體名稱。日誌會自動上傳至指定的儲存貯體。
8. 選擇 Execute (執行)。
9. 若要監控執行進度，請在 Execution status (執行狀態) 中，等待狀態從 Pending (待定) 變成 Success (成功)。展開輸出以檢視結果。若要檢視個別步驟的輸出，請在 Executed Steps (執行的步驟) 中選擇 Step ID (步驟 ID)。

在具有遠端登錄的EC2執行個體上啟用遠端桌面

如果您無法存取的執行個體不是由 AWS 系統管理員工作階段管理員所管理，則您可以使用遠端登錄來啟用遠端桌面。

1. 從EC2主控台停止無法連線的執行個體。
2. 將無法連線之執行個體的根磁碟區，連接到相同可用區域中的另一個執行個體以做為儲存磁碟區。如果同一可用區域中沒有可連線的執行個體，請啟動一個執行個體。記下無法連線的執行個體上根磁碟區的裝置名稱。
3. 在可連線的執行個體上，開啟「磁碟管理」。在「命令提示」視窗中執行下列命令，即可執行這項操作。

   ```
   diskmgmt.msc
   ```

4. 以滑鼠右鍵按一下來自無法連線之執行個體的新連接磁碟區，然後選擇線上。
5. 開啟 Windows 登錄編輯程式。在「命令提示」視窗中執行下列命令，即可執行這項操作。

   ```
   regedit
   ```

6. 在登錄編輯程式中，選擇 HKEYLOCAL_ _ MACHINE，然後選取檔案，載入 Hive。
7. 選取連接磁碟區的磁碟機，導覽至 `\Windows\System32\config\`，選取 `SYSTEM`，然後選擇 Open (開啟)。
8. 對於 Key Name (鍵值名稱)，請輸入 hive 的唯一名稱，並選擇 OK (確定)。
9. 請先備份登錄 hive，再對登錄進行任何變更。
   a. 在登錄編輯程式主控台樹狀目錄中，選取您載入的 Hive：HKEYLOCAL_ _ MACHINE*your-key-name*.
   b. 依序選擇檔案、匯出。

c. 在「匯出登錄檔案」對話方塊中，選擇您要儲存備份複本的位置，然後在 File name (檔案名稱) 欄位中輸入備份檔案的名稱。

d. 選擇 Save (儲存)。

10. 在「登錄編輯程式」中`HKEY_LOCAL_MACHINE\your key name\ControlSet001\Control\Terminal Server`，瀏覽至，然後在詳細資料窗格中按兩下fDenyTSConnections。
11. 在「編輯DWORD值」方塊`0`中，於「值資料」欄位中輸入。
12. 選擇 OK (確定)。

> **Note**
>
> 如果 Value data (值資料) 欄位中的值為 `1`，則執行個體將拒絕遠端桌面連線。值 `0` 允許遠端桌面連線。

13. 在「登錄編輯程式」中，選擇 HKEYLOCAL_ _ MACHINE*your-key-name*，然後選擇文件，卸載配置單元。
14. 關閉「「登錄編輯程式」」和「磁碟管理」。
15. 從EC2主控台中卸離磁碟區與可連線的執行個體，然後將其重新連接至無法連線的執行個體。將磁碟區連接到無法連線的執行個體時，請在裝置欄位中輸入您稍早儲存的裝置名稱。
16. 重新啟動無法連線的執行個體。

我的私有金鑰遺失。如何連接到我的 Windows 執行個體？

當您連線到新啟動的 Windows 執行個體時，使用您在啟動執行個體時所指定金鑰對中的私有金鑰，將管理員帳戶的密碼解密。

如果您遺失了管理員密碼，而且也不再擁有私有金鑰，則必須重設密碼或建立新的執行個體。如需詳細資訊，請參閱 重設 Amazon EC2 視窗執行個體的管理員密碼。如需使用「Systems Manager」文件重設密碼的步驟，請參閱「使用AWS Systems Manager 者指南」中的「重設EC2執行個體的密碼和SSH金鑰」。

Amazon EC2 視窗執行個體啟動問題

以下是可協助您解決 Amazon EC2 Windows 執行個體的密碼和啟用問題的疑難排解秘訣。

問題

- 「未提供密碼」
- 「尚未提供密碼」
- 「無法擷取 Windows 密碼」
- 等待中繼資料服務
- 無法啟動 Windows
- Windows 不是正版 (0x80070005)
- 無終端機伺服器的授權伺服器可提供授權
- 「部分設定是由您的組織所管理」

「未提供密碼」

若要使用遠端桌面連線到 Windows 執行個體，您必須指定帳戶和密碼。所提供的帳戶和密碼是根據您用來啟動執行個體的帳戶和密碼而定。AMI您可以擷取系統管理員帳戶的自動產生密碼，也可以使用在建立該帳戶的原始執行個體中使用的帳戶和密碼。AMI

您可以為使用自訂 Windows 啟動的執行個體產生管理員帳戶的密碼AMI。要生成密碼，您需要在創建之前在操作系統中配置一些設置。AMI如需詳細資訊，請參閱創建一個 Amazon 支EBS持 AMI。

如果您的 Windows 執行個體未設定產生隨機密碼，當您使用主控台擷取自動產生的密碼時，就會收到以下訊息：

```
Password is not available.
The instance was launched from a custom AMI, or the default password has changed. A
password cannot be retrieved for this instance. If you have forgotten your password,
 you can
reset it using the Amazon EC2 configuration service. For more information, see
 Passwords for a
Windows Server instance.
```

檢查執行個體的主控台輸出，查看您用來啟動執行個體的輸出是否已停用密碼產生功AMI能的情況下建立。如果停用產生密碼，主控台輸出會包含以下內容：

```
Ec2SetPassword: Disabled
```

如果停用產生密碼而您不記得原始執行個體的密碼，您可重設此執行個體的密碼。如需詳細資訊，請參閱重設 Amazon EC2 視窗執行個體的管理員密碼。

「尚未提供密碼」

若要使用遠端桌面連線到 Windows 執行個體，您必須指定帳戶和密碼。所提供的帳戶和密碼是根據您用來啟動執行個體的帳戶和密碼而定。AMI您可以擷取系統管理員帳戶的自動產生密碼，也可以使用在建立該帳戶的原始執行個體中使用的帳戶和密碼。AMI

您應該會在幾分鐘內收到密碼。如果未提供密碼，當您使用主控台擷取自動產生的密碼時，就會收到以下訊息：

```
Password not available yet.
Please wait at least 4 minutes after launching an instance before trying to retrieve
 the
auto-generated password.
```

如果超過四分鐘仍未收到密碼，則可能是執行個體的啟動代理程式未設定為產生密碼。請檢查主控台輸出是否為空以確認。如需詳細資訊，請參閱無法取得主控台輸出。

同時確認用來存取管理入口網站的 AWS Identity and Access Management (IAM) 帳戶是否具有允許的`ec2:GetPasswordData`動作。如需有關IAM權限的詳細資訊，請參閱什麼是IAM？ 。

「無法擷取 Windows 密碼」

若要擷取管理員帳戶的自動產生密碼，您必須使用啟動執行個體時所指定之金鑰對的私有金鑰。如果您在啟動執行個體時未指定金鑰對，就會收到下列訊息。

```
Cannot retrieve Windows password
```

您可以終止此執行個體並使用相同的執行個體啟動新執行個體AMI，並確定指定 key pair。

等待中繼資料服務

Windows 執行個體必須先從自己的執行個體中繼資料取得資訊，才能自行啟動。依預設，此`WaitForMetaDataAvailable`設定可確保EC2Config服務會等待執行個體中繼資料存取，然後再繼續開機程序。如需詳細資訊，請參閱使用執行個體中繼資料管理EC2執行個。

如果執行個體的執行個體連線能力測試失敗，請嘗試下列方法解決此問題。

- 檢查CIDR塊為您的VPC. 如果 Windows 執行個體啟動到的 IP 位址範圍從`224.0.0.0`到`255.255.255.255` (類別 D 和類別 E IP 位址範圍)，則無法正確開機。VPC這些是保留的 IP 地址範圍，不應該指派給主機裝置。我們建議您使VPC用 RFC19 18 中指定的私有（非公開路由）IP 位址範圍建立一個CIDR區塊。

- 系統可能已以靜態 IP 地址設定。請嘗試建立網路介面並連接至執行個體。
- 若要DHCP在您無法連線到的 Windows 執行個體上啟用

1. 停止受影響的執行個體並分離其根磁碟區。
2. 啟動與受影響執行個體同一可用區域中的暫時執行個體。

> **Warning**
>
> 如果您的暫存執行個體是以原始執行AMI個體所依據的相同為基礎，您必須完成其他步驟，否則您將無法在還原其根磁碟區後，因為磁碟簽章衝突而無法啟動原始執行個體。或者，為暫存例證選AMI取不同的執行個體。例如，如果原始執行個體使用 AWS 視窗AMI伺服器 2016 年版，請使用 AWS Windows AMI 伺服器 2019 年版啟動暫存執行個體。

3. 從受影響的執行個體將根磁碟區連接到此暫時執行個體。連線至暫時執行個體、開啟 Disk Management (磁碟管理) 公用程式，然後讓磁碟機上線。
4. 從臨時實例中，打開註冊表編輯器，然後選擇 HKEY_ LOCAL _ MACHINE。從 File (檔案) 功能表中，選擇 Load Hive (載入 Hive)。選取磁碟機、開啟檔案 `Windows\System32\config\SYSTEM`，然後在提示時指定金鑰名稱 (您可使用任何名稱)。
5. 選取剛載入的機碼並導覽至 `ControlSet001\Services\Tcpip\Parameters\Interfaces`。每個網路介面都以GUID. 選取正確的網路介面。如果DHCP停用且指派靜態 IP 位址，`EnableDHCP`則會設定為 0。若要啟用DHCP，請設`EnableDHCP`定為 1，並刪除下列金鑰 (如果存在)：`NameServerSubnetMaskIPAddress`、、和`DefaultGateway`。再次選取該金鑰，然後從 File (檔案) 功能表中選擇 Unload Hive (取消載入 Hive)。

> **Note**
>
> 如果您有多個網路介面，則需要識別要啟用的正確介面DHCP。若要找出正確的網路界面，請檢閱下列機碼值：`NameServer`、`SubnetMask`、`IPAddress` 和 `DefaultGateway`。這些值會顯示前一個執行個體的靜態組態。

6. (選擇性) 如果DHCP已啟用，則可能是您沒有前往中繼資料服務的路由。更新EC2Config可以解決此問題。
   a. 下載並安裝最新版本的EC2Config服務。如需安裝此服務的詳細資訊，請參閱 the section called “安裝 EC2Config”。
   b. 將 `.zip` 檔案中的檔案解壓縮到所連接之磁碟機上的 `Temp` 目錄中。

c. 打開註冊表編輯器，然後選擇 HKEY_ LOCAL _ MACHINE。從 File (檔案) 功能表中，選擇 Load Hive (載入 Hive)。選取磁碟機、開啟檔案 `Windows\System32\config\SOFTWARE`，然後在提示時指定金鑰名稱 (您可使用任何名稱)。

d. 選取剛載入的機碼並導覽至 `Microsoft\Windows\CurrentVersion`。選取 `RunOnce` 機碼。(如果此機碼不存在，請在 `CurrentVersion` 按一下滑鼠右鍵、指向 New (新增)、選取 Key (機碼) 並命名機碼 `RunOnce`。) 按一下滑鼠右鍵、指向 New (新增)，然後選取 String Value (字串值)。在名稱輸入 `Ec2Install`，在資料輸入 `C:\Temp\Ec2Install.exe -q`。

e. 再次選取該金鑰，然後從 File (檔案) 功能表中選擇 Unload Hive (取消載入 Hive)。

7. (選擇性) 如果您的暫存執行個體是以原始執行AMI個體所依據的相同為基礎，您必須完成下列步驟，否則您將無法在還原其根磁碟區後，因為磁碟簽章衝突而無法啟動原始執行個體。

> **Warning**
>
> 下列程序說明如何使用登錄編輯程式來編輯 Windows 登錄檔。如果您不熟悉 Windows 登錄檔或如何使用登錄編輯程式來安全地進行變更，請參閱 設定登錄檔。

a. 開啟命令提示，輸入 regedit.exe，然後按 Enter。

b. 在 [登錄編輯程式] 中，MACHINE從內容功能表中選擇 [HKEYLOCAL_ _] (按一下滑鼠右鍵)，然後選擇 [尋找]。

c. 輸入 Windows Boot Manager，然後選擇 Find Next (尋找下一個)。

d. 選擇名為 `11000001` 的金鑰。此金鑰是您在上一個步驟中找到的金鑰的同級。

e. 在右窗格中，選擇 `Element`，然後從內容功能表中選擇 Modify (修改)(按一下右鍵)。

f. 在資料中的位移為 0x38 處找到四位元組的磁碟簽章。反轉位元組以建立磁碟簽章，然後寫下來。例如，下列資料所代表的磁碟簽章是 `E9EB3AA5`：

```
...
0030  00 00 00 00 01 00 00 00
0038  A5 3A EB E9 00 00 00 00
0040  00 00 00 00 00 00 00 00
...
```

g. 在 [命令提示字元] 視窗中，執行下列命令以啟動 Microsoft DiskPart。

```
diskpart
```

h. 執行下列 DiskPart 命令以選取磁碟區。(可以使用磁碟管理公用程式，確認磁碟編號為 1)。

```
DISKPART> select disk 1

Disk 1 is now the selected disk.
```

i. 運行以下 DiskPart 命令以獲取磁盤簽名。

```
DISKPART>  uniqueid disk

Disk ID: 0C764FA8
```

j. 如果上一步中顯示的磁碟簽章與之前寫下BCD的磁碟簽章不符，請使用下列 DiskPart 命令來變更磁碟簽章，使其符合：

```
DISKPART> uniqueid disk id=E9EB3AA5
```

8. 使用 Disk Management (磁碟管理) 公用程式讓磁碟機離線。

> **Note**
>
> 如果暫時執行個體執行的作業系統與受影響執行個體的相同，磁碟機會自動離線，因此您不需要手動將其離線。

9. 分離磁碟區與暫時執行個體。如果不再需要使用此暫時執行個體，您可終止它。
10. 將磁碟區連接為 `/dev/sda1` 來還原受影響執行個體的根磁碟區。
11. 啟動受影響的執行個體。

如果您已連線至執行個體，請從執行個體開啟網際網路瀏覽器，並URL為中繼資料伺服器輸入下列資料：

```
http://169.254.169.254/latest/meta-data/
```

如果無法聯絡中繼資料伺服器，請嘗試下列方法解決問題：

- 下載並安裝最新版本的EC2Config服務。如需安裝此服務的詳細資訊，請參閱 the section called “安裝 EC2Config”。
- 檢查 Windows 執行個體是否正在執行 RedHat 光伏驅動程式。如果是，請更新至 Citrix PV 驅動程式。如需詳細資訊，請參閱the section called “升級 PV 驅動程式”。

- 確認防火牆IPSec、和 Proxy 設定不會封鎖傳送至中繼資料服務 (`169.254.169.254`) 或 AWS KMS 伺服器的傳出流量 (位址在中的`TargetKMSServer`元素中指定`C:\Program Files\Amazon\Ec2ConfigService\Settings\ActivationSettings.xml`)。
- 使用下列命令確認您有中繼資料服務的路由 (`169.254.169.254`)。

```
route print
```

- 檢查是否有可能影響您執行個體可用區域的網路問題。請前往 http://status.aws.amazon.com/。

無法啟動 Windows

視窗執行個體使用視窗 AWS KMS 啟用。您可以收到以下訊息：`A problem occurred when Windows tried to activate. Error Code 0xC004F074`，如果您的執行個體無法連線到 AWS KMS 伺服器。Windows 必須每 180 天啟動一次。EC2Config嘗試在啟用期限到期之前聯絡 AWS KMS 伺服器，以確保 Windows 保持啟動狀態。

如果您發生 Windows 啟用問題，請使用下列程序予以解決。

對於 EC2Config (視窗伺服器 2012 R2 AMIs 及更早版本)

1. 下載並安裝最新版本的EC2Config服務。如需安裝此服務的詳細資訊，請參閱 the section called “安裝 EC2Config”。
2. 登入執行個體並開啟下列檔案：`C:\Program Files\Amazon\Ec2ConfigService\Settings\config.xml`。
3. 在檔案中找到 EC2 WindowsActivate 外掛程`config.xml`式。將狀態變更為 Enabled (已啟用) 並儲存變更。
4. 在 Windows 服務嵌入式管理單元中，重新啟動EC2Config服務或重新啟動執行個體。

如果這無法解決啟用問題，請遵循下列其他步驟。

1. 設定 AWS KMS 目標：C:\> slmgr.vbs /skms 169.254.169.250:1688
2. 啟動 Windows：C:\> slmgr.vbs /ato

適用於 EC2Launch (視窗伺服器 2016 AMIs 及更新版本)

1. 從具有管理權限的 PowerShell 提示中，匯入EC2Launch模組：

```
PS C:\> Import-Module "C:\ProgramData\Amazon\EC2-Windows\Launch\Module
\Ec2Launch.psd1"
```

2. 呼叫新增路由函數以查看新路由清單：

```
PS C:\> Add-Routes
```

3. 調用 Set-ActivationSettings 函數：

```
PS C:\> Set-Activationsettings
```

4. 接著執行下列指令碼以啟動 Windows：

```
PS C:\> cscript "${env:SYSTEMROOT}\system32\slmgr.vbs" /ato
```

對於EC2Config和EC2Launch，如果您仍然收到激活錯誤，請驗證以下信息。

- 確認您具有通往 AWS KMS 伺服器的路由。開啟 `C:\Program Files\Amazon\Ec2ConfigService\Settings\ActivationSettings.xml` 並找到 `TargetKMSServer` 元素。執行下列命令，並檢查是否列出這些 AWS KMS 伺服器的位址。

```
route print
```

- 確認已設定 AWS KMS 用戶端金鑰。執行下列命令並檢查輸出。

```
C:\Windows\System32\slmgr.vbs /dlv
```

  如果輸出包含「錯誤：找不到產品金鑰」，表示未設定 AWS KMS 用戶端金鑰。如果未設定 AWS KMS 用戶端金鑰，請依照以下 Microsoft 文章：用戶端安裝金鑰所述查詢AWS KMS 用戶端金鑰，然後執行下列命令來設定 AWS KMS 用戶端金鑰。

```
C:\Windows\System32\slmgr.vbs /ipk client_key
```

- 確認系統的時間和時區是否正確。如果您使用的時區不是UTC，請新增下列登錄機碼並將其設定，1以確保時間正確無誤：`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\TimeZoneInformation\RealTimeIsUniversal`
- 如已啟用 Windows 防火牆，請使用下列命令暫時停用它。

```
netsh advfirewall set allprofiles state off
```

Windows 不是正版 (0x80070005)

視窗執行個體使用視窗 AWS KMS 啟用。如果執行個體無法完成啟用程序，會回報所用 Windows 不是正版。

請嘗試無法啟動 Windows的建議。

無終端機伺服器的授權伺服器可提供授權

Windows Server 預設透過遠端桌面可同時授權兩名使用者使用。如果您需要讓兩個以上的使用者透過遠端桌面同時存取您的 Windows 執行個體，您可以購買遠端桌面服務用戶端存取授權 (CAL)，並安裝遠端桌面工作階段主機和遠端桌面授權伺服器角色。

檢查是否有下列問題：

- 您已超過同時RDP工作階段的數目上限。
- 您已安裝 Windows 遠端桌面服務角色。
- 授權已過期。如果授權已過期，您就無法以使用者身分連線到您的 Windows 執行個體。您可以嘗試下列方法：
  - 使用 `/admin` 參數從命令列連線至執行個體，例如：

    ```
    mstsc /v:instance /admin
    ```

    如需詳細資訊，請參閱下列 Microsoft 文章：Access Remote Desktop Via Command Line。
  - 停止執行個體、分離其 Amazon EBS 磁碟區，然後將它們連接到相同可用區域中的另一個執行個體以復原資料。

「部分設定是由您的組織所管理」

從最新 Windows 伺服器啟動的執行個體AMIs可能會顯示 Windows Update 對話方塊訊息，指出「某些設定是由您的組織管理」。此訊息是因為 Windows Server 中的變更而出現，不影響 Windows Update 的行為或造成您無法管理更新設定。

如何移除警告

1. 開啟 `gpedit.msc` 並導覽至電腦組態 > 系統管理範本 > Windows 元件 > Windows 更新。編輯設定自動更新，然後將其設為已啟用。
2. 在命令提示字元中，使用 gpupdate /force 更新群組政策。
3. 關閉「Windows Update 設定」再重新開啟。您將看到上述訊息表示您的設定正由您的組織所管理，後面接著「我們將自動下載更新，付費連線的情況除外 (因為可能需要支付費用)。在該種情況下，我們會自動下載保持 Windows 順暢運作所需的這些更新。」
4. 返回 `gpedit.msc`，然後將群組政策設回尚未設定。再次執行 gpupdate /force。
5. 關閉命令提示字元並等待幾分鐘。
6. 重新開啟「Windows Update 設定」。您應該不會看到「部分設定是由您的組織所管理」這個訊息。

解決 Amazon EC2 視窗執行個體問題

以下是可協助您解決 Amazon EC2 Windows 執行個體相關問題的疑難排解秘訣。

問題

- EBS磁碟區不會在視窗伺服器 2016 和 2019 上初始化
- 將 EC2 Windows 執行個體啟動至目錄服務還原模式 (DSRM)
- 執行個體遺失網路連線能力或排程任務未如預期執行
- 無法取得主控台輸出
- 網路不提供 Windows Server 2012 R2
- 磁碟簽章衝突

EBS磁碟區不會在視窗伺服器 2016 和 2019 上初始化

從 Amazon 機器映像 (AMIs) 建立的執行個體，適用於 Windows 伺服器 2016 年和 2019 年，會使用 EC2Launch v1 代理程式執行各種啟動任務，包括初始化EBS磁碟區。依預設，EC2Launchv1 不會初始化輔助磁碟區。不過，您可以設定 EC2Launch v1 來自動初始化這些磁碟，如下所示。

將磁碟機代號映射到磁碟區

1. 連線至執行個體，以設定 `C:\ProgramData\Amazon\EC2-Windows\Launch\Config\DriveLetterMappingConfig.json` 檔案並在文字編輯器中開啟。

2. 指定磁碟區設定如下：

```
{
"driveLetterMapping": [
 {
   "volumeName": "sample volume",
   "driveLetter": "H"
 }]
}
```

3. 儲存您的變更並結束檔案。
4. 開啟 Windows PowerShell 並使用下列命令來執行初始化磁碟的 EC2Launch v1 指令碼：

```
PS C:\>  C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeDisks.ps1
```

若要在每次執行個體開機時初始化磁碟，請新增 `-Schedule` 標記，如下所示：

```
PS C:\>  C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeDisks.ps1 -
Schedule
```

EC2Launchv1 代理程式可以執行執行個體初始化指令碼，例如與指`InitializeInstance.ps1`令碼 parallel。`initializeDisks.ps1`如果 `InitializeInstance.ps1` 指令碼會重新啟動執行個體，它可能會中斷執行個體啟動時執行的其他排程任務。為了避免任何潛在的衝突，我們建議您將邏輯新增至 `initializeDisks.ps1` 指令碼，以確保執行個體初始化已先完成。

> **Note**
>
> 如果指EC2Launch令碼未初始化磁碟區，請確定磁碟區已連線。如果磁碟區離線，請執行下列命令以使所有磁碟上線。
>
> ```
> PS C:\> Get-Disk | Where-Object IsOffline -Eq $True | Set-Disk -IsOffline
> $False
> ```

將 EC2 Windows 執行個體啟動至目錄服務還原模式 (DSRM)

如果執行 Microsoft Active Directory 的執行個體遇到系統故障或其他重大問題，您可以開機進入特殊版本的安全模式 (稱為目錄服務還原模式 (DSRM)，藉此疑難排解執行個體。在DSRM您可以修復或恢復活動目錄。

驅動程式支援 DSRM

啟用DSRM和開機進入執行個體的方式取決於執行個體執行的驅動程式。在EC2主控台中，您可以從系統記錄檢視執行個體的驅動程式版本詳細資料。下表顯示支援的驅動程式DSRM。

驅動程式版本	DSRM支援？	後續步驟
Citrix PV 5.9	否	從備份還原執行個體。您無法啟用DSRM。
AWS 太陽能光伏	否	雖DSRM然此驅動程式不支援，但您仍然可以從執行個體中分離根磁碟區、擷取磁碟區的快照或建立AMI磁碟區，然後將其作為輔助磁碟區附加至相同可用區域中的另一個執行個體。然後您可以啟用 DSRM (如本節所述)。
AWS PV 7.2.2 及更高版本	是	分離根磁碟區，將其連接到另一個執行個體，然後啟用 DSRM (如本節所述)。
增強型聯網	是	分離根磁碟區，將其連接到另一個執行個體，然後啟用 DSRM (如本節所述)。

如需如何啟用增強型網路功能的相關資訊，請參閱the section called “彈性網路介面卡 (ENA)”。如需升級 AWS PV 驅動程式的相關資訊，請參閱升級 Windows 執行個體上的 PV 驅

設定要開機進入的執行個體 DSRM

EC2Windows 執行個體在作業系統執行之前沒有網路連線。因此，您無法按鍵盤的 F8 按鈕選取開機選項。您必須使用下列其中一個程序，將 EC2 Windows 伺服器執行個體啟動到中DSRM。

如果您懷疑 Active Directory 已損毀，且執行個體仍在執行中，您可以DSRM使用 [系統設定] 對話方塊或命令提示字元，將執行個體設定為開機。

使用系統組態對話方塊將DSRM線上執行個體啟動至

1. 在 Run (執行) 對話方塊中輸入 `msconfig`，然後按 Enter。

2. 選擇 Boot (開機) 標籤。
3. 在 Boot options (開機選項) 下選擇 Safe boot (安全開機)。
4. 選擇 Active Directory repair (Active Directory 修復)，然後選擇 OK (確定)。系統會提示您重新開機伺服器。

DSRM使用指令列將線上執行個體啟動至

從命令提示視窗執行下列命令：

```
bcdedit /set safeboot dsrepair
```

如果執行個體離線且無法連線，您必須卸離根磁碟區並將其連接至另一個執行個體以啟用DSRM模式。

若要將離線執行個體啟動至 DSRM

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 找出並選取受影響的執行個體。選擇 Instance state (執行個體狀態)、Stop instance (停止執行個體)。
4. 選擇 Launch instances (啟動執行個體)，並在與受影響執行個體相同的可用區域中，建立暫時性的執行個體。選擇使用不同 Windows 版本的執行個體類型。例如，如果您的執行個體是視窗伺服器 2016 年，請選擇視窗伺服器 2019 執行個體。

   > **Important**
   > 若您沒有在與受影響的執行個體相同的可用區域內建立執行個體，您將無法將受影響執行個體的根磁碟區連接到新的執行個體。

5. 在導覽窗格中，選擇 Volumes (磁碟區)。
6. 尋找受影響執行個體的根磁碟區。分離磁碟區，再將其連接至先前建立的暫時性執行個體。使用預設裝置名稱 (xvdf) 連接它。
7. 使用遠端桌面連接至暫時性執行個體，然後使用 Disk Management 公用程式將磁碟區變成可用。
8. 開啟命令提示並執行下列命令。使用您剛連接之輔助磁碟區的實際磁碟機代號取代 D：

   ```
   bcdedit /store D:\Boot\BCD /set {default} safeboot dsrepair
   ```

9. 在磁碟管理公用程式中，選擇先前連接的磁碟機，開啟內容 (按右鍵) 選單，然後選擇 Offline (離線)。
10. 在EC2主控台中，將受影響的磁碟區與暫存執行個體分離，然後使用裝置名稱`/dev/sda1`將其重新連接到原始執行個體。您必須指定此裝置名稱，將磁碟區指定為根磁碟區。
11. 啟動實例。
12. 執行個體通過EC2主控台中的健康狀態檢查之後，請使用遠端桌面連線至執行個體，並確認執行個體已開機進入DSRM模式。
13. (選用) 刪除或停止您在此程序中建立的暫時性執行個體。

執行個體遺失網路連線能力或排程任務未如預期執行

如果您重新啟動您的執行個體但它遺失網路連線能力，可能是執行個體的時間發生錯誤。

依預設，Windows 執行個體使用協調世界時間 (UTC)。如果您將執行個體的時間設至不同的時區然後重新啟動，就會發生時間位移，且執行個體會暫時遺失 IP 地址。執行個體最後會在重新取得網路連線能力，但這可能需要數小時。執行個體重新獲得網路連線所需的時間，取決於UTC與其他時區之間的差異。

此相同的時間問題也會導致排程任務未如預期執行。在本例中，因為執行個體的時間不正確，所以排程的任務未如預期執行。

若要使用UTC永久性以外的時區，您必須設定RealTimeIsUniversal登錄機碼。如果沒有這個金鑰，執行個體會UTC在您重新啟動後使用。

解決造成網路連線能力遺失的時間問題

1. 請務必執行建議的 PV 驅動程式。如需詳細資訊，請參閱 the section called “升級 PV 驅動程式”。
2. 確認下列登錄機碼存在且已設定為1：HKEY_ LOCAL _MACHINE\\ SYSTEMCurrentControlSet\控制\\TimeZoneInformation\ RealTimeIsUniversal

無法取得主控台輸出

若為 Windows 執行個體，執行個體主控台會顯示在 Windows 開機程序期間執行的任務輸出。如果 Windows 成功開機，最後記錄的訊息為 `Windows is Ready to use`。您也可以在主控台中顯示事件記錄訊息，但根據您的 Windows 版本，預設情況下可能不會啟用此功能。如需詳細資訊，請參閱 the section called “啟動代理程式”。

若要使用 Amazon 主控台取得執行個體的EC2主控台輸出，請選取執行個體，然後選擇 [動作]、[監控和疑難排解]、[取得系統記錄]。若要使用命令列取得主控台輸出，請使用下列命令之一：get-console-output(AWS CLI) 或 Get-EC2ConsoleOutput(AWS Tools for Windows PowerShell)。

對於執行 Windows Server 2012 R2 及更早版本的執行個體，如果主控台輸出為空白，則可能表示EC2Config服務發生問題，例如設定錯誤的設定檔，或 Windows 無法正常開機。若要修正此問題，請下載並安裝最新版本的EC2Config. 如需詳細資訊，請參閱 the section called “安裝 EC2Config”。

網路不提供 Windows Server 2012 R2

如需疑難排解無法在網路上使用的 Windows 伺服器 2012 R2 執行個體的相關資訊，請參閱 Windows 伺服器 2012 R2 在執行個體重新開機後中斷網路和儲存區連線。

磁碟簽章衝突

您可以使EC2Rescue用 Windows 伺服器檢查並解決磁碟簽章衝突。或者，您可以執行下列步驟，手動解決磁碟簽章問題。

> **Warning**
>
> 下列程序說明如何使用登錄編輯程式來編輯 Windows 登錄檔。如果您不熟悉 Windows 登錄檔或如何使用登錄編輯程式來安全地進行變更，請參閱 設定登錄檔。

1. 開啟命令提示，輸入 regedit.exe，然後按 Enter。
2. 在 [登錄編輯程式] 中，MACHINE從內容功能表中選擇 [HKEYLOCAL_ _] (按一下滑鼠右鍵)，然後選擇 [尋找]。
3. 輸入 Windows Boot Manager，然後選擇 Find Next (尋找下一個)。
4. 選擇名為 `11000001` 的金鑰。此金鑰是您在上一個步驟中找到的金鑰的同級。
5. 在右窗格中，選擇 `Element`，然後從內容功能表中選擇 Modify (修改)(按一下右鍵)。
6. 在資料中的位移為 0x38 處找到四位元組的磁碟簽章。這是「開機設定資料庫」簽章 (BCD)。反轉位元組以建立磁碟簽章，然後寫下來。例如，下列資料所代表的磁碟簽章是 `E9EB3AA5`：

```
...
0030  00 00 00 00 01 00 00 00
0038  A5 3A EB E9 00 00 00 00
0040  00 00 00 00 00 00 00 00
```

```
...
```

7. 在 [命令提示字元] 視窗中，執行下列命令以啟動 Microsoft DiskPart。

```
diskpart
```

8. 執行指`select disk` DiskPart 令並指定發生磁碟簽章衝突的磁碟區的磁碟編號。

> **Tip**
>
> 若要查看發生磁碟簽章衝突的磁碟區的磁碟編號，請使用磁碟管理公用程式。開啟命令提示，輸入 `compmgmt.msc`，然後按 Enter。在左側導覽面板中，按兩下磁碟管理。在磁碟管理公用程式中，查看發生磁碟簽章衝突的離線磁碟區的磁碟編號。

```
DISKPART> select disk 1
Disk 1 is now the selected disk.
```

9. 運行以下 DiskPart 命令以獲取磁盤簽名。

```
DISKPART>  uniqueid disk
Disk ID: 0C764FA8
```

10. 如果上一步中顯示的磁碟簽章與您先前寫下的磁碟簽章不符，請使用下列 DiskPart 命令來變更磁碟簽章，使其符合：

```
DISKPART> uniqueid disk id=E9EB3AA5
```

重設 Amazon EC2 視窗執行個體的管理員密碼

如果您因為 Windows 管理員密碼遺失或過期而無法連線到 Amazon EC2 Windows 執行個體，您可以重設密碼。

> **Note**
>
> 自動 AWS Systems Manager 化文件會自動套用重設本機管理員密碼所需的手動步驟。如需詳細資訊，請參閱《AWS Systems Manager 使用指南》中的「重設EC2執行個體的密碼和SSH金鑰」。

重設管理員密碼的手動方法使用 EC2Launch v2 EC2Config、、或EC2Launch。

- 對於包含 EC2Launch v2 代理程式的所有受支援視窗AMIs，請使用 EC2Launch v2。
- 對於視窗伺服器 2016 年AMIs之前的視窗，請使用該EC2Config服務。
- 對於視窗伺服器 2016 及更新版本AMIs，請使用此EC2Launch服務。

這些程序還說明在遺失用來建立執行個體的金鑰對時如何連線至執行個體。Amazon EC2 使用公開金鑰加密一段資料 (例如密碼) 和私密金鑰來解密資料。公有金鑰和私有金鑰稱為金鑰對。在 Windows 執行個體中，您可以使用 key pair 取得系統管理員密碼，然後使用登入RDP。

> **Note**
>
> 如果您已停用執行個體上的本機管理員帳戶，且您的執行個體已設定為 Systems Manager，您也可以使用和執行命令來重新啟用EC2Rescue和重設本機管理員密碼。如需詳細資訊，請參閱搭EC2Rescue配 Systems Manager 執行命令的 Windows 伺服器使用。

目錄

- 使用 EC2Launch v2 重置EC2實例的 Windows 管理員密碼
- 重置 Windows 管理員密碼，EC2例如使用 EC2Launch
- 重置 Windows 管理員密碼，EC2例如使用 EC2Config

使用 EC2Launch v2 重置EC2實例的 Windows 管理員密碼

如果您遺失 Windows 系統管理員密碼，而且正在使用包AMI含 EC2Launch v2 代理程式的受支援 Windows，您可以使用 EC2Launch v2 來產生新密碼。

如果您使用的是不包含 EC2Launch v2 代理程式AMI的 Windows 伺服器 2016 或更新版本，請參閱重置 Windows 管理員密碼，EC2例如使用 EC2Launch。

如果您使用的是不包含 EC2Launch v2 代理程式的 Windows 伺服器 2016 年AMI之前的版本，請參閱重置 Windows 管理員密碼，EC2例如使用 EC2Config。

> **Note**
>
> 如果您已停用執行個體上的本機管理員帳戶，且您的執行個體已設定為 Systems Manager，您也可以使用和執行命令來重新啟用EC2Rescue和重設本機管理員密碼。如需詳細資訊，請參閱搭EC2Rescue配 Systems Manager 執行命令的 Windows 伺服器使用。

> **Note**
>
> 自動 AWS Systems Manager 化文件會自動套用重設本機管理員密碼所需的手動步驟。如需詳細資訊，請參閱《AWS Systems Manager 使用指南》中的「重設EC2執行個體的密碼和SSH金鑰」。

要使用 EC2Launch v2 重置 Windows 管理員密碼，您需要執行以下操作：

- 步驟 1：確認 EC2Launch v2 代理程式正在執行
- 步驟 2：將根磁碟區從執行個體中分離
- 步驟 3：將磁碟區連接至暫時執行個體
- 步驟 4：刪除 .run-once 檔案
- 步驟 5：重新啟動原始執行個體

步驟 1：確認 EC2Launch v2 代理程式正在執行

嘗試重設系統管理員密碼之前，請確認 EC2Launch v2 代理程式已安裝並執行。您可以使用 EC2Launch v2 代理程式來重設本節稍後的系統管理員密碼。

驗證 EC2Launch v2 代理程式是否正在執行

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中選擇「Instances (執行個體)」，然後選擇需要重設密碼的執行個體。此執行個體即為本程序中的原始執行個體。
3. 選擇 Actions (動作)、Monitor and troubleshoot (監視和故障診斷)、Get system log (取得系統日誌)。
4. 找出EC2啟動項目，例如，啟動：EC2Launchv2 服務 v2.0. 124。如果您看到此項目，表示 EC2Launch v2 服務正在執行中。

如果系統記錄檔輸出為空白，或 EC2Launch v2 代理程式未執行，請使用執行個體主控台螢幕擷取畫面服務對執行個體進行疑難排解。如需詳細資訊，請參閱擷取無法連線執行個體的螢幕擷取畫面。

步驟 2：將根磁碟區從執行個體中分離

如果儲存密碼的磁碟區以根磁碟區的形式連接至執行個體，則無法使用 EC2Launch v2 重設管理員密碼。您必須將磁碟區從原始執行個體中分離，才能將磁碟區連接至暫時執行個體做為次要磁碟區。

若要將根磁碟區從執行個體中分離

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取需要重設密碼的執行個體，然後選擇執行個體狀態、停止執行個體。執行個體的狀態變更為 Stopped (已停止) 之後，請繼續下一個步驟。
4. (選用) 如果您擁有在啟動此執行個體時已指定的私有金鑰，請繼續下一個步驟。否則，請使用下列步驟將執行個體取代為使用新金鑰對啟動的新執行個體。
   a. 使用 Amazon EC2 主控台建立新的 key pair。若要使新金鑰對的名稱與已遺失私有金鑰的金鑰對的名稱相同，您必須先刪除現有金鑰對。
   b. 選取要取代的執行個體。請記下執行個體類型VPC、子網路、安全群組和執行個體的IAM角色。
   c. 選取執行個體後，選擇 [動作]、[映像和範本]、[建立映像]。輸入映像的名稱和說明，然後選擇 Create image (建立映像)。
   d. 在導覽窗格中，選擇AMIs。等待圖像狀態變更為可用。然後，選擇映像並從中選擇啟動實例 AMI。
   e. 填寫欄位以啟動執行個體，確定選取與要取代的執行個體相同的執行個體類型VPC、子網路、安全性群組和IAM角色，然後選擇 Launch 執行個體。
   f. 出現提示時，請選擇您為新執行個體建立的 key pair，然後選擇 Launch 執行個體。
   g. (選用) 如果原始執行個體有關聯的 Elastic IP 地址，將其傳輸至新執行個體。如果原始執行個體除了根EBS磁碟區之外還有磁碟區，請將它們傳輸到新的執行個體。
5. 將根磁碟區從原始執行個體中分離，如下所示：
   a. 選取原始執行個體並選擇儲存空間索引標籤。記下根裝置名稱下的根裝置名稱。在 [封鎖裝置] 底下找到具有此裝置名稱的磁碟區，並記下磁碟區 ID。

b. 在導覽窗格中，選擇 Volumes (磁碟區)。

c. 在磁碟區清單中，選取您記下為根裝置的磁碟區，然後選擇 [動作] > [卸離磁碟區]。在將磁碟區狀態變更為 available (可用) 後，繼續下一步驟。

6. 如果您建立了新執行個體來取代原始執行個體，您可以立即終止原始執行個體。它不再需要。對於此程序的其餘部分，原始例證的所有參照都會套用至您建立的新例證。

步驟 3：將磁碟區連接至暫時執行個體

接下來，啟動暫時執行個體，並將磁碟區連接至此執行個體做為次要磁碟區。這是您用來修改組態檔案的執行個體。

啟動暫時執行個體並連接磁碟區

1. 啟動暫時執行個體，如下所示：

   a. 在瀏覽窗格中，選擇 [執行個體]，選擇 [啟動執行個體]，然後選取AMI。

   > **Important**
   >
   > 若要避免磁碟簽章衝突，您必須AMI為不同版本的 Windows 選取。例如，如果原始執行個體執行 Windows 伺服器 2019，請使用視窗伺服器 2016 年的基礎AMI啟動暫存執行個體。

   b. 保留預設執行個體類型，然後選擇 Next: Configure Instance Details (下一步：設定執行個體的詳細資訊)。

   c. 在 Configure Instance Details (設定執行個體詳細資訊) 頁面中，對於 Subnet (子網路)，選取與原始執行個體相同的可用區域，然後選擇 Review and Launch (檢閱和啟動)。

   > **Important**
   >
   > 暫時執行個體與原始執行個體必須位於相同可用區域。如果您的暫時執行個體位於不同的可用區域，則無法將原始執行個體的根磁碟區與它連接。

   d. 在 Review Instance Launch (檢閱執行個體啟動) 頁面，選擇 Launch (啟動)。

   e. 出現提示時，建立新金鑰對，將其下載到電腦上的安全位置，然後選擇 Launch Instances (啟動執行個體)。

2. 將磁碟區連接到暫時執行個體作為輔助磁碟區，如下所示：

a. 在導覽窗格中，選擇 Volumes (磁碟區)，選取從原始執行個體分離的磁碟區，然後選擇 Actions (動作)、Attach Volume (連接磁碟區)。
b. 在 Attach Volume (連接磁碟區) 對話方塊中，對於 Instances (執行個體)，開始輸入暫時執行個體的名稱或 ID，然後從清單中選取執行個體。
c. 對於 Device (裝置)，輸入 **xvdf** (如果尚未出現在此)，選擇 Attach (連接)。

步驟 4：刪除 .run-once 檔案

現在必須從附接至執行個體的離線磁碟區刪除 `.run-once` 檔案。這會指示 EC2Launch v2 以的頻率執行所有工作once，其中包括設定管理員密碼。您連接之輔助磁碟區中的檔案路徑會類似 `D:\ProgramData\Amazon\EC2Launch\state\.run-once`。

若要刪除 .run-once 檔案

1. 開啟磁碟管理公用程式，然後使用以下說明將磁碟機連線：讓 Amazon 磁EBS碟區可供使用。
2. 在已連線的磁碟中找到 `.run-once` 檔案。
3. 刪除 .run-once 檔案。

> **Important**
> 設定為執行一次的任何指令碼都會由此動作觸發。

步驟 5：重新啟動原始執行個體

刪除 `.run-once` 檔案之後，將磁碟區重新連接至原始執行個體做為根磁碟區，然後使用金鑰對來連接至執行個體，以擷取管理員密碼。

1. 將磁碟區重新連接到原始執行個體，如下所示：
   a. 在導覽窗格中，選擇 Volumes (磁碟區)，選取從暫時執行個體分離的磁碟區，然後選擇 Actions (動作)、Attach Volume (連接磁碟區)。
   b. 在 Attach Volume (連接磁碟區) 對話方塊中，對於 Instances (執行個體)，開始輸入您的原始執行個體的名稱或 ID，然後選取執行個體。
   c. 對於 Device (裝置)，輸入 **/dev/sda1**。
   d. 選擇 Attach (連接)。在磁碟區狀態變更為 `in-use` 之後，繼續下一個步驟。

2. 在導覽窗格中，選擇 Instances (執行個體)。選取原始執行個體，然後依序選擇 Instance state (執行個體狀態)、Start instance (啟動執行個體)。在執行個體狀態變更為 Running 之後，繼續下一個步驟。
3. 使用新金鑰對的私有金鑰擷取新的 Windows 系統管理員密碼，然後連線到執行個體。如需詳細資訊，請參閱 使用 Connect 至您的 Windows 執行個體 RDP。

> **Important**
>
> 先停止執行個體，接著再啟動後，該執行個體將得到新的公有 IP 地址。請務必使用執行個體目前的公用名DNS稱連線至執行個體。如需詳細資訊，請參閱Amazon EC2 實例狀態更改。

4. (選用) 如果不再使用暫時執行個體，您可以終止它。選取暫時執行個體，再依序選擇 Instance State (執行個體狀態)、Terminate instance (終止執行個體)。

重置 Windows 管理員密碼，EC2例如使用 EC2Launch

如果您丟失了 Windows 管理員密碼，並且正在使用 Windows 服務器 2016 年或更高版本AMI，則可以使用該EC2Rescue工具，該工具使用該EC2Launch服務來生成新密碼。

如果您使用的是不包含 EC2Launch v2 代理程式AMI的 Windows 伺服器 2016 年或更新版本，您可以使用 EC2Launch v2 來產生新密碼。

如果您使用的視窗伺服器AMI早於視窗伺服器 2016 年，請參閱重置 Windows 管理員密碼，EC2例如使用 EC2Config。

> **Warning**
>
> 停止執行個體時，在任何執行個體存放區磁碟區的資料都會清除。若要保留執行個體存放區磁碟區的資料，請將資料備份至持久性儲存。

> **Note**
>
> 如果您已停用執行個體上的本機管理員帳戶，且您的執行個體已設定為 Systems Manager，您也可以使用和執行命令來重新啟用EC2Rescue和重設本機管理員密碼。如需詳細資訊，請參閱搭EC2Rescue配 Systems Manager 執行命令的 Windows 伺服器使用。

> **Note**
>
> 自動 AWS Systems Manager 化文件會自動套用重設本機管理員密碼所需的手動步驟。如需詳細資訊，請參閱《AWS Systems Manager 使用指南》中的「重設EC2執行個體的密碼和SSH金鑰」。

要使用重置 Windows 管理員密碼EC2Launch，您需要執行以下操作：

- 步驟 1：將根磁碟區從執行個體中分離
- 步驟 2：將磁碟區連接至暫時執行個體
- 步驟 3：重設管理員密碼
- 步驟 4：重新啟動原始執行個體

步驟 1：將根磁碟區從執行個體中分離

如果儲存密碼EC2Launch的磁碟區以根磁碟區的形式連接至執行個體，您就無法使用重設管理員密碼。您必須將磁碟區從原始執行個體中分離，才能將磁碟區連接至暫時執行個體做為次要磁碟區。

若要將根磁碟區從執行個體中分離

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取需要重設密碼的執行個體，然後選擇執行個體狀態、停止執行個體。執行個體的狀態變更為 Stopped (已停止) 之後，請繼續下一個步驟。
4. (選用) 如果您擁有在啟動此執行個體時已指定的私有金鑰，請繼續下一個步驟。否則，請使用下列步驟將執行個體取代為使用新金鑰對啟動的新執行個體。

    a. 使用 Amazon EC2 主控台建立新的 key pair。若要使新金鑰對的名稱與已遺失私有金鑰的金鑰對的名稱相同，您必須先刪除現有金鑰對。

    b. 選取要取代的執行個體。請記下執行個體類型VPC、子網路、安全群組和執行個體的IAM角色。

    c. 選取執行個體後，選擇 [動作]、[映像和範本]、[建立映像]。輸入映像的名稱和說明，然後選擇 Create image (建立映像)。

    d. 在導覽窗格中，選擇AMIs。等待圖像狀態變更為可用。然後，選擇映像並從中選擇啟動實例 AMI。

e. 填寫欄位以啟動執行個體，確定選取與要取代的執行個體相同的執行個體類型VPC、子網路、安全性群組和IAM角色，然後選擇 Launch 執行個體。

f. 出現提示時，請選擇您為新執行個體建立的 key pair，然後選擇 Launch 執行個體。

g. (選用) 如果原始執行個體有關聯的 Elastic IP 地址，將其傳輸至新執行個體。如果原始執行個體除了根EBS磁碟區之外還有磁碟區，請將它們傳輸到新的執行個體。

5. 將根磁碟區從原始執行個體中分離，如下所示：

   a. 選取原始執行個體並選擇儲存空間索引標籤。記下根裝置名稱下的根裝置名稱。在 [封鎖裝置] 底下找到具有此裝置名稱的磁碟區，並記下磁碟區 ID。

   b. 在導覽窗格中，選擇 Volumes (磁碟區)。

   c. 在磁碟區清單中，選取您記下為根裝置的磁碟區，然後選擇 [動作] > [卸離磁碟區]。在將磁碟區狀態變更為 available (可用) 後，繼續下一步驟。

6. 如果您建立了新執行個體來取代原始執行個體，您可以立即終止原始執行個體。它不再需要。對於此程序的其餘部分，原始例證的所有參照都會套用至您建立的新例證。

步驟 2：將磁碟區連接至暫時執行個體

接下來，啟動暫時執行個體，並將磁碟區連接至此執行個體做為次要磁碟區。這是您用來執行的執行個體EC2Launch。

啟動暫時執行個體並連接磁碟區

1. 啟動暫時執行個體，如下所示：

   a. 在瀏覽窗格中，選擇 [執行個體]，選擇 [啟動執行個體]，然後選取AMI。

   > **Important**
   >
   > 若要避免磁碟簽章衝突，您必須AMI為不同版本的 Windows 選取。例如，如果原始執行個體執行 Windows 伺服器 2019，請使用視窗伺服器 2016 年的基礎AMI啟動暫存執行個體。

   b. 保留預設執行個體類型，然後選擇 Next: Configure Instance Details (下一步：設定執行個體的詳細資訊)。

   c. 在 Configure Instance Details (設定執行個體詳細資訊) 頁面中，對於 Subnet (子網路)，選取與原始執行個體相同的可用區域，然後選擇 Review and Launch (檢閱和啟動)。

> **Important**
>
> 暫時執行個體與原始執行個體必須位於相同可用區域。如果您的暫時執行個體位於不同的可用區域，則無法將原始執行個體的根磁碟區與它連接。

d. 在 Review Instance Launch (檢閱執行個體啟動) 頁面，選擇 Launch (啟動)。

e. 出現提示時，建立新金鑰對，將其下載到電腦上的安全位置，然後選擇 Launch Instances (啟動執行個體)。

2. 將磁碟區連接到暫時執行個體作為輔助磁碟區，如下所示：

   a. 在導覽窗格中，選擇 Volumes (磁碟區)，選取從原始執行個體分離的磁碟區，然後選擇 Actions (動作)、Attach Volume (連接磁碟區)。

   b. 在 Attach Volume (連接磁碟區) 對話方塊中，對於 Instances (執行個體)，開始輸入暫時執行個體的名稱或 ID，然後從清單中選取執行個體。

   c. 對於 Device (裝置)，輸入 **xvdf** (如果尚未出現在此)，選擇 Attach (連接)。

步驟 3：重設管理員密碼

接下來，連接到臨時實例並使EC2Launch用重置管理員密碼。

重設管理員密碼

1. Connect 至暫存執行個體，並使EC2Rescue用執行個體上的 Windows Server 工具來重設管理員密碼，如下所示：

   a. 下載EC2Rescue適用於視窗伺服器的 zip 檔案、解壓縮內容，然後執行 EC2Rescue.exe。

   b. 在 License Agreement (授權合約) 畫面上閱讀授權合約，如果接受條款，請選擇 I Agree (我同意)。

   c. 在 [歡迎使EC2Rescue用 Windows 伺服器] 畫面上，選擇 [下一步]。

   d. 在 Select mode (選取模式) 畫面上，選擇 Offline instance (離線執行個體)。

   e. 在 Select a disk (選取磁碟) 畫面上選取 xvdf 設備，然後選擇 Next (下一步)。

   f. 確認選取磁碟，然後選擇 Yes (是)。

   g. 載入磁碟區之後，選擇 OK (確定)。

   h. 在 Select Offline Instance Option (選取離線執行個體選項) 畫面上，選擇 Diagnose and Rescue (診斷與救援)。

i. 在 Summary (摘要) 畫面上檢閱資訊，然後選擇 Next (下一步)。

j. 在 Detected possible issues (偵測到的可能問題) 畫面上，選取 Reset Administrator Password (重設管理員密碼)，然後選擇 Next (下一步)。

k. 在 Confirm (確認) 畫面上選擇 Rescue (救援)、OK (確定)。

l. 在 Done (完成) 畫面上選擇 Finish (完成)。

m. 關閉 Windows 伺服器工具，中斷與暫存執行個體的連線，然後返回 Amazon EC2 主控台。EC2Rescue

2. 將次要 (`xvdf`) 磁碟區從暫時執行個體中分離，如下所示：

   a. 在導覽窗格中選擇 Instances (執行個體)，然後選取暫時執行個體。

   b. 在暫存執行個體的 [儲存] 索引標籤上，記下列為 xv df 的EBS磁碟區識別碼。

   c. 在導覽窗格中，選擇 Volumes (磁碟區)。

   d. 在磁碟區清單中，請勾選之前步驟中所記下的磁碟區，並選擇 Actions (動作)、Detach Volume (分離磁碟區)。在將磁碟區狀態變更為 available (可用) 後，繼續下一步驟。

步驟 4：重新啟動原始執行個體

使用重設管理員密碼後EC2Launch，請將磁碟區重新連接到原始執行個體做為根磁碟區，並使用其 key pair 連線至執行個體以擷取管理員密碼。

重新啟動原始執行個體

1. 將磁碟區重新連接到原始執行個體，如下所示：

   a. 在導覽窗格中，選擇 Volumes (磁碟區)，選取從暫時執行個體分離的磁碟區，然後選擇 Actions (動作)、Attach Volume (連接磁碟區)。

   b. 在 Attach Volume (連接磁碟區) 對話方塊中，對於 Instances (執行個體)，開始輸入您的原始執行個體的名稱或 ID，然後選取執行個體。

   c. 對於 Device (裝置)，輸入 **`/dev/sda1`**。

   d. 選擇 Attach (連接)。在磁碟區狀態變更為 `in-use` 之後，繼續下一個步驟。

2. 在導覽窗格中，選擇 Instances (執行個體)。選取原始執行個體，然後依序選擇 Instance state (執行個體狀態)、Start instance (啟動執行個體)。在執行個體狀態變更為 `Running` 之後，繼續下一個步驟。

3. 使用新金鑰對的私有金鑰擷取新的 Windows 系統管理員密碼，然後連線到執行個體。如需詳細資訊，請參閱 使用 Connect 至您的 Windows 執行個體 RDP。

4. (選用) 如果不再使用暫時執行個體，您可以終止它。選取暫時執行個體，再依序選擇 Instance State (執行個體狀態)、Terminate instance (終止執行個體)。

重置 Windows 管理員密碼，EC2例如使用 EC2Config

如果您遺失了 Windows 系統管理員密碼，而且AMI在 Windows 伺服器 2016 年之前使用的是 Windows，您可以使用EC2Config代理程式來產生新密碼。

如果您使用的是 Windows 伺服器 2016 年或更新版本AMI，請參閱重置 Windows 管理員密碼，EC2例如使用 EC2Launch或，您可以使用該EC2Rescue工具，該工具會使用該EC2Launch服務來產生新密碼。

> **Note**
>
> 如果您已停用執行個體上的本機管理員帳戶，且您的執行個體已設定為 Systems Manager，您也可以使用和執行命令來重新啟用EC2Rescue和重設本機管理員密碼。如需詳細資訊，請參閱搭EC2Rescue配 Systems Manager 執行命令的 Windows 伺服器使用。

> **Note**
>
> 自動 AWS Systems Manager 化文件會自動套用重設本機管理員密碼所需的手動步驟。如需詳細資訊，請參閱《AWS Systems Manager 使用指南》中的「重設EC2執行個體的密碼和SSH金鑰」。

要使用重置 Windows 管理員密碼EC2Config，您需要執行以下操作：

- 步驟 1：確認EC2Config服務正在執行
- 步驟 2：將根磁碟區從執行個體中分離
- 步驟 3：將磁碟區連接至暫時執行個體
- 步驟 4：修改組態檔案
- 步驟 5：重新啟動原始執行個體

步驟 1：確認EC2Config服務正在執行

嘗試重設系統管理員密碼之前，請確認EC2Config服務已安裝並執行。您可以使用EC2Config服務來重設本節稍後的系統管理員密碼。

確認EC2Config服務是否正在執行

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中選擇「Instances (執行個體)」，然後選擇需要重設密碼的執行個體。此執行個體即為本程序中的原始執行個體。
3. 選擇 Actions (動作)、Monitor and troubleshoot (監視和故障診斷)、Get system log (取得系統日誌)。
4. 找出「EC2代理程式」項目，例如「EC2代理程式：Ec2Config 服務」v3.18.1118。如果您看到此項目，表示EC2Config服務正在執行中。

   如果系統記錄檔輸出為空白，或是EC2Config服務未執行，請使用執行個體主控台螢幕擷取畫面服務對執行個體進行疑難排解。如需詳細資訊，請參閱擷取無法連線執行個體的螢幕擷取畫面。

步驟 2：將根磁碟區從執行個體中分離

如果儲存密碼EC2Config的磁碟區以根磁碟區的形式連接至執行個體，您就無法使用重設管理員密碼。您必須將磁碟區從原始執行個體中分離，才能將磁碟區連接至暫時執行個體做為次要磁碟區。

若要將根磁碟區從執行個體中分離

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取需要重設密碼的執行個體，然後選擇執行個體狀態、停止執行個體。執行個體的狀態變更為 Stopped (已停止) 之後，請繼續下一個步驟。
4. (選用) 如果您擁有在啟動此執行個體時已指定的私有金鑰，請繼續下一個步驟。否則，請使用下列步驟將執行個體取代為使用新金鑰對啟動的新執行個體。

   a. 使用 Amazon EC2 主控台建立新的 key pair。若要使新金鑰對的名稱與已遺失私有金鑰的金鑰對的名稱相同，您必須先刪除現有金鑰對。

   b. 選取要取代的執行個體。請記下執行個體類型VPC、子網路、安全群組和執行個體的IAM角色。

c. 選取執行個體後，選擇 [動作]、[映像和範本]、[建立映像]。輸入映像的名稱和說明，然後選擇 Create image (建立映像)。

d. 在導覽窗格中，選擇AMIs。等待圖像狀態變更為可用。然後，選擇映像並從中選擇啟動實例AMI。

e. 填寫欄位以啟動執行個體，確定選取與要取代的執行個體相同的執行個體類型VPC、子網路、安全性群組和IAM角色，然後選擇 Launch 執行個體。

f. 出現提示時，請選擇您為新執行個體建立的 key pair，然後選擇 Launch 執行個體。

g. (選用) 如果原始執行個體有關聯的 Elastic IP 地址，將其傳輸至新執行個體。如果原始執行個體除了根EBS磁碟區之外還有磁碟區，請將它們傳輸到新的執行個體。

5. 將根磁碟區從原始執行個體中分離，如下所示：

a. 選取原始執行個體並選擇儲存空間索引標籤。記下根裝置名稱下的根裝置名稱。在 [封鎖裝置] 底下找到具有此裝置名稱的磁碟區，並記下磁碟區 ID。

b. 在導覽窗格中，選擇 Volumes (磁碟區)。

c. 在磁碟區清單中，選取您記下為根裝置的磁碟區，然後選擇 [動作] > [卸離磁碟區]。在將磁碟區狀態變更為 available (可用) 後，繼續下一步驟。

6. 如果您建立了新執行個體來取代原始執行個體，您可以立即終止原始執行個體。它不再需要。對於此程序的其餘部分，原始例證的所有參照都會套用至您建立的新例證。

步驟 3：將磁碟區連接至暫時執行個體

接下來，啟動暫時執行個體，並將磁碟區連接至此執行個體做為次要磁碟區。這是您用來修改組態檔案的執行個體。

啟動暫時執行個體並連接磁碟區

1. 啟動暫時執行個體，如下所示：

a. 在瀏覽窗格中，選擇 [執行個體]，選擇 [啟動執行個體]，然後選取AMI。

> **Important**
>
> 若要避免磁碟簽章衝突，您必須AMI為不同版本的 Windows 選取。例如，如果原始執行個體執行 Windows 伺服器 2019，請使用視窗伺服器 2016 年的基礎AMI啟動暫存執行個體。

b. 保留預設執行個體類型，然後選擇 Next: Configure Instance Details (下一步：設定執行個體的詳細資訊)。

c. 在 Configure Instance Details (設定執行個體詳細資訊) 頁面中，對於 Subnet (子網路)，選取與原始執行個體相同的可用區域，然後選擇 Review and Launch (檢閱和啟動)。

> **Important**
>
> 暫時執行個體與原始執行個體必須位於相同可用區域。如果您的暫時執行個體位於不同的可用區域，則無法將原始執行個體的根磁碟區與它連接。

d. 在 Review Instance Launch (檢閱執行個體啟動) 頁面，選擇 Launch (啟動)。

e. 出現提示時，建立新金鑰對，將其下載到電腦上的安全位置，然後選擇 Launch Instances (啟動執行個體)。

2. 將磁碟區連接到暫時執行個體作為輔助磁碟區，如下所示：

a. 在導覽窗格中，選擇 Volumes (磁碟區)，選取從原始執行個體分離的磁碟區，然後選擇 Actions (動作)、Attach Volume (連接磁碟區)。

b. 在 Attach Volume (連接磁碟區) 對話方塊中，對於 Instances (執行個體)，開始輸入暫時執行個體的名稱或 ID，然後從清單中選取執行個體。

c. 對於 Device (裝置)，輸入 **`xvdf`** (如果尚未出現在此)，選擇 Attach (連接)。

步驟 4：修改組態檔案

將磁碟區連接至暫時執行個體做為次要磁碟區之後，請在組態檔案中修改 `Ec2SetPassword` 外掛程式。

修改組態檔案

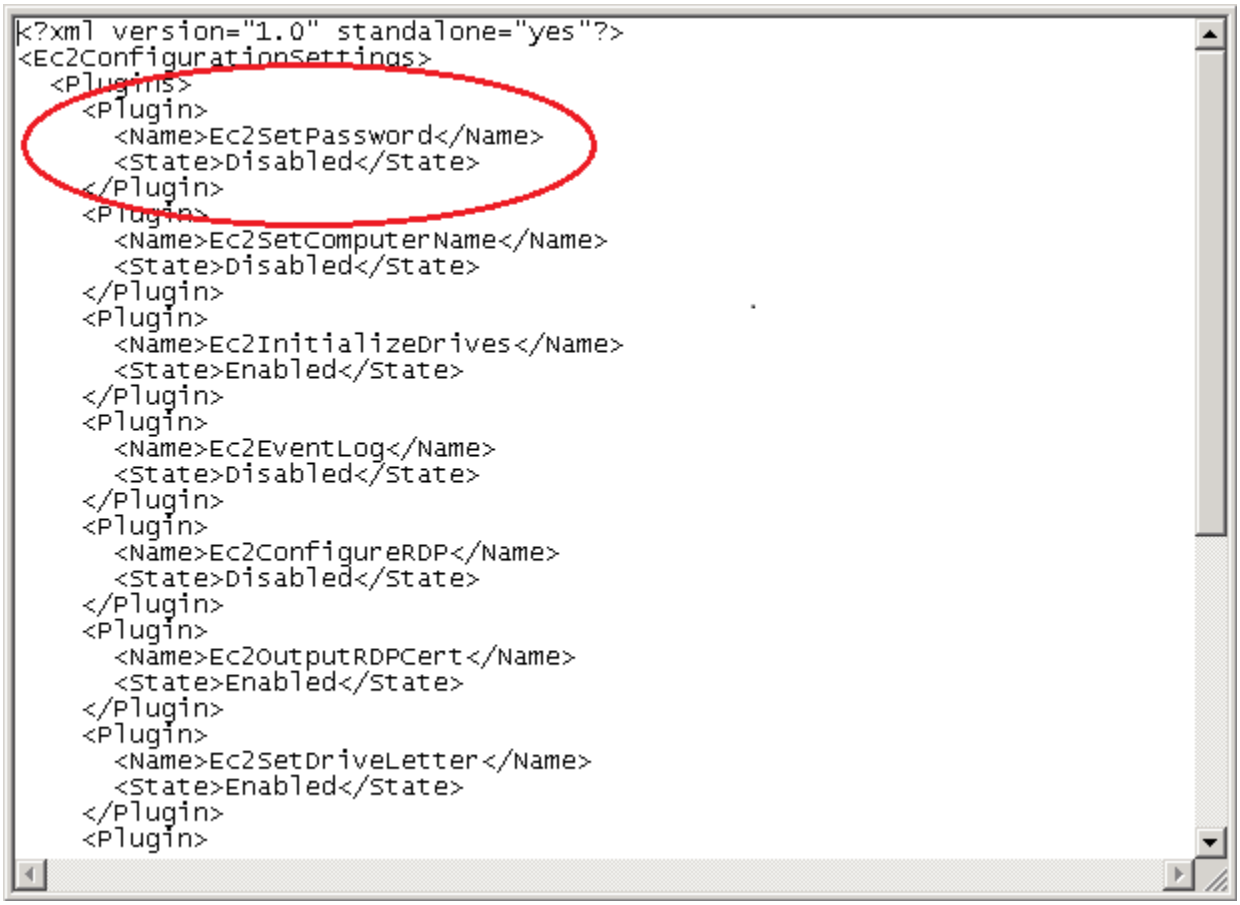
1. 從暫時執行個體，修改次要磁碟區上的組態檔案，如下所示：

a. 啟動並連接至暫時執行個體。

b. 使用下列指示將磁碟機上線：讓 Amazon EBS 磁碟區可供使用。

c. 導覽至次要磁碟區，然後使用「記事本」等文字編輯器開啟 `\Program Files\Amazon\Ec2ConfigService\Settings\config.xml`。

d. 在檔案頂端，找出名稱為 `Ec2SetPassword` 的外掛程式，如螢幕擷取畫面所示。將狀態從 `Disabled` 變更為 `Enabled` 並儲存檔案。



```
<?xml version="1.0" standalone="yes"?>
<Ec2ConfigurationSettings>
  <Plugins>
    <Plugin>
      <Name>Ec2SetPassword</Name>
      <State>Disabled</State>
    </Plugin>
    <Plugin>
      <Name>Ec2SetComputerName</Name>
      <State>Disabled</State>
    </Plugin>
    <Plugin>
      <Name>Ec2InitializeDrives</Name>
      <State>Enabled</State>
    </Plugin>
    <Plugin>
      <Name>Ec2EventLog</Name>
      <State>Disabled</State>
    </Plugin>
    <Plugin>
      <Name>Ec2ConfigureRDP</Name>
      <State>Disabled</State>
    </Plugin>
    <Plugin>
      <Name>Ec2OutputRDPCert</Name>
      <State>Enabled</State>
    </Plugin>
    <Plugin>
      <Name>Ec2SetDriveLetter</Name>
      <State>Enabled</State>
    </Plugin>
    <Plugin>
```

2. 修改組態檔案之後，將次要磁碟區從暫時執行個體中分離，如下所示：

   a. 使用 Disk Management (磁碟管理) 公用程式讓磁碟區離線。

   b. 中斷與臨時執行個體的連線，然後返回 Amazon EC2 主控台。

   c. 在導覽窗格中選擇 Volumes (磁碟區)、選取磁碟區，然後選擇 Actions (動作)、Detach Volume (分離磁碟區)。在將磁碟區狀態變更為 available (可用) 後，繼續下一步驟。

步驟 5：重新啟動原始執行個體

修改組態檔案之後，將磁碟區重新連接至原始執行個體做為根磁碟區，然後使用金鑰對來連接至執行個體，以擷取管理員密碼。

1. 將磁碟區重新連接到原始執行個體，如下所示：

   a. 在導覽窗格中，選擇 Volumes (磁碟區)，選取從暫時執行個體分離的磁碟區，然後選擇 Actions (動作)、Attach Volume (連接磁碟區)。

b. 在 Attach Volume (連接磁碟區) 對話方塊中，對於 Instances (執行個體)，開始輸入您的原始執行個體的名稱或 ID，然後選取執行個體。

c. 對於 Device (裝置)，輸入 **/dev/sda1**。

d. 選擇 Attach (連接)。在磁碟區狀態變更為 `in-use` 之後，繼續下一個步驟。

2. 在導覽窗格中，選擇 Instances (執行個體)。選取原始執行個體，然後依序選擇 Instance state (執行個體狀態)、Start instance (啟動執行個體)。在執行個體狀態變更為 `Running` 之後，繼續下一個步驟。

3. 使用新金鑰對的私有金鑰擷取新的 Windows 系統管理員密碼，然後連線到執行個體。如需詳細資訊，請參閱 使用 Connect 至您的 Windows 執行個體 RDP。

> **Important**
>
> 先停止執行個體，接著再啟動後，該執行個體將得到新的公有 IP 地址。請務必使用執行個體目前的公用名DNS稱連線至執行個體。如需詳細資訊，請參閱Amazon EC2 實例狀態更改。

4. (選用) 如果不再使用暫時執行個體，您可以終止它。選取暫時執行個體，再依序選擇 Instance State (執行個體狀態)、Terminate instance (終止執行個體)。

疑難排解 Amazon EC2 視窗執行個體的系統問題

如果您在映像準備期間發生問題或收到錯誤訊息，請檢閱下列日誌。記錄位置會因您使用 Sysprep 執行EC2Config、EC2Launch v1 或 EC2Launch v2 而有所不同。

- `%WINDIR%\Panther\Unattendgc`(EC2Config、EC2Launch第 1 版和第 EC2Launch 2 版)
- `%WINDIR%\System32\Sysprep\Panther`(EC2Config、EC2Launch第 1 版和第 EC2Launch 2 版)
- `C:\Program Files\Amazon\Ec2ConfigService\Logs\Ec2ConfigLog.txt`(EC2Config僅限)
- `C:\ProgramData\Amazon\Ec2Config\Logs`(EC2Config僅限)
- `C:\ProgramData\Amazon\EC2-Windows\Launch\Log\EC2Launch.log`(僅限EC2Launch第 1 版)
- `%ProgramData%\Amazon\EC2Launch\log\agent.log`(僅適用於第 EC2Launch 2 版)

如果您在使用 Sysprep 準備映像期間收到錯誤訊息，可能會無法連接 OS。若要檢閱日誌檔案，您必須停止執行個體，並將其根磁碟區連接到另一個運作狀態良好的執行個體做為輔助磁碟區，然後於次要磁碟區檢閱前述的日誌。如需依名稱排列日誌檔用途的詳細資訊，請參閱 Microsoft 文件中的 Windows 安裝程式相關日誌檔。

如果您在 Unattendgc 日誌檔案中找到錯誤，請使用 Microsoft Error Lookup Tool 取得錯誤的其他詳細資訊。Unattendgc 日誌檔案中所報告的下列問題，通常是因為執行個體上有一或多個毀損的使用者描述檔而導致的結果：

```
Error [Shell Unattend] _FindLatestProfile failed (0x80070003) [gle=0x00000003]
Error [Shell Unattend] CopyProfile failed (0x80070003) [gle=0x00000003]
```

有兩個選項可解決此問題：

選項 1

在執行個體上使用 Regedit，搜尋下列機碼。確認沒有已刪除之使用者的描述檔登錄機碼。

```
[HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion
\ProfileList\
```

選項 2

1. 編輯相關檔案，如下所示：
   - 視窗伺服器 2012 R2 及更早版本 — 編輯EC2Config回應檔案 (`C:\Program Files\Amazon\Ec2ConfigService\sysprep2008.xml`)。
   - Windows Server 2016 和 2019 — 編輯 unattend.xml 回應檔案 (`C:\ProgramData\Amazon\EC2-Windows\Launch\Sysprep\Unattend.xml`)。
   - Windows Server 2022 — 編輯 unattend.xml 回應檔案 (`C:\ProgramData\Amazon\EC2Launch\sysprep\unattend.xml`)。
2. 將 `<CopyProfile>true</CopyProfile>` 變更為 `<CopyProfile>false</CopyProfile>`。
3. 重新執行 Sysprep。請注意，此組態變更將會在 Sysprep 完成之後，刪除內建管理員使用者描述檔。

使用疑難排解受損的 Amazon EC2 Linux EC2Rescue

EC2Rescuefor Linux 是一種開放原始碼工具 easy-to-use，可在 Amazon EC2 Linux 執行個體上執行，使用其超過 100 個模組的程式庫來診斷、疑難排解和修復常見問題。模塊是YAML包含BASH或 Python 腳本以及必要元數據的文件。

Linux 執行個體的一些一般化使EC2Rescue用案例包括：

- 收集系統日誌和軟件包管理器日誌
- 收集資源使用率資料
- 診斷和修復已知有問題的核心參數和常見的「開啟」問題 SSH

> **Note**
>
> `AWSSupport-TroubleshootSSH` AWS Systems Manager 自動化執行手冊會安裝 EC2Rescue Linux，然後使用該工具檢查或嘗試修正阻止與 Linux 執行個體SSH連線的常見問題。如需詳細資訊，請參閱AWSSupport疑難排解SSH。

如果您使用的是 Windows 執行個體，請參閱the section called “EC2Rescue視窗執行個體”。

主題

- 在 Amazon EC2 Linux 執EC2Rescue行個體上安裝
- 在 Amazon EC2 Linux 實例上運行EC2Rescue命令
- 為 Amazon EC2 Linux 執行個體開發EC2Rescue模組

在 Amazon EC2 Linux 執EC2Rescue行個體上安裝

Linux EC2Rescue 專用工具可安裝在符合下列先決條件的 Amazon EC2 Linux 執行個體上。

必要條件

- 支援的作業系統：
  - Amazon Linux 2
  - Amazon Linux 2016.09+
  - SUSE企業伺服器 12 歲以上

- RHEL7 歲以上
- Ubuntu 16.04+

- 軟體要求：
  - Python 2.7.9+ 或 3.2+

安裝 EC2Rescue

`AWSSupport-TroubleshootSSH`runbook 會EC2Rescue為 Linux 安裝，然後使用該工具檢查或嘗試修復阻止遠端連線到 Linux 電SSH腦的常見問題。有關更多信息，並運行此自動化，請參閱 AWS Support-故障排除SSH。

如果您的系統具有所需的 Python 版本，則您可以安裝標準組建。否則，您可以安裝隨附的組建，其中包括最小的 Python 複本。

安裝標準組建

1. 從運作中的 Linux 執行個體下載EC2Rescue適用於 Linux 的工具：

```
curl -O https://s3.amazonaws.com/ec2rescuelinux/ec2rl.tgz
```

2. (選擇性) 驗證 Linux 安裝檔案的簽章。EC2Rescue如需詳細資訊，請參閱(選擇性) 確認適EC2Rescue用於 Linux 的簽章。
3. 下載 sha256 雜湊檔案：

```
curl -O https://s3.amazonaws.com/ec2rescuelinux/ec2rl.tgz.sha256
```

4. 確認 tarball 的完整性：

```
sha256sum -c ec2rl.tgz.sha256
```

5. 解壓縮 tarball：

```
tar -xzvf ec2rl.tgz
```

6. 列出說明檔確認安裝成功：

```
cd ec2rl-<version_number>
./ec2rl help
```

安裝隨附的組建

有關下載的鏈接和限制列表，請參EC2Rescue閱 github 上的 Linux。

(選擇性) 確認適EC2Rescue用於 Linux 的簽章

以下是驗證 Linux 套件適用於 Linux 作業系統EC2Rescue之有效性的建議程序。

當您從網際網路下載應用程式時，建議您驗證軟體發佈者的身分，並檢查應用程式在發佈之後並未遭更改或損毀。如此可保護您，避免安裝到包含病毒或其他惡意程式碼的應用程式版本。

如果在執行本主題中的步驟之後，您判斷 Linux EC2Rescue 適用的軟體已變更或損毀，請勿執行安裝檔案。反之，請聯絡 Amazon Web Services。

EC2Rescue對於 Linux 的基於 Linux 的操作系統的文件使用 GnuPG，一個開源實現相當不錯的隱私（開放PGP）標準的安全數字簽名進行簽名。GnuPG（也稱為GPG）通過數字簽名提供身份驗證和完整性檢查。 AWS 發佈公開金鑰和簽章，您可以用來驗證 Linux 套件下載EC2Rescue的檔案。如需有關PGP與 GnuPG (GPG) 的詳細資訊，請參閱 http://www.gnupg.org。

第一步是與軟體發佈者建立信任。下載軟體發佈者的公開金鑰，檢查公開金鑰的擁有者是否為聲稱的擁有者，然後將公開金鑰新增至您的 keyring。您的 keyring 是一組已知的公開金鑰。在您建立公開金鑰的真實性之後，即可用它來驗證應用程式的簽章。

任務

- 驗證和匯入公開金鑰
- 驗證套件的簽章

驗證和匯入公開金鑰

此程序的下一個步驟是驗證 Linux EC2Rescue 的公開金鑰，並將其新增為您的金鑰GPG圈中的受信任金鑰。

若要驗證並匯入EC2Rescue適用於 Linux 的公開金鑰

1. 在命令提示字元中，使用下列命令取得公開GPG建置金鑰的副本：

```
curl -O https://s3.amazonaws.com/ec2rescuelinux/ec2rl.key
```

2. 在您儲存的目錄中的命令提示字元中`ec2rl.key`，使用下列命令將 Linux 公開金鑰匯入金鑰圈：EC2Rescue

```
gpg2 --import ec2rl.key
```

此命令會傳回類似以下的結果：

```
gpg: /home/ec2-user/.gnupg/trustdb.gpg: trustdb created
gpg: key 2FAE2A1C: public key "ec2autodiag@amazon.com <EC2 Rescue for Linux>"
 imported
gpg: Total number processed: 1
gpg:               imported: 1  (RSA: 1)
```

> **Tip**
>
> 如果您看到指令無法找到的錯誤訊息，請使用 (以 Debian 為基礎的 Linux) 或 `apt-get install gnupg2` `yum install gnupg2` (以 Red Hat 為基礎的 Linux) 來安裝 GnuPG 公用程式。

驗證套件的簽章

在您安裝GPG工具、驗證並匯入EC2Rescue適用於 Linux 的公開金鑰，並驗證 Linux 公開金鑰受信任之後，您就可以準備好驗證 Linux 安裝指令碼的簽章。EC2Rescue EC2Rescue

若要驗證 Linux 安裝指令碼簽章的EC2Rescue步驟

1. 在命令提示字元上，執行以下命令以下載安裝指令碼的簽章檔案：

```
curl -O https://s3.amazonaws.com/ec2rescuelinux/ec2rl.tgz.sig
```

2. 在您儲存的目錄`ec2rl.tgz.sig`和 Linux 安裝檔案的命令提示字元中執行下列命令，以驗證簽章。EC2Rescue兩個檔案都必須存在。

```
gpg2 --verify ./ec2rl.tgz.sig
```

輸出應類似以下所示：

```
gpg: Signature made Thu 12 Jul 2018 01:57:51 AM UTC using RSA key ID 6991ED45
gpg: Good signature from "ec2autodiag@amazon.com <EC2 Rescue for Linux>"
gpg: WARNING: This key is not certified with a trusted signature!
```

```
gpg:          There is no indication that the signature belongs to the owner.
Primary key fingerprint: E528 BCC9 0DBF 5AFA 0F6C  C36A F780 4843 2FAE 2A1C
     Subkey fingerprint: 966B 0D27 85E9 AEEC 1146  7A9D 8851 1153 6991 ED45
```

如果輸出包含片語`Good signature from "ec2autodiag@amazon.com <EC2 Rescue for Linux>"`，表示簽章已成功驗證，您可以繼續執行 EC2Rescue for Linux 安裝程序檔。

如果輸出包含 `BAD signature` 片語，請檢查您是否已正確執行程序。如果您持續收到此回應，請聯絡 Amazon Web Services 並且不要執行您先前下載的安裝檔。

以下是您可能會看到的警告的詳細資訊：

- WARNING: This key is not certified with a trusted signature! There is no indication that the signature belongs to the owner.這是指您個人對您的信任程度，即您擁有適用EC2Rescue於 Linux 的真實公鑰。在理想世界中，您請會前往 Amazon Web Services 辦公室並獲得該金鑰。不過，通常您會從網站下載。在此情況下，該網站是 Amazon Web Services 網站。
- gpg2: no ultimately trusted keys found. 這表示該特定金鑰未獲得您 (或您信任的其他人) 的「最終信任」。

如需詳細資訊，請參閱 http://www.gnupg.org。

在 Amazon EC2 Linux 實例上運行EC2Rescue命令

EC2Rescue是一個命令行工具。在 Linux 執行個體EC2Rescue上安裝之後，您可以透過執行來取得如何使用此工具的一般說明`./ec2rl help`。您可以透過執行來檢視可用的模組`./ec2rl list`，也可以透過執行取得特定模組的說明`./ec2rl help` *`module_name`*。

下列是您可執行以開始使用本工具的常用任務。

任務

- 執行EC2Rescue模組
- 上傳模EC2Rescue組結果
- 建立 Amazon EC2 Linux 執行個體的備份

執行EC2Rescue模組

執行所有EC2Rescue模組

使用指./ec2rl run令而不指定任何其他參數。部分模組需要根存取權。如果您不是 root 使用者，請在執行指令sudo時使用。

```
./ec2rl run
```

執行特定EC2Rescue模組的步驟

使用./ec2rl run命令`--only-modules`，並指定要執行的模組名稱。某些模塊需要參數才能使用它們。

```
./ec2rl run --only-modules=module_name --arguments
```

例如，若要執行dig模組來查詢`amazon.com`網域，請使用下列命令。

```
./ec2rl run --only-modules=dig --domain=amazon.com
```

若要檢視EC2Rescue模組的結果

運行該模塊，然後查看日誌文件`cat /var/tmp/ec2rl/logfile_location`。例如，您可以在下列位置找到dig模組的記錄檔：

```
cat /var/tmp/ec2rl/timestamp/mod_out/run/dig.log
```

上傳模EC2Rescue組結果

如果 AWS Support 已要求EC2Rescue模組的結果，您可以使用EC2Rescue工具上傳記錄檔。您可以將結果上傳到您擁有的 Amazon S3 儲存貯體提供的位置， AWS Support 或將結果上傳到您擁有的 Amazon S3 儲存貯體。

將結果上載至由下列人員提供的位置： AWS Support

使用 ./ec2rl upload 命令。對於`--upload-directory`，指定記錄檔的位置。對於`--support-url`，指定URL提供者 AWS Support。

```
./ec2rl upload --upload-directory=/var/tmp/ec2rl/logfile_location --support-
url="url_provided_by_aws_support"
```

將結果上傳到 Amazon S3 存儲桶

使用 ./ec2rl upload 命令。對於`--upload-directory`，指定記錄檔的位置。對於`--presigned-url`，指定 S3 儲存貯體URL的預先簽署。如需針對 Amazon S3 產生預先簽署URLs的詳細資訊，請參閱使用預先URLs簽章上傳物件。

```
./ec2rl upload --upload-directory=/var/tmp/ec2rl/logfile_location --presigned-
url="presigned_s3_url"
```

建立 Amazon EC2 Linux 執行個體的備份

您可EC2Rescue以透過建立AMI或建立連接磁碟區的快照來備份 Linux 執行個體。

若要建立 AMI

使用指`./ec2rl run`令並指定`-backup-ami`。

```
./ec2rl run --backup=ami
```

建立所有連結磁碟區的多磁碟區快照

使用指`./ec2rl run`令並指定`-backup-allvolumes`。

```
./ec2rl run --backup=allvolumes
```

建立特定連接磁碟區的快照

使用指`./ec2rl run`令，對於`--backup`，指定要備份的磁碟區 ID。

```
./ec2rl run --backup=volume_id
```

為 Amazon EC2 Linux 執行個體開發EC2Rescue模組

模塊被寫入YAML，一個數據序列化標準。一個模塊的YAML文件由一個單一的文檔，代表模塊及其屬性。

新增模組屬性

下表列出可用的模組屬性。

屬性	描述
name	模組的名稱。此名稱長度應小於或等於 18 個字元。
version	模組的版本編號。

屬性	描述
title	模組的簡述標題。此值長度應小於或等於 50 個字元。

helptext	模組的延伸描述。各行長度應小於或等於 75 個字元。如果模組使用引數 (無論必要或選用)，請以 helptext 值包含它們。
	
	例如：
	
	`helptext: !!str |`
	`  Collect output from ps for system analysis`
	`  Consumes --times= for number of times to repeat`
	`  Consumes --period= for time period between repetition`

placement	應該執行模組的階段。支援的值為： • prediagnostic • run • postdiagnostic
語言	撰寫模組程式碼的語言。支援的值為： • bash • Python **Note** Python 程式碼必須與 Python 2.7.9+ 和 Python 3.2+ 都相容。

屬性	描述
remediation	指出模組是否支援修補。支援的值為 `True` 或 `False`。 如果無此屬性，模組將預設為 `False`，讓其成為不支援修補之模組的選用屬性。
content	完整的指令碼程式碼。
限制條件	包含限制值的物件名稱。
domain	說明模組如何分組或分類的描述項。包含的模組集使用下列網域： • application • net • os • performance
class	模組執行的任務類型描述項。包含的模組集使用下列類別： • collect (收集程式的輸出) • diagnose (根據條件集通過/失敗) • gather (複製檔案並寫入特定檔案)
distro	此模組支援的 Linux 發行版本清單。包含的模組集使用下列發行版本： • alami (Amazon Linux) • rhel • ubuntu • suse
必要	模組從CLI選項中使用的必要引數。
選擇性	模組可以使用的選用引數。

屬性	描述
software	在模組中使用的軟體可執行檔。此屬性旨在指定非預設安裝的軟體。EC2Rescuefor Linux 邏輯可確保這些程式在執行模組之前存在且可執行。
package	可執行檔的來源軟體套件。此屬性旨在提供軟體套件的延伸詳細資訊，包括下載或取得進一URL步資訊。
sudo	指出是否一定要有根存取才能執行模組。 您不必在模組指令碼中實作 sudo 檢查。如果值為 true，那麼 EC2Rescue for Linux 邏輯僅在執行中的使用者具有 root 存取權限時才會執行模組。
perfimpact	指出模組是否會顯著影響其執行所在環境的效能。如果此值為 true 且無 `--perfimpact=true` 引數，則跳過模組。
parallelexclusive	指定需要互斥的程式。例如，所有指定 "bpf" 的模組都依序執行。

新增環境變數

下表列出可用的環境變數。

環境變數	描述
`EC2RL_CALLPATH`	`ec2rl.py` 的路徑。此路徑可用以尋找 lib 目錄及使用經過廠商處理的 Python 模組。
`EC2RL_WORKDIR`	診斷工具的主要 tmp 目錄。 預設值：`/var/tmp/ec2rl` 。
`EC2RL_RUNDIR`	存放所有輸出的目錄。

環境變數	描述
	預設值：`/var/tmp/ec2rl/<date&timestamp>` 。
`EC2RL_GATHEREDDIR`	放置所收集模組資料的根目錄。 預設值：`/var/tmp/ec2rl/<date&timestamp>/mod_out/gathered/` 。
`EC2RL_NET_DRIVER`	執行個體上按字母順序第一個非虛擬網路界面使用的驅動程式。 範例: • xen_netfront • ixgbevf • ena
`EC2RL_SUDO`	如果 Linux 是以根使EC2Rescue用者身分執行，則為 true；否則為 false。
`EC2RL_VIRT_TYPE`	如執行個體中繼資料所提供的虛擬化類型。 範例： • default-hvm • default-paravirtual
`EC2RL_INTERFACES`	系統上的界面列舉清單。值是包含名稱的字串，例如 `eth0`、`eth1` 等等。這是透過 `functions.bash` 產生的，僅供其來源模組使用。

使用YAML語法

構建模塊YAML文件時應注意以下內容：

- 三個連字號 (`---`) 表示文件的明確開頭。

- 該`!ec2rlcore.module.Module`標籤告訴YAML解析器從數據流創建對象時調用哪個構造函數。您可於 `module.py` 檔案中找到建構函數。
- 該`!!str`標籤告訴解YAML析器不要嘗試確定數據的類型，而是將內容解釋為字符串文字。
- 管道字符（`|`）告訴YAML解析器該值是一個文字樣式的標量。在本例中，剖析器包含所有空格。這對模組而言很重要，因為會保留縮排和換行字元。
- 標YAML準縮排是兩個空格，可以在下面的例子中看到。請確定您的指令碼保持標準縮排 (例如，Python 四格)，然後在模組檔案中將完整內容縮排兩格。

範例模組

範例一 (`mod.d/ps.yaml`)：

```
--- !ec2rlcore.module.Module
# Module document. Translates directly into an almost-complete Module object
name: !!str ps
path: !!str
version: !!str 1.0
title: !!str Collect output from ps for system analysis
helptext: !!str |
  Collect output from ps for system analysis
  Requires --times= for number of times to repeat
  Requires --period= for time period between repetition
placement: !!str run
package:
  - !!str
language: !!str bash
content: !!str |
  #!/bin/bash
  error_trap()
  {
      printf "%0.s=" {1..80}
      echo -e "\nERROR: "$BASH_COMMAND" exited with an error on line ${BASH_LINENO[0]}"
      exit 0
  }
  trap error_trap ERR

  # read-in shared function
  source functions.bash
  echo "I will collect ps output from this $EC2RL_DISTRO box for $times times every
 $period seconds."
  for i in $(seq 1 $times); do
```

```
      ps auxww
      sleep $period
  done
constraint:
  requires_ec2: !!str False
  domain: !!str performance
  class: !!str collect
  distro: !!str alami ubuntu rhel suse
  required: !!str period times
  optional: !!str
  software: !!str
  sudo: !!str False
  perfimpact: !!str False
  parallelexclusive: !!str
```

使用疑難排解受損的 Amazon EC2 Windows EC2Rescue

EC2RescueWindows 伺服器是您在 Amazon EC2 Windows 伺服器執行個體上執行的 easy-to-use 工具，用於診斷和疑難排解可能出現的問題。收集日誌檔案和故障診斷問題以及主動搜尋可能發生問題的區域，極其重要。它甚至可以從其他執行個體檢查 Amazon EBS 根磁碟區，並收集相關日誌，以便使用該磁碟區進行 Windows Server 執行個體的疑難 以下是一些EC2Rescue可以解決的常見問題：

- 由於防火牆、遠端桌面通訊協定 (RDP) 或網路介面設定造成的執行個體連線問題
- 由於停止錯誤、開機迴圈或登錄損毀而導致的作業系統開機問題
- 可能需要進階記錄分析和疑難排解的問題

EC2Rescue對於 Windows 服務器有兩個不同的模塊：

- 從所有不同來源收集資料的資料收集器模組
- 一種分析器模組，可根據一系列預先定義的規則來剖析收集的資料，以識別問題並提供建議

EC2Rescue適用於 Windows 伺服器工具僅在執行 Windows 伺服器 2012 及更新版本的 Amazon EC2 執行個體上執行。當工具啟動時，它會檢查它是否在 Amazon EC2 執行個體上執行。

> **Note**
>
> `AWSSupport-ExecuteEC2Rescue` AWS Systems Manager Automation Runbook 會使用此EC2Rescue工具進行疑難排解，並在可能的情況下修正指定EC2執行個體的常見連線問題。如需詳細資訊，並執行此自動化操作，請參閱 > AWSSupport-Execute EC2Rescue。

如果您使用的是 Linux 執行個體，請參閱the section called “EC2Rescue針對 Linux 執行個體”。

主題

- 使用 EC2Rescue GUI
- 使用 EC2Rescue CLI
- 使用EC2Rescue和 Systems Manager 對受損的 Windows 執行個體

使用 EC2Rescue GUI

EC2RescueWindows 伺服器可以在離線執行個體上執行下列分析：

選項	描述

診斷與救援	EC2RescueWindows 服務器可以檢測並解決以下服務設置的問題：
	• 系統時間
	  • RealTimeisUniversal-偵測`RealTimeisUniversal` 登錄機碼是否已啟用。如果停用，當時區設定為以外UTC的值時，Windows 系統時間會漂移。
	• Windows 防火牆
	  • Domain networks (網域網路) - 偵測此 Windows 防火牆描述檔為啟用或停用。
	  • Private networks (私有網路) - 偵測此 Windows 防火牆描述檔為啟用或停用。

選項	描述

	• Guest or public networks (訪客或公有網路) - 偵測此 Windows 防火牆描述檔為啟用或停用。
	
	• 遠端桌面
	  • Service Start (服務啟動) - 偵測是否啟用遠端桌面服務。
	  • Remote Desktop Connections (遠端桌面連線) - 偵測是否啟用此服務。
	  • TCP連接埠-偵測遠端桌面服務正在偵聽的連接埠。
	
	• EC2Config(視窗伺服器 2012 年 R2 及更早版本)
	  • 安裝-偵測已安裝的EC2Config版本。
	  • 服務啟動-偵測EC2Config服務是否已啟用。
	  • EC2 SetPassword-產生新的管理員密碼。
	  • EC2 HandleUserData-允許您在下次啟動實例時運行用戶數據腳本。
	
	• EC2Launch(視窗伺服器 2016 及更新版本)
	  • 安裝-偵測已安裝的EC2Launch版本。
	  • EC2 SetPassword-產生新的管理員密碼。
	
	• 網路介面
	  • DHCP服務啟動-偵測DHCP服務是否已啟用。
	  • Ethernet detail (乙太網路詳細資訊) - 如果偵測到，顯示網路驅動程式版本的相關資訊。

選項	描述
	• DHCP在乙太網路上-偵測DHCP是否已啟用。 • 磁碟簽章狀態 • 磁碟上的簽章和開機組態資料庫上的簽章(BCD)-偵測磁碟簽章和BCD簽章是否相同。如果值不同，請EC2Rescue嘗試在開啟簽章的情況下覆寫磁碟簽章BCD。
還原	執行下列動作之一： • Last Known Good Configuration (上一個已知的良好組態) - 嘗試將執行個體開機為上一個已知的可開機狀態。 • Restore registry from backup (從備份還原登錄) - 從 `\Windows\System32\config\RegBack` 還原登錄。
擷取日誌	允許您擷取執行個體的日誌用於分析。

EC2RescueWindows 伺服器可以從使用中和離線執行個體收集下列資料：

項目	描述
事件日誌	收集應用程式、系統和EC2Config事件記錄。
登錄	收集 `SYSTEM` 和 `SOFTWARE` Hive。
Windows Update 日誌	收集 Windows Update 產生的日誌檔案。 **Note** 在視窗伺服器 2016 及更新版本中，記錄檔會以視窗事件追蹤 (ETW) 格式收集。

項目	描述
Sysprep 日誌	收集 Windows 系統準備工具產生的日誌檔案。
驅動程式設定日誌	收集視窗安裝程式API記錄檔 (`setupapi.dev.log` 和`setupapi.setup.log`)。
開機組態	收集 `HKEY_LOCAL_MACHINE\BCD00000000` Hive。
記憶體傾印	收集執行個體上所有的記憶體傾印檔案。
EC2Config檔案	收集EC2Config服務產生的記錄檔。
EC2Launch檔案	收集指令碼所產生的記錄EC2Launch檔。
SSM代理程式檔	收集SSM代理程式和修補程式管理員記錄檔產生的記錄檔
EC2E lasticGPUs 檔案	收集與彈性相關的事件日誌GPUs。
ECS	收集與 Amazon 相關的日誌ECS。
CloudEndure	收集與 CloudEndure 代理程式相關的記錄檔。

EC2RescueWindows 伺服器可以從使用中執行個體收集下列其他資料：

項目	描述
系統資訊	收集MSInfo32。
群組政策結果	收集 Group Policy (群組政策) 報告。

分析離線的執行個體

Offline Instance (離線的執行個體) 選項對偵錯 Windows 執行個體的開機問題很有幫助。

在離線的執行個體上執行動作

1. 從運作中的 Windows 伺服器執行個體中，下載EC2Rescue適用於 Windows 伺服器的工具並解壓縮檔案。

   您可以執行下列 PowerShell 命令來下載，EC2Rescue而不必變更 Internet Explorer 增強型安全性設定 (ESC)：

   ```
   Invoke-WebRequest https://s3.amazonaws.com/ec2rescue/windows/EC2Rescue_latest.zip -
   OutFile $env:USERPROFILE\Desktop\EC2Rescue_latest.zip
   ```

   此命令會將 EC2Rescue .zip 檔案下載至目前登入使用者的桌面。

   > **Note**
   >
   > 如果您在下載檔案時收到錯誤訊息，而且您使用的是 Windows 伺服器 2016 或更早版本，則可能需要為 PowerShell 終端機啟用 TLS 1.2。您可以使用以下命令為當前 PowerShell 會話啟用 TLS 1.2，然後再試一次：
   >
   > ```
   > [Net.ServicePointManager]::SecurityProtocol =
   > [Net.SecurityProtocolType]::Tls12
   > ```

2. 如果故障的執行個體尚未停止，請停止它。
3. 從錯誤的執行個體中分離EBS根磁碟區，並將磁碟區連接到已EC2Rescue安裝 Windows 伺服器的運作中 Windows 執行個體。
4. 在工作執EC2Rescue行個體上執行 Windows 伺服器工具，然後選擇 [離線執行個體]。
5. 選取新掛載磁碟區上的磁碟，然後選擇 Next (下一步)。
6. 確認選取磁碟，然後選擇 Yes (是)。
7. 選擇要執行的離線執行個體選項並選擇 Next (下一步)。

Windows Server 工具會掃描磁碟區，並根據選取的記錄檔收集疑難排解資訊。EC2Rescue

收集作用中執行個體的資料

您可收集作用中執行個體的日誌和其他資料。

收集作用中執行個體的資料

1. 連接至 Windows 執行個體。
2. 將EC2Rescue適用於 Windows 伺服器的工具下載到您的 Windows 執行個體並解壓縮檔案。

   您可以執行下列 PowerShell 命令來下載，EC2Rescue而不必變更 Internet Explorer 增強型安全性設定 (ESC)：

   ```
   Invoke-WebRequest https://s3.amazonaws.com/ec2rescue/windows/EC2Rescue_latest.zip -
   OutFile $env:USERPROFILE\Desktop\EC2Rescue_latest.zip
   ```

   此命令會將 EC2Rescue .zip 檔案下載至目前登入使用者的桌面。

   > **Note**
   >
   > 如果您在下載檔案時收到錯誤訊息，而且您使用的是 Windows 伺服器 2016 或更早版本，則可能需要為 PowerShell 終端機啟用 TLS 1.2。您可以使用以下命令為當前 PowerShell 會話啟用 TLS 1.2，然後再試一次：
   >
   > ```
   > [Net.ServicePointManager]::SecurityProtocol =
   > [Net.SecurityProtocolType]::Tls12
   > ```

3. 開啟 Windows 伺服器應用程式並接受授權合約。EC2Rescue
4. 選擇 Next (下一步)、Current instance (目前的執行個體)、Capture logs (擷取日誌)。
5. 選取要收集的資料項目並選擇 Collect... (收集)。閱讀警告內容，然後選擇 Yes (是) 繼續。
6. 選擇檔案的檔案名稱和位置，然後ZIP選擇 [儲存]。
7. Windows 伺服器完成後EC2Rescue，選擇「開啟包含資料夾」以檢視ZIP檔案。
8. 選擇 Finish (完成)。

使用 EC2Rescue CLI

Windows 伺服器命令列介面 (CLI) 可讓您以程式設計方式執EC2Rescue行 Windows 伺服器外掛程式 (稱為「動作」)。EC2Rescue

EC2Rescue適用於 Windows 伺服器工具有兩種執行模式：

- /online — 這可讓您對安裝 Windows 伺服器的執EC2Rescue行個體採取動作，例如收集記錄檔。
- /離線：<device_id>這可讓您EC2Rescue針對已安裝 Windows 伺服器的個別 Amazon EC2 Windows 執行個體所連接的離線根磁碟區採取動作。

將EC2Rescue適用於 Windows 伺服器視窗伺服EC2Rescue器工具下載到您的 Windows 執行個體，然後解壓縮檔案。您可以使用下列命令來檢視說明檔案：

```
EC2RescueCmd.exe /help
```

EC2Rescue視窗伺服器可以在 Amazon 視EC2窗執行個體上執行下列動作：

- 收集動作
- 救援動作
- 還原動作

收集動作

> **Note**
> 您可收集所有日誌、完整的日誌群組，或群組內的個別日誌。

EC2RescueWindows 伺服器可以從使用中和離線執行個體收集下列資料。

日誌群組	可用日誌	描述
`all`		收集所有可用的日誌。
`eventlog`	• `'Application'` • `'System'` • `'EC2ConfigService'`	收集應用程式、系統和EC2Config事件記錄。
`memory-dump`	• `'Memory Dump File'` • `'Mini Dump Files'`	收集執行個體上所有的記憶體傾印檔案。
`ec2config`	• `'Log Files'`	收集EC2Config服務產生的記錄檔。

日誌群組	可用日誌	描述
	• 'Configuration Files'	
ec2launch	• 'Logs' • 'Config'	收集指令碼所產生的記錄EC2Launch檔。
ssm-agent	• 'Log Files' • 'Patch Baseline Logs' • 'InstanceData'	收集SSM代理程式和修補程式管理員記錄檔產生的記錄檔
sysprep	'Log Files'	收集 Windows 系統準備工具產生的日誌檔案。
driver-setup	• 'SetupAPI Log Files' • 'DPInst Log File' • 'AWS PV Setup Log File'	收集視窗安裝程式API記錄檔 (setupapi.dev.log 和setupapi.setup.log)。
registry	• 'SYSTEM' • 'SOFTWARE' • 'BCD'	收集 SYSTEM 和 SOFTWARE Hive。
egpu	• 'Event Log' • 'System Files'	收集與彈性相關的事件日誌GPUs。
boot-config	'BCDEDIT Output'	收集 HKEY_LOCAL_MACHINE\BCD00000000 Hive。

日誌群組	可用日誌	描述
windows-update	'Log Files'	收集 Windows Update 產生的日誌檔案。 **Note** 在視窗伺服器 2016 及更新版本中，記錄檔會以視窗事件追蹤 (ETW) 格式收集。
cloudendure	• 'Migrate Script Logs' • 'Driver Logs' • 'CloudEndure File List'	收集與 CloudEndure 代理程式相關的記錄檔。

EC2RescueWindows 伺服器可以從使用中執行個體收集下列其他資料。

日誌群組	可用日誌	描述
system-info	'MSInfo32 Output'	收集 MSInfo32。
gpresult	'GPResult Output'	收集 Group Policy (群組政策) 報告。

以下為可用的選項：

- /output: < outputFilePath >-以 zip 格式儲存收集的記錄檔所需的目標檔案路徑位置。
- /no-offline - 離線模式中使用的選用屬性。完成動作後不會將磁碟區設為離線。
- /no-fix-signature-在離線模式下使用的可選屬性。完成動作後不會修正可能的磁碟簽章衝突。

範例

以下是使用 Windows 伺服器的EC2Rescue範例CLI。

線上模式範例

收集所有可用的日誌：

```
EC2RescueCmd /accepteula /online /collect:all /output:<outputFilePath>
```

只收集特定的日誌群組：

```
EC2RescueCmd /accepteula /online /collect:ec2config /output:<outputFilePath>
```

收集日誌群組內的個別日誌：

```
EC2RescueCmd /accepteula /online /collect:'ec2config.Log Files,driver-setup.SetupAPI
 Log Files' /output:<outputFilePath>
```

離線模式範例

從EBS磁碟區收集所有可用的記錄檔。磁碟區依 device_id 值指定。

```
EC2RescueCmd /accepteula /offline:xvdf /collect:all /output:<outputFilePath>
```

只收集特定的日誌群組：

```
EC2RescueCmd /accepteula /offline:xvdf /collect:ec2config /output:<outputFilePath>
```

救援動作

EC2RescueWindows 服務器可以檢測並解決以下服務設置的問題：

服務群組	可用動作	描述
all		
system-time	'RealTimeIsUniversal'	系統時間 • RealTimeisUniversal-偵測RealTimeisUniversal 登錄機碼是否已啟用。如果停用，當時區設定為以外

服務群組	可用動作	描述
		UTC的值時，Windows 系統時間會漂移。
`firewall`	• `'Domain networks'` • `'Private networks'` • `'Guest or public networks'`	Windows 防火牆 • Domain networks (網域網路) - 偵測此 Windows 防火牆描述檔為啟用或停用。 • Private networks (私有網路) - 偵測此 Windows 防火牆描述檔為啟用或停用。 • Guest or public networks (訪客或公有網路) - 偵測此 Windows 防火牆描述檔為啟用或停用。
rdp	• `'Service Start'` • `'Remote Desktop Connections'` • `'TCP Port'`	遠端桌面 • Service Start (服務啟動) - 偵測是否啟用遠端桌面服務。 • Remote Desktop Connections (遠端桌面連線) - 偵測是否啟用此服務。 • TCP連接埠-偵測遠端桌面服務正在偵聽的連接埠。
`ec2config`	• `'Service Start'` • `'Ec2SetPassword'` • `'Ec2HandleUserData'`	EC2Config • 服務啟動-偵測EC2Config服務是否已啟用。 • EC2 SetPassword-產生新的管理員密碼。 • EC2 HandleUserData-允許您在下次啟動實例時運行用戶數據腳本。

服務群組	可用動作	描述
ec2launch	'Reset Administrator Password'	產生新的 Windows 管理員密碼。
network	'DHCP Service Startup'	網路介面 • DHCP服務啟動-偵測DHCP服務是否已啟用。

以下為可用的選項：

- /level:<level> - 動作應該觸發的檢查層級選用屬性。允許的值為：information、warning、error、all。預設會設定為 error。
- /check-only - 會產生報告但不修改離線磁碟區的選用屬性。

> **Note**
>
> 如果 EC2Rescue Windows Server 偵測到可能的磁碟簽章衝突，它會在預設情況下完成離線程序後更正簽章，即使您使用該/check-only選項也是如此。您必須使用/no-fix-signature此選項來防止修正。

- /no-offline - 在動作完成後防止將磁碟區設為離線的選用屬性。
- /no-fix-signature-選用屬性，在完成動作後不會修正可能的磁碟簽章衝突。

救援範例

以下是使用 Windows 伺服器的EC2Rescue範例CLI。磁碟區使用 device_id 值指定。

嘗試修正磁碟區中找到的所有問題：

```
EC2RescueCmd /accepteula /offline:xvdf /rescue:all
```

嘗試修正磁碟區服務群組內的所有問題：

```
EC2RescueCmd /accepteula /offline:xvdf /rescue:firewall
```

嘗試修正磁碟區服務群組內的特定項目：

```
EC2RescueCmd /accepteula /offline:xvdf /rescue:rdp.'Service Start'
```

指定磁碟區中嘗試修正的多個問題：

```
EC2RescueCmd /accepteula /offline:xvdf /rescue:'system-
time.RealTimeIsUniversal,ec2config.Service Start'
```

還原動作

EC2RescueWindows 服務器可以檢測並解決以下服務設置的問題：

服務群組	可用的動作	描述
還原上一個已知的良好組態	`lkgc`	Last Known Good Configuration (上一個已知的良好組態) - 嘗試將執行個體開機為上一個已知的可開機狀態。
從最新的備份還原 Windows 登錄	`regback`	Restore registry from backup (從備份還原登錄) - 從 `\Windows\System32\config\RegBack` 還原登錄。

以下為可用的選項：

- /no-offline—在動作完成後防止將磁碟區設為離線的選用屬性。
- /no-fix-signature-可選屬性，不會在完成動作後修復可能的磁碟簽章衝突。

還原範例

以下是使用 Windows 伺服器的EC2Rescue範例CLI。磁碟區使用 device_id 值指定。

還原磁碟區中上一個已知的良好組態：

```
EC2RescueCmd /accepteula /offline:xvdf /restore:lkgc
```

在磁碟區上還原上一個 Windows 登錄備份：

```
EC2RescueCmd /accepteula /offline:xvdf /restore:regback
```

使用EC2Rescue和 Systems Manager 對受損的 Windows 執行個體

AWS Support 提供「Systems Manager 執行命令」文件，以連接啟用系統管理員的執行個體，以便在 Windows Server 上執EC2Rescue行。執行命令文件稱為 `AWSSupport-RunEC2RescueForWindowsTool`。

此 Systems Manager 執行命令文件會執行下列任務：

- 下載並驗證EC2Rescue視窗伺服器。
- 匯入 PowerShell 模組以簡化您與工具的互動。
- EC2RescueCmd使用提供的命令和參數運行。

Systems Manager 執行命令文件接受三個參數：

- 命令-EC2Rescue 用於 Windows 伺服器動作。目前允許的值為：
  - ResetAccess重設本機管理員密碼。會重設目前執行個體的本機管理員密碼，隨機產生的密碼則以 `/EC2Rescue/Password/<INSTANCE_ID>` 安全存放在參數存放區中。如果您選取此動作且不提供任何參數，則會使用預設KMS金鑰自動加密密碼。或者，您可以在參數中指定KMS金鑰 ID，以您自己的金鑰加密密碼。
  - CollectLogs使用此`/collect:all`動作EC2Rescue針對 Windows 伺服器執行。如果您選取此動作，`Parameters` 必須包含要將日誌上傳到該處的 Amazon S3 儲存貯體名稱。
  - FixAll使用此`/rescue:all`動作EC2Rescue針對 Windows 伺服器執行。如果您選取此動作，`Parameters` 必須包含要救援的區塊型儲存設備名稱。
- 參數-要為指定指令傳遞的 PowerShell 參數。

> **Note**
>
> 為了使ResetAccess動作正常運作，您的 Amazon EC2 執行個體需要附加下列政策，才能將加密的密碼寫入參數存放區。將此原則附加到相關IAM角色之後，請稍候幾分鐘，然後再嘗試重設執行個體的密碼。
> 使用預設KMS金鑰：
>
> ```
> {
>  "Version": "2012-10-17",
> ```

```
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ssm:PutParameter"
      ],
      "Resource": [
        "arn:aws:ssm:region:account_id:parameter/EC2Rescue/
Passwords/<instanceid>"
      ]
    }
  ]
}
```

使用自定義KMS密鑰：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ssm:PutParameter"
      ],
      "Resource": [
        "arn:aws:ssm:region:account_id:parameter/EC2Rescue/
Passwords/<instanceid>"
        ]
      },
      {
      "Effect": "Allow",
      "Action": [
      "kms:Encrypt"
      ],
      "Resource": [
        "arn:aws:kms:region:account_id:key/<kmskeyid>"
      ]
    }
  ]
}
```

下列程序說明如何在 JSON Amazon EC2 主控台中檢視此文件。

若要檢視JSON系 Systems Manager 執行命令文件

1. 在開啟 AWS Systems Manager 主控台https://console.aws.amazon.com/systems-manager/。
2. 在導覽窗格中，展開 Shared Services (系統管理員共享服務) 並選擇 Documents (文件)。
3. 在搜尋列中，將 Owner (擁有者) 設定為 Owned by Me or Amazon (由我或 Amazon 所有) 並將 Document name prefix (文件名稱前綴) 設定為 `AWSSupport-RunEC2RescueForWindowsTool`。
4. 選取`AWSSupport-RunEC2RescueForWindowsTool`文件，選擇「內容」，然後檢視JSON。

範例

以下是如何使用 Systems Manager 執行命令文件來執行 Windows 伺服器EC2Rescue的一些範例，使用 AWS CLI. 若要取得有關使用傳送指令的更多資訊 AWS CLI，請參閱《指AWS CLI 令參考》。

嘗試修正離線根磁碟區內的所有已識別問題

嘗試修正連接到 Amazon EC2 Windows 執行個體的離線根磁碟區上所有已識別的問題：

```
aws ssm send-command --instance-ids "i-0cb2b964d3e14fd9f" --document-name "AWSSupport-
RunEC2RescueForWindowsTool" --comment "EC2Rescue offline volume xvdf" --parameters
 "Command=FixAll, Parameters='xvdf'" --output text
```

從目前的 Amazon EC2 視窗執行個體收集日誌

從目前線上 Amazon EC2 Windows 執行個體收集所有日誌，並將其上傳到 Amazon S3 儲存貯體：

```
aws ssm send-command --instance-ids "i-0cb2b964d3e14fd9f" --document-name "AWSSupport-
RunEC2RescueForWindowsTool" --comment "EC2Rescue online log collection to S3" --
parameters "Command=CollectLogs, Parameters='YOURS3BUCKETNAME'" --output text
```

從離線的 Amazon EC2 Windows 執行個體磁碟區收集日誌

從連接到 Amazon EC2 Windows 執行個體的離線磁碟區收集所有日誌，並使用預先簽署URL的方式將它們上傳到 Amazon S3：

```
aws ssm send-command --instance-ids "i-0cb2b964d3e14fd9f" --document-name "AWSSupport-
RunEC2RescueForWindowsTool" --comment "EC2Rescue offline log collection to S3"
```

```
 --parameters "Command=CollectLogs, Parameters=\"-Offline -BlockDeviceName xvdf -
S3PreSignedUrl 'YOURS3PRESIGNEDURL'\"" --output text
```

重設本機管理員密碼

下列範例示範您可用以重設本機管理員密碼的方法。輸出提供參數存放區的連結，您可以在其中找到隨機產生的安全密碼，然後以本機管理員身分用於 Amazon EC2 Windows 執行個體。RDP

使用預設的 AWS KMS key 別名/aws/ssm 重設線上執行個體的本機管理員密碼：

```
aws ssm send-command --instance-ids "i-0cb2b964d3e14fd9f" --document-name "AWSSupport-
RunEC2RescueForWindowsTool" --comment "EC2Rescue online password reset" --parameters
 "Command=ResetAccess" --output text
```

使用金鑰重設線上執行個體的本機管理員密KMS碼：

```
aws ssm send-command --instance-ids "i-0cb2b964d3e14fd9f" --document-name "AWSSupport-
RunEC2RescueForWindowsTool" --comment "EC2Rescue online password reset" --parameters
 "Command=ResetAccess, Parameters=a133dc3c-a2g4-4fc6-a873-6c0720104bf0" --output text
```

> **Note**
>
> 在此範例中，索KMS引鍵為a133dc3c-a2g4-4fc6-a873-6c0720104bf0。

EC2實例的序列主控台

透過EC2序列主控台，您可以存取 Amazon EC2 執行個體的序列埠，用於疑難排解開機、網路組態和其他問題。序列主控台不需要您的執行個體具有任何網路功能。使用序列主控台，您可以為執行個體輸入命令，就好像鍵盤和監視器直接連接到執行個體的序列埠一樣。序列主控台工作階段在執行個體重新開機和停止期間會持續。在重新開機期間，您可以從一開始檢視所有的開機訊息。

依預設，無法存取序列主控台。您的組織必須授與序列主控台的帳戶存取權，並設定IAM原則，以授與使用者存取序列主控台。序列主控台存取可透過使用執行個體IDs、資源標籤和其他IAM槓桿，在細微層級控制。如需詳細資訊，請參閱設定EC2序列主控台的存取權。

序列主控台可以使用主EC2控台或 AWS CLI.

序列主控台可供免費使用。

主題

- EC2序列主控台的先決條件
- 設定EC2序列主控台的存取權
- Connect 至EC2序列主控台
- 中斷與EC2序列主控台的連線
- 使用EC2序列主控台疑難排解 Amazon EC2 執行個體

EC2序列主控台的先決條件

若要連線到EC2序列主控台並使用您選擇的工具進行疑難排解，必須符合下列先決條件：

- AWS 區域
- Wavelength 區和 AWS Outposts
- 本機區域
- 執行個體類型
- 授與存取權
- 支援瀏覽器型用戶端
- 執行個體狀態
- Amazon EC2 Systems Manager
- 設定所選的疑難排解工具

AWS 區域

全部支持 AWS 區域。

Wavelength 區和 AWS Outposts

不支援。

本機區域

所有本地區域皆支援。

執行個體類型

支援的執行個體類型：

- Linux

- 在 Nitro 系統上建置的所有虛擬化執行個體。
- 所有裸機執行個體，除了：
  - 一般用途：`a1.metal`、`mac1.metal`、`mac2.metal`
  - 加速運算：`g5g.metal`
  - 記憶體最佳化：`u-6tb1.metal`、`u-9tb1.metal`、`u-12tb1.metal`、`u-18tb1.metal`、`u-24tb1.metal`
- Windows

  在 Nitro 系統上建置的所有虛擬化執行個體。在裸機執行個體上不受支援。

授與存取權

您必須完成組態工作，才能授與EC2序列主控台的存取權。如需詳細資訊，請參閱設定EC2序列主控台的存取權。

支援瀏覽器型用戶端

若要使用以瀏覽器為基礎的用戶端連線到序列主控台，您的瀏覽器必須支援 WebSocket。如果您的瀏覽器不支援 WebSocket，請使用您自己的金鑰和用戶SSH端連線到序列主控台。

執行個體狀態

必須為 `running`。

若執行個體處於 `pending`、`stopping`、`stopped`、`shutting-down`，或 `terminated` 狀態，將無法連線到序列主控台。

如需執行個體狀態的詳細資訊，請參閱 Amazon EC2 實例狀態更改。

Amazon EC2 Systems Manager

如果執行個體使用 Amazon EC2 Systems Manager，則必須在執行個體上安裝SSM代理程式版本 3.0.854.0 或更新版本。如需有關SSM代理程式的資訊，請參閱使用指南中的AWS Systems Manager使用SSM代理程式。

設定所選的疑難排解工具

若要透過序列主控台疑難排解執行個體，您可以 SysRq 在 Linux 執行個體上使用GRUB或，並在 Windows 執行個體上使用特殊管理主控台 (SAC)。在使用這些工具之前，您必須先對要使用工具所在的每個執行個體執行設定步驟。

使用執行個體作業系統的指示，設定您選擇的疑難排解工具。

(Linux 執行個體) 設定 GRUB

若要設定GRUB，請根據用來啟動執行個體的程序AMI，選擇下列其中一個程序。

Amazon Linux 2

若要GRUB在 Amazon Linux 2 執行個體上進行設定

1. 使用 Connect 到您的 Linux 執行個體 SSH
2. 在 `/etc/default/grub` 中新增或變更下列選項：
   - 設定 `GRUB_TIMEOUT=1`。
   - 新增 `GRUB_TERMINAL="console serial"`。
   - 新增 `GRUB_SERIAL_COMMAND="serial --speed=115200"`。

   以下是 `/etc/default/grub` 的範例。您可能需要根據系統設定來變更組態。

```
GRUB_CMDLINE_LINUX_DEFAULT="console=tty0 console=ttyS0,115200n8 net.ifnames=0
 biosdevname=0 nvme_core.io_timeout=4294967295 rd.emergency=poweroff rd.shell=0"
GRUB_TIMEOUT=1
GRUB_DISABLE_RECOVERY="true"
GRUB_TERMINAL="console serial"
GRUB_SERIAL_COMMAND="serial --speed=115200"
```

3. 執行下列命令來套用更新的組態。

```
[ec2-user ~]$ sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

Ubuntu

若要在 Ubuntu 執GRUB行個體上設定

1. 連線到您的執行個體。
2. 在 `/etc/default/grub.d/50-cloudimg-settings.cfg` 中新增或變更下列選項：
   - 設定 `GRUB_TIMEOUT=1`。
   - 新增 `GRUB_TIMEOUT_STYLE=menu`。

- 新增 GRUB_TERMINAL="console serial"。
- 移除 GRUB_HIDDEN_TIMEOUT。
- 新增 GRUB_SERIAL_COMMAND="serial --speed=115200"。

以下是 /etc/default/grub.d/50-cloudimg-settings.cfg 的範例。您可能需要根據系統設定來變更組態。

```
# Cloud Image specific Grub settings for Generic Cloud Images
# CLOUD_IMG: This file was created/modified by the Cloud Image build process

# Set the recordfail timeout
GRUB_RECORDFAIL_TIMEOUT=0

# Do not wait on grub prompt
GRUB_TIMEOUT=1
GRUB_TIMEOUT_STYLE=menu

# Set the default commandline
GRUB_CMDLINE_LINUX_DEFAULT="console=tty1 console=ttyS0
 nvme_core.io_timeout=4294967295"

# Set the grub console type
GRUB_TERMINAL="console serial"
GRUB_SERIAL_COMMAND="serial --speed 115200"
```

3. 執行下列命令來套用更新的組態。

```
[ec2-user ~]$ sudo update-grub
```

RHEL

若要設定執GRUB行個RHEL體

1. 連線到您的執行個體。
2. 在 /etc/default/grub 中新增或變更下列選項：
   - 移除 GRUB_TERMINAL_OUTPUT。
   - 新增 GRUB_TERMINAL="console serial"。

- 新增 `GRUB_SERIAL_COMMAND="serial --speed=115200"`。

以下是 `/etc/default/grub` 的範例。您可能需要根據系統設定來變更組態。

```
GRUB_TIMEOUT=1
GRUB_DISTRIBUTOR="$(sed 's, release .*$,,g' /etc/system-release)"
GRUB_DEFAULT=saved
GRUB_DISABLE_SUBMENU=true
GRUB_CMDLINE_LINUX="console=tty0 console=ttyS0,115200n8 net.ifnames=0
 rd.blacklist=nouveau nvme_core.io_timeout=4294967295 crashkernel=auto"
GRUB_DISABLE_RECOVERY="true"
GRUB_ENABLE_BLSCFG=true
GRUB_TERMINAL="console serial"
GRUB_SERIAL_COMMAND="serial --speed=115200"
```

3. 執行下列命令來套用更新的組態。

```
[ec2-user ~]$ sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

CentOS

對於使用 CentOS 啟動的執行個體AMI，GRUB預設會為序列主控台設定。

以下是 `/etc/default/grub` 的範例。您的組態可能會視您的系統設定而有所不同。

```
GRUB_TIMEOUT=1
GRUB_DISTRIBUTOR="$(sed 's, release .*$,,g' /etc/system-release)"
GRUB_DEFAULT=saved
GRUB_DISABLE_SUBMENU=true
GRUB_TERMINAL="serial console"
GRUB_SERIAL_COMMAND="serial --speed=115200"
GRUB_CMDLINE_LINUX="console=tty0 crashkernel=auto console=ttyS0,115200"
GRUB_DISABLE_RECOVERY="true"
```

(Linux 執行個體) 設定 SysRq

若要設定 SysRq，請啟用目前開機週期的 SysRq 指令。若要使組態持續存在，您也可以為後續開機啟用這些 SysRq 指令。

啟用目前開機週期的所有 SysRq 指令

1. 連線到您的執行個體。
2. 執行下列命令。

```
[ec2-user ~]$ sudo sysctl -w kernel.sysrq=1
```

> **Note**
> 此設定會在下次重新開機時清除。

為後續啟動啟用所有 SysRq 指令的步驟

1. 建立文件 `/etc/sysctl.d/99-sysrq.conf` 並在您最喜歡的編輯器中開啟。

```
[ec2-user ~]$ sudo vi /etc/sysctl.d/99-sysrq.conf
```

2. 新增以下這一行。

```
kernel.sysrq=1
```

3. 重新啟動執行個體以套用變更。

```
[ec2-user ~]$ sudo reboot
```

4. 出現`login`提示時，輸入您先前設定之密碼型使用者的使用者名稱，然後按 Enter。
5. 出現 `Password` 提示時，輸入密碼，然後按 Enter。

(視窗執行個體) 啟用SAC和開機功能表

> **Note**
> 如果您在執行個體SAC上啟用，依賴密碼擷取的EC2服務將無法從 Amazon EC2 主控台運作。Amazon 上的 Windows EC2 啟動代理程式 (EC2Config、EC2Launch v1 和 EC2Launch v2) 依賴序列主控台來執行各種任務。當您在執行個體SAC上啟用時，這些工作無法順利執行。如需有關 Amazon EC2 啟動代理程式上 Windows 的詳細資訊，請參閱the section called

"設定執行個體"。如果啟用SAC，您可以稍後將其停用。如需詳細資訊，請參閱禁用SAC和啟動菜單。

使用下列其中一種方法在執行個體上啟用SAC和開機功能表。

PowerShell

在 Windows 實例上啟用SAC和啟動菜單

1. Connect 線至執行個體，並從提高權限的 PowerShell 命令列執行下列步驟。
2. 啟用SAC。

```
bcdedit /ems '{current}' on
bcdedit /emssettings EMSPORT:1 EMSBAUDRATE:115200
```

3. 啟用開機功能表。

```
bcdedit /set '{bootmgr}' displaybootmenu yes
bcdedit /set '{bootmgr}' timeout 15
bcdedit /set '{bootmgr}' bootems yes
```

4. 透過重新啟動執行個體來套用更新的組態。

```
shutdown -r -t 0
```

Command prompt

在 Windows 實例上啟用SAC和啟動菜單

1. 連接到您的執行個體，並從命令提示字元執行下列步驟。
2. 啟用SAC。

```
bcdedit /ems {current} on
bcdedit /emssettings EMSPORT:1 EMSBAUDRATE:115200
```

3. 啟用開機功能表。

```
bcdedit /set {bootmgr} displaybootmenu yes
bcdedit /set {bootmgr} timeout 15
```

```
bcdedit /set {bootmgr} bootems yes
```

4. 透過重新啟動執行個體來套用更新的組態。

```
shutdown -r -t 0
```

設定EC2序列主控台的存取權

若要設定序列主控台的存取權，您必須在帳戶層級授與序列主控台存取權，然後設定IAM原則以授與使用者存取權。對於 Linux 執行個體，您還必須在每個執行個體上設定以密碼為基礎的使用者，以便使用者可以使用序列主控台進行疑難排解。

在開始之前，請務必檢查先決條件。

主題

- EC2序列主控台的存取層級
- 管理EC2序列主控台的帳戶存取權
- 設IAM定EC2序列主控台存取的原則
- 在 Linux 執行個體上設定作業系統使用者密碼

EC2序列主控台的存取層級

依預設，帳戶層級無法存取序列主控台。您需要在帳戶層級明確授予對序列主控台的存取權。如需詳細資訊，請參閱管理EC2序列主控台的帳戶存取權。

您可以使用服務控制原則 (SCP) 來允許存取組織內的序列主控台。然後，您可以使用IAM政策來控制存取，在使用者層級擁有精細的存取控制。透過使用SCP和IAM原則的組合，您可以對序列主控台擁有不同層級的存取控制。

組織層級

您可以使用服務控制策略 (SCP) 來允許存取組織內成員帳戶的序列主控台。如需有關的詳細資訊 SCPs，請參閱AWS Organizations 使用指南中的服務控制策略。

執行個體層級

您可以使用IAM PrincipalTag 和 ResourceTag 建構，並透過 ID 指定執行個體，來設定序列主控台存取原則。如需詳細資訊，請參閱設IAM定EC2序列主控台存取的原則。

使用者層級

您可以將IAM政策設定為允許或拒絕指定使用者將SSH公開金鑰推送至特定執行個體之序列主控台服務的權限，以便在使用者層級設定存取權。如需詳細資訊，請參閱設IAM定EC2序列主控台存取的原則。

作業系統層級 (僅限 Linux 執行個體)

您可以在客體作業系統層級設定使用者密碼。在某些使用案例下，這可讓您存取序列主控台。不過，若要監控日誌，您不需要以密碼為基礎的使用者。如需詳細資訊，請參閱在 Linux 執行個體上設定作業系統使用者密碼。

管理EC2序列主控台的帳戶存取權

依預設，帳戶層級無法存取序列主控台。您需要在帳戶層級明確授予對序列主控台的存取權。

主題

- 授予使用者管理帳戶存取的許可
- 檢視序列主控台的帳戶存取狀態
- 授與序列主控台的帳戶存取權
- 拒絕帳戶存取序列主控台

授予使用者管理帳戶存取的許可

若要允許使用者管理EC2序列主控台的帳戶存取權，您必須授予他們必要的IAM權限。

下列策略授與檢視帳號狀態的權限，以及允許和阻止帳號存取EC2序列主控台。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:GetSerialConsoleAccessStatus",
                "ec2:EnableSerialConsoleAccess",
                "ec2:DisableSerialConsoleAccess"
            ],
            "Resource": "*"
```

```
        }
    ]
}
```

如需詳細資訊，請參閱《使用指南》中的IAM〈建立IAM策略〉。

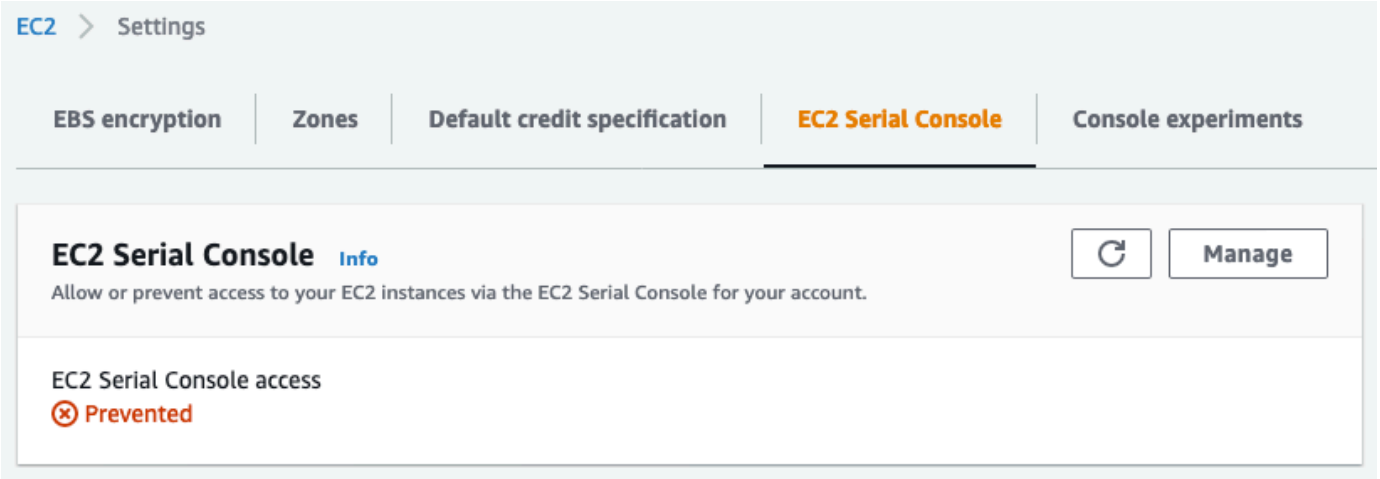
檢視序列主控台的帳戶存取狀態

檢視序列主控台 (主控台) 的帳戶存取狀態

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側導覽窗格中，選擇 [EC2儀表板]。
3. 在帳號屬性中，選擇EC2序列主控台。

   「EC2序列主控台存取」欄位會指出「允許」還是「禁止」帳戶存取。

   下列螢幕擷取畫面顯示帳戶無法使用EC2序列主控台。



檢視序列主控台的帳戶存取狀態 (AWS CLI)

使用 get-serial-console-access-status 指令可檢視序列主控台的帳戶存取狀態。

```
aws ec2 get-serial-console-access-status --region us-east-1
```

在下面的輸出中，`true` 表示允許該帳戶存取序列控制台。

```
{
    "SerialConsoleAccessEnabled": true
```

```
}
```

授與序列主控台的帳戶存取權

授與序列主控台 (主控台) 的帳戶存取權

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側導覽窗格中，選擇 [EC2儀表板]。
3. 在帳號屬性中，選擇EC2序列主控台。
4. 選擇管理。
5. 若要允許存取帳戶中所有執行個體的EC2序列主控台，請選取 [允許] 核取方塊。
6. 選擇 Update (更新)。

授與帳戶存取序列主控台 (AWS CLI)

使用指enable-serial-console-access令可允許帳戶存取序列主控台。

```
aws ec2 enable-serial-console-access --region us-east-1
```

在下面的輸出中，`true` 表示允許該帳戶存取序列控制台。

```
{
    "SerialConsoleAccessEnabled": true
}
```

拒絕帳戶存取序列主控台

拒絕帳戶存取序列主控台 (主控台)

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在左側導覽窗格中，選擇 [EC2儀表板]。
3. 在帳號屬性中，選擇EC2序列主控台。
4. 選擇管理。
5. 若要防止存取帳戶中所有執行個體的EC2序列主控台，請清除 [允許] 核取方塊。
6. 選擇 Update (更新)。

拒絕帳戶存取序列主控台 (AWS CLI)

使用此指disable-serial-console-access令可防止帳戶存取序列主控台。

```
aws ec2 disable-serial-console-access --region us-east-1
```

在下面的輸出中，`false` 表示該帳戶被拒絕存取序列控制台。

```
{
    "SerialConsoleAccessEnabled": false
}
```

設IAM定EC2序列主控台存取的原則

依預設，使用者無法存取序列主控台。您的組織必須設定IAM原則，才能授與您的使用者必要的存取權。如需詳細資訊，請參閱《使用指南》中的IAM〈建立IAM策略〉。

對於序列主控台存取，請建立包含`ec2-instance-connect:SendSerialConsoleSSHPublicKey`動作的JSON政策文件。此動作會授予使用者將公有金鑰推送至序列主控台服務的許可，此服務會啟動序列主控台工作階段。建議您限制對特定EC2執行個體的存取權。否則，所有具有此權限的使用者都可以連線到所有EC2執行個體的序列主控台。

IAM原則範例

- 明確允許存取序列主控台
- 明確拒絕存取序列主控台
- 使用資源標籤來控制對序列主控台的存取

明確允許存取序列主控台

依預設，沒有人可以存取序列主控台。若要授與對序列主控台的存取權，您必須設定政策以明確允許存取。建議您設定限制存取特定執行個體的政策。

下列政策允許存取特定執行個體的序列主控台，並依其執行個體 ID 進行識別。

請注意，`DescribeInstances`、`DescribeInstanceTypes`、和 `GetSerialConsoleAccessStatus` 動作不支援資源層級許可，因此為這些動作指定所有資源，以 `*` (星號) 表示。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowDescribeInstances",
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeInstances",
                "ec2:DescribeInstanceTypes",
                "ec2:GetSerialConsoleAccessStatus"
            ],
             "Resource": "*"
        },
        {
            "Sid": "AllowinstanceBasedSerialConsoleAccess",
            "Effect": "Allow",
            "Action": [
                "ec2-instance-connect:SendSerialConsoleSSHPublicKey"
            ],
            "Resource": "arn:aws:ec2:region:account-id:instance/i-0598c7d356eba48d7"
        }
    ]
}
```

明確拒絕存取序列主控台

下列IAM原則可讓您存取以 * (星號) 表示的所有執行個體的序列主控台，並明確拒絕存取特定執行處理的序列主控台 (以其 ID 識別)。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowSerialConsoleAccess",
            "Effect": "Allow",
            "Action": [
                "ec2-instance-connect:SendSerialConsoleSSHPublicKey",
                "ec2:DescribeInstances",
                "ec2:DescribeInstanceTypes",
                "ec2:GetSerialConsoleAccessStatus"
            ],
            "Resource": "*"
        },
```

```
        {
            "Sid": "DenySerialConsoleAccess",
            "Effect": "Deny",
            "Action": [
                "ec2-instance-connect:SendSerialConsoleSSHPublicKey"
            ],
            "Resource": "arn:aws:ec2:region:account-id:instance/i-0598c7d356eba48d7"
        }
    ]
}
```

使用資源標籤來控制對序列主控台的存取

您可以使用資源標籤來控制對執行個體序列主控台的存取。

以屬性為基礎的存取控制是一種授權策略，根據可附加至使用者和 AWS 資源的標籤來定義權限。例如，下列政策允許使用者啟動執行個體的序列主控台連線，只有當該執行個體的資源標籤和主參與者的標籤具有相同的標籤鍵 `SerialConsole` 值時，才能啟動執行個體的序列主控台連線。

如需有關使用標籤來控制 AWS 資源存取權的詳細資訊，請參閱《使IAM用指南》中的〈控制 AWS 資源存取〉。

請注意，`DescribeInstances`、`DescribeInstanceTypes`、和 `GetSerialConsoleAccessStatus` 動作不支援資源層級許可，因此為這些動作指定所有資源，以 `*` (星號) 表示。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowDescribeInstances",
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeInstances",
                "ec2:DescribeInstanceTypes",
                "ec2:GetSerialConsoleAccessStatus"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AllowTagBasedSerialConsoleAccess",
            "Effect": "Allow",
```

```
            "Action": [
                "ec2-instance-connect:SendSerialConsoleSSHPublicKey"
            ],
            "Resource": "arn:aws:ec2:region:account-id:instance/*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/SerialConsole":
 "${aws:PrincipalTag/SerialConsole}"
                }
            }
        }
    ]
}
```

在 Linux 執行個體上設定作業系統使用者密碼

> **Note**
>
> 本節僅適用於 Linux 執行個體。

您無需密碼即可連線到序列主控台。不過，若要使用序列主控台對 Linux 執行個體進行疑難排解，執行個體必須有以密碼為基礎的 OS 使用者。

您可以為任何 OS 使用者 (包括根使用者) 設定密碼。請注意，根使用者可以修改所有檔案，而每個 OS 使用者的許可有限制。

您必須為每個要使用序列主控台的執行個體設定使用者密碼。這是每個執行個體都有的一次性要求。

> **Note**
>
> 下列指示僅適用於您使用 Linux AMI 提供的執行個體啟動， AWS 因為依預設，AMIs提供者並未設定以密碼為基礎的使用者。 AWS 如果您使用已設定 root 使用者密碼AMI的執行個體啟動執行個體，則可以略過這些指示。

若要在 Linux 執行個體上設定作業系統使用者密碼

1. 連線到您的執行個體。您可以使用任何方法連線至執行個體，EC2序列主控台連線方法除外。
2. 若要設定使用者的密碼，請使用 passwd 命令。在下列範例中，使用者是 `root`。

```
[ec2-user ~]$ sudo passwd root
```

下列為範例輸出。

```
Changing password for user root.
New password:
```

3. 出現 New password 提示時，輸入新密碼。
4. 出現提示時，重新輸入密碼。

Connect 至EC2序列主控台

您可以使用 Amazon 主控台或透過連線到EC2執行個體的序列EC2主控台SSH。連線到序列主控台後，您可以使用它來疑難排解開機、網路設定和其他問題。如需疑難排解的詳細資訊，請參閱 使用EC2序列主控台疑難排解 Amazon EC2 執行個體。

考量事項

- 每個執行個體僅支援 1 個使用中的序列主控台連線。
- 除非您終止連線，否則序列主控台連線通常會持續 1 個小時。不過，在系統維護期間，Amazon EC2 會中斷序列主控台工作階段的連線。
- 從序列主控台中斷連線後，需要 30 秒才能縮減工作階段，以便允許新的工作階段。
- 支援的序列主控台連接埠：ttyS0(Linux 執行個體) 和 COM1 (Windows 執行個體)
- 當您連線到序列主控台時，可能會發現執行個體的輸送量略有下降。

主題

- 使用以瀏覽器為基礎的用戶端連接
- 使 Connect 您自己的金鑰和用SSH戶端連線
- EC2序列主控台端點和指紋

使用以瀏覽器為基礎的用戶端連接

您可以使用瀏覽器型用戶端連線至EC2執行個體的序列主控台。您可以在 Amazon EC2 主控台中選取執行個體，然後選擇連線到序列主控台來執行此操作。以瀏覽器為基礎的用戶端會處理權限，並提供成功的連線。

EC2串行控制台適用於大多數瀏覽器，並支持鍵盤和鼠標輸入。

連線之前，請確認您已完成事前準備。

使用以瀏覽器為基礎的用戶端 (Amazon EC2 主控台) 連線至執行個體的序列埠

1. 在打開 Amazon EC2 控制台https://console.aws.amazon.com/ec2/。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取執行個體，然後選擇 [動作]、[監視及疑難排解]、[EC2序列主控台]

   或者，選取執行個體並選擇「Connect」、「EC2序列主控台」、「Connect」

   瀏覽器內的終端機視窗隨即開啟。
4. 按 Enter。如果傳回登入提示，表示您已連線到序列主控台。

   如果螢幕保持黑色，您可以使用下列資訊來協助解決連線到序列主控台的問題：

   - 檢查您是否已設定序列主控台的存取權。如需詳細資訊，請參閱設定EC2序列主控台的存取權。
   - (僅限 Linux 執行個體) 用 SysRq 於連線到序列主控台。 SysRq 不需要您透過以瀏覽器為基礎的用戶端進行連線。如需詳細資訊，請參閱(Linux 執行個體) 用 SysRq 於疑難排解執行個體。
   - (僅限 Linux 執行個體) 重新啟動裝置。如果您有SSH權存取執行個體，請使用連線至執行個體SSH，然後使用下列指令重新啟動 getty。

     ```
     [ec2-user ~]$ sudo systemctl restart serial-getty@ttyS0
     ```
   - 將執行個體重新開機。您可以使用 SysRq (Linux 執行個體)、EC2主控台或 AWS CLI. 如需詳細資訊，請參閱 (Linux 執行個體) 用 SysRq 於疑難排解執行個體 (Linux 執行個體) 或重新啟動您的執行個體。
5. (僅限 Linux 執行個體) 出現**login**提示時，輸入您先前設定之密碼型使用者的使用者名稱，然後按 Enter。
6. (僅限 Linux 執行個體) 出現`Password`提示時，輸入密碼，然後按 Enter 鍵。

您現在已登入執行個體，並可使用序列主控台進行疑難排解。

使 Connect 您自己的金鑰和用SSH戶端連線

使用序列主控台時，您可以使用自己的SSH金鑰，並從您選擇的用SSH戶端連線至執行個體API。這可讓您從發佈公有的金鑰至執行個體的序列主控台功能受益。

連線之前，請確認您已完成事前準備。

若要使用連線至執行個體的序列主控台 SSH

1. 將SSH公開金鑰推送至執行個體以啟動序列主控台工作階段

   使用 send-serial-console-ssh-public-key 指令將您的SSH公開金鑰推送至執行個體。這會啟動序列主控台工作階段。

   如果此執行個體已經啟動序列主控台工作階段，則命令會失敗，因為您一次只能開啟一個工作階段。從序列主控台中斷連線後，需要 30 秒才能縮減工作階段，以便允許新的工作階段。

```
aws ec2-instance-connect send-serial-console-ssh-public-key \
    --instance-id i-001234a4bf70dec41EXAMPLE \
    --serial-port 0 \
    --ssh-public-key file://my_key.pub \
    --region us-east-1
```

2. 使用私有金鑰連線到序列主控台

   在從序列主控台服務中移除公開金鑰之前，請使用 ssh 命令連線到序列主控台。在將其移除之前，您有 60 秒時間。

   使用對應於公開金鑰的私有金鑰。

   使用者名稱格式為`instance-id.port0`，其中包含執行個體 ID 和連接埠 0。在下列範例中，使用者名稱為`i-001234a4bf70dec41EXAMPLE.port0`。

   每個區域的序列主控台服務端點各不相同。請參閱每個區域端點的 EC2序列主控台端點和指紋 資料表。在下列範例中，序列主控台服務位於 *us-east-1* 區域。

```
ssh -i my_key i-001234a4bf70dec41EXAMPLE.port0@serial-console.ec2-instance-
connect.us-east-1.aws
```

3. (選擇性) 驗證指紋

   當您第一次連線到序列主控台時，系統會提示您確認指紋。您可以將序列主控台指紋與顯示進行驗證的指紋進行比較。如果這些指紋不相符，可能是有人試圖攻擊 man-in-the-middle「」。如果它們匹配，您可以放心地連接到序列主控台。

   以下指紋適用於 us-east-1 區域的序列控制台服務。如需每個區域的指紋，請參閱 EC2序列主控台端點和指紋。

```
SHA256:dXwn5ma/xadVMeBZGEru5l2gx+yI5LDiJaLUcz0FMmw
```

> **Note**
> 指紋只會在您第一次連線到序列主控台時出現。

4. 按 Enter。如果傳回提示，表示您已連線到序列主控台。

   如果螢幕保持黑色，您可以使用下列資訊來協助解決連線到序列主控台的問題：

   - 檢查您是否已設定序列主控台的存取權。如需詳細資訊，請參閱設定EC2序列主控台的存取權。
   - (僅限 Linux 執行個體) 用 SysRq 於連線到序列主控台。 SysRq 不需要您通過連接SSH。如需詳細資訊，請參閱(Linux 執行個體) 用 SysRq 於疑難排解執行個體。
   - (僅限 Linux 執行個體) 重新啟動裝置。如果您有SSH權存取執行個體，請使用連線至執行個體 SSH，然後使用下列指令重新啟動 getty。

     ```
     [ec2-user ~]$ sudo systemctl restart serial-getty@ttyS0
     ```

   - 將執行個體重新開機。您可以使用 SysRq (僅限 Linux 執行個體)、EC2主控台或 AWS CLI. 如需詳細資訊，請參閱 (Linux 執行個體) 用 SysRq 於疑難排解執行個體 (僅限 Linux 執行個體) 或重新啟動您的執行個體。
5. (僅限 Linux 執行個體) 出現`login`提示時，輸入您先前設定之密碼型使用者的使用者名稱，然後按 Enter。
6. (僅限 Linux 執行個體) 出現`Password`提示時，輸入密碼，然後按 Enter 鍵。

您現在已登入執行個體，並可使用序列主控台進行疑難排解。

EC2序列主控台端點和指紋

以下是EC2序列主控台的服務端點和指紋。若要以程式設計方式連線至執行個體的序列主控台，請使用EC2序列主控台端點。每個 AWS 區域的EC2序列主控台端點和指紋都是唯一的。

區域名稱	區域	端點	指紋
美國東部 (俄亥俄)	us-east-2	serial-console.ec2-instance-connect.us-east-2.aws	SHA256: EhwPkTzRtTY7TRSzz26XbB

區域名稱	區域	端點	指紋
			0/高V9j 0x瓦特/D/0 RM7mCZN
美國東部 (維吉尼亞北部)	us-east-1	serial-console.ec2-instance-connect.us-east-1.aws	SHA256: dXwn 五五五月 0 xadVMe BZGEru5l2gx LDiJaLUcz FMmw
美國西部 (加利佛尼亞北部)	us-west-1	serial-console.ec2-instance-connect.us-west-1.aws	SHA256:OH ldlcMET8u 7QLSX3jmR TRAPFHVtq byoLZBMUCqiH3Y
美國西部 (奧勒岡)	us-west-2	serial-console.ec2-instance-connect.us-west-2.aws	SHA256:EM CIe23TqKaBI6yGHain qZcMwqNkD hhAVHa1O2JxVUc
非洲 (開普敦)	af-south-1	ec2-serial-console.af-south-1.api.aws	SHA256: RMWWZ2fVe PeJUqzjO5jL2KIgXsc zoHlz21Ed 00 比威
亞太區域 (香港)	ap-east-1	ec2-serial-console.ap-east-1.api.aws	SHA256: 第一季 lpiXxCho ZHplnAkjb P7tkm2xXViC9bJFsjY nifk
亞太區域 (海德拉巴)	ap-south-2	ec2-serial-console.ap-south-2.api.aws	SHA256:WJ gPBSwV4/小姐 + OPITValoe wAuYj15DV W845JEhDKRs

區域名稱	區域	端點	指紋
亞太區域 (雅加達)	ap-southeast-3	ec2-serial-console.ap-southeast-3.api.aws	SHA256：5 英寸 ZwgrCh四氧化四 XlTqL zIfbx bZgs YFqy3o8mIk
亞太區域 (墨爾本)	ap-southeast-4	ec2-serial-console.ap-southeast-4.api.aws	SHA256: 阿瓦奇 5 Z0O7H90p0 hFgLvjn gTSSh GG46wfOeT 6ZJvM
亞太區域 (孟買)	ap-south-1	serial-console.ec2-instance-connect.ap-south-1.aws	SHA256: oBLXc Y mklqHHEbli ARxEgH8IsO51rezTPi SM35BsU4 0
亞太區域 (大阪)	ap-northeast-3	ec2-serial-console.ap-northeast-3.api.aws	SHA256: 上午/9 /v /3 jiBKBn BuFnHr aXsg EV3G8Tu VHFXE
亞太區域 (首爾)	ap-northeast-2	serial-console.ec2-instance-connect.ap-northeast-2.aws	SHA256: 照片 +DZ +G 9 WXNX + uNTztg 聯邦匯報 PK49WYMqBX ZM2dSrqrI
亞太區域 (新加坡)	ap-southeast-1	serial-console.ec2-instance-connect.ap-southeast-1.aws	SHA256: PLFNn7WnC QDHx3qmwL u1Gy /8 TUX7LQgZu aC6L45CoY
亞太區域 (悉尼)	ap-southeast-2	serial-console.ec2-instance-connect.ap-southeast-2.aws	SHA256: yFvMw UK9lEUQjQ TRoXXzuN +CW9/ VSe9W984Cf5Tgzo4

區域名稱	區域	端點	指紋
亞太區域 (東京)	ap-northeast-1	序列控制台 .ec2 實例連接 .ap-東北 -1.aws	SHA256:RQ fsDCZTOfQ awewTRDV1t9Em/ HMrFQe+CRlIOT 5um4k
加拿大 (中部)	ca-central-1	serial-console.ec2-instance-connect.ca-central-1.aws	SHA256: P2O2 兆波 6 jOZwmp YW738FIOT HdUTyEv2g czYMMO7s4
中國 (北京)	cn-north-1	ec2-serial-console .cn-north-1.api.amazonwebservices.com.cn	SHA256：2 gHVFy 4 小時 U3 + 瓦 aFUx D28 伏特/長靴 + Y ggMeqjvSlgngpg
中國 (寧夏)	cn-northwest-1	ec2-serial-console .cn-northwest-1.api.amazonwebservices.com.cn	SHA256: T dgrNZki QOdVfYEBU hO4SzUA VWI5rYOZG TogpwmiM
歐洲 (法蘭克福)	eu-central-1	serial-console.ec2-instance-connect.eu-central-1.aws	SHA256: 一個CMFS/ yIcOdOlkXvOl八點一腳趾 + bBnr JJ3Fy
歐洲 (愛爾蘭)	eu-west-1	serial-console.ec2-instance-connect.eu-west-1.aws	SHA256: H2AA GAWO4Hath htm6ezs3Bj7udgUxi2 qTrHjZAwCW6E
歐洲 (倫敦)	eu-west-2	serial-console.ec2-instance-connect.eu-west-2.aws	SHA256: A69RD5CE/ AEG4Amm53 I6lkD1ZPv SBCV3tTPW2RnJg8

區域名稱	區域	端點	指紋
歐洲 (米蘭)	eu-south-1	ec2-serial-console.eu-south-1.api.aws	SHA256: LC0 kOVJnpg F 0A7N99 yBVrxn 0 ecLb XSX95cuuS7X7QK3
歐洲 (巴黎)	eu-west-3	serial-console.ec2-instance-connect.eu-west-3.aws	SHA256：八方 ldnAf 9 pymeNe 8 nFVng 乙年 3/RPArkxs wJUzfrlxeEWs
歐洲 (西班牙)	eu-south-2	ec2-serial-console.eu-south-2.api.aws	SHA256: 走 CW2DFRlu6 69QNxqFxEcsR6fZUz ZcwoEc
歐洲 (斯德哥爾摩)	eu-north-1	serial-console.ec2-instance-connect.eu-north-1.aws	SHA256: tkGFFUVUD voc 迪 GSS3Cu8Gd l6w2uI32E PNpKFKLwX84
歐洲 (蘇黎世)	eu-central-2	ec2-serial-console.eu-central-2.api.aws	SHA256: 8 件 mBMf 6 WdCw 0 /4O NUlzKfwM4 IfRz aXFut QXWp6mk
以色列 (特拉維夫)	il-central-1	ec2-序列控制台中央-1.api.aws	SHA256:JR 6q8v6kNNP i8+QSFQ4d j5dimNmZP TgwgsM1SNvtYyU
中東 (巴林)	me-south-1	ec2-serial-console .me-south-1.api.aws	SHA256:nP jLLKHu2Qn LdUq2kVAr soK5xvPJO MRJKCBzCDqC3k8

區域名稱	區域	端點	指紋
中東 (UAE)	me-central-1	ec2-serial-console .me-central-1.api.aws	SHA256：百兆平方米 + L0 B4 /X duKi dFwPeyyk MPBYhI zXNe FSDKBvLE
南美洲 (聖保羅)	sa-east-1	序列控制台 .ec2 實例連接 .sa-東 1.aws	SHA256: RD2+32-波特比第六二 O yVIem ENaQzC qAPDq
AWS GovCloud (美國東部)	us-gov-east-1	序列控制台 .ec2-實例連接。 us-gov-east-1. 亞馬遜	SHA256：tI we19 GWsoyLClr tvu38YEEh 歲以上 DHIkqnDcZ nmtebvF28
AWS GovCloud (美國西部)	us-gov-west-1	序列控制台 .ec2-實例連接。 us-gov-west-1. 亞馬遜	SHA256: kfOFRWLa OZfB + 輸出 3 8 8 晚 bRf OIPf GO2YZLqXZ ilw5DQ

中斷與EC2序列主控台的連線

如果您不再需要連線至執行個體的EC2序列主控台，可以中斷連線。當您終止與序列主控台的連線時，執行個體上執行的任何 shell 工作階段都會繼續執行。如果想要結束 shell 工作階段，您必須先將其結束，然後才能終止與序列主控台的連線。

考量事項

- 除非您終止連線，否則序列主控台連線通常會持續 1 個小時。不過，在系統維護期間，Amazon EC2 會中斷序列主控台工作階段的連線。
- 從序列主控台中斷連線後，需要 30 秒才能縮減工作階段，以便允許新的工作階段。

終止與序列主控台連線的方式取決於用戶端。

瀏覽器型用戶端

若要終止與序列主控台的連線，請關閉序列主控台的瀏覽器內終端機視窗。

標準開放SSH用戶端

若要中斷與序列主控台的連線，請使用下列指令來關閉SSH連線。必須在新行之後立即執行此命令。

```
~.
```

您用來關閉SSH連線的命令可能會因您使用的用SSH戶端而有所不同。

使用EC2序列主控台疑難排解 Amazon EC2 執行個體

透過使用EC2序列主控台，您可以透過連線至執行個體的序列埠來疑難排解開機、網路設定和其他問題。

請使用執行個體作業系統以及您在執行個體上設定之工具的指示。

> **Note**
> 開始之前，請確定您已完成先決條件，包括設定您選擇的疑難排解工具。

(Linux 執行個體) 用GRUB於疑難排解執行個體

GNUGRUB（GNUGRand統一引導加載程序的簡稱，通常稱為GRUB）是大多數 Linux 操作系統的默認引導加載程序。您可以從GRUB功能表中選擇要開機的核心，或修改功能表項目以變更核心的開機方式。這在疑難排解失敗的執行個體時非常有用。

此GRUB功能表會在開機程序期間顯示。此功能表無法透過正常存取SSH，但您可以透過EC2序列主控台存取該功能表。

您可以開機進入單一使用者模式或緊急模式。單一使用者模式會以較低的執行層級啟動核心。例如，它可能掛載檔案系統，但不會啟動網路，讓您有機會執行修復執行個體所需的維護。緊急模式與單一使用者模式類似，不同之處在於核心以最低的執行層級執行。

開機至單一使用者模式

1. 連線到執行個體的序列主控台。
2. 使用下列命令重新啟動執行個體。

```
[ec2-user ~]$ sudo reboot
```

3. 在重新啟動期間，當GRUB菜單出現時，按任意鍵停止引導過程。
4. 在GRUB菜單中，使用箭頭鍵選擇要啟動的內核，然後按鍵盤e上的。
5. 使用方向鍵將游標定位在包含核心的行上。該行以`linux`或開頭，`linux16`視用於啟動執行個體的項目而定。AMI對於 Ubuntu，兩行以 `linux` 開頭，必須在下一個步驟中修改這兩行。
6. 在該行的末尾，添加單詞 `single`。

   以下是 Amazon Linux 2 的範例。

   ```
   linux /boot/vmlinuz-4.14.193-149.317.amzn2.aarch64 root=UUID=d33f9c9a-\
   dadd-4499-938d-ebbf42c3e499 ro  console=tty0 console=ttyS0,115200n8 net.ifname\
   s=0 biosdevname=0 nvme_core.io_timeout=4294967295 rd.emergency=poweroff rd.she\
   ll=0 single
   ```

7. 按 Ctrl+X 以開機進入單一使用者模式。
8. 出現`login`提示時，輸入您先前設定之密碼型使用者的使用者名稱，然後按 Enter。
9. 出現 `Password` 提示時，輸入密碼，然後按 Enter。

開機進入緊急模式

按照與單用戶模式相同的步驟進行操作，但是在步驟 6 中，添加單詞`emergency`而不是`single`。

(Linux 執行個體) 用 SysRq 於疑難排解執行個體

System Request (SysRq) 密鑰有時也被稱為 SysRq「魔術」，可以用來在 shell 之外直接向內核發送命令，無論內核在做什麼，內核都會響應。例如，如果執行個體已停止回應，您可以使用 SysRq 金鑰告知核心損毀或重新啟動。有關更多信息，請參見維基百科中的魔術 SysRq 密鑰。

您可以在EC2序列主控台瀏覽器型用戶端或用戶端中使用 SysRq 指令SSH。每個用戶端傳送中斷請求的命令是不同的。

若要使用 SysRq，請根據您使用的用戶端選擇下列其中一個程序。

Browser-based client

SysRq 在序列主控台瀏覽器型用戶端中使用

1. 連線到執行個體的序列主控台。

2. 若要傳送中斷請求，請按 CTRL+0 (零)。如果您的鍵盤支援，您也可以使用「暫停」或「Break」鍵來傳送中斷請求。

```
[ec2-user ~]$ CTRL+0
```

3. 若要發出 SysRq 指令，請按鍵盤上與所需指令相對應的按鍵。例如，若要顯示指 SysRq 令清單，請按h。

```
[ec2-user ~]$ h
```

h 命令會輸出類似下列內容。

```
[ 1169.389495] sysrq: HELP : loglevel(0-9) reboot(b) crash(c) terminate-all-
tasks(e) memory-full-oom-kill(f) kill-all-tasks(i) thaw-filesystems
(j) sak(k) show-backtrace-all-active-cpus(l) show-memory-usage(m) nice-all-RT-
tasks(n) poweroff(o) show-registers(p) show-all-timers(q) unraw(r
) sync(s) show-task-states(t) unmount(u) show-blocked-tasks(w) dump-ftrace-
buffer(z)
```

SSH client

若要 SysRq 在用SSH戶端中使用

1. 連線到執行個體的序列主控台。
2. 若要傳送中斷請求，請按下 ~B (波狀符號，後面接著大寫 B)。

```
[ec2-user ~]$ ~B
```

3. 若要發出 SysRq 指令，請按鍵盤上與所需指令相對應的按鍵。例如，若要顯示指 SysRq 令清單，請按h。

```
[ec2-user ~]$ h
```

h 命令會輸出類似下列內容。

```
[ 1169.389495] sysrq: HELP : loglevel(0-9) reboot(b) crash(c) terminate-all-
tasks(e) memory-full-oom-kill(f) kill-all-tasks(i) thaw-filesystems
(j) sak(k) show-backtrace-all-active-cpus(l) show-memory-usage(m) nice-all-RT-
tasks(n) poweroff(o) show-registers(p) show-all-timers(q) unraw(r
```

```
) sync(s) show-task-states(t) unmount(u) show-blocked-tasks(w) dump-ftrace-
buffer(z)
```

> **Note**
>
> 您用來傳送休息要求的命令可能會因您使用的用SSH戶端而有所不同。

(Windows 執行個體) 用SAC於疑難排解執行個體

Windows 的特殊管理控制台 (SAC) 功能提供了一種對 Windows 執行個體進行疑難排解的方法。透過連線至執行個體的序列主控台並使用SAC，您可以中斷開機程序並以安全模式啟動 Windows。

> **Note**
>
> 如果您在執行個體SAC上啟用，依賴密碼擷取的EC2服務將無法從 Amazon EC2 主控台運作。Amazon 上的 Windows EC2 啟動代理程式 (EC2Config、EC2Launch v1 和 EC2Launch v2) 依賴序列主控台來執行各種任務。當您在執行個體SAC上啟用時，這些工作無法順利執行。如需有關 Amazon EC2 啟動代理程式上 Windows 的詳細資訊，請參閱the section called “設定執行個體”。如果啟用SAC，您可以稍後將其停用。如需詳細資訊，請參閱禁用SAC和啟動菜單。

主題

- 使用 SAC
- 使用開機功能表
- 禁用SAC和啟動菜單

使用 SAC

若要使用 SAC

1. 連接至序列主控台。

   如果SAC在執行個體上啟用，序列主控台會顯示SAC>提示。

```
Computer is booting, SAC started and initialized.

Use the "ch -?" command for information about using channels.
Use the "?" command for general help.


SAC>?
EVENT: The CMD command is now available.
SAC
```

2. 若要顯示SAC指令，請輸入**?**，然後按 Enter。

   預期的輸出結果

```
SAC>?
ch                    Channel management commands.  Use ch -? for more help.
cmd                   Create a Command Prompt channel.
d                     Dump the current kernel log.
f                     Toggle detailed or abbreviated tlist info.
? or help             Display this list.
i                     List all IP network numbers and their IP addresses.
i <#> <ip> <subnet> <gateway> Set IPv4 addr., subnet and gateway.
id                    Display the computer identification information.
k <pid>               Kill the given process.
l <pid>               Lower the priority of a process to the lowest possible.
lock                  Lock access to Command Prompt channels.
m <pid> <MB-allow>    Limit the memory usage of a process to <MB-allow>.
p                     Toggle paging the display.
r <pid>               Raise the priority of a process by one.
s                     Display the current time and date (24 hour clock used).
s mm/dd/yyyy hh:mm    Set the current time and date (24 hour clock used).
t                     Tlist.
restart               Restart the system immediately.
shutdown              Shutdown the system immediately.
crashdump             Crash the system. You must have crash dump enabled.
```

3. 若要建立命令提示通道 (例如 `cmd0001` 或 `cmd0002`)，請輸入 **`cmd`**，然後按 Enter。
4. 若要檢視命令提示字元頻道，請按 ESC，然後按下TAB。

   預期的輸出結果

```
Name:                  Cmd0001
Description:           Command
Type:                  VT-UTF8
Channel GUID:          ef9f20a0-1287-11eb-82b0-0e4ba51872e5
Application Type GUID: 63d02271-8aa4-11d5-bccf-00b0d014a2d0

Press <esc><tab> for next channel.
Press <esc><tab>0 to return to the SAC channel.
Use any other key to view this channel.
```

5. 要切換頻道，請同時按 ESC+ TAB + 頻道號碼。例如，要切換到cmd0002頻道（如果已經創建），請按 ESC+ TAB +2。
6. 輸入命令提示通道所需的憑證。

```
Please enter login credentials.
Username: Administrator
Domain  : .
Password: *******
```

命令提示字元與您在桌面上取得的完整功能命令 Shell 相同，但是除了它，不允許讀取已經輸出的字元。

```
Microsoft Windows [Version 10.0.17763.1457]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>diskpart

Microsoft DiskPart version 10.0.17763.1

Copyright (C) Microsoft Corporation.
On computer: EC2AMAZ-ASR4SAI

DISKPART> list disk

  Disk ###  Status         Size     Free     Dyn  Gpt
  --------  -------------  -------  -------  ---  ---
  Disk 0    Online           30 GB      0 B
  Disk 1    Online           46 GB    46 GB

DISKPART>
```

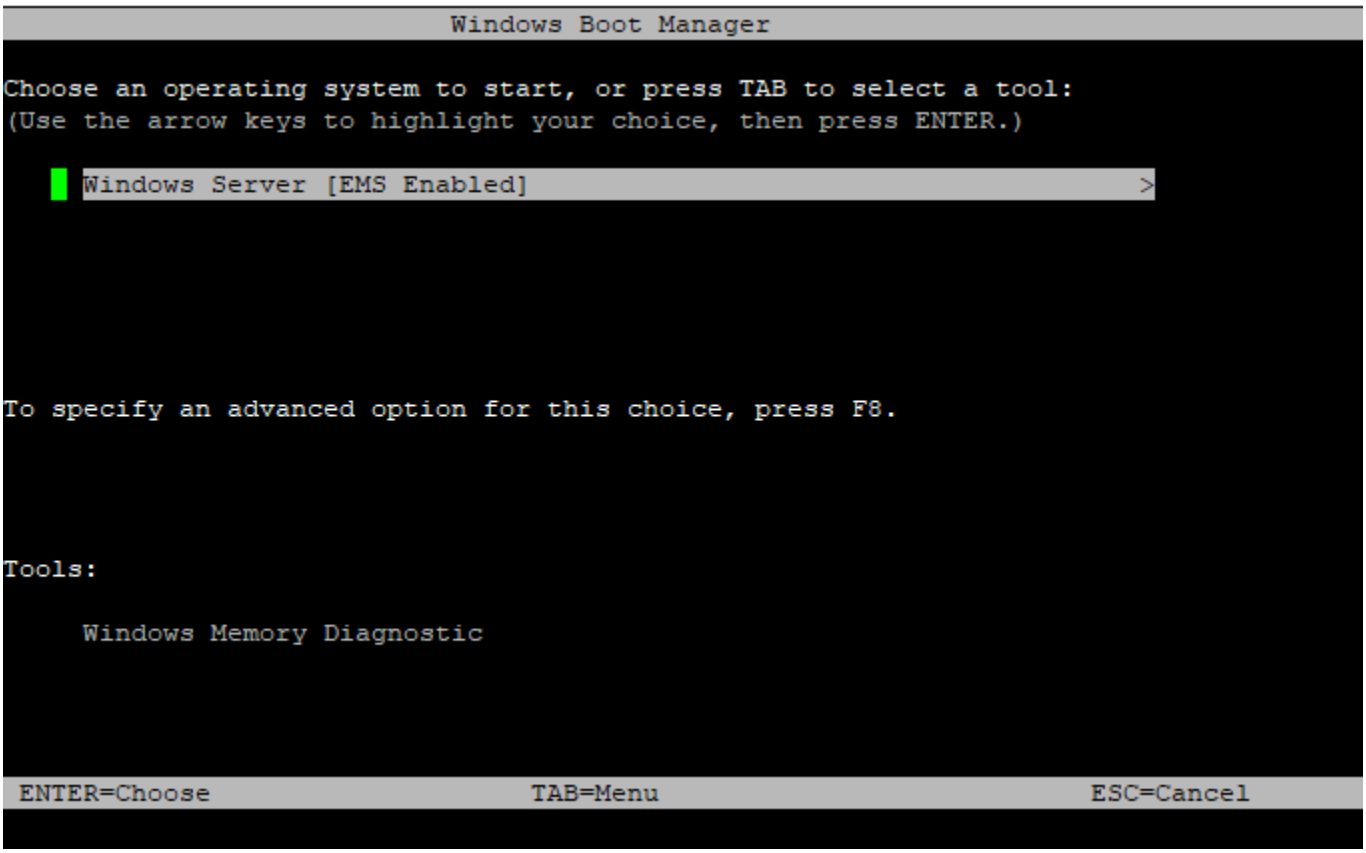
PowerShell 也可以從命令提示符使用。

請注意，您可能需要將進度偏好設定設定為無訊息模式。

```
PS C:\Windows\system32> $ProgressPreference="SilentlyContinue"
PS C:\Windows\system32> $computerInfo = Get-ComputerInfo
PS C:\Windows\system32> $computerInfo.Csprocessors[0].Name
Intel(R) Xeon(R) Platinum 8124M CPU @ 3.00GHz
PS C:\Windows\system32> $computerInfo.Csprocessors[0].Description
Intel64 Family 6 Model 85 Stepping 4
PS C:\Windows\system32>
```

使用開機功能表

如果執行個體啟用了開機功能表，並且在連線後重新啟動SSH，您應該會看到開機功能表，如下所示。



開機功能表命令

ENTER

啟動選取的作業系統項目。

TAB

切換至「工具」功能表。

ESC

取消並重新啟動執行個體。

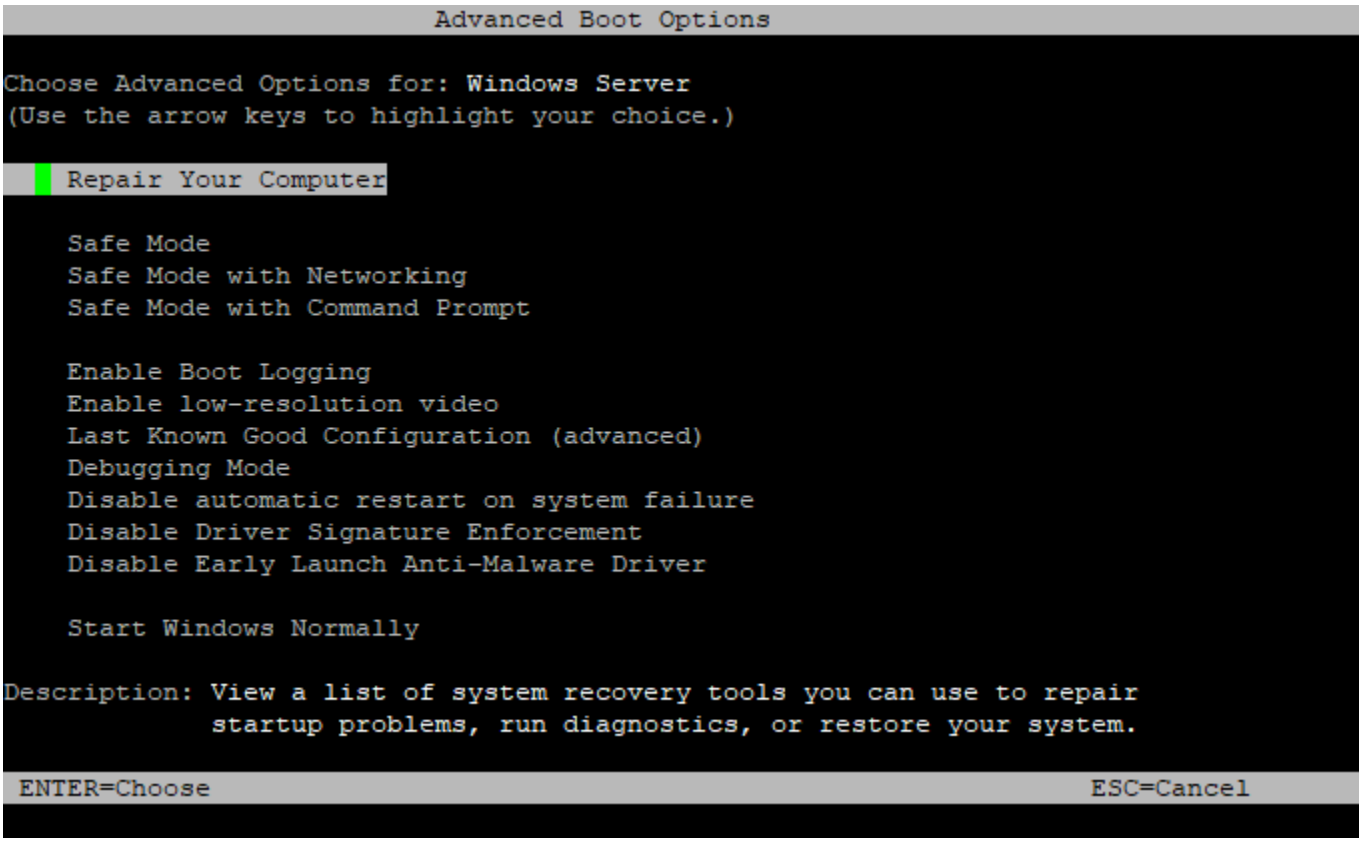
ESC其次是 8

相當於按下 F8 。顯示所選項目的進階選項。

ESC鍵 + 左箭頭

回到初始開始功能表。

> **Note**
>
> 單獨的ESC鍵不會將您帶回主菜單，因為 Windows 正在等待查看是否正在進行轉義序列。



禁用SAC和啟動菜單

如果您啟用SAC和開機功能表，您可以稍後停用這些功能。

使用下列其中一種方法來停SAC用執行個體上的開機功能表。

PowerShell

在 Windows 實例上禁用SAC和啟動菜單

1. Connect 線至執行個體，並從提高權限的 PowerShell 命令列執行下列步驟。
2. 首先透過將值變更為 no 來停用開機選單。

```
bcdedit /set '{bootmgr}' displaybootmenu no
```

3. 然後SAC通過將值更改為禁用off。

```
bcdedit /ems '{current}' off
```

4. 透過重新啟動執行個體來套用更新的組態。

```
shutdown -r -t 0
```

Command prompt

在 Windows 實例上禁用SAC和啟動菜單

1. 連接到您的執行個體，並從命令提示字元執行下列步驟。
2. 首先透過將值變更為 no 來停用開機選單。

```
bcdedit /set {bootmgr} displaybootmenu no
```

3. 然後SAC通過將值更改為禁用off。

```
bcdedit /ems {current} off
```

4. 透過重新啟動執行個體來套用更新的組態。

```
shutdown -r -t 0
```

傳送診斷中斷以偵錯無法連線的 Amazon EC2 執行個體

> **Warning**
> 診斷中斷專供進階使用者使用。不正確的使用可能會對執行個體造成負面影響。將診斷中斷傳送至執行個體可能會觸發執行個體當機並重新開機，這會導致資料遺失。

您可以傳送診斷中斷給無法連線或無回應的執行個體，以手動觸發 Linux 執行個體的核心錯誤，或針對 Windows 執行個體觸發停止錯誤 (通常稱為藍色畫面錯誤)。

Linux 執行個體

發生核心錯誤時，Linux 作業系統通常會當機並重新開機。作業系統的具體行為取決於其組態。核心錯誤也可用來促使執行個體的作業系統核心執行任務，例如產生損毀傾印檔案。然後，您可以利用損毀傾印檔案中的資訊來分析根本原因，對執行個體進行除錯。作業系統會代表執行個體本身在本機產生損毀傾印資料。

Windows 執行個體

一般而言，Windows 作業系統在停止錯誤發生時會當機並重新開機，但具體行為取決於其組態。停止錯誤也會導致作業系統將除錯資訊 (例如核心記憶體傾印) 寫入檔案。然後，您可以利用此資訊來分析根本原因，以進行執行個體的除錯。作業系統會代表執行個體本身在本機產生記憶體傾印資料。

將診斷中斷傳送至執行個體之前，建議您先參閱作業系統的說明文件，然後進行必要的組態變更。

內容

- 支援的執行個體類型
- 必要條件
- 傳送診斷中斷

支援的執行個體類型

所有 Nitro 型執行個體類型都支援診斷中斷，但由 AWS Graviton 處理器驅動的執行個體類型除外。如需詳細資訊，請參閱以 AWS Nitro 系統建置的執行個體和AWS 重力子。

必要條件

使用診斷中斷之前，您必須設定執行個體的作業系統。這可確保它在發生核心錯誤 (Linux 執行個體) 或停止錯誤 (Windows 執行個體) 時執行您需要的動作。

Linux 執行個體

設定 Amazon Linux 2 在核心錯誤發生時產生損毀傾印

1. 連線到您的執行個體。
2. 安裝 kexec 和 kdump。

```
[ec2-user ~]$ sudo yum install kexec-tools -y
```

3. 設定核心保留適當的記憶體數量給次要核心。該保留的記憶體數量取決於執行個體可用的記憶體總數。使用您偏好的文字編輯器開啟 `/etc/default/grub` 檔案，找出開頭為

GRUB_CMDLINE_LINUX_DEFAULT 的那一行，然後依下列格式新增 crashkernel 參數：crashkernel=*memory_to_reserve*。例如，若要保留 160MB，請如下修改 grub 檔案：

```
GRUB_CMDLINE_LINUX_DEFAULT="crashkernel=160M console=tty0 console=ttyS0,115200n8
 net.ifnames=0 biosdevname=0 nvme_core.io_timeout=4294967295 rd.emergency=poweroff
 rd.shell=0"
GRUB_TIMEOUT=0
GRUB_DISABLE_RECOVERY="true"
```

4. 儲存變更並關閉 grub 檔案。
5. 重建組GRUB2態檔案。

```
[ec2-user ~]$ sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

6. 在以 Intel 和AMD處理器為基礎的執行個體上，此send-diagnostic-interrupt命令會傳送未知的不可遮罩岔斷 (NMI) 給執行個體。您必須配置內核在收到未知的內核時崩潰NMI。使用您偏好的文字編輯器開啟 /etc/sysctl.conf 檔案，然後新增下列命令。

```
kernel.unknown_nmi_panic=1
```

7. 重新開機並重新連接您的執行個體。
8. 確認核心已使用正確的 crashkernel 參數開機。

```
$ grep crashkernel /proc/cmdline
```

下列輸出範例指出組態成功。

```
BOOT_IMAGE=/boot/vmlinuz-4.14.128-112.105.amzn2.x86_64 root=UUID=a1e1011e-
e38f-408e-878b-fed395b47ad6 ro crashkernel=160M console=tty0 console=ttyS0,115200n8
 net.ifnames=0 biosdevname=0 nvme_core.io_timeout=4294967295 rd.emergency=poweroff
 rd.shell=0
```

9. 確認 kdump 服務在執行中。

```
[ec2-user ~]$ systemctl status kdump.service
```

下列輸出範例顯示 kdump 服務在執行中的結果。

```
kdump.service - Crash recovery kernel arming
```

```
   Loaded: loaded (/usr/lib/systemd/system/kdump.service; enabled; vendor preset:
 enabled)
   Active: active (exited) since Fri 2019-05-24 23:29:13 UTC; 22s ago
  Process: 2503 ExecStart=/usr/bin/kdumpctl start (code=exited, status=0/SUCCESS)
 Main PID: 2503 (code=exited, status=0/SUCCESS)
```

> **Note**
>
> 根據預設，損毀傾印檔案會儲存至 `/var/crash/`。若要變更位置，請使用您偏好的文字編輯器來修改 `/etc/kdump.conf` 檔案。

設定 Amazon Linux 在核心錯誤發生時產生損毀傾印

1. 連線到您的執行個體。
2. 安裝 kexec 和 kdump。

   ```
   [ec2-user ~]$ sudo yum install kexec-tools -y
   ```

3. 設定核心保留適當的記憶體數量給次要核心。該保留的記憶體數量取決於執行個體可用的記憶體總數。

   ```
   $ sudo grubby --args="crashkernel=memory_to_reserve" --update-kernel=ALL
   ```

   例如，若要保留 160MB 給損毀核心，請使用下列命令。

   ```
   $ sudo grubby --args="crashkernel=160M" --update-kernel=ALL
   ```

4. 在以 Intel 和AMD處理器為基礎的執行個體上，此`send-diagnostic-interrupt`命令會傳送未知的不可遮罩岔斷 (NMI) 給執行個體。您必須配置內核在收到未知的內核時崩潰NMI。使用您偏好的文字編輯器開啟 `/etc/sysctl.conf` 檔案，然後新增下列命令。

   ```
   kernel.unknown_nmi_panic=1
   ```

5. 重新開機並重新連接您的執行個體。
6. 確認核心已使用正確的 `crashkernel` 參數開機。

   ```
   $ grep crashkernel /proc/cmdline
   ```

下列輸出範例指出組態成功。

```
root=LABEL=/ console=tty1 console=ttyS0 selinux=0 nvme_core.io_timeout=4294967295
 LANG=en_US.UTF-8 KEYTABLE=us crashkernel=160M
```

7. 確認 kdump 服務在執行中。

```
[ec2-user ~]$ sudo service kdump status
```

如果服務在執行中，命令會傳回 `Kdump is operational` 回應。

> **Note**
>
> 根據預設，損毀傾印檔案會儲存至 `/var/crash/`。若要變更位置，請使用您偏好的文字編輯器來修改 `/etc/kdump.conf` 檔案。

若要設定 SUSE Linux 企業版、Ubuntu 或紅帽企業版 Linux

在以 Intel 和AMD處理器為基礎的執行個體上，此`send-diagnostic-interrupt`命令會傳送未知的不可遮罩岔斷 (NMI) 給執行個體。您必須透NMI過調整作業系統的組態檔案，將核心設定為在收到未知資訊時損毀。如需如何設定核心損毀的詳細資訊，請參閱您作業系統的說明文件：

- SUSELinux 企業版
- Ubuntu
- 紅帽企業版 (RHEL)

Windows 執行個體

設定 Windows 在停止錯誤發生時產生記憶體傾印

1. 連線到您的執行個體。
2. 開啟控制台，選擇系統、進階系統設定。
3. 在系統內容對話方塊中，選擇進階索引標籤。
4. 在啟動及修復區段中，選擇設定...。
5. 在系統失敗區段中，視需要進行設定，然後選擇確定。

如需有關設定 Windows 停止錯誤的詳細資訊，請參閱 Windows 記憶體傾印檔案選項的概觀。

傳送診斷中斷

完成必要的組態變更後，您可以使用 AWS CLI 或 Amazon 將診斷中斷傳送到執行個體EC2API。

AWS CLI

將診斷中斷傳送至執行個體 (AWS CLI)

使用指send-diagnostic-interrupt令並指定執行個體 ID。

```
aws ec2 send-diagnostic-interrupt --instance-id i-1234567890abcdef0
```

PowerShell

將診斷中斷傳送至執行個體 (AWS Tools for Windows PowerShell)

使用指Send-EC2DiagnosticInterrupt令程式並指定執行個體識別碼。

```
PS C:\> Send-EC2DiagnosticInterrupt -InstanceId i-1234567890abcdef0
```

Amazon EC2 用戶指南的文檔歷史記錄

下表說明從 2019 年開始對 Amazon EC2 使用者指南的重要新增內容。我們也會經常更新指南，以解決您發送給我們的反饋。

變更	描述	日期
管理隨需容量保留的新功能	您現在可以分割「產能保留」、在「產能預留」之間移轉產能，以及修改「產能保留」的執行環境資格屬性。	2024年8月14日
支援 C6 公克、六分之六指數、C6gD、七克、兆六克、M6gD、M7gD、M7gD、R6 克及 R6gD 的休眠支援	讓在 C6g、C6GN、C6gD、C7G、C7gD、M6gD、M6gD、M7g、M7gD、R6g 和 R6gD 執行個體類型上執行的新啟動執行個體休眠。	2024年7月30日
支援 Graviton 執行AMIs個體類型的休眠支援	從支援重力子執行個體類型的 Amazon Linux 或 Ubuntu 啟動的新啟動執行AMI個體進行休眠。	2024年7月30日
認證保全支援的其他執行個體類型	您現在可以為 C7i、C7-彈性、M7i、M7i-彈性、R7i、R7i-彈性和 T3 執行個體啟用認證保護。	2024年6月26日
EC2M1 超級 Mac 实例	配備 Apple M1 Ultra 處理器的全新一般用途執行個體類型。	2024年6月17日
EC2例證類型搜尋工具 — 其他參數	EC2執行個體類型尋找工具現在會提供額外的參數，讓您為工作負載指定更詳細的需求。	2024年6月5 日
U7 至 12 TB、U7 英吋 -16 TB、烏七合二十四 TB 和三十二 TB 的 U7 執行個體	新增記憶體增強型執行個體類型，配備第 4 代 Intel Xeon 可擴充處理器	2024年5月28日

適用於EC2快速啟動的新受管政策	已新增EC2FastLaunchFullAccess政策，以從執API行個體執行與EC2快速啟動功能相關的動作。	2024 年 5 月 14 日
AMI撤銷註冊保護	您可以開啟取消註冊保護，以防AMI止意外或惡意刪除。	2024年4月23 日
PTP硬體時鐘 — 執行個體類型支援	PTP硬體時鐘現在可在 C7a、C7i、M7a、M7g、M7i、R7a 和 R7i 執行個體類型上使用。	2024年4月22 日
新增強型網路功能的 Nitro 效能考量	本頁面著重於網路考量，以協助調整 Nitro 型 Amazon EC2 執行個體的效能。	2024年4月4日
VSS基於EBS快照的新受管理策略	Amazon 提供EC2VSS了新的IAM受管政策，您可以將其新增至執行個體設定檔角色，以確保您的許可保留 up-to-date 並遵循最佳實務。	2024年3月28日
PTP硬體時鐘 — 美國東部 (維吉尼亞北部)	現在美國東部 (維吉尼亞北部) 區域提供PTP硬體時鐘。	2024年3月26日
設IMDSv2為帳戶預設值	您可以將帳戶中的所有新EC2執行個體啟動設定為預設使用執行個體中繼資料服務版本 2 (IMDSv2)。	2024年3月25日
標記從快照AMIs建立的新Linux	當您AMI從快照建立 Linux 時，您可以標記新的快照AMI。	2024年3月7日
複製時標記新AMIs的快照和快照	複製快照時AMI，您可以使用相同的標籤來標記新快照AMI和新快照，也可以使用不同的標籤來標記它們。	2024年3月7日

移除 AWS 管理組件頁面	AWS 管理組件主要與視窗伺服器 2012 年及之前版本搭配使用。這些舊版作業系統平台版本已不再受支援。若要管理和疑難排解在內部部署上 AWS 執行的伺服器群組，請參閱 AWS Systems Manager Fleet Manager。	2024年2月12日
EC2在 macOS 上預先安裝執行個體 Connect AMIs	EC2實例 Connect 現在自帶預裝在 macOS 索諾瑪 14.2.1 或更高版本, macOS 文圖拉 13.6.3 或更高版本, 和 macOS 蒙特雷 12.7.2 或更高版本. AMIs	2024年1月26日
EC2執行個體 Connect 支援 CentOS、macOS 和 RHEL	你現在可以在支援的 CentOS、macOS 和 RHELAMIs上安裝EC2執行個體 Connect。	2023 年 12 月 6 日
C7a、C7i、R7a、R7i 和 R7iz 的休眠支援	對您新啟動且在 C7a、C7i、R7a、R7i 和 R7iz 執行個體類型上執行的執行個體進行休眠。	2023 年 12 月 1 日
Amazon Q EC2 實例類型選擇器	Amazon Q EC2 執行個體類型選擇器會考量您的使用案例、工作負載類型和CPU製造商偏好設定，以及您如何設定價格和效能的優先順序。然後，它會使用這些資料，針對最適合您的新工作負載的 Amazon EC2 執行個體類型提供指導和建議。	2023 年 11 月 28 日
EC2 免費方案	您可以從EC2儀表板追蹤EC2免費方案的使用情況。	2023 年 11 月 26 日

Console-to-Code	Console-to-Code 可協助您開始使用自動化程式碼。主控台到程式碼會記錄您的主控台動作，然後使用生成式 AI，以您偏好的基礎架構即程式碼格式建議程式碼。您可以使用該程式碼作為起點，對其進行自訂以使其針對特定使用案例準備好投入生產。	2023 年 11 月 26 日
可設定的閒置連線追蹤逾時	處於閒置狀態的安全群組連線可能會導致連線追蹤耗盡，並造成不追蹤連線及捨棄封包。您現在可以在彈性網路介面上設定安全群組連線追蹤的逾時 (以秒為單位)。	2023 年 11 月 17 日
PTP硬體時鐘	支援的執行個體現在具有精確時間通訊協定 (PTP) 硬體時脈。PTP硬件時鐘支持NTP或直接PTP連接。	2023 年 11 月 16 日
為已啟用休眠的執行個體變更執行個體類型	您現在可以為已啟用休眠的執行個體變更執行個體類型 (處於 `stopped` 狀態的情況下)。	2023 年 11 月 16 日
執行個體拓撲	您可以使用偵測執行個體的位置，然後使用這項資訊來最佳化HPC和 ML 工作，方法是在實際上彼此較近的執行個體上執行這些工作。 DescribeInstanceTopology API	2023 年 11 月 13 日

EC2快速啟動共用AMI支援	您現在可以在與您共享AMI的上啟用EC2快速啟動。當您在共用上啟用EC2快速啟動時AMI，會在您的帳戶中建立預先佈建的快照以加快啟動速度。	2023 年 11 月 6 日
ML 的容量區塊	您現在可以在 future 日期預留GPU執行個體，以支援短期機器學習 (ML) 工作負載。	2023 年 10 月 31 日
Spot 執行個體休眠	現在您可以讓 Spot 執行個體進入休眠，方法與隨需執行個體目前可使用的休眠體驗和執行個體系列相同。	2023 年 10 月 24 日
封鎖公用存取的預設設定 AMIs	對於所有新帳戶和沒AMIs有公開的現有帳戶，預設為啟用的封鎖公開存取AMIs。	2023 年 10 月 20 日
Amazon EC2 全球視圖	Amazon EC2 全球檢視支援其他資源類型和可自訂的顯示選項。	2023 年 10 月 18 日
對於 Ubuntu 的休眠支持 22.04.2LTS（果醬水母）	休眠您從 Ubuntu 22.04.2LTS（卡米水母）啟動的新啟動實例。AMI	2023 年 10 月 16 日
禁用 AMI	您可以禁用AMI以防止將其用於實例啟動。	2023 年 10 月 12 日
附加EBS狀態檢查	您可以使用附加的EBS狀態檢查來監控連接到執行個體的Amazon EBS 磁碟區是否可存取。	2023 年 10 月 11 日
對 Red Hat Enterprise Linux 9 的休眠支援	讓您從 RHEL 9 啟動的新啟動執行個體休眠。AMI	2023 年 10 月 2 日

對 Microsoft Windows Server 2022 的休眠支援	讓從 Microsoft 視窗伺服器 2022 啟動的新啟動執行個體休眠。AMI	2023 年 10 月 2 日
023 的休眠支援 AL2	讓從 023 啟動的新啟動執行個體休眠。AL2 AMI	2023 年 10 月 2 日
啟動 Spot 機群中的 Spot 執行個體中斷	您可以在 Amazon EC2 主控台中選取 Spot 叢集，並啟動叢集中的 Spot 執行個體中斷，以便測試 Spot 執行個體上的應用程式如何處理中斷。	2023 年 9 月 21 日
封鎖公眾存取 AMIs	您可以AMIs在帳戶級別啟用阻止公共訪問，以阻止任何嘗試AMIs公開。	2023 年 9 月 12 日
M7i 和 M7i-flex 休眠支援	對您新啟動且在 M7i 和 M7i-flex 執行個體類型上執行的執行個體進行休眠。	2023 年 8 月 22 日
EC2-經典已被棄用	透過 EC2-Classic，EC2執行個體會在與其他客戶共用的單一扁平網路中執行。Amazon VPC 取代 EC2-經典。使用 AmazonVPC，您的執行個體在邏輯上與您的 AWS 帳戶隔離的虛擬私有雲 (VPC) 中執行。	2023 年 8 月 8 日
專用主機	您可以在 Outpost 上的特定硬體資產中分配專用執行個體。	2023 年 6 月 20 日
EC2實例 Connect 端點	您現在可以透過SSH或RDP不要求執行個體擁有公共IPv4位址連線至執行個體。	2023 年 6 月 13 日

IMDSPackage 分析儀	您現在可以使用IMDS封包分析器來識別EC2執行個體上的IMDSv1呼叫來源。	2023 年 6 月 1 日
EC2序列主控台裸機執行個體	EC2序列主控台現在支援連線到所選裸機執行個體的序列埠。	2023 年 4 月 11 日
啟動範本配額	您現在可以在「Service Quotas」主控台中檢視啟動範本的配額，以及啟動範本版本，並使用「Service Quotas」CLI。	2023 年 4 月 3 日
容量保留使用率通知	AWS Health 現在，當您帳戶中「容量保留」的容量使用率降至 20% 以下時，會傳送通知。	2023 年 4 月 3 日
容量保留群組	您現在可以將與您共用的容量保留新增到您擁有的容量保留群組。	2023 年 3 月 30 日
修改執行個體中繼資料選項	您現在可以使用 Amazon 主 EC2控台修改執行個體中繼資料選項。	2023 年 3 月 20 日
就地 macOS 作業系統更新	現在，您可以在 M1 Mac 執行個體上執行就地 Apple macOS 作業系統更新。	2023 年 3 月 14 日
UEFI首選	您現在可以建立支援統一AMI可延伸韌體介面 (UEFI) 和舊版BIOS開機模式的單一介面。	2023 年 3 月 3 日
修改AMI的 IMDSv2	修改現有實例，AMI以便默認情況下從 AMI require 啟動IMDSv2的實例。	2023 年 2 月 28 日

基於 Windows 虛擬化的安全性 - Credential Guard	您可以在支援的 Amazon 執行個體上啟用登入資料保全，這是一項虛擬化安全性 (VBS) 功能。EC2	2023 年 1 月 31 日
AMI啟動範本中的別名	您可以在啟動範本中指定 AWS Systems Manager 參數而不是 AMI ID，以避免每次 AMI ID 變更時都必須更新範本。	2023 年 1 月 19 日
C6i、I3en 和 M6i 休眠支援	對您新啟動且在 C6i、I3en 和 M6i 執行個體類型上執行的執行個體進行休眠。	2022 年 12 月 19 日
預防未完成的寫入	利用區塊儲存功能，預防未完成的寫入，提升 I/O 密集型關聯式資料庫工作負載的效能，並減少延遲，而不會對資料彈性產生負面影響。	2022 年 11 月 29 日
ENA快遞	使用 ENA Express 增加輸送量並將EC2執行個體之間的網路流量尾端延遲降至最低	2022 年 11 月 28 日
複製AMI標籤	複製時AMI，您可以同時複製使用者定義的AMI標籤。	2022 年 11 月 18 日
AMI儲存和還原的大小	可以在 Amazon S3 儲存貯體和從 Amazon S3 儲存貯體存放和還原的 AMI (壓縮前) 大小現在最高可達 5,000 GB。	2022 年 11 月 16 日
priceCapacityOptimizedSpot 執行個體的配置策略	使用 `priceCapacityOptimized` 分配策略的 Spot 機群會考慮價格和容量，來選取最不可能中斷且價格最低的 Spot 執行個體集區。	2022 年 11 月 10 日

price-capacity-optimizedSpot 執行個體的配置策略	使用`price-capacity-optimized` 配置策略的EC2叢集會同時查看價格和容量，以選擇最不可能中斷且價格最低的 Spot 執行個體集區。	2022 年 11 月 10 日
取消與您的帳戶AMI共享	如果AMI已與您共享，但您不想再 AWS 帳戶 與您的帳戶共享該帳戶，則可以從啟動權限中刪除您AMI的帳戶。	2022 年 11 月 4 日
轉移彈性 IP 地址	您現在可以將彈性 IP 地址從一個地址傳輸 AWS 帳戶 到另一個。	2022 年 10 月 31 日
取代根磁碟區	您可以使用替換執行中執行個體的根 Amazon EBS 磁碟區 AMI。	2022 年 10 月 27 日
自動將執行個體連線至資料庫	使用自動連線功能，將一或多個EC2執行個體快速連線至RDS資料庫，以允許它們之間的流量。	2022 年 10 月 10 日
AMI配額	配額現在適用於建立和共用 AMIs。	2022 年 10 月 10 日
設AMI定 IMDSv2	配置您的AMI以便默認情況下從AMI需要啟動IMDSv2的實例。	2022 年 10 月 3 日
啟動 Spot 執行個體中斷	您可以在 Amazon EC2 主控台中選取競價型執行個體並啟動中斷，以便測試 Spot 執行個體上的應用程式如何處理被中斷。	2022 年 9 月 26 日

驗證AMI供應商	在 Amazon EC2 控制台中，由 Amazon 或經過驗證的 Amazon 合作夥伴擁有的公共 AMIs標記為「已驗證的提供商」。	2022 年 7 月 22 日
放置群組 AWS Outposts	新增在 Outpost 上放置群組的主機分散策略。	2022 年 6 月 30 日
io2 Block Express 磁碟區	您可以修改 io2 Block Express 磁碟區IOPS的大小和佈建，也可以啟用它們以進行快速快照還原。	2022 年 5 月 31 日
專用主機 AWS Outposts	您可以配置 AWS Outposts上的專用執行個體。	2022 年 5 月 31 日
執行個體停止保護	若要避免意外停止執行個體，您可啟用執行個體的停止保護。	2022 年 5 月 24 日
UEFI安全開機	UEFI安全開機 defense-in-depth 是建立在 Amazon 長期安全開機程序的基礎上，EC2 並提供其他功能，協助客戶保護軟體免於在重新開機期間持續存在的威脅。	2022 年 5 月 10 日
硝基 TPM	硝基可信平台模塊（NitroTPM）是由 N AWS itro 系統提供並符合 2.0 規範的虛擬設備。TPM	2022 年 5 月 10 日
AMI狀態變更事件	Amazon EC2 現在在AMI改變狀態時生成一個事件。您可以使用 Amazon EventBridge 偵測這些事件並做出回應。	2022 年 5 月 9 日

描述公有金鑰	您可以查詢 Amazon key pair 的公開金EC2鑰和建立日期。	2022 年 4 月 28 日
建立金鑰對	您可以在建立新 key pair 時指定金鑰格式 (PEM或PPK)。	2022 年 4 月 28 日
啟動時掛載 Amazon FSx 檔案系統	您可以使用新的啟動執行個體精靈，在啟動時掛載新的 NetApp ONTAP或現有的 Amazon FSx for 或 Amazon 用於開啟ZFS檔案系統。FSx	2022 年 4 月 12 日
新的啟動執行個體精靈	Amazon EC2 主控台提供全新且經過改良的啟動體驗，提供更快速、更輕鬆的啟動EC2執行個體的方式。	2022 年 4 月 5 日
自動棄用公開 AMIs	默認情況下，所有公共的棄用日期AMIs都設置為從AMI創建日期起兩年。	2022 年 3 月 31 日
執行個體中繼資料類別：自動擴充/target-lifecycle-state	使用 Auto Scaling 群組時，您可以從執行個體中繼資料中存取執行個體的目標生命週期狀態。	2022 年 3 月 24 日
AMI上次啟動時間	`lastLaunchedTime` 表示您上次用於啟動執行個體的時AMI間。	2022 年 2 月 28 日
ED25519鑰匙	ED25519EC2執行個體 Connect 和EC2序列主控台現在支援金鑰。	2022 年 1 月 20 日
容量保留的其他RHEL平台	適用於按需容量保留的其他 Red Hat Enterprise Linux 平台。	2022 年 1 月 11 日

設定視窗AMIs以加快啟動速度	使用預先佈建的快照，將 Windows AMIs 設定為啟動執行個體的速度提高達 65%。	2022 年 1 月 10 日
執行個體中繼資料中的執行個體標籤	您可以從執行個體中繼資料存取執行個體的標籤。	2022 年 1 月 6 日
集群放置群組中的容量保留	可以在集群放置群組中建立容量保留。	2022 年 1 月 6 日
Spot 機群 launch-before-terminate	Spot 機群可以在新的替代 Spot 執行個體啟動之後，終止收到重新平衡通知的 Spot 執行個體。	2021 年 11 月 4 日
EC2艦隊 launch-before-terminate	EC2叢集可以終止在新的替代 Spot 執行個體啟動後收到重新平衡通知的 Spot 執行個體。	2021 年 11 月 4 日
比較時間戳記	您可以透過比較 Amazon EC2 Linux 執行個體的時間戳記與來判斷事件的真實時間 ClockBound。	2021 年 11 月 2 日
AMIs與組織共享 OUs	您現在可以AMIs與下列 AWS 資源共用：組織和組織單位 (OUs)。	2021 年 10 月 29 日
Spot 配置分數	根據您的 Spot 容量需求取得區 AWS 域或可用區域的建議。	2021 年 10 月 27 日
Spot 機群的屬性型執行個體類型選取	指定執行個體必須具有的屬性，Amazon EC2 會識別具有這些屬性的所有執行個體類型。	2021 年 10 月 27 日

叢集的屬性型執行個體類型選擇 EC2	指定執行個體必須具有的屬性，Amazon EC2 會識別具有這些屬性的所有執行個體類型。	2021 年 10 月 27 日
隨需容量預留機群	您可以使用容量預留機群，啟動一個群組或機群的容量預留。	2021 年 10 月 5 日
休眠支援對於 Ubuntu 20.04-焦LTS點	休眠您從 Ubuntu 20. LTS 04 啟動的新啟動的實例-焦點。AMI	2021 年 10 月 4 日
EC2叢集和目標隨需容量保留	EC2叢集可以將隨需執行個體啟動至`targeted`容量保留。	2021 年 9 月 22 日
專用主機上的 T3 執行個體	Support Amazon EC2 專用主機上的 T3 執行個體。	2021 年 9 月 14 日
軟呢帽RHEL和 CentOS 的休眠支援	讓你從 RHEL Fedora 和 CentOS 啟動的新啟動的執行個體休眠。AMIs	2021 年 9 月 9 日
Amazon EC2 全球視圖	Amazon EC2 Global View 可讓您在單一主控台中跨多個 AWS 區域檢VPCs視子網路、執行個體、安全群組和磁碟區。	2021 年 9 月 1 日
AMIAmazon Data Lifecycle Manager 的棄用支援	Amazon Data Lifecycle Manager EBS 支援的AMI政策可以棄用AMIs。受 AWSDataLifecycleManagerServiceRoleForAMIManagement AWS 管理策略已更新以支援此功能。	2021 年 8 月 23 日

C5d、M5d 和 R5d 的休眠支援	對您新啟動且在 C5d、M5d 和 R5d 執行個體類型上執行的執行個體進行休眠。	2021 年 8 月 19 日
Amazon EC2 金鑰對	Amazon EC2 現在支持 Linux 和 Mac 實例上的ED25519密鑰。	2021 年 8 月 17 日
網路介面字首	您可以自動IPv4或手動將私有或IPv6CIDR範圍指派給您的網路介面。	2021 年 7 月 22 日
事件時段	您可以為重新開機、停止或終止 Amazon EC2 執行個體的排程事件定義自訂的每週重複事件視窗。	2021 年 7 月 15 日
安全性群組規則的資源IDs和標籤支援	您可以依資源 ID 參考安全群組規則。您也可以為安全群組規則新增標籤。	2021 年 7 月 7 日
棄用 AMI	您現在可以指定何時AMI被取代。	2021 年 6 月 11 日
Windows 每秒計費	Amazon 以 Windows 和SQL伺服器為基礎的使用量EC2收取第二個費用，最低費用為一分鐘。	2021 年 6 月 10 日
容量保留 AWS Outposts	您現在可以在 AWS Outposts 上使用容量預留。	2021 年 5 月 24 日
共用容量保留	您現在可以共用在 Local Zones 和 Wavelength 區域中建立的容量預留。	2021 年 5 月 24 日
根磁碟區更換	您現在可以使用根磁碟區取代工作來取代執行中執行個體的根EBS磁碟區。	2021 年 4 月 22 日

AMI使用 S3 存放和恢復	AMIs在 S3 中存放EBS支援並從 S3 還原它們，以啟用跨分區複製. AMIs	2021 年 4 月 6 日
EC2序列主控台	透過建立執行個體的序列埠連線，疑難排解開機和網路連線問題。	2021 年 3 月 30 日
開機模式	Amazon EC2 現在支援選定AMD的執行個體和以 Intel 為基礎的EC2執行個體UEFI開機	2021 年 3 月 22 日
創建一個反向DNS記錄	您現在可以為彈性 IP 地址設置反向DNS查找。	2021 年 2 月 3 日
AMI建立時標記AMIs和快照	建立快照時AMI，您可以使用相同的標籤來標記AMI和快照，也可以使用不同的標籤來標記快照。	2020 年 12 月 4 日
使用 Amazon EventBridge 監控競價型叢集事件	建立可觸發程式設計動作的EventBridge 規則，以回應Spot 叢集狀態變更和錯誤。	2020 年 11 月 20 日
使用 Amazon EventBridge 監控EC2車隊事件	建立 EventBridge 規則來觸發程式設計動作，以回應EC2叢集狀態變更和錯誤。	2020 年 11 月 20 日
刪除 instant 機群	刪除EC2叢集類型`instant`並在單一API呼叫中終止叢集中的所有執行個體。	2020 年 11 月 18 日
對 T3 和 T3a 的休眠支援	對新啟動且在 T3 和 T3a 執行個體類型上執行的執行個體進行休眠。	2020 年 11 月 17 日

Amazon EFS 快速創建	您可以使用 Amazon EFS 快速建立，在啟動時建立 Amazon EFS 檔案系統並將其掛接到執行個體。	2020 年 11 月 9 日
例項中繼資料類別：events/recommendations/rebalance	針對執行處理發出EC2執行處理重新平衡建議通知時的大約時間 (以表示)。UTC	2020 年 11 月 4 日
EC2實例重新平衡建議	當 Spot 執行個體的中斷風險提高時，會以訊號來通知您。	2020 年 11 月 4 日
在 Wavelength 區域中的容量保留	容量預留 現在可以在 Wavelength 區域中建立和使用。	2020 年 11 月 4 日
容量重新平衡	設定 Spot 叢集或EC2叢集，以在 Amazon EC2 發出重新平衡建議時啟動替代 Spot 執行個體。	2020 年 11 月 4 日
對 I3、M5ad 和 R5ad 的休眠支援	對您新啟動且在 I3、M5ad 和 R5ad 執行個體類型上執行的執行個體進行休眠。	2020 年 10 月 21 日
Spot 執行個體 v CPU 限制	現在，Spot 執行個體限制是根據執行中的 Spot 執行 vCPUs 個體正在使用或將使用等待履行未結請求的數量來管理。	2020 年 10 月 1 日
Local Zones 中的容量保留	容量預留現在可以在 Local Zones 中建立和使用。	2020 年 9 月 30 日
對 M5a 和 R5a 的休眠支援	對您新啟動且在 M5a 和 R5a 執行個體類型上執行的執行個體進行休眠。	2020 年 8 月 28 日

執行個體中繼資料提供執行個體位置和放置資料	`placement` 類別下的新執行個體中繼資料欄位：區域、置放群組名稱、磁碟分割號碼、主機 ID 和可用區域 ID。	2020 年 8 月 24 日
容量保留群組	您可以使用 AWS Resource Groups 建立容量保留的邏輯集合，然後將目標執行個體啟動到這些群組中。	2020 年 7 月 29 日
EC2LaunchV2	您可以在執行個體啟動期間使用 EC2Launch v2 來執行工作，如果執行個體停止並稍後啟動，如果執行個體重新啟動，以及視需要執行個體。EC2LaunchV2 支持所有版本的視窗服務器和替換EC2 Launch和EC2Config.	2020 年 6 月 30 日
帶上自己的IPv6地址	您可以將內部部署網路中的部分或全部IPv6地址範圍帶入您的 AWS 帳戶。	2020 年 5 月 21 日
使用 Systems Manager 參數啟動執行個體	啟動執行個體AMI時，您可以指定 AWS Systems Manager 參數而不是參數。	2020 年 5 月 5 日
自訂排程的事件通知	您可以自訂排程事件通知，以便在電子郵件通知中包含標籤。	2020 年 5 月 4 日
Amazon Linux 2 Kernel Live Patching	Amazon Linux 2 的 Kernel Live Patching 可讓您將安全性漏洞和重大錯誤修補程式套用至執行中的 Linux 核心，而不需要重新啟動或中斷執行中的應用程式。	2020 年 4 月 28 日

在專用主機上的 Windows Server	您可以使用 Amazon AMIs 提供的 Windows 服務器在專用主機上運行最新版本的 Windows 服務器。	2020 年 4 月 7 日
停止並啟動 Spot 執行個體	停止 Amazon 支援的 Spot 執行個體EBS並隨意啟動它們，而不是依賴停止中斷行為。	2020 年 1 月 13 日
資源標記	您可以標記僅限輸出的網際網路閘道、本機閘道、本機閘道路由表、本機閘道虛擬介面、本機閘道虛擬介面群組、本機閘道路由表VPC關聯，以及本機閘道路由表虛擬介面群組關聯。	2020 年 1 月 10 日
使用工作階段管理員連線至執行個體	您可以從 Amazon EC2 主控台使用執行個體啟動工作階段管理員工作階段。	2019 年 12 月 18 日
專用主機和主機資源群組	專用主機現在可與主機資源群組搭配使用。	2019 年 12 月 2 日
專用主機共用	您現在可以跨 AWS 帳戶共用專用主機。	2019 年 12 月 2 日
帳戶層級的預設額度規格	您可以在每個 AWS 區域的帳戶層次，設定每個高載績效執行環境系列的預設信用規格。	2019 年 11 月 25 日
執行個體類型探索	您可以尋找符合您需求的執行個體類型。	2019 年 11 月 22 日
專用主機	您現在可以將 專用主機 設定為支援執行個體系列中的多個執行個體類型。	2019 年 11 月 21 日

執行個體中繼資料服務第 2 版	您可以使用 執行個體中繼資料服務第 2 版，這種工作階段導向的方法，可讓您請求執行個體中繼資料。	2019 年 11 月 19 日
Elastic Fabric Adapter	彈性光纖介面卡現在可搭配 Intel MPI 2019 更新 6 使用。	2019 年 11 月 15 日
休眠支援隨需 Windows 執行個體	您可以讓隨需 Windows 執行個體進入休眠。	2019 年 10 月 14 日
已將預留執行個體購買放入佇列	最早可提前三年將 預留執行個體 購買放入佇列。	2019 年 10 月 4 日
診斷中斷	您可以將診斷中斷傳送至無法連線或沒有回應的執行個體，以觸發核心錯誤。	2019 年 8 月 14 日
容量最佳化分配策略	使用EC2叢集或 Spot 叢集，您可以從 Spot 集區啟動 Spot 執行個體，並針對正在啟動的執行個體數量使用最佳容量。	2019 年 8 月 12 日
隨需容量保留共用	您現在可以跨 AWS 帳戶共用容量保留。	2019 年 7 月 29 日
Elastic Fabric Adapter	EFA現在支援開放式 MPI 3.1.4 和英特爾 MPI 2019 年更新 4。	2019 年 7 月 26 日
EC2實例 Connect	EC2執行個體 Connect 是使用安全殼層 (SSH) 連線至執行個體的簡單且安全的方式。	2019 年 6 月 27 日
主機復原	當 專用主機 上發生非預期的硬體失敗時，在新主機上自動重新啟動您的執行個體。	2019 年 6 月 5 日

VSS應用程式一致性	使用執 AWS Systems Manager 行命令，為連接到 Windows 執行個體的所有 Amazon EBS 磁碟區拍攝應用程式一致的快照。	2019 年 5 月 13 日
Microsoft SQL 伺服器資料庫的視窗到 Linux 重新平台輔助程式	將現有的 Microsoft SQL 伺服器工作負載從視窗移至 Linux 作業系統。更新鏈接指向 Microsoft SQL 服務器在 Amazon EC2 用戶指南。	2019 年 5 月 8 日
Windows 自動升級	使用執行 EC2 Windows 執行個體的自動升級 AWS Systems Manager。	2019 年 5 月 6 日
Elastic Fabric Adapter	您可以將彈性網狀架構介面卡連接至執行個體，以加速高效能運算 (HPC) 應用程式。	2019 年 4 月 29 日

如需 Amazon 執行個體類型版本的相關資訊EC2，請參閱 Amazon EC2 執行個體類型指南中的文件歷史記錄。

2018 年及更早版本的歷史

下表說明 2018 年和更早年份 Amazon EC2 使用者指南的重要新增內容。

功能	API版本	描述	發行日期
分區置放群組	2016-11-15	分區置放群組會將執行個體分散到邏輯分區，確保一個分區中的執行個體不會與其他分區中的執行個體共用底層硬體。如需詳細資訊，請參閱分區置放群組。	2018 年 12 月 20 日
讓 EC2 Linux 執行個體休眠	2016-11-15	當您的 Linux 執行個體已啟用休眠並符合休眠必要條件時，您才能讓執行個體進入休眠。如需詳細資訊，請參閱 休眠您的 Amazon EC2 實例。	2018 年 11 月 28 日

功能	API版本	描述	發行日期
Amazon Elastic Inference Accelerator	2016-11-15	您可以將 Amazon EI 加速器連接到執行個體，以新增GPU支援的加速功能，以降低執行深度學習推論的成本。	2018 年 11 月 28 日
Spot 主控台建議使用眾多執行個體	2016-11-15	Spot 主控台根據 Spot 最佳實務 (執行個體多樣化) 建議一組執行個體，以滿足您應用程式所需的最低硬體規格 (vCPUs、記憶體和儲存)。如需詳細資訊，請參閱建立 Spot 機群。	2018 年 11 月 20 日
新EC2車隊請求類型：instant	2016-11-15	EC2叢集現在支援新的要求類型`instant`，您可以使用此類型同步佈建執行個體類型和購買模式的容量。要`instant`求會在回API應中傳回已啟動的執行個體，而且不會採取進一步的動作，讓您控制是否啟動執行個體以及何時啟動執行個體。如需詳細資訊，請參閱EC2叢集和 Spot 叢集請求類型。	2018 年 11 月 14 日
Spot 節省資訊	2016-11-15	您可以檢視針對單一 Spot Fleet 或所有 Spot 執行個體使用 Spot 執行個體所節省的成本。如需詳細資訊，請參閱 購買 競價型執行個體省下的成本。	2018 年 11 月 5 日
主控台支援最佳化CPU選項	2016-11-15	啟動執行個體時，您可以使用 Amazon EC2 主控台優化CPU選項，以滿足特定工作負載或業務需求。如需詳細資訊，請參閱CPUAmazon EC2 實例的選項。	2018 年 10 月 31 日
主控台支援從執行個體建立啟動範本	2016-11-15	您可以使用執行個體建立啟動範本，做為使用 Amazon EC2 主控台新啟動範本的基礎。如需詳細資訊，請參閱創建一個 Amazon EC2 啟動模板。	2018 年 10 月 30 日

功能	API版本	描述	發行日期
On-Demand Capacity Reservations	2016-11-15	您可以在任何時間內為特定可用區域中的 Amazon EC2 執行個體預留容量。這可讓您建立和管理容量保留，而不受預留執行個體 (RI) 的帳單折扣影響。如需詳細資訊，請參閱 透過隨需容量保留來保留運算容量。	2018 年 10 月 25 日
攜帶您自己的 IP 位址 (BYOIP)	2016-11-15	您可以將內部部署網路中的部分或全部公用IPv4位址範圍帶入您的 AWS 帳戶。將地址範圍帶到之後 AWS，它會以地址集區的形式出現在您的帳戶中。您可以從您的地址集區建立彈性 IP 地址，並將它用於您的 AWS 資源。如需詳細資訊，請參閱 將您自己的 IP 地址（BYOIP）帶到 Amazon EC2。	2018 年 10 月 23 日
在建立專用主機時套用標籤和主控台支援	2016-11-15	您可以在建立專用主機時標記，也可以使用 Amazon 主EC2控台管理專用主機標籤。如需詳細資訊，請參閱分配 Amazon EC2 專用主機以在您的帳戶中使用。	2018 年 10 月 8 日
對 Spot Fleet 的排程擴展的主控台支援	2016-11-15	根據日期和時間，來增加或減少機群目前的容量。如需詳細資訊，請參閱 排定的擴展：按排程擴展 Spot 叢集。	2018 年 9 月 20 日
EC2艦隊的配置策略	2016-11-15	您可以指定隨需容量是由價格 (最低價格優先) 或優先順序 (最高優先順序優先) 履行。您可以指定要配置目標 Spot 容量的 Spot 集區數量。如需詳細資訊，請參閱 使用配置策略來確定 EC2叢集或 Spot 叢集如何滿足 Spot 和隨需容量。	2018 年 7 月 26 日
Spot Fleets 的分配策略	2016-11-15	您可以指定隨需容量是由價格 (最低價格優先) 或優先順序 (最高優先順序優先) 履行。您可以指定要配置目標 Spot 容量的 Spot 集區數量。如需詳細資訊，請參閱 使用配置策略來確定 EC2叢集或 Spot 叢集如何滿足 Spot 和隨需容量。	2018 年 7 月 26 日

功能	API版本	描述	發行日期
自動快照生命週期	2016-11-15	您可以使用 Amazon Data Lifecycle Manager 為EBS磁碟區自動建立和刪除快照。如需詳細資訊，請參閱 Amazon Data Lifecycle Manager。	2018 年 7 月 12 日
啟動範本CPU選項	2016-11-15	使用命令列工具建立啟動範本時，您可以最佳化CPU選項以符合特定工作負載或業務需求。如需詳細資訊，請參閱創建一個 Amazon EC2 啟動模板。	2018 年 7 月 11 日
標記 專用主機	2016-11-15	您可以標記您的 專用主機。	2018 年 7 月 3 日
取得最新主控台輸出	2016-11-15	您可以在使用get-console-output AWS CLI 指令時擷取某些執行個體類型的最新主控台輸出。	2018 年 5 月 9 日
最佳化CPU選項	2016-11-15	啟動執行個體時，您可以最佳化CPU選項以符合特定工作負載或業務需求。如需詳細資訊，請參閱CPUAmazon EC2 實例的選項。	2018 年 5 月 8 日
EC2艦隊	2016-11-15	您可以使用EC2叢集，跨不同執行個體類型和可用區域，以及隨需EC2執行個體、預留執行個體和 Spot 執行個體購買模式啟動一組執行個體。如需詳細資訊，請參閱EC2艦隊和現貨艦隊。	2018 年 5 月 2 日
Spot Fleets 中的隨需執行個體	2016-11-15	您可以在 Spot Fleet 請求中，加入對隨需容量的請求，來確保永遠有執行個體容量。如需詳細資訊，請參閱EC2艦隊和現貨艦隊。	2018 年 5 月 2 日
建立EBS快照時標記	2016-11-15	您可以在建立期間將標籤套用至快照。	2018 年 4 月 2 日
變更置放群組	2016-11-15	您可以將執行個體移入或移出置放群組，或變更其置放群組。如需詳細資訊，請參閱 變更 Amazon EC2 執行個體的放置位置。	2018 年 3 月 1 日
更長的資源 IDs	2016-11-15	您可以為更多資源類型啟用較長的 ID 格式。	2018 年 2 月 9 日

功能	API版本	描述	發行日期
網路效能提升	2016-11-15	在其他執行個體或 Amazon S3 之間傳送或接收網路流量時，集群放置群組之外的執行個體現在可享有更高的頻寬。	2018 年 1 月 24 日
標記彈性 IP 地址	2016-11-15	您可以標記彈性 IP 地址。	2017 年 12 月 21 日
Amazon Time Sync Service	2016-11-15	您可以使用 Amazon Time Sync Service 來確保執行個體的時間正確。如需詳細資訊，請參閱EC2執行個體上的精確時脈和時間同步。	2017 年 11 月 29 日
無限制 T2	2016-11-15	T2 效能不受限執行個體可隨時按需要大幅提升效能以超越基準。如需詳細資訊，請參閱 爆量效能執行個體。	2017 年 11 月 29 日
啟動範本	2016-11-15	啟動範本可包含用以啟動執行個體的所有或部分參數，因此您不需要在每次啟動執行個體時指定。如需詳細資訊，請參閱 在 Amazon 啟動範本中存放執行個體EC2啟動參數。	2017 年 11 月 29 日
分散置放	2016-11-15	若應用程式具有應該分開保存的少量重要執行個體，建議使用分散置放群組。如需詳細資訊，請參閱 分散放置群組。	2017 年 11 月 29 日
Spot 執行個體休眠	2016-11-15	Spot 服務可以在發生中斷時讓 Spot 執行個體休眠。如需詳細資訊，請參閱 休眠中斷的 競價型執行個體。	2017 年 11 月 28 日
Spot Fleet 目標追蹤	2016-11-15	您可以為 Spot Fleet 設定目標追蹤擴展政策。如需詳細資訊，請參閱 目標追蹤擴展：透過定位特定量度的值來擴展 Spot 叢集。	2017 年 11 月 17 日
Spot Fleet 與 Elastic Load Balancing 整合	2016-11-15	您可以將一或多個負載平衡器連接至 Spot Fleet。	2017 年 11 月 10 日

功能	API版本	描述	發行日期
合併及分割 可轉換預留執行個體	2016-11-15	您可以為新的 可轉換預留執行個體 交換 (合併) 兩個或多個 可轉換預留執行個體。您也可以使用修改程序將可轉換預留執行個體分割為較小的保留。如需詳細資訊，請參閱 交換 可轉換預留執行個體。	2017 年 11 月 6 日
修改VPC租賃	2016-11-15	您可以將執行個體租用屬性VPC從變更 `dedicated` 為`default`。如需詳細資訊，請參閱變更執行個體租用 VPC。	2017 年 10 月 16 日
每秒計費	2016-11-15	Amazon 以 Linux 為基礎的用量EC2收取第二個費用，最低費用為一分鐘。	2017 年 10 月 2 日
中斷時停止	2016-11-15	您可以指定 Amazon 是否EC2應在 Spot 執行個體中斷時停止或終止它們。如需詳細資訊，請參閱Spot 執行個體中斷的行為。	2017 年 9 月 18 日
標記NAT閘道	2016-11-15	您可以標記NAT閘道。如需詳細資訊，請參閱標記您的 資源。	2017 年 9 月 7 日
安全群組規則說明	2016-11-15	您可以為安全群組規則新增說明。	2017 年 8 月 31 日
Elastic Graphics	2016-11-15	將 Elastic Graphics 加速器連接至您的執行個體，來加速應用程式的圖形效能。	2017 年 8 月 29 日
復原彈性 IP 地址	2016-11-15	如果您釋放彈性 IP 位址供在中使用VPC，您可能可以復原它。	2017 年 8 月 11 日
標記 Spot Fleet 執行個體	2016-11-15	您可以設定 Spot Fleet 以自動標記其啟動的執行個體。	2017 年 7 月 24 日

功能	API版本	描述	發行日期
在建立期間標記資源	2016-11-15	您可以在建立期間將標籤套用至執行個體和磁碟區。如需詳細資訊，請參閱 標記您的 資源。此外，您也可以使用標籤型資源層級許可，來控制套用的標籤。如需更多詳細資訊，請參閱 授予在建立期間標記 Amazon EC2 資源的權限。	2017 年 3 月 28 日
對連結的EBS磁碟區執行修改	2016-11-15	將大多數EBS磁碟區連接到大多數EC2執行個體後，您可以修改磁碟區大小、類型，IOPS而無需分離磁碟區或停止執行個體。	2017 年 2 月 13 日
連接 IAM 角色	2016-11-15	您可以附加、卸離或取代現有執行個體的IAM角色。如需詳細資訊，請參閱IAMAmazon 的角色 EC2。	2017 年 2 月 9 日
專用的 競價型執行個體	2016-11-15	您可以在虛擬私有雲中的單租用戶硬體上執行 Spot 執行個體 (VPC)。如需詳細資訊，請參閱在單租戶硬體上啟動。	2017 年 1 月 19 日
IPv6支持	2016-11-15	您可以將一個IPv6CIDR與您的VPC和子網路相關聯，並將IPv6位址指派給您中的VPC執行個體。如需詳細資訊，請參閱Amazon EC2 實例 IP 定址。	2016 年 12 月 1 日
Spot Fleet 的自動擴展		您現在可以為 Spot Fleet 設定擴展政策。如需詳細資訊，請參閱 瞭解 Spot 叢集的自動擴充。	2016 年 9 月 1 日
彈性網路介面卡 (ENA)	2016-04-01	您現在可以使ENA用增強型網路功能。如需詳細資訊，請參閱Amazon EC2 執行個體上的增強聯網。	2016 年 6 月 28 日
增強對查看和修改時間更長的支持 IDs	2016-04-01	您現在可以檢視和修改其他使IAM用者、IAM角色或根使用者的較長 ID 設定。	2016 年 6 月 23 日
在 AWS 帳戶之間複製加密的 Amazon EBS 快照	2016-04-01	您現在可以在 AWS 帳戶之間複製加密的EBS快照。	2016 年 6 月 21 日

功能	API版本	描述	發行日期
擷取執行個體主控台的螢幕擷取畫面	2015-10-01	當您對無法連線的執行個體除錯時，您現在可以取得其他資訊。如需詳細資訊，請參閱擷取無法連線執行個體的螢幕擷取畫面。	2016 年 5 月 24 日
兩種新EBS磁碟區類型	2015-10-01	您現在可以建立輸送量最佳化 HDD (st1) 和冷 HDD (sc1) 磁碟區。	2016 年 4 月 19 日
為 Amazon 添加了新的 NetworkPacketsIn 和 NetworkPacketsOut 指標 EC2		為 Amazon 添加了新的 NetworkPacketsIn 和 NetworkPacketsOut 指標EC2。如需詳細資訊，請參閱執行個體指標。	2016 年 3 月 23 日
CloudWatch 競價型艦隊的指標		您現在可以取得 Spot 叢集的 CloudWatch 指標。如需詳細資訊，請參閱使用以下方式監控您的車EC2隊或 Spot CloudWatch。	2016 年 3 月 21 日
排程執行個體	2015-10-01	排程的預留執行個體 (排程執行個體) 可讓您以指定的開始時間和期間，購買每日、每週或每月的容量保留。	2016 年 1 月 13 日
更長的資源 IDs	2015-10-01	我們逐漸IDs為某些 Amazon EC2 和 Amazon EBS 資源類型引入更長的長度。在選擇期間，您可以為支援的資源類型啟用較長的 ID 格式。	2016 年 1 月 13 日
ClassicLink DNS支持	2015-10-01	您可以啟用對您的 ClassicLink DNS支援，以VPC便在連結 EC2-Classic 執行個體和執行個體之間尋址的DNS主機名稱VPC解析為私有 IP 位址，而非公用 IP 位址。	2016 年 1 月 11 日
專用主機	2015-10-01	Amazon EC2 專用主機是具有專供您使用的執行個體容量的實體伺服器。如需詳細資訊，請參閱 Amazon EC2 專用主機。	2015 年 11 月 23 日
Spot 執行個體持續時間	2015-10-01	您現在可以為 競價型執行個體指定持續時間。不支援 Spot 區塊 (2023 年 1 月)。	2015 年 10 月 6 日

功能	API版本	描述	發行日期
Spot Fleet 修改請求	2015-10-01	您現在可以修改 Spot Fleet 請求的目標容量。如需詳細資訊，請參閱 修改 Spot Fleet 請求。	2015 年 9 月 29 日
Spot Fleet 多樣化分配策略	2015-04-15	您現在可以使用單一 Spot Fleet 請求，將 Spot 執行個體配置到多個 Spot 集區。如需詳細資訊，請參閱使用配置策略來確定EC2叢集或 Spot 叢集如何滿足 Spot 和隨需容量。	2015 年 9 月 15 日
Spot Fleet 執行個體權重	2015-04-15	您現在可以定義每個執行個體類型對應用程式效能所貢獻的容量單位，並據以調整每個 Spot 集區您願意為 競價型執行個體支付的價格。如需詳細資訊，請參閱使用執行個體加權來管理EC2叢集或 Spot 叢集的成本和效能。	2015 年 8 月 31 日
新的重新啟動警報動作和用於警報動作的新IAM角色		添加了重新啟動警報動作和用於警報操作的新IAM角色。如需詳細資訊，請參閱建立警示以停止、終止、重新啟動或復原執行個體。	2015 年 7 月 23 日
Spot Fleets	2015-04-15	您可以管理 Spot 執行個體集合或機群，而不是管理個別 Spot 執行個體請求。如需詳細資訊，請參閱EC2艦隊和現貨艦隊。	2015 年 5 月 18 日
將彈性 IP 地址遷移到 EC2-典型	2015-04-15	您可以遷移已分配給在 EC2-Classic 中使用的彈性 IP 地址，以便在VPC..	2015 年 5 月 15 日
將多VMs個磁碟匯入為 AMIs	2015-03-01	VM Import 入程序現在支援將多VMs個磁碟匯入為AMIs。如需詳細資訊，請參閱VM Import/Export 使用者指南中的使用 VM 匯入/匯出將 VM 匯入為映像。	2015 年 4 月 23 日
Systems Manager		Systems Manager 可讓您設定和管理EC2執行個體。	2015 年 2 月 17 日

功能	API版本	描述	發行日期
Microsoft SCVMM 1.5 的 Systems Manager		您現在可以使用 Microsoft SCVMM 的 Systems Manager 器啟動執行個體，並將虛擬機器從 SCVMM Amazon 匯入EC2。	2015 年 1 月 21 日
EC2執行個體的自動復原		您可以建立 Amazon CloudWatch 警示來監控 Amazon EC2 執行個體，並在執行個體因基礎硬體故障或需要 AWS 參與修復的問題而受損時，自動復原執行個體。復原的執行個體與原始執行個體相同，包括執行個體 ID、IP 地址及所有執行個體中繼資料。 如需詳細資訊，請參閱 執行個體恢復。	2015 年 1 月 12 日
ClassicLink	2014-10-01	ClassicLink 可讓您將 EC2-Classic 執行個體連結至帳戶VPC中的執行個體。您可以將VPC安全群組與 EC2-Classic 執行個體建立關聯，以便在您的 EC2-Classic 執行個體與VPC使用私有 IP 位址的執行個體之間進行通訊。	2015 年 1 月 7 日
Spot 執行個體終止通知		防止 Spot 執行個體中斷的最佳方法就是建立容錯的應用程式架構。此外，您可以利用競價型執行個體終止通知，該通知在 Amazon EC2 必須終止 Spot 執行個體之前提供兩分鐘的警告。 如需詳細資訊，請參閱Spot 執行個體中斷通知。	2015 年 1 月 5 日
Microsoft Systems Manager SCVMM		Microsoft 的 Systems Manager SCVMM 提供了一個簡單的， easy-to-use接口來管理 AWS 資源，如EC2實例，從 Microsoft SCVMM。	2014 年 10 月 29 日
`DescribeVolumes` 分頁支援	2014-09-01	`DescribeVolumes` API呼叫現在支援使用`MaxResults` 和`NextToken` 參數對結果進行分頁。如需詳細資訊，請參閱 Amazon 參 EC2API考資料DescribeVolumes中的。	2014 年 10 月 23 日

功能	API版本	描述	發行日期
增加了對 Amazon CloudWatch 日誌支持		您可以使用 Amazon CloudWatch Logs 監控、存放和存取執行個體或其他來源的系統、應用程式和自訂日誌檔。然後，您可以使用 Amazon CloudWatch 主控台、中的 CloudWatch 日誌命令或日誌從 CloudWatch 日誌擷取關聯的 CloudWatch 日誌資料SDK。 AWS CLI	2014 年 7 月 10 日
新增EC2服務限制頁面		使用 Amazon EC2 主控台中的「EC2服務限制」頁面，以每個區域為基礎VPC，檢視 Amazon EC2 和 Amazon 提供的資源目前限制。	2014 年 6 月 19 日
Amazon EBS 通用 SSD卷	2014-05-01	一般用途SSD磁碟區提供符合成本效益的儲存，非常適合廣泛的工作負載。這些磁碟區提供個位數毫秒的延遲，能夠長時間突增至 3,000IOPS，以及 3 /GiB 的基本效能。IOPS一般用途SSD磁碟區的大小範圍從 1 GiB 到 1 TiB。	2014 年 6 月 16 日
AWS 管理組件		AWS 管理組件現在支援系統中心作業管理員 2012 年 R2。	2014 年 5 月 22 日
Amazon EBS 加密	2014-05-01	Amazon EBS 加密提供EBS資料磁碟區和快照的無縫加密，無需建立和維護安全的金鑰管理基礎設施。EBS加密通過使 AWS 受管金鑰用加密數據來實現靜態數據的安全性。加密會在託管執行個體的伺服器上進EC2行，在執行個體和EBS儲存EC2體之間移動時提供資料加密。	2014 年 5 月 21 日
Amazon EC2 用量報告		Amazon EC2 用量報告是一組報告，顯示您的使用成本和用量資料EC2。	2014 年 1 月 28 日
匯入 Linux 虛擬機器	2013-10-15	VM Import 程序現在支援匯入 Linux 執行個體。如需詳細資訊，請參閱 VM Import/Export 使用者指南。	2013 年 12 月 16 日

功能	API版本	描述	發行日期
下列項目的資源層級權限 RunInstances	2013-10-15	您現在可以在中建立政策， AWS Identity and Access Management 以控制 Amazon EC2 RunInstances API 動作的資源層級許可。如需詳細資訊和範例政策，請參閱Amazon 的身份和訪問管理 EC2。	2013 年 11 月 20 日
從啟動執行個體 AWS Marketplace		您現在可以從 AWS Marketplace 使用 Amazon 啟動精靈EC2啟動執行個體。如需詳細資訊，請參閱從一個啟動 Amazon EC2 實例 AWS Marketplace AMI。	2013 年 11 月 11 日
新的 Launch Wizard		有一個新的和重新設計的EC2啟動嚮導。如需詳細資訊，請參閱使用 Amazon 主控台中的啟動EC2執行個體精靈啟動 Amazon EC2 執行個體。	2013 年 10 月 10 日
修改預留執行個體的例項類型	2013-10-01	您現在可以修改相同系列之 Linux 預留執行個體的執行個體類型 (例如 M1、M2、M3、C1)。如需詳細資訊，請參閱 修改 預留執行個體。	2013 年 10 月 9 日
修改 Amazon EC2 預留執行個體	2013-08-15	您現在可以修改區域中的預留執行個體。如需更多詳細資訊，請參閱 修改 預留執行個體。	2013 年 9 月 11 日
指派公有 IP 地址	2013-07-15	您現在可以在. 中啟動執行個體時指派公用 IP 位址VPC。如需詳細資訊，請參閱在執行個體啟動期間指派公用IPv4地址。	2013 年 8 月 20 日
授予資源層級許可	2013-06-15	Amazon EC2 支持新的 Amazon 資源名稱（ARNs）和條件密鑰。如需詳細資訊，請參閱Amazon 的基於身份的政策 EC2。	2013 年 7 月 8 日
增量式快照複本	2013-02-01	您現在可以執行增量式快照複本。	2013 年 6 月 11 日

功能	API版本	描述	發行日期
AWS 管理組件		AWS 管理組件會連結 Amazon EC2 執行個體以及其中執行的 Windows 或 Linux 作業系統。AWS 管理包是 Microsoft 系統中心操作管理器的擴展。	2013 年 5 月 8 日
新的 Tags (標籤) 頁面		Amazon EC2 控制台中有一個新的標籤頁面。如需詳細資訊，請參閱標記您的 Amazon EC2 資源。	2013 年 4 月 4 日
AMI從一個區域複製到另一個區域	2013-02-01	您可以AMI從一個區域複製到另一個區域，以便在多個 AWS 區域中快速輕鬆地啟動一致的執行個體。 如需詳細資訊，請參閱複製 Amazon EC2 AMI。	2013 年 3 月 11 日
將執行個體啟動為預設值 VPC	2013-02-01	您的 AWS 帳戶能夠將執行個體啟動到 EC2-Classic 或 a 中VPC，或者只能在 region-by-region 基礎上啟動執行個體。VPC如果您只能將執行個體啟動到 aVPC，我們會VPC為您建立預設值。當您啟動執行個體時，我們會將其啟動為預設值VPC，除非您建立非預設值VPC並在啟動執行個體時指定該執行個體。	2013 年 3 月 11 日
EBS快照複製	2012-12-01	您可以使用快照複本來建立資料備份、建立新的 Amazon EBS 磁碟區或建立 Amazon 機器映像 (AMIs)。	2012 年 12 月 17 日
已更新已佈建IOP SSSD磁碟區的EBS指標和狀態檢查	2012-10-01	已更新指EBS標以包含已佈建IOPSSSD磁碟區的兩個新指標。也新增已佈建IOPSSSD磁碟區的新狀態檢查。	2012 年 11 月 20 日
Spot 執行個體請求狀態	2012-10-01	Spot 執行個體請求狀態可讓您輕鬆判斷 Spot 請求的狀態。	2012 年 10 月 14 日

功能	API版本	描述	發行日期
Amazon EC2 預留實例 Marketplace	2012-08-15	預留執行個體 Marketplace 會將擁有不再需要的 Amazon EC2 預留執行個體的賣家與希望購買額外容量的買家進行比對。經由預留執行個體 Marketplace 買賣的預留執行個體運作起來與任何其他預留執行個體相同，不同之處在於可以有低於完整標準的剩餘期限，並能以不同的價格出售。	2012 年 9 月 11 日
IOPSSSD為 Amazon 佈建 EBS	2012-07-20	佈建的IOPSSSD磁碟區可為依賴於一致且快速回應時間的 I/O 密集型工作負載 (例如資料庫應用程式) 提供可預測的高效能。	2012 年 7 月 31 日
IAMAmazon EC2 實例上的角色	2012-06-01	IAMAmazon 的角色EC2提供： • AWS 存取 Amazon EC2 執行個體上執行之應用程式的金鑰。 • 自動輪替 Amazon EC2 執行個體上的 AWS 存取金鑰。 • 針對在 Amazon EC2 執行個體上執行的應用程式提出 AWS 服務請求的精細許可。	2012 年 6 月 11 日

功能	API版本	描述	發行日期
Spot 執行個體功能可讓您更輕鬆地開始處理可能的中斷情況。		您現在可用下列方式管理 競價型執行個體： • 使用 Auto Scaling 啟動組態指定您願意為 競價型執行個體 支付的金額，並設定排程以指定願意為 競價型執行個體 支付的金額。如需詳細資訊，請參閱 Amazon EC2 Auto Scaling 使用者指南中的在自動擴展群組中啟動 Spot 執行個體。 • 啟動或終止執行個體時取得通知。 • 使用 AWS CloudFormation 範本在包含 AWS 資源的堆疊中啟動 Spot 執行個體。	2012 年 6 月 7 日
EC2針對 Amazon 進行狀態檢查的執行個體匯出和時間戳記 EC2	2012-05-01	已新增對匯出您原先匯入的 Windows 伺服器執行個體的支援EC2。 新增對執行個體狀態和系統狀態時間戳記的支援，以指出狀態檢查失敗的日期和時間。	2012 年 5 月 25 日
EC2Amazon 執行個體匯出、執行個體中的時間戳記和系統狀態檢查 VPC	2012-05-01	增加了對EC2實例導出到 Citrix Xen，Microsoft 超 V 和. VMware vSphere 新增對執行個體和系統狀態檢查中時間戳記的支援。	2012 年 5 月 25 日
AWS Marketplace AMIs	2012-04-01	增加了對 AWS Marketplace AMIs.	2012 年 4 月 19 日
預留執行個體定價方案	2011-12-15	新增說明如何利用預留執行個體定價方案內建的折扣定價之章節。	2012 年 3 月 5 日

功能	API版本	描述	發行日期
適用於 Amazon Virtual Private Cloud 中EC2執行個體的彈性網路界面 (ENIs)	2011-12-01	已針對VPC. 中的執行個EC2體新增彈性網路介面 (ENIs) 區段 如需詳細資訊，請參閱彈性網絡介面。	2011 年 12 月 21 日
Amazon EC2 預留執行個體的新產品項目類型	2011-11-01	您可以從各種預留執行個體方案中選擇因應您預期使用的執行個體。	2011 年 12 月 01 日
Amazon EC2 實例狀態	2011-11-01	您可以檢視執行個體狀態的其他詳細資訊，包括可能對執行個體產生影響的排程事件。 AWS 這些操作活動包括套用軟體更新或安全性修補程式所需的執行個體重新開機，或發生硬體問題時所需的執行個體淘汰。如需詳細資訊，請參閱 監控 Amazon EC2 執行個體的狀態。	2011 年 11 月 16 日
Amazon 競價型執行個VPC	2011-07-15	已新增 Amazon 中 Spot 執行個體支援的相關資訊VPC。透過此更新，使用者可以將 Spot 執行個體啟動虛擬私有雲 (VPC)。透過在中啟動 Spot 執行個體VPC，Spot 執行個體的使用者可以享受 Amazon 的好處VPC。	2011 年 10 月 11 日
簡化CLI工具使用者的 VM 匯入程序	2011-07-15	VM Import 程序已透過`ImportInstance` 和的增強功能進行簡化`ImportVolume` ，現在會在建立匯入任務EC2後將映像檔上傳到 Amazon。此外，藉由引進 `ResumeImport` ，使用者可以在任務停止位置重新啟動未完成的上傳。	2011 年 9 月 15 日
Support 以VHD檔案格式匯入		虛擬機 VM Import 現在可以以格式匯入虛擬機器映像VHD檔。該VHD文件格式與思傑 Xen 和 Microsoft Hyper-V 虛擬化平台兼容。在此版本中，VM Import 現在支援RAW以VHD及 VMDK (VMwareESX相容) 映像格式。如需詳細資訊，請參閱 VM Import/Export 使用者指南。	2011 年 8 月 24 日

功能	API版本	描述	發行日期
Amazon EC2 虛擬機器導入連接器的更新 VMware vCenter		已新增 1.1 版適用於VMware vCenter 虛擬設備之 Amazon EC2 VM Import 連接器 (連接器) 的相關資訊。此更新包括對網際網路存取權的代理支援、更佳的錯誤處理、更精確的任務進度列，以及數種錯誤修正。	2011 年 6 月 27 日
競價型執行個體可用區域定價變更	2011-05-15	新增 競價型執行個體可用區域定價功能的相關資訊。在此版本中，我們新增了可用區域定價選項，做為您查詢 Spot 執行個體請求和 Spot 價格歷史記錄時傳回資訊的一部分。這些新增功能可讓您更輕鬆地判斷將 Spot 執行個體啟動至特定可用區域中所需的價格。	2011 年 5 月 26 日
AWS Identity and Access Management		已新增有關 AWS Identity and Access Management (IAM) 的資訊，可讓使用者指定使用者一般可與 Amazon EC2 資源搭配使用的 Amazon EC2 動作。如需詳細資訊，請參閱Amazon 的身份和訪問管理 EC2。	2011 年 4 月 26 日
專用執行個體		在 Amazon 虛擬私有雲 (AmazonVPC) 內啟動的專用執行個體是在主機硬體層級實體隔離的執行個體。專用執行個體可讓您充分利用 Amazon VPC 和 AWS 雲端的優勢，包括隨需彈性佈建和僅按使用量付費，同時在硬體層級隔離 Amazon EC2 運算執行個體。如需詳細資訊，請參閱Amazon EC2 專用實例。	2011 年 3 月 27 日
AWS 管理主控台的預留執行個體更新		AWS 管理主控台的更新可讓使用者更輕鬆地檢視其預留執行個體並購買額外的預留執行個體，包括專用預留執行個體。	2011 年 3 月 27 日
中繼資料資訊	2011-01-01	新增中繼資料的相關資訊，以反映 2011-01-01 版本中的變更。如需詳細資訊，請參閱 使用執行個體中繼資料管理EC2執行個 和 執行個體中繼資料分類。	2011 年 3 月 11 日

功能	API版本	描述	發行日期
Amazon EC2 VM Import 連接器 VMware vCenter		已新增VMwarevCenter 虛擬設備的 Amazon EC2 VM Import 連接器 (連接器) 的相關資訊。連接器是與用VMware vSphere 戶端整合VMware vCenter 的外掛程式，並提供圖形化使用者介面，您可以用來將VMware虛擬機器匯入 Amazon EC2。	2011 年 3 月 3 日
強制磁碟區分離		您現在可以使 AWS Management Console 用強制從執行個體分離 Amazon EBS 磁碟區。	2011 年 2 月 23 日
執行個體終止保護		您現在可以使用 AWS 管理主控台來防止執行個體終止。如需詳細資訊，請參閱啟用終止保護。	2011 年 2 月 23 日
VM Import	2010-11-15	已新增虛擬機 VM Import 的相關資訊，可讓您將虛擬機器或磁碟區匯入 Amazon EC2。如需詳細資訊，請參閱 VM Import/Export 使用者指南。	2010 年 12 月 15 日
執行個體的基本監控	2010-08-31	已新增執行個體基本監視的EC2相關資訊。	2010 年 12 月 12 日
篩選條件和標籤	2010-08-31	新增列出、篩選及建立標籤資源的相關資訊。如需詳細資訊，請參閱 尋找您的 Amazon EC2 資源 和 標記您的 Amazon EC2 資源。	2010 年 9 月 19 日
等冪執行個體啟動	2010-08-31	新增在執行執行個體時確保等冪的相關資訊。	2010 年 9 月 19 日
AWS Identity and Access Management 對於 Amazon EC2		Amazon EC2 現在與 AWS Identity and Access Management （IAM）集成。如需詳細資訊，請參閱Amazon 的身份和訪問管理 EC2。	2010 年 9 月 2 日

功能	API版本	描述	發行日期
Amazon VPC IP 地址指定	2010-06-15	Amazon VPC 使用者現在可以指定 IP 地址，以指派在VPC.	2010 年 7 月 12 日
Amazon CloudWatch 監控 Amazon EBS 卷		Amazon CloudWatch 監控現在可以自動用於 Amazon EBS 卷。	2010 年 6 月 14 日
Windows 預留執行個體		Amazon EC2 現在支援 Windows 的預留執行個體。	2010 年 2 月 22 日

本文為英文版的機器翻譯版本，如內容有任何歧義或不一致之處，概以英文版為準。