使用者指南

AWS CloudShell

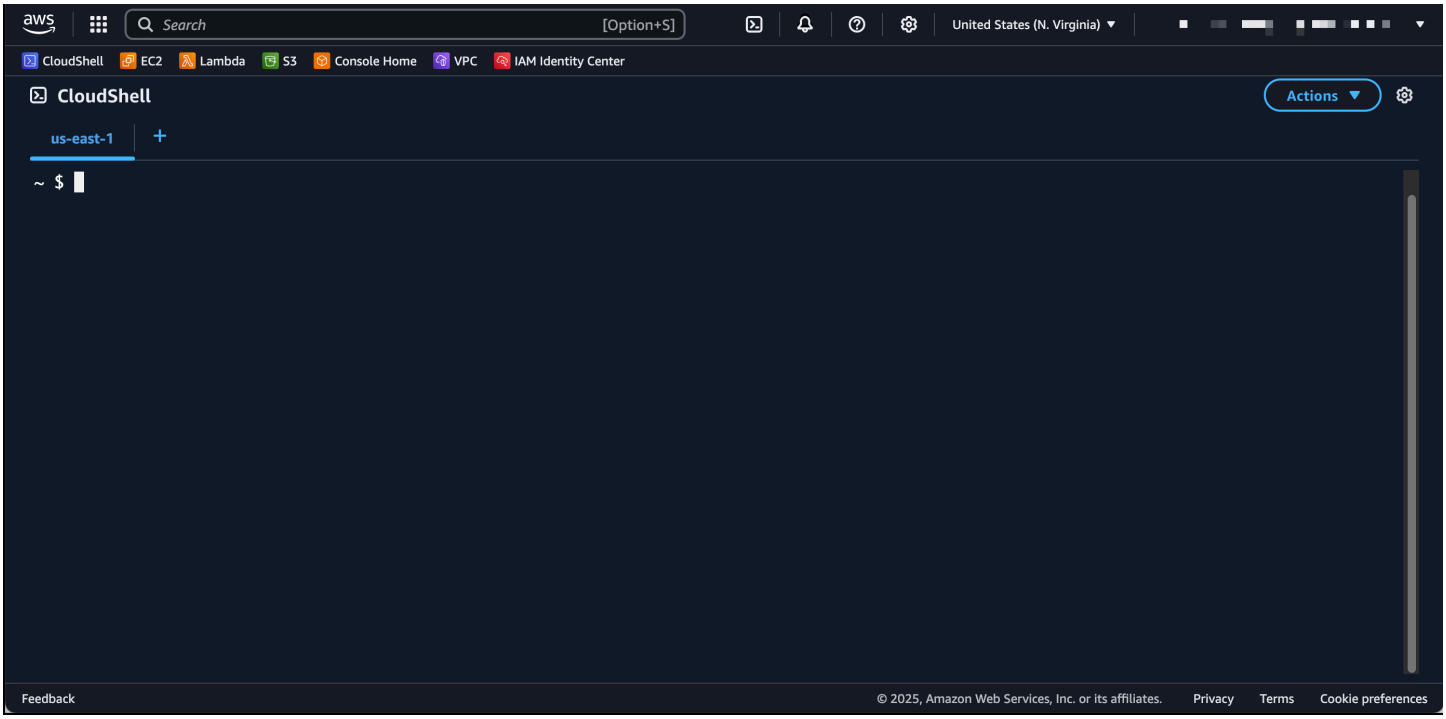
AWS CloudShell: 使用者指南

Table of Contents

什麼是 AWS CloudShell？

AWS CloudShell 是以瀏覽器為基礎的預先驗證 Shell，您可以直接從 啟動 AWS Management Console。您可以從 AWS Management Console 幾種不同的方式導覽至 CloudShell。如需詳細資訊，請參閱 入門 AWS CloudShell

您可以使用您偏好的 shell 執行 AWS CLI 命令，例如 、BashPowerShell 或 Z shell。您也可以執行此操作，而無需下載或安裝命令列工具。



啟動時 AWS CloudShell，會建立以 Amazon Linux 2023 為基礎的運算環境。在此環境中，您可以存取各種預先安裝的開發工具、上傳和下載檔案的選項，以及工作階段之間持續存在的檔案儲存。您可以在最新版本的 Google Chrome、Mozilla Firefox、Microsoft Edge 和 Apple Safari 瀏覽器中使用 CloudShell。

（立即嘗試：入門 AWS CloudShell)

AWS CloudShell的功能

AWS CloudShell 提供以下功能：

AWS Command Line Interface

您可以從 AWS CloudShell 啟動 AWS Management Console。您用來登入主控台的 AWS 登入資料會自動在新的 shell 工作階段中使用。由於 AWS CloudShell 使用者已預先驗證，因此使用 第 2 AWS CLI 版與 AWS 服務 互動時，您不需要設定登入資料。已 AWS CLI 預先安裝在 shell 的運算環境中。

如需 AWS 服務 使用命令列界面與 互動的詳細資訊，請參閱 在 CloudShell 中從 CLI 管理 AWS 服務。

Shell 和開發工具

使用為 AWS CloudShell 工作階段建立的 shell，您可以在偏好的命令列 Shell 之間無縫切換。具體而言，您可以在 Bash、PowerShell 和 之間切換Z shell。您也可以存取預先安裝的工具和公用程式。這些包括 git、make、pip、sudo、tartmux、vim、wget、 和 zip。

shell 環境已預先設定支援多種主要軟體語言，例如 Node.js和 Python。這表示例如，您可以執行 Node.js和 Python專案，而不需要先執行執行時間安裝。PowerShell 使用者可以使用.NET Core執行時間。

如需詳細資訊，請參閱AWS CloudShell 運算環境：規格和軟體。

持久性儲存

使用 AWS CloudShell時，您最多可以在每個 中使用 1 GB 的持久性儲存 AWS 區域 體，無需額外費用。持續性儲存空間位於主目錄 (`$HOME`) 中，而且對您來說是私有的。與每個 shell 工作階段結束後回收的暫時性環境資源不同，主目錄中的資料會保留在工作階段之間。

如需在持久性儲存中保留資料的詳細資訊，請參閱 持久性儲存。

> **Note**
>
> CloudShell VPC 環境沒有持久性儲存。當您的 VPC 環境逾時 （ 閒置 20-30 分鐘後 ），或是您刪除或重新啟動環境時，$HOME 目錄會被刪除。

CloudShell VPC 環境

AWS CloudShell 虛擬私有雲端 (VPC) 可讓您在 VPC 中建立 CloudShell 環境。對於每個 VPC 環境，您可以指派 VPC、新增子網路，並關聯一或多個安全群組。 AWS CloudShell 會繼承 VPC 的網路組態，並可讓您在與 VPC 中其他資源相同的子網路內 AWS CloudShell 安全地使用 。

安全

AWS CloudShell 環境及其使用者受到特定安全功能的保護。這包括 IAM 許可管理、 shell 工作階段限制和文字輸入的安全貼上等功能。

使用 IAM 進行許可管理

身為管理員，您可以使用 IAM 政策授予和拒絕 AWS CloudShell 使用者許可。您也可以建立政策，指定使用者可以使用 shell 環境執行的特定動作。如需詳細資訊，請參閱使用 IAM 政策管理 AWS CloudShell 存取和用量。

Shell 工作階段管理

非作用中和長時間執行的工作階段會自動停止和回收。如需詳細資訊，請參閱Shell 工作階段。

文字輸入的安全貼上

預設會啟用安全貼上。此安全功能需要您驗證要貼入 shell 的多行文字不包含惡意指令碼。如需詳細資訊，請參閱對多行文字使用安全貼上。

自訂選項

您可以根據您的確切偏好自訂 AWS CloudShell 體驗。例如，您可以變更畫面配置 （多個標籤）、顯示的文字大小，並在淺色和深色界面主題之間切換。如需詳細資訊，請參閱自訂您的 AWS CloudShell 體驗。

您也可以安裝自己的軟體，並使用指令碼修改 Shell，以擴展您的 shell 環境。

工作階段還原

工作階段還原功能會還原您在 CloudShell 終端機中單一或多個瀏覽器索引標籤上執行的工作階段。如果您重新整理或重新開啟最近關閉的瀏覽器索引標籤，此功能會繼續工作階段，直到 shell 因非作用中工作階段而停止為止。若要繼續使用 CloudShell 工作階段，請按終端機視窗中的任何鍵。如需 Shell 工作階段的詳細資訊，請參閱 Shell 工作階段。

工作階段還原也會在每個終端機標籤中還原最新的終端機輸出和執行中的程序。

> **Note**
>
> 行動應用程式無法使用工作階段還原。

的定價 AWS CloudShell

AWS CloudShell 是可免費 AWS 服務 使用的 。不過，您需為您執行的其他 AWS 資源付費 AWS CloudShell。此外，也適用標準資料傳輸費率。如需詳細資訊，請參閱 AWS CloudShell 定價。

如需詳細資訊，請參閱的服務配額和限制 AWS CloudShell。

關鍵 AWS CloudShell 主題

- 入門 AWS CloudShell
- AWS CloudShell 概念
- 在 CloudShell 中從 CLI 管理 AWS 服務
- 自訂您的 AWS CloudShell 體驗
- AWS CloudShell 運算環境：規格和軟體

入門 AWS CloudShell

本簡介教學課程說明如何使用 shell 命令列界面啟動 AWS CloudShell 和執行關鍵任務。

首先，您會登入 AWS Management Console 並選取 AWS 區域。然後，在新的瀏覽器視窗中啟動 CloudShell，並在要使用的 Shell 類型中啟動。

接下來，您可以在主目錄中建立新的資料夾，並從本機電腦上傳檔案到其中。您會先使用預先安裝的編輯器處理該檔案，再從命令列將其做為程式執行。最後，您呼叫 AWS CLI 命令來建立 Amazon S3 儲存貯體，並將 檔案新增為儲存貯體的物件。

先決條件

IAM 許可

您可以透過將下列 AWS 受管政策連接至 IAM 身分 （例如使用者、角色或群組） AWS CloudShell 來取得 的許可：

- AWSCloudShellFullAccess：提供使用者對 AWS CloudShell 及其功能的完整存取權。

在本教學課程中，您也可以與 互動 AWS 服務。具體而言，您可以透過建立 Amazon S3 S3 互動。您的 IAM 身分需要至少授予 `s3:CreateBucket`和 `s3:PutObject`許可的政策。

如需詳細資訊，請參閱《Amazon Simple Storage Service 使用者指南》中的 Amazon S3 動作。

練習檔案

本練習也涉及上傳和編輯檔案，然後從命令列界面以程式形式執行。在本機電腦上開啟文字編輯器，並新增下列程式碼片段。

```
import sys
x=int(sys.argv[1])
y=int(sys.argv[2])
sum=x+y
print("The sum is",sum)
```

儲存檔案，並將其命名為 `add_prog.py`。

目錄

步驟 1：登入 AWS Management Console

此步驟涉及輸入您的 IAM 使用者資訊以存取 AWS Management Console。如果您已在主控台中，請跳至步驟 2。

- 您可以使用 IAM AWS Management Console 使用者登入 URL 或前往主登入頁面來存取 。

  IAM user sign-in URL

  - 開啟瀏覽器並輸入下列登入 URL。`account_alias_or_id` 以管理員提供的帳戶別名或帳戶 ID 取代 。

    ```
    https://account_alias_or_id.signin.aws.amazon.com/console/
    ```

  - 輸入您的 IAM 登入憑證，然後選擇登入。

  Main sign-in page

  - 開啟 https://aws.amazon.com/console/。
  - 如果您先前未使用此瀏覽器登入，則會出現主要登入頁面。選擇 IAM 使用者，輸入帳戶別名或帳戶 ID，然後選擇下一步。

- 如果您之前已以 IAM 使用者身分登入。您的瀏覽器可能會記住 的帳戶別名或帳戶 ID AWS 帳戶。若是如此，請輸入您的 IAM 登入憑證，然後選擇登入。

> **Note**
>
> 您也可以以根使用者身分登入。此身分可完整存取帳戶中的所有 AWS 服務 和資源。強烈建議您不要將根使用者用於日常任務，即使是管理任務。反之，請遵循僅以根使用者建立您第一個 IAM 使用者的最佳實務。

步驟 2：選取區域、啟動 AWS CloudShell，然後選擇 shell

在此步驟中，您可以從主控台界面啟動 CloudShell、選擇可用的 AWS 區域，然後切換到您偏好的 shell，例如 Bash、PowerShell 或 Z shell。

1. 若要選擇要 AWS 區域 使用的 ，請前往選取區域選單，然後選取支援的 AWS 區域。（可用區域會反白顯示。）

> **Important**
>
> 如果您切換區域，界面會重新整理，且選取的名稱 AWS 區域 會顯示在命令列文字上方。您新增至持久性儲存體的任何檔案都只能在此相同位置使用 AWS 區域。如果您變更區域，則可以存取不同的儲存體和檔案。

> **Important**
>
> 如果您在 上啟動 CloudShell 時，在所選區域中無法使用 CloudShellConsole Toolbar，則在主控台左下角，則預設區域會設定為最接近所選區域的區域。您可以執行 命令，該命令提供許可來管理與預設區域不同的 區域中的資源。如需詳細資訊，請參閱使用 AWS 區域。
>
> Example
>
> 範例

如果您選擇歐洲（西班牙），eu-south-2但歐洲（西班牙）無法使用 CloudShelleu-south-2，則預設區域會設定為歐洲（愛爾蘭）eu-west-1，最接近歐洲（西班牙）eu-south-2。

您將針對預設區域使用服務配額，歐洲（愛爾蘭），eu-west-1且所有區域都會還原相同的 CloudShell 工作階段。預設區域可能會變更，而且您會在 CloudShell 瀏覽器視窗中收到通知。

2. 從 中 AWS Management Console，您可以選擇下列其中一個選項來啟動 CloudShell：

   1. 在導覽列上，選擇 CloudShell 圖示。
   2. 在搜尋方塊中，輸入「CloudShell」，然後選擇 CloudShell。
   3. 在最近造訪的小工具中，選擇 CloudShell。
   4. 在 主控台左Console Toolbar下角的 上選擇 CloudShell。
      - 您可以拖曳 來調整 CloudShell 工作階段的高度=。
      - 您可以按一下在新瀏覽器索引標籤中開啟，將 CloudShell 工作階段切換為全螢幕。

   出現命令提示時，表示 Shell 已準備好開始互動。

   > **Note**
   >
   > 如果您遇到無法成功啟動或與 互動的問題 AWS CloudShell，請檢查 中的資訊以識別和解決這些問題故障診斷 AWS CloudShell。

3. 若要選擇要使用的預先安裝 Shell，請在命令列提示中輸入其程式名稱。

   Bash

   ```
   bash
   ```

   如果您切換到 Bash，命令提示字元的 符號會更新為 $。

   > **Note**
   >
   > Bash 是啟動時執行的預設 Shell AWS CloudShell。

PowerShell

```
pwsh
```

如果切換至 PowerShell，則命令提示中的符號會更新為 PS>。

Z shell

```
zsh
```

如果您切換到 Z shell，命令提示字元的 符號會更新為 %。

如需有關預先安裝在 shell 環境中的版本的資訊，請參閱 AWS CloudShell 運算環境區段中的 shell 資料表。

步驟 3：從 下載檔案 AWS CloudShell

> **Note**
>
> 此選項不適用於 VPC 環境。

此步驟會逐步引導您下載檔案。

1. 若要下載檔案，請前往動作，然後從選單中選擇下載檔案。

   隨即顯示下載檔案對話方塊。
2. 在下載檔案對話方塊中，輸入要下載的檔案路徑。

> **Note**
>
> 您可以在指定要下載的檔案時使用絕對或相對路徑。使用相對路徑名稱時， 預設會自動/home/cloudshell-user/新增至開頭。因此，若要下載名為 的檔案mydownload-file，下列兩個都是有效的路徑：
>
> - 絕對路徑： /home/cloudshell-user/subfolder/mydownloadfile.txt
> - 相對路徑： subfolder/mydownloadfile.txt

3. 選擇 Download (下載)。

   如果檔案路徑正確，則會顯示對話方塊。您可以使用此對話方塊來開啟具有預設應用程式的檔案。或者，您可以將檔案儲存到本機電腦上的資料夾。

> **Note**
>
> 當您在 上啟動 CloudShell 時，無法使用下載選項Console Toolbar。您可以從 CloudShell 主控台或使用 Chrome Web 瀏覽器下載檔案。

步驟 4：將檔案上傳至 AWS CloudShell

> **Note**
>
> 此選項不適用於 VPC 環境。

此步驟說明如何上傳檔案，然後將其移至主目錄中的新目錄。

1. 若要檢查您目前的工作目錄，請在提示字元輸入下列命令：

   pwd

   當您按下 Enter 時， shell 會傳回您目前的工作目錄 （ 例如，`/home/cloudshell-user`)。
2. 若要將檔案上傳至此目錄，請前往動作，然後從功能表中選擇上傳檔案。

   上傳檔案對話方塊隨即顯示。
3. 選擇 Browse (瀏覽)。
4. 在系統的檔案上傳對話方塊中，選取您為此教學課程 (`add_prog.py`) 建立的文字檔案，然後選擇開啟。
5. 在上傳檔案對話方塊中，選擇上傳。

   進度列會追蹤上傳。如果上傳成功，訊息會確認`add_prog.py`已新增至主目錄的根目錄。
6. 若要建立 檔案的目錄，請輸入 make directoryies 命令：`mkdir mysub_dir`。
7. 若要將上傳的檔案從主目錄的根目錄移至新目錄，請使用 `mv`命令：

```
mv add_prog.py mysub_dir.
```

8. 若要將工作目錄變更為新目錄，請輸入 `cd mysub_dir`。

   命令提示更新，表示您已變更工作目錄。

9. 若要檢視目前目錄的內容，`mysub_dir`請輸入 `ls`命令。

   工作目錄的內容會列出。這包括您剛上傳的檔案。

步驟 5：從 移除檔案 AWS CloudShell

此步驟說明如何從中移除檔案 AWS CloudShell。

1. 若要從 移除檔案 AWS CloudShell，請使用標準 shell 命令，例如 `rm`（移除）。

   ```
   rm my-file-for-removal
   ```

2. 若要移除符合指定條件的多個檔案，請執行 `find`命令。

   下列範例會移除名稱中包含尾碼 ".pdf" 的所有檔案。

   ```
   find -type f -name '*.pdf' -delete
   ```

> **Note**
>
> 假設您在特定 AWS CloudShell 中停止使用 AWS 區域。然後，該區域中的持久性儲存體中的資料會在指定的期間之後自動移除。如需詳細資訊，請參閱持久性儲存。

步驟 6：建立主目錄備份

此步驟說明如何建立主目錄備份。

1. 建立備份檔案

   在主目錄外建立暫存資料夾。

   ```
   HOME_BACKUP_DIR=$(mktemp --directory)
   ```

您可以使用下列其中一個選項來建立備份：

a. 使用 tar 建立備份檔案

若要使用 tar 建立備份檔案，請輸入下列命令：

```
tar \
    --create \
    --gzip \
    --verbose \
    --file=${HOME_BACKUP_DIR}/home.tar.gz \
    [--exclude ${HOME}/.cache] \ // Optional
    ${HOME}/
echo "Home directory backed up to this file: ${HOME_BACKUP_DIR}/home.tar.gz"
```

b. 使用 zip 建立備份檔案

若要使用 zip 建立備份檔案，請輸入下列命令：

```
zip \
    --recurse-paths \
    ${HOME_BACKUP_DIR}/home.zip \
    ${HOME} \
    [--exclude ${HOME}/.cache/\*] // Optional
echo "Home directory backed up to this file: ${HOME_BACKUP_DIR}/home.zip"
```

2. 將備份檔案傳輸至 CloudShell 外部

您可以使用下列其中一個選項，將備份檔案傳輸至 CloudShell 外部：

a. 在本機電腦上下載備份檔案

您可以下載在上一個步驟中建立的檔案。如需如何從 CloudShell 下載檔案的詳細資訊，請參閱從 下載檔案 AWS CloudShell。

在下載檔案對話方塊中，輸入要下載的檔案路徑（例如 `/tmp/tmp.iA99tD9L98/home.tar.gz`)。

b. 將備份檔案傳輸至 S3

若要產生儲存貯體，請輸入下列命令：

```
aws s3 mb s3://${BUCKET_NAME}
```

使用 AWS CLI 將檔案複製到 S3 儲存貯體：

```
aws s3 cp ${HOME_BACKUP_DIR}/home.tar.gz s3://${BUCKET_NAME}
```

> **Note**
>
> 可能需要支付資料傳輸費用。

3. 直接備份至 S3 儲存貯體

   若要直接備份至 S3 儲存貯體，請輸入下列命令：

```
aws s3 cp \
    ${HOME}/ \
    s3://${BUCKET_NAME} \
    --recursive \
    [--exclude .cache/\*] // Optional
```

步驟 7：重新啟動 shell 工作階段

此步驟說明如何重新啟動 shell 工作階段。

> **Note**
>
> 作為安全措施，如果您長時間不使用鍵盤或指標與 shell 互動，工作階段會自動停止。長時間執行的工作階段也會自動停止。如需詳細資訊，請參閱Shell 工作階段。

1. 若要重新啟動 shell 工作階段，請選擇動作、重新啟動。

   您會收到重新啟動 AWS CloudShell 停止目前 中所有作用中工作階段的通知 AWS 區域。

2. 若要確認，請選擇重新啟動。

   界面會顯示 CloudShell 運算環境正在停止的訊息。環境停止並重新啟動後，您可以在新的工作階段中開始使用命令列。

> **Note**
>
> 在某些情況下，您的環境可能需要幾分鐘的時間才能重新啟動。

步驟 8：刪除 shell 工作階段主目錄

此步驟說明如何刪除 shell 工作階段。

> **Note**
>
> 此選項不適用於 VPC 環境。當您重新啟動 VPC 環境時，會刪除其主目錄。

> **Warning**
>
> 刪除主目錄是不可復原的動作，其中存放在主目錄中的所有資料都會永久刪除。不過，在下列情況下，您可能想要考慮此選項：
>
> - 您不正確地修改了檔案，且無法存取 AWS CloudShell 運算環境。刪除主目錄 AWS CloudShell 會返回其預設設定。
> - 您想要 AWS CloudShell 立即從 移除所有資料。如果您在 AWS CloudShell AWS 區域中停止使用 ，除非您在 區域中 AWS CloudShell 再次啟動，否則持久性儲存會在保留期間結束時自動刪除。
>
> 如果您需要長期儲存檔案，請考慮使用 Amazon S3 之類的服務。

1. 若要刪除 Shell 工作階段，請選擇動作、刪除。

   系統會通知您，刪除 AWS CloudShell 主目錄會刪除目前存放在您 AWS CloudShell 環境中的所有資料。

   > **Note**
   >
   > 您無法復原此動作。

2. 若要確認刪除，請在文字輸入欄位中輸入 Delete，然後選擇 Delete。

   AWS CloudShell 會停止目前 中的所有作用中工作階段 AWS 區域。您可以建立新的環境或設定 CloudShell VPC 環境。

3. 若要建立新環境，請選擇開啟標籤。
4. 若要建立 CloudShell VPC 環境，請選擇建立 VPC 環境。

   手動結束 shell 工作階段

   使用命令列，您可以離開 shell 工作階段並使用 `exit`命令登出。然後，您可以按任何鍵重新連線並繼續使用 AWS CloudShell。

步驟 9：編輯檔案的程式碼，並使用命令列執行

此步驟示範如何使用預先安裝的Vim編輯器來使用 檔案。然後，您可以從命令列將該檔案做為程式執行。

1. 若要編輯您在上一個步驟中上傳的檔案，請輸入下列命令：

   ```
   vim add_prog.py
   ```

   shell 界面會重新整理以顯示Vim編輯器。
2. 若要編輯 中的檔案Vim，請按 I鍵。現在編輯內容，讓程式加起來三個數字，而不是兩個數字。

   ```
   import sys
   x=int(sys.argv[1])
   y=int(sys.argv[2])
   z=int(sys.argv[3])
   sum=x+y+z
   print("The sum is",sum)
   ```

   > **Note**
   >
   > 如果您將文字貼入編輯器並啟用安全貼圖功能，則會顯示警告。複製的多行文字可能包含惡意指令碼。使用安全貼上功能，您可以在貼上前驗證完整文字。如果您確信文字是安全的，請選擇貼上。

3. 編輯程式後，按 Esc 進入Vim命令模式。然後，輸入 `:wq`命令來儲存檔案並結束編輯器。

> **Note**
>
> 如果您是初次使用 Vim 命令模式，一開始您可能會發現在命令模式和插入模式之間切換具有挑戰性。儲存檔案和結束應用程式時，會使用命令模式。插入新文字時使用插入模式。若要進入插入模式，請按 I，若要進入命令模式，請按 Esc。如需 Vim 和其他 中可用工具的詳細資訊 AWS CloudShell，請參閱 開發工具和 shell 公用程式。

4. 在主命令列界面上，執行下列程式，並指定三個數字進行輸入。語法如下。

   ```
   python3 add_prog.py 4 5 6
   ```

   命令列會顯示程式輸出：`The sum is 15`。

步驟 10：使用 將檔案 AWS CLI 新增為 Amazon S3 儲存貯體中的物件

在此步驟中，您會建立 Amazon S3 儲存貯體，然後使用 PutObject 方法將程式碼檔案新增為該儲存貯體中的物件。

> **Note**
>
> 本教學課程說明如何在 AWS CLI 中使用 AWS CloudShell 與其他 AWS 服務互動。使用此方法，您不需要下載或安裝任何其他資源。此外，因為您已經在 Shell 中驗證身分，因此無需設定憑證即可呼叫。

1. 若要在指定的 中建立儲存貯體 AWS 區域，請輸入下列命令：

   ```
   aws s3api create-bucket --bucket insert-unique-bucket-name-here --region us-east-1
   ```

   > **Note**
   >
   > 如果您要在 `us-east-1`區域之外建立儲存貯體，`create-bucket-configuration`請使用 `LocationConstraint` 參數新增 以指定 區域。以下為範例語法。

```
$ aws s3api create-bucket --bucket my-bucket --region eu-west-1 --create-
bucket-configuration LocationConstraint=eu-west-1
```

如果呼叫成功，命令列會顯示來自 服務的回應，類似於下列輸出。

```
{
    "Location": "/insert-unique-bucket-name-here"
}
```

> **Note**
>
> 如果您不遵守命名儲存貯體的規則，則會顯示下列錯誤：呼叫 CreateBucket InvalidBucketName) 錯誤：指定的儲存貯體無效。

2. 若要上傳檔案並將檔案新增為物件至您剛建立的儲存貯體，請呼叫 PutObject方法。

```
aws s3api put-object --bucket insert-unique-bucket-name-here --key add_prog --body
 add_prog.py
```

物件上傳到 Amazon S3 儲存貯體後，命令列會顯示來自服務的回應，類似下列輸出：

```
{"ETag": "\"ab123c1:w:wad4a567d8bfd9a1234ebeea56\""}
```

ETag 是存放物件的雜湊。您可以使用此雜湊來檢查上傳至 Amazon S3 之物件的完整性。

相關主題

- 在 CloudShell 中從 CLI 管理 AWS 服務
- 在本機電腦和 CloudShell 之間複製多個檔案
- AWS CloudShell 概念
- 自訂您的 AWS CloudShell 體驗

AWS CloudShell 教學課程

下列教學課程說明如何在使用 時，進行實驗和測試不同的功能和整合 AWS CloudShell。

教學課程概觀	進一步了解
複製多個檔案	the section called “教學課程：複製多個檔案”
建立預先簽章URLs	???
在 AWS CloudShell 內建置 Docker 容器並推送至 Amazon ECR	???
使用 AWS CDK 部署 Lambda 函數	???

在本機電腦和 CloudShell 之間複製多個檔案

本教學課程說明如何在本機電腦和 CloudShell 之間複製多個檔案。

使用 AWS CloudShell 界面，您可以一次在本機電腦和 shell 環境之間上傳或下載單一檔案。若要同時在 CloudShell 和本機電腦之間複製多個檔案，請使用下列其中一個選項：

- Amazon S3：在本機電腦和 CloudShell 之間複製檔案時，使用 S3 儲存貯體做為媒介。
- 壓縮檔案：壓縮單一壓縮資料夾中的多個檔案，可使用 CloudShell 介面上傳或下載。

> **Note**
>
> 由於 CloudShell 不允許傳入網際網路流量，因此目前無法使用 `scp`或 等命令`rsync`，在本機機器和 CloudShell 運算環境之間複製多個檔案。

使用 Amazon S3 上傳和下載多個檔案

此步驟說明如何使用 Amazon S3 上傳和下載多個檔案。

先決條件

若要使用儲存貯體和物件，您需要 IAM 政策來授予執行下列 Amazon S3 API 動作的許可：

- `s3:CreateBucket`
- `s3:PutObject`
- `s3:GetObject`
- `s3:ListBucket`

如需 Amazon S3 動作的完整清單，請參閱 Amazon Simple Storage Service API 參考中的動作。

AWS CloudShell 使用 Amazon S3 將多個檔案上傳至

此步驟說明如何使用 Amazon S3 上傳多個檔案。

1. 在 中 AWS CloudShell，執行下列`s3`命令來建立 S3 儲存貯體：

   ```
   aws s3api create-bucket --bucket your-bucket-name --region us-east-1
   ```

   如果呼叫成功，命令列會顯示來自 S3 服務的回應：

   ```
   {
       "Location": "/your-bucket-name"
   }
   ```

2. 將目錄中的檔案從本機電腦上傳至儲存貯體。選擇下列其中一個選項來上傳檔案：

   - AWS Management Console：使用drag-and-drop將檔案和資料夾上傳至儲存貯體。
   - AWS CLI：使用安裝在本機電腦上的工具版本，使用命令列將檔案和資料夾上傳至儲存貯體。

   Using the console

   - 開啟 Amazon S3 主控台，網址為 http：//https://s3.console.aws.amazon.com/s3/。

     （如果您使用的是 AWS CloudShell，您應該已經登入 主控台。）
   - 在左側導覽窗格中，選擇儲存貯體，然後選擇您要上傳資料夾或檔案的儲存貯體名稱。您也可以選擇建立儲存貯體來建立您選擇的儲存貯體。

- 若要選取您要上傳的檔案和資料夾，請選擇上傳。然後，將您選取的檔案和資料夾拖放到主控台視窗中，其中列出目的地儲存貯體中的物件，或選擇新增檔案或新增資料夾。

  您選擇的檔案會列在 Upload (上傳) 頁面上。
- 選取核取方塊以指出要新增的檔案。
- 若要將選取的檔案新增至儲存貯體，請選擇上傳。

> **Note**
>
> 如需使用主控台時完整範圍組態選項的相關資訊，請參閱《Amazon Simple Storage Service 使用者指南》中的如何將檔案和資料夾上傳至 S3 儲存貯體？。

Using AWS CLI

> **Note**
>
> 針對此選項，您需要在本機電腦上安裝 AWS CLI 工具，並設定 憑證以呼叫 AWS 服務。如需詳細資訊，請參閱《AWS Command Line Interface 使用者指南》https://docs.aws.amazon.com/cli/latest/userguide/。

- 啟動 AWS CLI 工具並執行下列`aws s3`命令，將指定的儲存貯體與本機電腦上目前目錄的內容同步：

```
aws s3 sync folder-path s3://your-bucket-name
```

如果同步成功，則會針對新增至儲存貯體的每個物件顯示上傳訊息。

3. 返回 CloudShell 命令列並輸入下列命令，以同步 shell 環境中的目錄與 S3 儲存貯體的內容：

```
aws s3 sync  s3://your-bucket-name folder-path
```

> **Note**
>
> 您也可以將 `--exclude "<value>"`和 `--include "<value>"` 參數新增至sync命令，以執行模式比對以排除或包含特定檔案或物件。
> 如需詳細資訊，請參閱《 AWS CLI 命令參考》中的使用排除和包含篩選條件。

如果同步成功，則會顯示從 儲存貯體下載到 目錄的每個檔案的下載訊息。

> **Note**
>
> 使用 同步命令時，只會以遞迴方式將新的和更新的檔案從來源目錄複製到目的地。

AWS CloudShell 使用 Amazon S3 從 下載多個檔案

此步驟說明如何使用 Amazon S3 下載多個檔案。

1. 使用 AWS CloudShell 命令列，輸入下列aws s3命令來同步 S3 儲存貯體與 shell 環境中目前目錄的內容：

```
aws s3 sync folder-path s3://your-bucket-name
```

> **Note**
>
> 您也可以將 `--exclude "<value>"`和 `--include "<value>"` 參數新增至sync命令，以執行模式比對以排除或包含特定檔案或物件。
> 如需詳細資訊，請參閱《 AWS CLI 命令參考》中的使用排除和包含篩選條件。

如果同步成功，則會針對新增至儲存貯體的每個物件顯示上傳訊息。

2. 將儲存貯體的內容下載至本機電腦。由於 Amazon S3 主控台不支援下載多個物件，因此您需要使用 AWS CLI 安裝在本機電腦上的工具。

從 AWS CLI 工具的命令列，執行下列命令：

```
aws s3 sync s3://your-bucket-name folder-path
```

如果同步成功，命令列會顯示每個已更新或新增至目的地目錄中的檔案的下載訊息。

> **Note**
>
> 針對此選項，您需要在本機電腦上安裝 AWS CLI 工具，並設定 憑證以呼叫 AWS 服務。如需詳細資訊，請參閱《AWS Command Line Interface 使用者指南》https://docs.aws.amazon.com/cli/latest/userguide/。

使用壓縮資料夾上傳和下載多個檔案

此步驟說明如何使用壓縮資料夾上傳和下載多個檔案。

使用 zip/unzip 公用程式，您可以在 封存中壓縮多個檔案，這些檔案可視為單一檔案。公用程式會預先安裝在 CloudShell 運算環境中。

如需預先安裝工具的詳細資訊，請參閱 開發工具和 shell 公用程式。

AWS CloudShell 使用壓縮資料夾將多個檔案上傳至

此步驟說明如何使用壓縮資料夾上傳多個檔案。

1. 在本機電腦上，新增要上傳至壓縮資料夾的檔案。
2. 啟動 CloudShell，然後選擇動作、上傳檔案。
3. 在上傳檔案對話方塊中，選擇選取檔案，然後選擇您剛建立的壓縮資料夾。
4. 在上傳檔案對話方塊中，選擇上傳，將選取的檔案新增至 shell 環境。
5. 在 CloudShell 命令列中，執行下列命令，將 zip 封存的內容解壓縮至指定的目錄：

```
unzip zipped-files.zip -d my-unzipped-folder
```

AWS CloudShell 使用壓縮資料夾從 下載多個檔案

此步驟說明如何使用壓縮資料夾下載多個檔案。

1. 在 CloudShell 命令列中，執行下列命令，將目前目錄中的所有檔案新增至壓縮資料夾：

```
zip -r zipped-archive.zip *
```

2. 選擇動作、下載檔案。
3. 在下載檔案對話方塊中，輸入壓縮資料夾的路徑（例如，)/home/cloudshell-user/zip-folder/zipped-archive.zip，然後選擇下載。

   如果路徑正確，瀏覽器對話方塊會提供開啟壓縮資料夾或將其儲存至本機電腦的選項。
4. 在本機電腦上，您現在可以解壓縮下載的壓縮資料夾內容。

使用 CloudShell 建立 Amazon S3 物件的預先簽章 URL

本教學課程說明如何建立預先簽章的 URL，以與他人共用 Amazon S3 物件。由於物件擁有者在共用時指定自己的安全登入資料，因此接收預先簽章 URL 的任何人都可以在有限的時間內存取物件。

先決條件

- 具有 AWSCloudShellFullAccess 政策所提供存取許可的 IAM 使用者。
- 如需建立預先簽章 URL 所需的 IAM 許可，請參閱《Amazon Simple Storage Service 使用者指南》中的與他人共用物件。

步驟 1：建立 IAM 角色以授予對 Amazon S3 儲存貯體的存取權

此步驟說明如何建立 IAM 角色來授予 Amazon S3 儲存貯體的存取權。

1. 若要取得可共用的 IAM 詳細資訊，請呼叫 get-caller-identity命令 AWS CloudShell。

   ```
   aws sts get-caller-identity
   ```

   如果呼叫成功，命令列會顯示類似以下的回應。

   ```
   {
       "Account": "123456789012",
       "UserId": "AROAXXOZUUOTTWDCVIDZ2:redirect_session",
       "Arn": "arn:aws:sts::531421766567:assumed-role/Feder08/redirect_session"
   }
   ```
2. 取得您在上一個步驟中取得的使用者資訊，並將其新增至 AWS CloudFormation 範本。此範本會建立 IAM 角色。此角色會授予您協作者共用資源的最低權限許可。

   ```
   Resources:
   ```

```
  CollaboratorRole:
    Type: AWS::IAM::Role
    Properties:
      AssumeRolePolicyDocument:
        Version: 2012-10-17
        Statement:
          - Effect: Allow
            Principal:
              AWS: "arn:aws:iam::531421766567:role/Feder08"
            Action: "sts:AssumeRole"
      Description: Role used by my collaborators
      MaxSessionDuration: 7200
  CollaboratorPolicy:
    Type: AWS::IAM::Policy
    Properties:
      PolicyDocument:
        Version: 2012-10-17
        Statement:
          - Effect: Allow
            Action:
              - 's3:*'
            Resource: 'arn:aws:s3:::<YOUR_BUCKET_FOR_FILE_TRANSFER>'
            Condition:
              StringEquals:
                s3:prefix:
                 - "myfolder/*"
      PolicyName: S3ReadSpecificFolder
      Roles:
        - !Ref CollaboratorRole
Outputs:
  CollaboratorRoleArn:
    Description: Arn for the Collaborator's Role
    Value: !GetAtt CollaboratorRole.Arn
```

3. 將 AWS CloudFormation 範本儲存在名為 的檔案中`template.yaml`。
4. 使用 範本來部署堆疊，並透過呼叫 `deploy`命令來建立 IAM 角色。

```
aws cloudformation deploy --template-file ./template.yaml --stack-name
 CollaboratorRole --capabilities CAPABILITY_IAM
```

產生預先簽章的 URL

此步驟說明如何產生預先簽章的 URL。

1. 在 中使用編輯器 AWS CloudShell，新增下列程式碼。此程式碼會建立 URL，讓聯合身分使用者能夠直接存取 AWS Management Console。

```
import urllib, json, sys
import requests
import boto3
import os

def main():
  sts_client = boto3.client('sts')
  assume_role_response = sts_client.assume_role(
      RoleArn=os.environ.get(ROLE_ARN),
      RoleSessionName="collaborator-session"
  )
  credentials = assume_role_response['Credentials']
  url_credentials = {}
  url_credentials['sessionId'] = credentials.get('AccessKeyId')
  url_credentials['sessionKey'] = credentials.get('SecretAccessKey')
  url_credentials['sessionToken'] = credentials.get('SessionToken')
  json_string_with_temp_credentials = json.dumps(url_credentials)
  print(f"json string {json_string_with_temp_credentials}")

  request_parameters = f"?
Action=getSigninToken&Session={urllib.parse.quote(json_string_with_temp_credentials)}"
  request_url = "https://signin.aws.amazon.com/federation" + request_parameters
  r = requests.get(request_url)
  signin_token = json.loads(r.text)
  request_parameters = "?Action=login"
  request_parameters += "&Issuer=Example.org"
  request_parameters += "&Destination=" + urllib.parse.quote("https://us-
west-2.console.aws.amazon.com/cloudshell")
  request_parameters += "&SigninToken=" + signin_token["SigninToken"]
  request_url = "https://signin.aws.amazon.com/federation" + request_parameters

  # Send final URL to stdout
  print (request_url)

if __name__ == "__main__":
```

```
  main()
```

2. 將程式碼儲存在名為 的檔案中share.py。
3. 從命令列執行下列命令，從中擷取 IAM 角色的 Amazon Resource Name (ARN) AWS CloudFormation。然後，在Python指令碼中使用它來取得臨時安全登入資料。

```
ROLE_ARN=$(aws cloudformation describe-stacks --stack-name CollaboratorRole --query
 "Stacks[*].Outputs[?OutputKey=='CollaboratorRoleArn'].OutputValue" --output text)
 python3 ./share.py
```

指令碼會傳回協作者可以按一下的 URL，以便在 AWS CloudShell 中將其帶至 AWS Management Console。協作者可以在接下來的 3，600 秒 (1 小時） 內完全控制 Amazon S3 儲存貯體中的myfolder/資料夾。登入資料會在一小時後過期。經過這段時間後，協作者就無法再存取儲存貯體。

在 CloudShell 內建置 Docker 容器，並將其推送至 Amazon ECR 儲存庫

本教學課程說明如何在 中定義和建置 Docker 容器， AWS CloudShell 並將其推送至 Amazon ECR 儲存庫。

先決條件

- 您必須擁有必要的許可，才能建立和推送至 Amazon ECR 儲存庫。如需使用 Amazon ECR 儲存庫的詳細資訊，請參閱《Amazon ECR 使用者指南》中的 Amazon ECR 私有儲存庫。 如需使用 Amazon ECR 推送映像所需的許可的詳細資訊，請參閱《Amazon ECR 使用者指南》中的推送映像所需的 IAM 許可。

教學課程程序

下列教學課程概述如何使用 CloudShell 介面建置 Docker 容器，並將其推送至 Amazon ECR 儲存庫。

1. 在主目錄中建立新的資料夾。

```
mkdir ~/docker-cli-tutorial
```

2. 導覽至您建立的資料夾。

```
cd ~/docker-cli-tutorial
```

3. 建立空的 Dockerfile。

```
touch Dockerfile
```

4. 使用文字編輯器，例如 `nano Dockerfile`，開啟 檔案並將下列內容貼到其中。

```
# Dockerfile

# Base this container on the latest Amazon Linux version
FROM public.ecr.aws/amazonlinux/amazonlinux:latest

# Install the cowsay binary
RUN dnf install --assumeyes cowsay

# Default entrypoint binary
ENTRYPOINT [ "cowsay" ]

# Default argument for the cowsay entrypoint
CMD [ "Hello, World!" ]
```

5. Dockerfile 現在已準備好建置。執行 來建置容器`docker build`。使用easy-to-type的名稱標記容器，以供未來命令使用。

```
docker build --tag test-container .
```

請務必包含結尾句號 (`.`)。

6. 您現在可以測試容器，以檢查其是否正確執行 AWS CloudShell。

```
docker container run test-container
```

7. 現在您已擁有正常運作的 Docker 容器，您需要將其推送至 Amazon ECR 儲存庫。如果您有現有的 Amazon ECR 儲存庫，您可以略過此步驟。

執行下列命令來建立本教學課程的 Amazon ECR 儲存庫。

```
ECR_REPO_NAME=docker-tutorial-repo
aws ecr create-repository --repository-name ${ECR_REPO_NAME}
```

8. 建立 Amazon ECR 儲存庫之後，您可以將 Docker 容器推送到該儲存庫。

   執行下列命令以取得 Docker 的 Amazon ECR 登入憑證。

```
AWS_ACCOUNT_ID=$(aws sts get-caller-identity --query "Account" --output text)
ECR_URL=${AWS_ACCOUNT_ID}.dkr.ecr.${AWS_REGION}.amazonaws.com
aws ecr get-login-password | docker login --username AWS --password-stdin
 ${ECR_URL}
```

   > **Note**
   >
   > 如果未在 CloudShell 中設定AWS_REGION環境變數，或您想要與其他 中的資源互動 AWS 區域，請執行下列命令：
   >
   > ```
   > AWS_REGION=<your-desired-region>
   > ```

9. 使用目標 Amazon ECR 儲存庫標記映像，然後將其推送至該儲存庫。

```
docker tag test-container ${ECR_URL}/${ECR_REPO_NAME}
docker push ${ECR_URL}/${ECR_REPO_NAME}
```

   如果您在嘗試完成本教學課程時遇到錯誤或遇到問題，請參閱本指南的疑難排解一節以取得協助。

清除

您現在已成功將 Docker 容器部署至 Amazon ECR 儲存庫。若要從 AWS CloudShell 環境中移除您在本教學課程中建立的檔案，請執行下列命令。

- ```
  cd ~
  rm -rf ~/docker-cli-tutorial
  ```

- 刪除 Amazon ECR 儲存庫。

```
aws ecr delete-repository --force --repository-name ${ECR_REPO_NAME}
```
```

在 CloudShell AWS CDK 中使用 部署 Lambda 函數

本教學課程說明如何使用 CloudShell AWS Cloud Development Kit (AWS CDK) 中的 將 Lambda 函數部署至您的帳戶。

先決條件

- 引導您的帳戶以搭配 使用 AWS CDK。如需使用 引導的詳細資訊 AWS CDK，請參閱《 AWS CDK v2 開發人員指南》中的引導。如果您尚未引導帳戶，您可以在 CloudShell `cdk bootstrap`中執行 。

- 請確定您擁有將資源部署到 帳戶的適當許可。建議使用管理員許可。

教學課程程序

下列教學課程概述如何在 CloudShell AWS CDK 中使用 部署 Docker 容器型 Lambda 函數。

1. 在主目錄中建立新的資料夾。

```
mkdir ~/docker-cdk-tutorial
```

2. 導覽至您建立的資料夾。

```
cd ~/docker-cdk-tutorial
```

3. 在本機安裝 AWS CDK 相依性。

```
npm install aws-cdk aws-cdk-lib
```

4. 在您建立的資料夾中建立骨架 AWS CDK 專案。

```
touch cdk.json
mkdir lib
touch lib/docker-tutorial.js lib/Dockerfile lib/hello.js
```

5. 使用文字編輯器，例如 `nano cdk.json`，開啟 檔案並將下列內容貼到其中。

```
{
  "app": "node lib/docker-tutorial.js"
```

```
}
```

6. 開啟 lib/docker-tutorial.js 檔案，並將下列內容貼入其中。

```
// this file defines the CDK constructs we want to deploy

const { App, Stack } = require('aws-cdk-lib');
const { DockerImageFunction, DockerImageCode } = require('aws-cdk-lib/aws-lambda');
const path = require('path');

// create an application
const app = new App();

// define stack
class DockerTutorialStack extends Stack {
  constructor(scope, id, props) {
    super(scope, id, props);

    // define lambda that uses a Docker container
    const dockerfileDir = path.join(__dirname);
    new DockerImageFunction(this, 'DockerTutorialFunction', {
      code: DockerImageCode.fromImageAsset(dockerfileDir),
      functionName: 'DockerTutorialFunction',
    });
  }
}

// instantiate stack
new DockerTutorialStack(app, 'DockerTutorialStack');
```

7. 開啟 lib/Dockerfile，並將下列內容貼入其中。

```
# Use a NodeJS 20.x runtime
FROM public.ecr.aws/lambda/nodejs:20

# Copy the function code to the LAMBDA_TASK_ROOT directory
# This environment variable is provided by the lambda base image
COPY hello.js ${LAMBDA_TASK_ROOT}

# Set the CMD to the function handler
CMD [ "hello.handler" ]
```

8. 開啟 lib/hello.js 檔案，並將下列內容貼入其中。

```
// define the handler
exports.handler = async (event) => {
  // simply return a friendly success response
  const response = {
    statusCode: 200,
    body: JSON.stringify('Hello, World!'),
  };
  return response;
};
```

9. 使用 AWS CDK CLI 合成專案並部署資源。您必須引導您的帳戶。

```
npx cdk synth
npx cdk deploy --require-approval never
```

10. 叫用 Lambda 函數來確認和驗證。

```
aws lambda invoke --function-name DockerTutorialFunction out.json
jq . out.json
```

您現在已成功使用 部署 Docker 容器型 Lambda 函數 AWS CDK。如需詳細資訊 AWS CDK，請參閱 AWS CDK v2 開發人員指南。如果您在嘗試完成本教學課程時遇到錯誤或遇到問題，請參閱本指南的疑難排解一節以取得協助。

清除

您現在已成功使用 部署 Docker 容器型 Lambda 函數 AWS CDK。在 AWS CDK 專案中，執行下列命令來刪除相關聯的資源。系統會提示您確認刪除。

-
```
npx cdk destroy DockerTutorialStack
```

- 若要從 AWS CloudShell 環境中移除您在本教學課程中建立的檔案和資源，請執行下列命令。

```
cd ~
rm -rf ~/docker-cli-tutorial
```

AWS CloudShell 概念

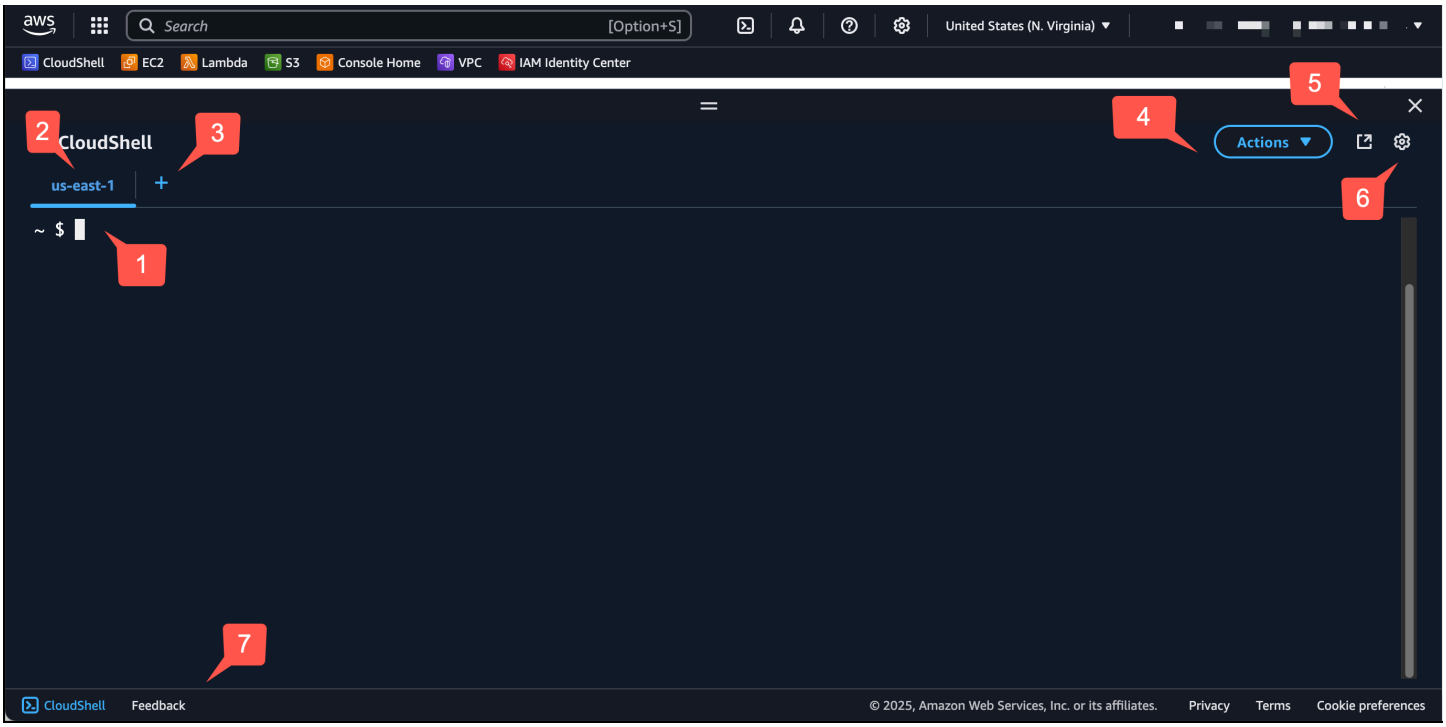
本節說明如何與支援的應用程式互動 AWS CloudShell 並執行特定動作。

主題

導覽 AWS CloudShell 界面

您可以從 AWS Management Console 和 導覽 CloudShell 介面功能Console Toolbar。

下列螢幕擷取畫面指出數個關鍵 AWS CloudShell 界面功能。



1. AWS CloudShell 命令列界面，您可以使用您偏好的 shell 來執行命令。命令提示字元會指出目前的 shell 類型。

2. 終端機標籤，使用 AWS CloudShell 目前執行 AWS 區域 的 。
3. + 圖示是一個下拉式選單，其中包含建立、重新啟動和刪除環境的選項。
4. 動作功能表提供變更畫面配置、下載和上傳檔案、重新啟動 AWS CloudShell和刪除 AWS CloudShell 主目錄的選項。

> **Note**
>
> 當您在 上啟動 CloudShell 時，無法使用下載選項Console Toolbar。

5. 在新的瀏覽器中開啟索引標籤，提供在全螢幕中存取 CloudShell 工作階段的選項。
6. 偏好設定選項，您可以用來自訂您的 shell 體驗。
7. 底部列提供下列選項：
   - 從 CloudShell 圖示啟動 CloudShell。
   - 從意見回饋圖示提供意見回饋。選擇您要提交的意見回饋類型、新增評論，然後選擇提交。
     - 若要提交 CloudShell 的意見回饋，請選擇下列其中一個選項：
       - 從主控台啟動 CloudShell，然後選擇意見回饋。新增您的評論，然後選擇提交。
       - 在 主控台左Console Toolbar下角選擇 CloudShell，然後選擇在新的瀏覽器索引標籤中開啟圖示，Feedback。新增您的評論，然後選擇提交。

> **Note**
>
> 當您在 上啟動 CloudShell 時，無法使用意見回饋選項Console Toolbar。

   - 了解我們的隱私權政策和使用條款，並自訂 Cookie 偏好設定。

使用 AWS 區域

您目前在 中執行 AWS 區域 的 會顯示為標籤。

您可以使用區域選取器選取特定區域，以選擇要在其中 AWS 區域 工作的 。變更區域後，界面會在 Shell 工作階段連線到所選區域中執行的不同運算環境時重新整理。

Important

- 每個 最多可使用 1 GB 的持久性儲存 AWS 區域。持久性儲存會存放在您的主目錄 () 中$HOME。這表示存放在您主目錄中的任何個人檔案、目錄、程式或指令碼都位於一個 中 AWS 區域。此外，它們與位於主目錄中並存放不同 區域的 不同。

  持久性儲存中檔案的長期保留也會按區域管理。如需詳細資訊，請參閱持久性儲存。
- 持久性儲存不適用於 AWS CloudShell VPC 環境。

指定 的預設值 AWS 區域AWS CLI

您可以使用環境變數來指定 AWS 服務 使用 存取所需的組態選項和登入資料 AWS CLI。當您從 中的特定區域啟動 AWS CloudShell 時， AWS Management Console 或當您在區域選擇器中選擇選項時，指定 AWS 區域 shell 工作階段預設值的環境變數會設定在 中。

環境變數的優先順序高於 更新的 AWS CLI 登入資料檔案aws configure。因此，您無法執行 aws configure命令來變更環境變數指定的區域。反之，若要變更 AWS CLI 命令的預設區域，請將值指派給AWS_REGION環境變數。在下列範例中，將 us-east-1取代為您所在的區域。

Bash or Zsh

```
$ export AWS_REGION=us-east-1
```

設定環境變數會變更所使用的值，直到 Shell 工作階段結束時或將變數設定為不同的值為止。您可以在 shell 的啟動指令碼中設定變數，讓變數在未來的工作階段中持續存在。

PowerShell

```
PS C:\> $Env:AWS_REGION="us-east-1"
```

如果您在 PowerShell 提示字元設定環境變數，環境變數只會在目前工作階段的持續時間內儲存該值。或者，您可以將變數新增至 PowerShell 設定檔，以設定所有未來 PowerShell 工作階段的變數。如需儲存環境變數的詳細資訊，請參閱 PowerShell 文件。

若要確認您已變更預設區域，請執行 aws configure list命令以顯示目前的 AWS CLI 組態資料。

> **Note**
>
> 對於特定 AWS CLI 命令，您可以使用命令列選項 覆寫預設區域`--region`。如需詳細資訊，請參閱AWS Command Line Interface 《 使用者指南》中的命令列選項。

使用檔案和儲存

使用 AWS CloudShell的界面，您可以將檔案上傳到 shell 環境並從中下載檔案。如需下載和上傳檔案的詳細資訊，請參閱 入門 AWS CloudShell。

為了確保您在工作階段結束後新增的任何檔案都可用，您應該知道持久性和暫時儲存之間的差異。

- 持久性儲存：您每個都有 1 GB 的持久性儲存 AWS 區域。持久性儲存位於您的主目錄中。
- 暫時儲存：暫時儲存會在工作階段結束時回收。暫時儲存位於主目錄以外的目錄中。

> **Important**
>
> 請務必保留要保留的檔案，並用於主目錄中的未來 shell 工作階段。例如，假設您執行 `mv`命令，將檔案移出主目錄。然後，當目前的 shell 工作階段結束時，該檔案會回收。

在主控台行動應用程式中存取 CloudShell

您可以從 AWS Console Mobile Application 主畫面存取 中的 CloudShell。從主畫面中，您可以檢視 CloudShell 和其他 AWS 服務的相關資訊。如需詳細資訊，請參閱 AWS Console Mobile Application入門。若要在 中啟動 CloudShell AWS Console Mobile Application，請選擇下列其中一個選項：

- 選取導覽列底部的 CloudShell 圖示。
- 選取服務功能表上的 CloudShell。

您可以隨時選擇 X 來結束 CloudShell。

如需在主控台行動應用程式中存取 CloudShell 的詳細資訊，請參閱存取 AWS CloudShell。

> **Note**
>
> 目前，您無法在 中建立或啟動 VPC 環境 AWS Console Mobile Application。

使用 Docker

AWS CloudShell 完全支援 Docker，無需安裝或設定。您可以在內部定義、建置和執行 Docker 容器 AWS CloudShell。您可以透過 AWS CDK Toolkit 部署以 Docker 為基礎的資源，例如以 Docker 容器為基礎的 Lambda 函數，以及建置 Docker 容器，並透過 Docker CLI 將其推送至 Amazon ECR 儲存庫。如需如何執行這兩個部署的詳細步驟，請參閱下列教學課程：

- 教學課程：使用 部署 Lambda 函數 AWS CDK
- 教學課程：在內部建置 Docker 容器 AWS CloudShell 並將其推送至 Amazon ECR 儲存庫

搭配下列項目使用 Docker 有特定限制 AWS CloudShell：

- Docker 在 環境中的空間有限。如果您有大型的個別映像，或預先存在太多的 Docker 映像，可能會導致您無法提取、建置或執行其他映像的問題。如需 Docker 的詳細資訊，請參閱 Docker 文件指南。
- Docker 適用於所有 AWS 區域，但 AWS GovCloud (US) 區域除外。如需可使用 Docker 的區域清單，請參閱 支援的 AWS 區域 AWS CloudShell。
- 如果您在搭配 使用 Docker 時遇到問題 AWS CloudShell，請參閱本指南的疑難排解一節，以取得如何可能解決這些問題的資訊。

的存取功能 AWS CloudShell

本主題說明如何使用 CloudShell 的存取功能。您可以使用鍵盤瀏覽頁面上的可聚焦元素。您也可以自訂 CloudShell 的外觀，包括字型大小和界面主題。

CloudShell 中的鍵盤導覽

若要瀏覽頁面上的可聚焦元素，請按 Tab。

CloudShell 終端機存取功能

您可以在下列模式中使用 Tab金鑰：

- 終端機模式（預設）– 在此模式中，終端機會擷取您的Tab金鑰項目。焦點在終端機上之後，按 Tab 以僅存取終端機的功能。
- 導覽模式 – 在此模式中，終端機不會擷取您的Tab金鑰項目。按下 Tab 以瀏覽頁面上的可聚焦元素。

若要在終端機模式和導覽模式之間切換，請按 Ctrl + M。切換回後，標籤：導覽會顯示在 標頭中，您可以使用 Tab金鑰導覽頁面。

若要返回終端機模式，請按 Ctrl + M。或者，選擇 Tab X旁的導覽。

> **Note**
>
> 目前，CloudShell 終端機存取功能不適用於行動裝置。

在 CloudShell 中選擇字型大小和界面主題

您可以自訂 CloudShell 的外觀，以適應您的視覺效果偏好設定。

- 字型大小 – 選擇終端機中最小、小型、中型、大型和最大的字型大小。如需變更字型大小的詳細資訊，請參閱 the section called “變更字型大小”。
- 佈景主題：選擇淺色和深色界面佈景主題。如需變更界面主題的詳細資訊，請參閱 the section called “變更界面主題”。

在 CloudShell 中從 CLI 管理 AWS 服務

的主要優點 AWS CloudShell 是您可以使用它從命令列界面管理您的 AWS 服務。這表示您不需要事先在本機下載和安裝工具或設定登入資料。當您啟動時 AWS CloudShell，會建立已安裝下列 AWS 命令列工具的運算環境：

- AWS CLI
- AWS Elastic Beanstalk CLI
- Amazon ECS CLI
- AWS SAM

由於您已登入 AWS，因此在使用 服務之前，不需要在本機設定您的登入資料。您用來登入 的登入資料 AWS Management Console 會轉送至 AWS CloudShell。

如果您想要變更用於 的預設 AWS 區域 AWS CLI，您可以變更指派給`AWS_REGION`環境變數的值。(如需詳細資訊，請參閱 指定 的預設值 AWS 區域AWS CLI。)

本主題的其餘部分示範如何開始使用 AWS CloudShell 從命令列與所選 AWS 服務互動。

AWS CLI 所選 AWS 服務的命令列範例

下列範例僅代表您可以使用第 2 AWS CLI 版提供的命令使用的一些 AWS 服務。如需完整清單，請參閱 AWS CLI 命令參考。

- DynamoDB
- Amazon EC2
- S3 Glacier

DynamoDB

DynamoDB 是全受管 NoSQL 資料庫服務，提供快速且可預期的效能，以及無縫的可擴展性。此服務的 NoSQL 模式實作支援鍵值和文件資料結構。

下列`create-table`命令會建立`MusicCollection`在您的 AWS 帳戶中名為 的 NoSQL 樣式資料表。

```
aws dynamodb create-table \
```

```
    --table-name MusicCollection \
    --attribute-definitions AttributeName=Artist,AttributeType=S
AttributeName=SongTitle,AttributeType=S \
    --key-schema AttributeName=Artist,KeyType=HASH
AttributeName=SongTitle,KeyType=RANGE \
    --provisioned-throughput ReadCapacityUnits=5,WriteCapacityUnits=5 \
    --tags Key=Owner,Value=blueTeam
```

如需詳細資訊，請參閱AWS Command Line Interface 《 使用者指南》中的搭配 使用 DynamoDB AWS CLI。

Amazon EC2

Amazon Elastic Compute Cloud (Amazon EC2) 是一種 Web 服務，可在雲端中提供安全且可調整大小的運算容量。它旨在讓 Web 規模的雲端運算更容易且更易於存取。

下列`run-instances`命令會在 VPC 的指定子網路中啟動 t2.micro 執行個體：

```
aws ec2 run-instances --image-id ami-xxxxxxxx --count 1 --instance-type t2.micro --key-
name MyKeyPair --security-group-ids sg-903004f8 --subnet-id subnet-6e7f829e
```

如需詳細資訊，請參閱AWS Command Line Interface 《 使用者指南》中的搭配使用 Amazon EC2 與 AWS CLI。

S3 Glacier

S3 Glacier 和 S3 Glacier Deep Archive 是安全、耐用且成本極低的 Amazon S3 雲端儲存類別，可用於資料封存和長期備份。

下列`create-vault`命令會建立保存庫 - 用於存放封存的容器：

```
aws glacier create-vault --vault-name my-vault --account-id -
```

如需詳細資訊，請參閱《 使用者指南》中的搭配 使用 Amazon S3 Glacier AWS CLI。 AWS Command Line Interface

AWS Elastic Beanstalk CLI

CLI AWS Elastic Beanstalk 提供命令列界面，可簡化從本機儲存庫建立、更新和監控環境。在這種情況下，環境是指執行應用程式版本 AWS 的資源集合。

下列`create`命令會在自訂 Amazon Virtual Private Cloud (VPC) 中建立新的環境。

```
$ eb create dev-vpc --vpc.id vpc-0ce8dd99 --vpc.elbsubnets subnet-
b356d7c6,subnet-02f74b0c --vpc.ec2subnets subnet-0bb7f0cd,subnet-3b6697c1 --
vpc.securitygroup sg-70cff265
```

如需詳細資訊，請參閱《 AWS Elastic Beanstalk 開發人員指南》中的 EB CLI 命令參考。

Amazon ECS CLI

Amazon Elastic Container Service (Amazon ECS) 命令列界面 (CLI) 提供數個高階命令。這些旨在簡化從本機開發環境建立、更新和監控叢集和任務的程序。(Amazon ECS 叢集是任務或服務邏輯分組。)

下列`configure`命令會將 Amazon ECS CLI 設定為建立名為 的叢集組態`ecs-cli-demo`。此叢集組態使用 `FARGATE`做為 中`ecs-cli-demo`叢集的預設啟動類型`us-east-1 region`。

```
ecs-cli configure --region us-east-1 --cluster ecs-cli-demo --default-launch-type
 FARGATE --config-name ecs-cli-demo
```

如需詳細資訊，請參閱 Amazon Elastic Container Service 開發人員指南中的 Amazon ECS 命令列參考。

AWS SAM CLI

AWS SAM CLI 是一種命令列工具，可在 AWS Serverless Application Model 範本和應用程式程式碼上操作。您可以使用它來執行數個任務。這包括在本機叫用 Lambda 函數、為無伺服器應用程式建立部署套件，以及將無伺服器應用程式部署至 AWS 雲端。

下列`init`命令會使用做為參數傳遞的必要參數來初始化新的 SAM 專案：

```
sam init --runtime python3.9 --dependency-manager pip --app-template hello-world --name
 sam-app
```

如需詳細資訊，請參閱《 AWS Serverless Application Model 開發人員指南》中的 AWS SAM CLI 命令參考。

在 CloudShell 中使用 Amazon Q CLI

> **Important**
>
> AWS CloudShell 由於內部問題， 暫時停用 Amazon Q 聊天功能。我們正在積極調查，並會盡快還原此功能。同時，您可以繼續使用 中的 Q 聊天 AWS Management Console。

Amazon Q CLI 是一種命令列界面，可讓您與 Amazon Q 互動。如需詳細資訊，請參閱《Amazon Q 開發人員使用者指南》中的在命令列上使用 Amazon Q 開發人員。

CloudShell 中的 Amazon Q CLI 可讓您在自然語言對話中互動、提出問題，並從終端機接收來自 Amazon Q 的回應。您可以取得相關的 shell 命令，減少在終端機中輸入 時搜尋、記住語法和接收命令建議的需求。

> **Note**
>
> 目前，CloudShell 中的 Amazon Q CLI 功能無法在 CloudShell VPC 環境中使用。

如果您在 CloudShell 中看不到 Amazon Q CLI 功能，請聯絡您的管理員以提供您 IAM 許可。如需詳細資訊，請參閱《Amazon Q 開發人員使用者指南》中的 Amazon Q 開發人員的身分型政策範例。

本章說明如何在 CloudShell 中使用 Amazon Q CLI 功能。

在 CloudShell 中使用 Amazon Q 內嵌建議

CloudShell 中的 Amazon Q 內嵌建議會在您輸入終端機時為您提供命令建議。如需詳細資訊，請參閱《Amazon Q 開發人員使用者指南》中的命令列上的內嵌 Amazon Q。

在 CloudShell 中使用 Amazon Q 內嵌建議

1. 從 AWS Management Console選擇 CloudShell。
2. 在 CloudShell 終端機上，切換到 Z shell，然後開始輸入。若要切換到 Z shell，請在終端機`zsh`中輸入 ，然後按 Enter 鍵。

> **Note**
>
> 目前，Amazon Q 內嵌僅支援 Z shell。

當您開始輸入命令時，Amazon Q 會根據您目前的輸入和先前的命令提出建議。內嵌建議會自動啟用。

若要停用內嵌建議，請執行下列命令：

```
q inline disable
```

若要啟用內嵌建議，請執行下列命令：

```
q inline enable
```

在 CloudShell 中使用 Q 聊天命令

`q chat` 命令可讓您從終端機提出問題並接收來自 Amazon Q 的回應。若要啟動與 Amazon Q 的對話，請在 CloudShell 終端機中執行`q chat`命令。如需詳細資訊，請參閱《Amazon Q 開發人員使用者指南》中的 CLI 中的與 Amazon Q 聊天。

在 CloudShell 中使用 Q translate 命令

`q translate` 命令可讓您撰寫自然語言指令。若要使用 Amazon Q 翻譯，請在 CloudShell 終端機中執行`q translate`命令。如需詳細資訊，請參閱《Amazon Q 開發人員使用者指南》中的從自然語言轉換為 bash。

CloudShell 中的 CLI 命令完成

當您在終端機中輸入 命令和選項時，CloudShell 中的 CLI 完成會提供建議。如需詳細資訊，請參閱《Amazon Q 開發人員使用者指南》中的產生命令列完成。

啟用或停用 Amazon Q CLI

您可以透過選擇偏好設定、啟用 Amazon Q CLI 和停用 Amazon Q CLI 來啟用或停用 Amazon Q CLI。Amazon Q CLI 可讓您與自然語言指示互動、提出問題，並從終端機取得 Amazon Q 的答案。當

您在終端機中輸入 時，它也會為您提供命令建議。當您開始在終端機中輸入 時，Amazon Q 會建議相關選項來完成您的命令。

CloudShell 中 Amazon Q CLI 的身分型政策

若要在 CloudShell 中使用 Amazon Q CLI，請確定您擁有必要的 IAM 許可。如需詳細資訊，請參閱《Amazon Q 開發人員使用者指南》中的 Amazon Q 開發人員的身分型政策範例。

從 AWS 服務主控台在 CloudShell 中執行命令

您可以透過 中的 Amazon ElastiCache 和 Amazon DocumentDB （與 MongoDB 相容） 主控台在 CloudShell 終端機中執行命令 AWS Management Console。

若要從其他 AWS Service 主控台在 CloudShell 中執行命令，指派給您的角色的 IAM 政策必須包含`cloudshell:approveCommand`許可。

CloudShell 會在主控台工具列上開啟，而執行命令快顯視窗會顯示在 CloudShell 中。在執行命令快顯視窗上，命令會出現在命令方塊中。

若要在 CloudShell 終端機中執行命令，請選擇下列其中一個步驟：

1. 如果您尚未在 CloudShell 中建立 VPC 環境，請在新環境名稱方塊中輸入名稱。

   您可以根據資源的 VPC 詳細資訊來檢視 VPC 環境詳細資訊。

   a. 選擇 Create and run (建立並執行)。

      此步驟將建立新的 CloudShell VPC 環境，並在 CloudShell 終端機中執行 命令。

2. 如果您已經建立 CloudShell VPC 環境，則可以檢視 CloudShell 環境名稱。

   > **Note**
   >
   > 如果您已有 CloudShell VPC 環境，則無法建立新的 VPC 環境。

   a. 選擇執行。

      此步驟會在所選 CloudShell VPC 環境中的 CloudShell 終端機中執行 命令。

      > **Note**
      >
      > 如果您沒有檢視已建立 VPC 環境的許可，請聯絡您的管理員以新增`cloudshell:describeEnvironments`許可。如需詳細資訊，請參閱使用 IAM 政策管理 AWS CloudShell 存取和使用。

您可以繼續在 CloudShell 終端機中執行命令。

自訂您的 AWS CloudShell 體驗

您可以自訂 AWS CloudShell 體驗的下列層面：

- 標籤配置：將命令列界面分割成多個資料欄和資料列。
- 字型大小：調整命令列文字的大小。
- 顏色佈景主題：在淺色和深色佈景主題之間切換。
- 安全貼上：開啟或關閉需要您在貼上前驗證多行文字的功能。
- Tmux 至工作階段還原：使用 tmux 會還原工作階段，直到工作階段變成非作用中為止。
- Amazon Q CLI：使用 Amazon Q CLI 可讓您使用 Amazon Q CLI 功能。

您也可以透過安裝自己的軟體和使用指令碼修改 Shell 來擴展 shell 環境。

將命令列顯示分割成多個索引標籤

透過將您的命令列界面分割為多個窗格來執行多個命令。

> **Note**
>
> 開啟多個標籤後，您可以按一下所選窗格中的任意位置來選取您要使用的標籤。您可以選擇區域名稱旁邊的 x 符號來關閉標籤。

- 從標籤配置中選擇動作和下列其中一個選項：
  - 新標籤：新增目前作用中標籤旁的新標籤。
  - 分割為資料列：在低於目前作用中標籤的資料列中新增新標籤。
  - 分割為資料欄：在目前作用中的資料欄旁新增標籤。

  如果沒有足夠的空間可完整顯示每個標籤，請捲動以查看整個標籤。您也可以選取分隔窗格的分割列，並使用指標來增加或減少窗格大小來拖曳它們。

變更字型大小

增加或減少命令列界面中顯示的文字大小。

1. 若要變更 AWS CloudShell 終端機設定，請前往設定、偏好設定。
2. 選擇文字大小。您的選項最小、小型、中型、大型和最大。

變更界面主題

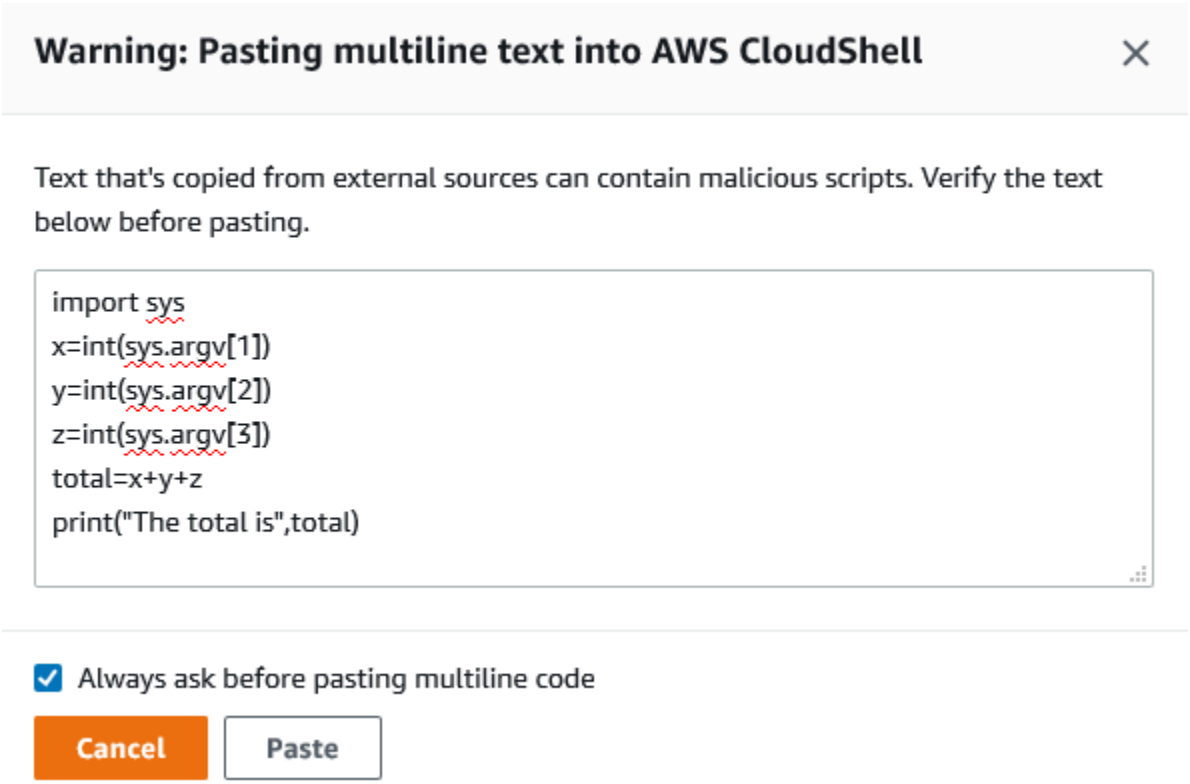
在命令列界面的淺色和深色佈景主題之間切換。

1. 若要變更 AWS CloudShell 佈景主題，請前往設定、偏好設定。
2. 選擇淺色或深色。

對多行文字使用安全貼上

Safe Paste 是一項安全功能，會提示您驗證即將貼入 shell 的多行文字不包含惡意指令碼。從第三方網站複製的文字可能包含隱藏程式碼，在您的 shell 環境中觸發意外行為。

安全貼上對話方塊會顯示您複製到剪貼簿的完整文字。如果您滿意沒有安全風險，請選擇貼上。



建議您啟用 Safe Paste 以捕捉指令碼中的潛在安全風險。您可以選擇偏好設定、啟用安全貼上和停用安全貼上，以開啟或關閉此功能。

使用 tmux 還原工作階段

AWS CloudShell 使用 tmux 在單一或多個瀏覽器索引標籤中還原工作階段。如果您重新整理瀏覽器索引標籤，它會繼續工作階段，直到工作階段變成非作用中為止。如需詳細資訊，請參閱工作階段還原。

使用 Amazon Q CLI

您可以透過選擇偏好設定、啟用 Amazon Q CLI 和停用 Amazon Q CLI 來啟用或停用 Amazon Q CLI。如需詳細資訊，請參閱啟用/停用 Amazon Q CLI。

在 Amazon VPC AWS CloudShell 中使用

AWS CloudShell 虛擬私有雲端 (VPC) 可讓您在 VPC 中建立 CloudShell 環境。對於每個 VPC 環境，您可以指派 VPC、新增子網路，以及建立最多五個安全群組的關聯。 AWS CloudShell 會繼承 VPC 的網路組態，並可讓您在與 VPC 中其他資源相同的子網路內 AWS CloudShell 安全地使用 ，並與其連線。

使用 Amazon VPC，您可以在您定義的邏輯隔離虛擬網路中啟動 AWS 資源。這個虛擬網路與您在資料中心中操作的傳統網路非常相似，且具備使用 AWS可擴展基礎設施的優勢。如需 VPC 的詳細資訊，請參閱 Amazon Virtual Private Cloud。

操作限制條件

AWS CloudShell VPC 環境有下列限制：

- 每個 IAM 主體最多可以建立兩個 VPC 環境。
- 您最多可以為 VPC 環境指派五個安全群組。
- 您無法在 VPC 環境的動作選單中使用 CloudShell 上傳和下載選項。

> **Note**
>
> 您可以從可透過其他 CLI 工具存取網際網路輸入/輸出的 VPC 環境上傳或下載檔案。

- VPC 環境不支援持久性儲存。儲存體是暫時性的。作用中環境工作階段結束時，會刪除資料和主目錄。
- 您的 AWS CloudShell 環境只能在私有 VPC 子網路中連線到網際網路。

> **Note**
>
> 根據預設，不會將公有 IP 地址配置給 CloudShell VPC 環境。在公有子網路中建立且路由表設定為將所有流量路由至網際網路閘道的 VPC 環境將無法存取公有網際網路，但使用網路位址轉譯 (NAT) 設定的私有子網路可存取公有網際網路。在此類私有子網路中建立的 VPC 環境將可存取公有網際網路。

- 若要為您的帳戶提供受管 CloudShell 環境， AWS 可為基礎運算主機佈建下列服務的網路存取權：
  - Amazon S3

- VPC 端點
  - com.amazonaws.<region>.ssmmessages
  - com.amazonaws.<region>.logs
  - com.amazonaws.<region>.kms
  - com.amazonaws.<region>.execute-api
  - com.amazonaws.<region>.ecs-telemetry
  - com.amazonaws.<region>.ecs-agent
  - com.amazonaws.<region>.ecs
  - com.amazonaws.<region>.ecr.dkr
  - com.amazonaws.<region>.ecr.api
  - com.amazonaws.<region>.codecatalyst.packages
  - com.amazonaws.<region>.codecatalyst.git
  - aws.api.global.codecatalyst

您無法透過修改 VPC 組態來限制對這些端點的存取。

CloudShell VPC 適用於所有 AWS 區域和 GovCloud 區域。如需可使用 CloudShell VPC 的區域清單，請參閱 支援的 AWS 區域 AWS CloudShell。

建立 CloudShell VPC 環境

本主題會逐步解說在 CloudShell 中建立 VPC 環境的步驟。

先決條件

您的管理員必須提供必要的 IAM 許可，您才能建立 VPC 環境。如需啟用建立 CloudShell VPC 環境許可的詳細資訊，請參閱 the section called “建立和使用 CloudShell VPC 環境所需的 IAM 許可”。

建立 CloudShell VPC 環境

1. 在 CloudShell 主控台頁面上，選擇 + 圖示，然後從下拉式功能表中選擇建立 VPC 環境。
2. 在建立 VPC 環境頁面上，在名稱方塊中輸入 VPC 環境的名稱。
3. 從虛擬私有雲端 (VPC) 下拉式清單中，選擇 VPC。
4. 從子網路下拉式清單中，選擇子網路。
5. 從安全群組下拉式清單中，選擇您要指派給 VPC 環境的一或多個安全群組。

> **Note**
>
> 您最多可以選擇五個安全群組。

6. 選擇建立以建立 VPC 環境。
7. （選用）選擇動作，然後選擇檢視詳細資訊以檢閱新建立 VPC 環境的詳細資訊。VPC 環境的 IP 地址會顯示在命令列提示字元中。

如需使用 VPC 環境的詳細資訊，請參閱 開始使用。

建立和使用 CloudShell VPC 環境所需的 IAM 許可

若要建立和使用 CloudShell VPC 環境，IAM 管理員必須啟用對 VPC 特定 Amazon EC2 許可的存取。本節列出建立和使用 VPC 環境所需的 Amazon EC2 許可。

若要建立 VPC 環境，指派給您角色的 IAM 政策必須包含下列 Amazon EC2 許可：

- `ec2:DescribeVpcs`
- `ec2:DescribeSubnets`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeDhcpOptions`
- `ec2:DescribeNetworkInterfaces`

- `ec2:CreateTags`
- `ec2:CreateNetworkInterface`
- `ec2:CreateNetworkInterfacePermission`

我們建議包含：

- ec2:DeleteNetworkInterface

> **Note**
>
> 此許可不是強制性的，但 CloudShell 需要此許可才能清除其建立的 ENI 資源（為 CloudShell VPC 環境建立ENIs 會加上ManagedByCloudShell金鑰標籤）。如果此許可未啟用，您必須在每次 CloudShell VPC 環境使用後手動清除 ENI 資源。

授予完整 CloudShell 存取權的 IAM 政策，包括對 VPC 的存取

下列範例顯示如何啟用 CloudShell 的完整許可，包括對 VPC 的存取：

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Sid": "AllowCloudShellOperations",
   "Effect": "Allow",
   "Action": [
    "cloudshell:*"
   ],
   "Resource": "*"
  },
  {
   "Sid": "AllowDescribeVPC",
   "Effect": "Allow",
   "Action": [
    "ec2:DescribeSubnets",
    "ec2:DescribeSecurityGroups",
    "ec2:DescribeVpcs"
   ],
   "Resource": "*"
  },
  {
   "Sid": "AllowInspectVPCConfigurationViaCloudShell",
   "Effect": "Allow",
   "Action": [
    "ec2:DescribeDhcpOptions",
    "ec2:DescribeNetworkInterfaces"
   ],
   "Resource": "*",
   "Condition": {
    "ForAnyValue:StringEquals": {
```

```
      "aws:CalledVia": "cloudshell.amazonaws.com"
     }
    }
   },
   {
    "Sid": "AllowCreateTagWithCloudShellKeyViaCloudShell",
    "Effect": "Allow",
    "Action": [
     "ec2:CreateTags"
    ],
    "Resource": "arn:aws:ec2:*:*:network-interface/*",
    "Condition": {
     "StringEquals": {
      "ec2:CreateAction": "CreateNetworkInterface"
     },
     "ForAnyValue:StringEquals": {
      "aws:TagKeys": "ManagedByCloudShell",
      "aws:CalledVia": "cloudshell.amazonaws.com"
     }
    }
   },
   {
    "Sid": "AllowCreateNetworkInterfaceWithSubnetsAndSGViaCloudShell",
    "Effect": "Allow",
    "Action": [
     "ec2:CreateNetworkInterface"
    ],
    "Resource": [
     "arn:aws:ec2:*:*:subnet/*",
     "arn:aws:ec2:*:*:security-group/*"
    ],
    "Condition": {
     "ForAnyValue:StringEquals": {
      "aws:CalledVia": "cloudshell.amazonaws.com"
     }
    }
   },
   {
    "Sid": "AllowCreateNetworkInterfaceWithCloudShellTagViaCloudShell",
    "Effect": "Allow",
    "Action": [
     "ec2:CreateNetworkInterface"
    ],
    "Resource": "arn:aws:ec2:*:*:network-interface/*",
```

```
    "Condition": {
     "ForAnyValue:StringEquals": {
      "aws:TagKeys": "ManagedByCloudShell",
      "aws:CalledVia": "cloudshell.amazonaws.com"
     }
    }
   },
   {
    "Sid": "AllowCreateNetworkInterfacePermissionWithCloudShellTagViaCloudShell",
    "Effect": "Allow",
    "Action": [
     "ec2:CreateNetworkInterfacePermission"
    ],
    "Resource": "arn:aws:ec2:*:*:network-interface/*",
    "Condition": {
     "StringEquals": {
      "aws:ResourceTag/ManagedByCloudShell": ""
     },
     "ForAnyValue:StringEquals": {
      "aws:CalledVia": "cloudshell.amazonaws.com"
     }
    }
   },
   {
    "Sid": "AllowDeleteNetworkInterfaceWithCloudShellTagViaCloudShell",
    "Effect": "Allow",
    "Action": [
     "ec2:DeleteNetworkInterface"
    ],
    "Resource": "arn:aws:ec2:*:*:network-interface/*",
    "Condition": {
     "StringEquals": {
      "aws:ResourceTag/ManagedByCloudShell": ""
     },
     "ForAnyValue:StringEquals": {
      "aws:CalledVia": "cloudshell.amazonaws.com"
     }
    }
   }
  ]
}
```

針對 VPC 環境使用 IAM 條件金鑰

您可以針對 VPC 設定使用 CloudShell 特定條件金鑰，為您的 VPC 環境提供額外的許可控制。您也可以指定 VPC 環境可以和不能使用的子網路和安全群組。

CloudShell 在 IAM 政策中支援下列條件金鑰：

- `CloudShell:VpcIds` – 允許或拒絕一或多個 VPCs
- `CloudShell:SubnetIds` – 允許或拒絕一或多個子網路
- `CloudShell:SecurityGroupIds` – 允許或拒絕一或多個安全群組

> **Note**
>
> 如果修改可存取公有 CloudShell 環境的使用者許可，以對`cloudshell:createEnvironment`動作新增限制，他們仍然可以存取其現有的公有環境。不過，如果您想要修改具有此限制的 IAM 政策，並停用其對現有公有環境的存取，您必須先更新具有限制的 IAM 政策，然後確保帳戶中的每個 CloudShell 使用者使用 CloudShell Web 使用者介面 (動作 → 刪除 CloudShell 環境) 手動刪除現有的公有環境。

具有 VPC 設定條件金鑰的範例政策

下列範例示範如何使用條件金鑰進行 VPC 設定。建立具有所需限制的政策陳述式之後，請附加目標 使用者或角色的政策陳述式。

確保使用者僅建立 VPC 環境，並拒絕建立公有環境

為了確保使用者只能建立 VPC 環境，請使用拒絕許可，如下列範例所示：

```
{
  "Statement": [
    {
      "Sid": "DenyCloudShellNonVpcEnvironments",
      "Action": [
        "cloudshell:CreateEnvironment"
      ],
      "Effect": "Deny",
      "Resource": "*",
```

```
      "Condition": {
        "Null": {
          "cloudshell:VpcIds": "true"
        }
      }
    }
  ]
}
```

拒絕使用者存取特定 VPC、子網路或安全群組

若要拒絕使用者存取特定 VPC，請使用 StringEquals 來檢查 cloudshell:VpcIds 條件的值。下列範例拒絕使用者存取 vpc-1和 vpc-2：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "EnforceOutOfVpc",
      "Action": [
        "cloudshell:CreateEnvironment"
      ],
      "Effect": "Deny",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "cloudshell:VpcIds": [
            "vpc-1",
            "vpc-2"
          ]
        }
      }
    }
  ]
}
```

若要拒絕使用者存取特定 VPC，請使用 StringEquals 來檢查 cloudshell:SubnetIds 條件的值。下列範例拒絕使用者存取 subnet-1和 subnet-2：

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

```
    {
      "Sid": "EnforceOutOfSubnet",
      "Action": [
        "cloudshell:CreateEnvironment"
      ],
      "Effect": "Deny",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "cloudshell:SubnetIds": [
            "subnet-1",
            "subnet-2"
          ]
        }
      }
    }
  ]
}
```

若要拒絕使用者存取特定 VPC，請使用 StringEquals 來檢查 cloudshell:SecurityGroupIds 條件的值。下列範例拒絕使用者存取 sg-1和 sg-2：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "EnforceOutOfSecurityGroups",
      "Action": [
        "cloudshell:CreateEnvironment"
      ],
      "Effect": "Deny",
      "Resource": "*",
      "Condition": {
        "ForAnyValue:StringEquals": {
          "cloudshell:SecurityGroupIds": [
            "sg-1",
            "sg-2"
          ]
        }
      }
    }
  ]
}
```

允許使用者使用特定 VPC 組態建立環境

若要允許使用者存取特定 VPCs，請使用 StringEquals檢查cloudshell:VpcIds條件的值。下列範例允許使用者存取 vpc-1和 vpc-2：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "EnforceStayInSpecificVpc",
      "Action": [
        "cloudshell:CreateEnvironment"
      ],
      "Effect": "Allow",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "cloudshell:VpcIds": [
            "vpc-1",
            "vpc-2"
          ]
        }
      }
    }
  ]
}
```

若要允許使用者存取特定 VPCs，請使用 StringEquals檢查cloudshell:SubnetIds條件的值。下列範例允許使用者存取 subnet-1和 subnet-2：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "EnforceStayInSpecificSubnets",
      "Action": [
        "cloudshell:CreateEnvironment"
      ],
      "Effect": "Allow",
      "Resource": "*",
      "Condition": {
        "ForAllValues:StringEquals": {
          "cloudshell:SubnetIds": [
```

```
            "subnet-1",
            "subnet-2"
          ]
        }
      }
    }
  ]
}
```

若要允許使用者存取特定 VPCs，請使用 StringEquals檢查cloudshell:SecurityGroupIds條件的值。下列範例允許使用者存取 sg-1和 sg-2：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "EnforceStayInSpecificSecurityGroup",
      "Action": [
        "cloudshell:CreateEnvironment"
      ],
      "Effect": "Allow",
      "Resource": "*",
      "Condition": {
        "ForAllValues:StringEquals": {
          "cloudshell:SecurityGroupIds": [
            "sg-1",
            "sg-2"
          ]
        }
      }
    }
  ]
}
```

的安全性 AWS CloudShell

雲端安全是 Amazon Web Services (AWS) 最重視的一環。身為 AWS 客戶，您可以受益於資料中心和網路架構，這些架構是為了滿足最安全敏感組織的需求而建置。安全性是 AWS 與您之間的共同責任。共同責任模型 將此描述為雲端本身的安全和雲端內部的安全。

雲端的安全性 – AWS 負責保護執行 AWS 雲端中提供的所有服務的基礎設施，並提供您可以安全使用的服務。我們的安全責任是 的最高優先順序 AWS，而第三方稽核人員會在AWS 合規計劃中定期測試和驗證我們的安全有效性。

雲端的安全性 – 您的責任取決於您使用 AWS 的服務，以及其他因素，包括資料的敏感度、組織的需求，以及適用的法律和法規。

AWS CloudShell 透過其支援的特定 AWS 服務遵循共同責任模型。如需 AWS 服務安全資訊，請參閱AWS 服務安全文件頁面AWS ，以及合規計劃在 AWS 合規工作範圍內的服務。

下列主題說明如何設定 AWS CloudShell 以符合您的安全與合規目標。

主題

- 中的資料保護 AWS CloudShell
- AWS CloudShell 的 Identity and Access Management
- 在 中記錄和監控 AWS CloudShell
- 的合規驗證 AWS CloudShell
- 中的彈性 AWS CloudShell
- 中的基礎設施安全 AWS CloudShell
- 的安全最佳實務 AWS CloudShell
- AWS CloudShell 安全性FAQs

中的資料保護 AWS CloudShell

AWS 共同責任模型適用於 中的資料保護 AWS CloudShell。如此模型所述， AWS 負責保護執行所有 的 全球基礎設施 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。如需資料隱私權的詳細資訊，請參閱資料隱私權常見問答集。如需有關歐洲資料保護的相關資訊，請參閱 AWS 安全性部落格上的 AWS 共同的責任模型和 GDPR 部落格文章。

基於資料保護目的，我們建議您保護 AWS 帳戶 登入資料，並使用 AWS IAM Identity Center 或 AWS Identity and Access Management (IAM) 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：

- 每個帳戶均要使用多重要素驗證 (MFA)。
- 使用 SSL/TLS 與 AWS 資源通訊。我們需要 TLS 1.2 並建議使用 TLS 1.3。
- 使用 設定 API 和使用者活動記錄 AWS CloudTrail。如需有關使用 CloudTrail 追蹤擷取 AWS 活動的資訊，請參閱AWS CloudTrail 《 使用者指南》中的使用 CloudTrail 追蹤。
- 使用 AWS 加密解決方案，以及其中的所有預設安全控制 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。
- 如果您在 AWS 透過命令列界面或 API 存取 時需要 FIPS 140-3 驗證的密碼編譯模組，請使用 FIPS 端點。如需有關 FIPS 和 FIPS 端點的更多相關資訊，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。這包括當您使用 AWS CloudShell 或使用主控台、API AWS CLI或其他 AWS 服務 AWS SDKs 時。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。如果您提供外部伺服器的 URL，我們強烈建議請勿在驗證您對該伺服器請求的 URL 中包含憑證資訊。

資料加密

資料加密是指在存放於 時保護靜態資料， AWS CloudShell 以及在傳輸時，資料會在 AWS CloudShell 和服務端點之間傳輸。

使用 進行靜態加密 AWS KMS

靜態加密是指在存放時對資料進行加密，以保護您的資料免受未經授權的存取。使用 時 AWS CloudShell，每個 AWS 區域可免費持續儲存 1 GB。持續性儲存空間位於主目錄 ($HOME) 中，而且對您來說是私有的。與每個 shell 工作階段結束後回收的暫時性環境資源不同，主目錄中的資料會持續存在。

存放在 中的資料加密 AWS CloudShell 會使用 AWS Key Management Service () 提供的密碼編譯金鑰實作AWS KMS。這是用於建立和控制的受管 AWS 服務 AWS KMS keys，用於加密存放在 AWS CloudShell 環境中的客戶資料的加密金鑰。 會 AWS CloudShell 產生和管理密碼編譯金鑰，以代表客戶加密資料。

傳輸中加密

傳輸中的加密指的是保護您的資料免於在通訊端點間移動時遭到攔截。

根據預設，用戶端的 Web 瀏覽器電腦與雲端型電腦之間的所有資料通訊 AWS CloudShell 都會透過 HTTPS/TLS 連線傳送所有內容來加密。

您不需要執行任何動作，即可使用 HTTPS/TLS 進行通訊。

AWS CloudShell 的 Identity and Access Management

AWS Identity and Access Management (IAM) 是 AWS 服務 ，可協助管理員安全地控制對 AWS 資源的存取。IAM 管理員可控制誰可以進行身分驗證 （ 登入 ） 和授權 （ 具有許可 ） 來使用 CloudShell 資源。IAM 是 AWS 服務 您可以免費使用的 。

主題

目標對象

使用方式 AWS Identity and Access Management (IAM) 會有所不同，取決於您在 CloudShell 中執行的工作。

服務使用者 – 如果您使用 CloudShell 服務執行任務，管理員會為您提供所需的登入資料和許可。當您使用更多 CloudShell 功能來執行工作時，您可能需要額外的許可。了解存取許可的管理方式可協助您向管理員請求正確的許可。如果您無法存取 CloudShell 中的功能，請參閱 對 AWS CloudShell 身分和存取進行故障診斷。

服務管理員 – 如果您在公司負責 CloudShell 資源，您可能擁有 CloudShell 的完整存取權。您的任務是判斷服務使用者應存取哪些 CloudShell 功能和資源。接著，您必須將請求提交給您的 IAM 管理員，來變更您服務使用者的許可。檢閱此頁面上的資訊，了解 IAM 的基本概念。若要進一步了解貴公司如何搭配 CloudShell 使用 IAM，請參閱 AWS CloudShell 如何與 IAM 搭配使用。

IAM 管理員 – 如果您是 IAM 管理員，建議您了解撰寫政策以管理 CloudShell 存取的詳細資訊。若要檢視您可以在 IAM 中使用的 CloudShell 身分型政策範例，請參閱 AWS CloudShell 的身分型政策範例。

使用身分驗證

身分驗證是您 AWS 使用身分憑證登入 的方式。您必須以 AWS 帳戶根使用者身分、IAM 使用者身分或擔任 IAM 角色來驗證 （ 登入 AWS)。

您可以使用透過身分來源提供的憑證，以聯合身分 AWS 身分身分登入 。 AWS IAM Identity Center (IAM Identity Center) 使用者、您公司的單一登入身分驗證，以及您的 Google 或 Facebook 登入資料，都是聯合身分的範例。您以聯合身分登入時，您的管理員先前已設定使用 IAM 角色的聯合身分。當您使用聯合 AWS 身分存取 時，您會間接擔任角色。

根據您的使用者類型，您可以登入 AWS Management Console 或 AWS 存取入口網站。如需登入的詳細資訊 AWS，請參閱AWS 登入 《 使用者指南》中的如何登入您的 AWS 帳戶 。

如果您以 AWS 程式設計方式存取 ， AWS 會提供軟體開發套件 (SDK) 和命令列界面 (CLI)，以使用您的 憑證以密碼編譯方式簽署您的請求。如果您不使用 AWS 工具，則必須自行簽署請求。如需使用建議的方法自行簽署請求的詳細資訊，請參閱《IAM 使用者指南》中的適用於 API 請求的AWS Signature 第 4 版。

無論您使用何種身分驗證方法，您可能都需要提供額外的安全性資訊。例如， AWS 建議您使用多重驗證 (MFA) 來提高帳戶的安全性。如需更多資訊，請參閱《AWS IAM Identity Center 使用者指南》中的多重要素驗證和《IAM 使用者指南》中的 IAM 中的AWS 多重要素驗證。

AWS 帳戶 根使用者

當您建立 時 AWS 帳戶，您會從一個登入身分開始，該身分可完整存取帳戶中的所有 AWS 服務 和資源。此身分稱為 AWS 帳戶 Theroot 使用者，可透過使用您用來建立帳戶的電子郵件地址和密碼登入來存取。強烈建議您不要以根使用者處理日常任務。保護您的根使用者憑證，並將其用來執行只能由根使用者執行的任務。如需這些任務的完整清單，了解需以根使用者登入的任務，請參閱 IAM 使用者指南中的需要根使用者憑證的任務。

聯合身分

最佳實務是， 要求人類使用者，包括需要管理員存取權的使用者，使用臨時 AWS 服務 憑證與身分提供者聯合來存取 。

聯合身分是來自您的企業使用者目錄、Web 身分提供者、 AWS Directory Service、Identity Center 目錄，或使用透過身分來源提供的 AWS 服務 登入資料存取的任何使用者。當聯合身分存取時 AWS 帳戶，它們會擔任 角色，而角色會提供臨時登入資料。

對於集中式存取權管理，我們建議您使用 AWS IAM Identity Center。您可以在 IAM Identity Center 中建立使用者和群組，也可以連接並同步到您自己的身分來源中的一組使用者 AWS 帳戶 和群組，以便在所有 和應用程式中使用。如需 IAM Identity Center 的詳細資訊，請參閱 AWS IAM Identity Center 使用者指南中的什麼是 IAM Identity Center？。

IAM 使用者和群組

IAM 使用者是 中的身分 AWS 帳戶 ，具有單一人員或應用程式的特定許可。建議您盡可能依賴臨時憑證，而不是擁有建立長期憑證 (例如密碼和存取金鑰) 的 IAM 使用者。但是如果特定使用案例需要擁有長期憑證的 IAM 使用者，建議您輪換存取金鑰。如需更多資訊，請參閱 IAM 使用者指南中的為需要長期憑證的使用案例定期輪換存取金鑰。

IAM 群組是一種指定 IAM 使用者集合的身分。您無法以群組身分簽署。您可以使用群組來一次為多名使用者指定許可。群組可讓管理大量使用者許可的程序變得更為容易。例如，您可以擁有一個名為 IAMAdmins 的群組，並給予該群組管理 IAM 資源的許可。

使用者與角色不同。使用者只會與單一人員或應用程式建立關聯，但角色的目的是在由任何需要它的人員取得。使用者擁有永久的長期憑證，但角色僅提供臨時憑證。如需更多資訊，請參閱《IAM 使用者指南》中的 IAM 使用者的使用案例。

IAM 角色

IAM 角色是 中具有特定許可 AWS 帳戶 的身分。它類似 IAM 使用者，但不與特定的人員相關聯。若要暫時在 中擔任 IAM 角色 AWS Management Console，您可以從使用者切換至 IAM 角色 （主控台）。您可以透過呼叫 AWS CLI 或 AWS API 操作或使用自訂 URL 來擔任角色。如需使用角色的方法詳細資訊，請參閱《IAM 使用者指南》中的擔任角色的方法。

使用臨時憑證的 IAM 角色在下列情況中非常有用：

- 聯合身分使用者存取 — 如需向聯合身分指派許可，請建立角色，並為角色定義許可。當聯合身分進行身分驗證時，該身分會與角色建立關聯，並獲授予由角色定義的許可。如需有關聯合角色的相關資訊，請參閱《IAM 使用者指南》中的為第三方身分提供者 (聯合) 建立角色。如果您使用 IAM Identity Center，則需要設定許可集。為控制身分驗證後可以存取的內容，IAM Identity Center 將許可集與 IAM 中的角色相關聯。如需有關許可集的資訊，請參閱 AWS IAM Identity Center 使用者指南中的許可集。
- 暫時 IAM 使用者許可 – IAM 使用者或角色可以擔任 IAM 角色來暫時針對特定任務採用不同的許可。
- 跨帳戶存取權：您可以使用 IAM 角色，允許不同帳戶中的某人 (信任的主體) 存取您帳戶的資源。角色是授予跨帳戶存取權的主要方式。不過，對於某些 AWS 服務，您可以直接將政策連接到資源

（而不是使用角色做為代理）。如需了解使用角色和資源型政策進行跨帳戶存取之間的差異，請參閱《IAM 使用者指南》中的 IAM 中的跨帳戶資源存取。

- 跨服務存取 – 有些 AWS 服務 使用其他 中的功能 AWS 服務。例如，當您在服務中進行呼叫時，該服務通常會在 Amazon EC2 中執行應用程式或將物件儲存在 Amazon Simple Storage Service (Amazon S3) 中。服務可能會使用呼叫主體的許可、使用服務角色或使用服務連結角色來執行此作業。
  - 轉送存取工作階段 (FAS) – 當您使用 IAM 使用者或角色在其中執行動作時 AWS，您會被視為委託人。使用某些服務時，您可能會執行某個動作，進而在不同服務中啟動另一個動作。FAS 使用呼叫 的委託人許可 AWS 服務，結合 AWS 服務 請求向下游服務提出請求。只有在服務收到需要與其他 AWS 服務 或 資源互動才能完成的請求時，才會提出 FAS 請求。在此情況下，您必須具有執行這兩個動作的許可。如需提出 FAS 請求時的政策詳細資訊，請參閱《轉發存取工作階段》。
  - 服務角色 – 服務角色是服務擔任的 IAM 角色，可代表您執行動作。IAM 管理員可以從 IAM 內建立、修改和刪除服務角色。如需詳細資訊，請參閱《IAM 使用者指南》中的建立角色以委派許可權給 AWS 服務。
  - 服務連結角色 – 服務連結角色是一種連結至 的服務角色。 AWS 服務服務可以擔任代表您執行動作的角色。服務連結角色會出現在您的 中 AWS 帳戶 ，並由服務擁有。IAM 管理員可以檢視，但不能編輯服務連結角色的許可。
- 在 Amazon EC2 上執行的應用程式 – 您可以使用 IAM 角色來管理在 EC2 執行個體上執行之應用程式的臨時登入資料，以及提出 AWS CLI 或 AWS API 請求。這是在 EC2 執行個體內儲存存取金鑰的較好方式。若要將 AWS 角色指派給 EC2 執行個體並將其提供給其所有應用程式，您可以建立連接至執行個體的執行個體描述檔。執行個體設定檔包含該角色，並且可讓 EC2 執行個體上執行的程式取得臨時憑證。如需詳細資訊，請參閱《IAM 使用者指南》中的使用 IAM 角色來授予許可權給 Amazon EC2 執行個體上執行的應用程式。

使用政策管理存取權

您可以透過建立政策並將其連接到身分或資源 AWS 來控制 AWS 中的存取。政策是 中的物件，當與身分或資源相關聯時， AWS 會定義其許可。當委託人（使用者、根使用者或角色工作階段）發出請求時， 會 AWS 評估這些政策。政策中的許可決定是否允許或拒絕請求。大多數政策會以 JSON 文件 AWS 的形式存放在 中。如需 JSON 政策文件結構和內容的詳細資訊，請參閱 IAM 使用者指南中的 JSON 政策概觀。

管理員可以使用 AWS JSON 政策來指定誰可以存取內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

預設情況下，使用者和角色沒有許可。若要授予使用者對其所需資源執行動作的許可，IAM 管理員可以建立 IAM 政策。然後，管理員可以將 IAM 政策新增至角色，使用者便能擔任這些角色。

IAM 政策定義該動作的許可，無論您使用何種方法來執行操作。例如，假設您有一個允許 `iam:GetRole` 動作的政策。具有該政策的使用者可以從 AWS Management Console AWS CLI、 或 AWS API 取得角色資訊。

身分型政策

身分型政策是可以附加到身分 (例如 IAM 使用者、使用者群組或角色) 的 JSON 許可政策文件。這些政策可控制身分在何種條件下能對哪些資源執行哪些動作。如需了解如何建立身分型政策，請參閱《IAM 使用者指南》中的透過客戶管理政策定義自訂 IAM 許可。

身分型政策可進一步分類成內嵌政策或受管政策。內嵌政策會直接內嵌到單一使用者、群組或角色。受管政策是獨立的政策，您可以連接到 中的多個使用者、群組和角色 AWS 帳戶。受管政策包括 AWS 受管政策和客戶受管政策。如需了解如何在受管政策及內嵌政策之間選擇，請參閱《IAM 使用者指南》中的在受管政策和內嵌政策間選擇。

資源型政策

資源型政策是連接到資源的 JSON 政策文件。資源型政策的最常見範例是 IAM 角色信任政策和 Amazon S3 儲存貯體政策。在支援資源型政策的服務中，服務管理員可以使用它們來控制對特定資源的存取權限。對於附加政策的資源，政策會定義指定的主體可以對該資源執行的動作以及在何種條件下執行的動作。您必須在資源型政策中指定主體。委託人可以包含帳戶、使用者、角色、聯合身分使用者或 AWS 服務。

資源型政策是位於該服務中的內嵌政策。您無法在資源型政策中使用來自 IAM 的 AWS 受管政策。

存取控制清單 (ACL)

存取控制清單 (ACL) 可控制哪些主體 (帳戶成員、使用者或角色) 擁有存取某資源的許可。ACL 類似於資源型政策，但它們不使用 JSON 政策文件格式。

Amazon S3 AWS WAF和 Amazon VPC 是支援 ACLs的服務範例。如需進一步了解 ACL，請參閱 Amazon Simple Storage Service 開發人員指南中的存取控制清單 (ACL) 概觀。

其他政策類型

AWS 支援其他較不常見的政策類型。這些政策類型可設定較常見政策類型授予您的最大許可。

- 許可界限 – 許可範圍是一種進階功能，可供您設定身分型政策能授予 IAM 實體 (IAM 使用者或角色) 的最大許可。您可以為實體設定許可界限。所產生的許可會是實體的身分型政策和其許可界限的交

集。會在 Principal 欄位中指定使用者或角色的資源型政策則不會受到許可界限限制。所有這類政策中的明確拒絕都會覆寫該允許。如需許可界限的詳細資訊，請參閱 IAM 使用者指南中的 IAM 實體許可界限。

- 服務控制政策 SCPs) – SCPs是 JSON 政策，可指定 中組織或組織單位 (OU) 的最大許可 AWS Organizations。 AWS Organizations 是一種用於分組和集中管理您企業擁有 AWS 帳戶 之多個 的服務。若您啟用組織中的所有功能，您可以將服務控制政策 (SCP) 套用到任何或所有帳戶。SCP 會限制成員帳戶中實體的許可，包括每個實體 AWS 帳戶根使用者。如需 Organizations 和 SCP 的詳細資訊，請參閱《AWS Organizations 使用者指南》中的服務控制政策。
- 資源控制政策 (RCP) - RCP 是 JSON 政策，可用來設定您帳戶中資源的可用許可上限，採取這種方式就不需要更新附加至您所擁有的每個資源的 IAM 政策。RCP 會限制成員帳戶中資源的許可，並可能影響身分的有效許可，包括 AWS 帳戶根使用者，無論它們是否屬於您的組織。如需 Organizations 和 RCPs的詳細資訊，包括支援 RCPs AWS 服務 的 清單，請參閱AWS Organizations 《 使用者指南》中的資源控制政策 (RCPs)。
- 工作階段政策 – 工作階段政策是一種進階政策，您可以在透過撰寫程式的方式建立角色或聯合使用者的暫時工作階段時，做為參數傳遞。所產生工作階段的許可會是使用者或角色的身分型政策和工作階段政策的交集。許可也可以來自資源型政策。所有這類政策中的明確拒絕都會覆寫該允許。如需詳細資訊，請參閱 IAM 使用者指南中的工作階段政策。

多種政策類型

將多種政策類型套用到請求時，其結果形成的許可會更為複雜、更加難以理解。若要了解如何 AWS 在涉及多個政策類型時決定是否允許請求，請參閱《IAM 使用者指南》中的政策評估邏輯。

AWS CloudShell 如何與 IAM 搭配使用

在您使用 IAM 管理 CloudShell 的存取權之前，請先了解哪些 IAM 功能可與 CloudShell 搭配使用。

您可以搭配 AWS CloudShell 使用的 IAM 功能

IAM 功能	CloudShell 支援
身分型政策	是
資源型政策	否
政策動作	是

IAM 功能	CloudShell 支援
政策資源	是
政策條件索引鍵 (服務特定)	是
ACL	否
ABAC(政策中的標籤)	否
臨時憑證	是
轉送存取工作階段 (FAS)	否
服務角色	否
服務連結角色	否

若要全面了解 CloudShell 和其他 AWS 服務如何與大多數 IAM 功能搭配使用，請參閱《AWS IAM 使用者指南》中的與 IAM 搭配使用的 服務。

CloudShell 的身分型政策

支援身分型政策：是

身分型政策是可以附加到身分 (例如 IAM 使用者、使用者群組或角色) 的 JSON 許可政策文件。這些政策可控制身分在何種條件下能對哪些資源執行哪些動作。如需了解如何建立身分型政策，請參閱《IAM 使用者指南》中的透過客戶管理政策定義自訂 IAM 許可。

使用 IAM 身分型政策，您可以指定允許或拒絕的動作和資源，以及在何種條件下允許或拒絕動作。您無法在身分型政策中指定主體，因為這會套用至連接的使用者或角色。如要了解您在 JSON 政策中使用的所有元素，請參閱《IAM 使用者指南》中的 IAM JSON 政策元素參考。

CloudShell 的身分型政策範例

若要檢視 CloudShell 身分型政策的範例，請參閱 AWS CloudShell 的身分型政策範例。

CloudShell 中的資源型政策

支援資源型政策：否

資源型政策是附加到資源的 JSON 政策文件。資源型政策的最常見範例是 IAM 角色信任政策和 Amazon S3 儲存貯體政策。在支援資源型政策的服務中，服務管理員可以使用它們來控制對特定資源的存取權限。對於附加政策的資源，政策會定義指定的主體可以對該資源執行的動作以及在何種條件下執行的動作。您必須在資源型政策中指定主體。委託人可以包含帳戶、使用者、角色、聯合身分使用者或 AWS 服務。

如需啟用跨帳戶存取權，您可以指定在其他帳戶內的所有帳戶或 IAM 實體，做為資源型政策的主體。新增跨帳戶主體至資源型政策，只是建立信任關係的一半。當委託人和資源位於不同位置時 AWS 帳戶，信任帳戶中的 IAM 管理員也必須授予委託人實體（使用者或角色）存取資源的許可。其透過將身分型政策連接到實體來授與許可。不過，如果資源型政策會為相同帳戶中的主體授予存取，這時就不需要額外的身分型政策。如需詳細資訊，請參閱《IAM 使用者指南》中的 IAM 中的快帳戶資源存取。

CloudShell 的政策動作

支援政策動作：是

管理員可以使用 AWS JSON 政策來指定誰可以存取內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

JSON 政策的 `Action` 元素描述您可以用來允許或拒絕政策中存取的動作。政策動作通常具有與相關聯 AWS API 操作相同的名稱。有一些例外狀況，例如沒有相符的 API 操作的僅限許可動作。也有一些作業需要政策中的多個動作。這些額外的動作稱為相依動作。

政策會使用動作來授予執行相關聯動作的許可。

若要查看 CloudShell 動作清單，請參閱《服務授權參考》中的 AWS CloudShell 定義的動作。有些動作可能有多個 API。

CloudShell 中的政策動作在動作之前使用以下字首：

```
cloudshell
```

若要在單一陳述式中指定多個動作，請用逗號分隔。

```
"Action": [
      "cloudshell:action1",
      "cloudshell:action2"
         ]
```

若要檢視 CloudShell 身分型政策的範例，請參閱 AWS CloudShell 的身分型政策範例。

CloudShell 的政策資源

支援政策資源：是

管理員可以使用 AWS JSON 政策來指定誰可以存取內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

Resource JSON 政策元素可指定要套用動作的物件。陳述式必須包含 Resource 或 NotResource 元素。最佳實務是使用其 Amazon Resource Name (ARN) 來指定資源。您可以針對支援特定資源類型的動作 (稱為資源層級許可) 來這麼做。

對於不支援資源層級許可的動作 (例如列出操作)，請使用萬用字元 (*) 來表示陳述式適用於所有資源。

```
"Resource": "*"
```

若要查看 CloudShell 資源類型及其 ARNs，請參閱《服務授權參考》中的 AWS CloudShell 定義的資源。若要了解您可以使用哪些動作指定每個資源的 ARN，請參閱 AWS CloudShell 定義的動作。

若要檢視 CloudShell 身分型政策的範例，請參閱 AWS CloudShell 的身分型政策範例。

CloudShell 的政策條件索引鍵

支援服務特定政策條件金鑰：是

管理員可以使用 AWS JSON 政策來指定誰可以存取內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

Condition 元素 (或 Condition 區塊) 可讓您指定使陳述式生效的條件。Condition 元素是選用項目。您可以建立使用條件運算子的條件運算式 (例如等於或小於)，來比對政策中的條件和請求中的值。

若您在陳述式中指定多個 Condition 元素，或是在單一 Condition 元素中指定多個索引鍵， AWS 會使用邏輯 AND 操作評估他們。如果您為單一條件索引鍵指定多個值， 會使用邏輯OR操作 AWS 評估條件。必須符合所有條件，才會授與陳述式的許可。

您也可以在指定條件時使用預留位置變數。例如，您可以只在使用者使用其 IAM 使用者名稱標記時，將存取資源的許可授予該 IAM 使用者。如需更多資訊，請參閱 IAM 使用者指南中的 IAM 政策元素：變數和標籤。

AWS 支援全域條件金鑰和服務特定的條件金鑰。若要查看所有 AWS 全域條件索引鍵，請參閱《IAM 使用者指南》中的AWS 全域條件內容索引鍵。

若要查看 CloudShell 條件金鑰清單，請參閱《服務授權參考》中的 AWS CloudShell 的條件金鑰。若要了解您可以使用條件金鑰的動作和資源，請參閱 AWS CloudShell 定義的動作。

若要檢視 CloudShell 身分型政策的範例，請參閱 AWS CloudShell 的身分型政策範例。

CloudShell ACLs

支援 ACL：否

存取控制清單 (ACL) 可控制哪些主體 (帳戶成員、使用者或角色) 擁有存取某資源的許可。ACL 類似於資源型政策，但它們不使用 JSON 政策文件格式。

ABAC 搭配 CloudShell

支援 ABAC （政策中的標籤）：否

屬性型存取控制 (ABAC) 是一種授權策略，可根據屬性來定義許可。在 中 AWS，這些屬性稱為標籤。您可以將標籤連接至 IAM 實體 （使用者或角色） 和許多 AWS 資源。為實體和資源加上標籤是 ABAC 的第一步。您接著要設計 ABAC 政策，允許在主體的標籤與其嘗試存取的資源標籤相符時操作。

ABAC 在成長快速的環境中相當有幫助，並能在政策管理變得繁瑣時提供協助。

如需根據標籤控制存取，請使用 `aws:ResourceTag/key-name`、`aws:RequestTag/key-name` 或 `aws:TagKeys` 條件索引鍵，在政策的條件元素中，提供標籤資訊。

如果服務支援每個資源類型的全部三個條件金鑰，則對該服務而言，值為 Yes。如果服務僅支援某些資源類型的全部三個條件金鑰，則值為 Partial。

如需 ABAC 的詳細資訊，請參閱《IAM 使用者指南》中的使用 ABAC 授權定義許可。如要查看含有設定 ABAC 步驟的教學課程，請參閱 IAM 使用者指南中的使用屬性型存取控制 (ABAC)。

搭配 CloudShell 使用臨時登入資料

支援臨時憑證：是

當您使用臨時登入資料登入時，有些 AWS 服務 無法運作。如需詳細資訊，包括哪些 AWS 服務 使用臨時登入資料，請參閱《AWS 服務 IAM 使用者指南》中的使用 IAM 的 。

如果您 AWS Management Console 使用使用者名稱和密碼以外的任何方法登入 ，則會使用暫時登入資料。例如，當您 AWS 使用公司的單一登入 (SSO) 連結存取 時，該程序會自動建立臨時登入資料。

當您以使用者身分登入主控台，然後切換角色時，也會自動建立臨時憑證。如需切換角色的詳細資訊，請參閱《IAM 使用者指南》中的從使用者切換至 IAM 角色 (主控台)。

您可以使用 AWS CLI 或 AWS API 手動建立臨時登入資料。然後，您可以使用這些臨時登入資料來存取 AWS。 AWS 建議您動態產生臨時登入資料，而不是使用長期存取金鑰。如需詳細資訊，請參閱 IAM 中的暫時性安全憑證。

當您切換角色時，您將使用不同的環境。您無法在相同 AWS CloudShell 環境中切換角色。

轉送 CloudShell 的存取工作階段

支援轉寄存取工作階段 (FAS)：否

當您使用 IAM 使用者或角色在 中執行動作時 AWS，您會被視為委託人。使用某些服務時，您可能會執行某個動作，進而在不同服務中啟動另一個動作。FAS 使用呼叫 的委託人許可 AWS 服務，結合 AWS 服務 請求向下游服務提出請求。只有當服務收到需要與其他 AWS 服務 或 資源互動才能完成的請求時，才會提出 FAS 請求。在此情況下，您必須具有執行這兩個動作的許可。如需提出 FAS 請求時的政策詳細資訊，請參閱轉發存取工作階段。

CloudShell 的服務角色

支援服務角色：否

服務角色是服務擔任的 IAM 角色，可代您執行動作。IAM 管理員可以從 IAM 內建立、修改和刪除服務角色。如需詳細資訊，請參閱《IAM 使用者指南》中的建立角色以委派許可權給 AWS 服務。

> **Warning**
>
> 變更服務角色的許可可能會中斷 CloudShell 功能。只有在 CloudShell 提供指引時，才能編輯服務角色。

CloudShell 的服務連結角色

支援服務連結角色：否

服務連結角色是連結至 的一種服務角色 AWS 服務。服務可以擔任代表您執行動作的角色。服務連結角色會出現在您的 中 AWS 帳戶 ，並由服務擁有。IAM 管理員可以檢視，但不能編輯服務連結角色的許可。

AWS CloudShell 的身分型政策範例

根據預設，使用者和角色沒有建立或修改 CloudShell 資源的許可。他們也無法使用 AWS Management Console、 AWS Command Line Interface (AWS CLI) 或 AWS API 來執行任務。若要授予使用者對其所需資源執行動作的許可，IAM 管理員可以建立 IAM 政策。然後，管理員可以將 IAM 政策新增至角色，使用者便能擔任這些角色。

如需了解如何使用這些範例 JSON 政策文件建立 IAM 身分型政策，請參閱《IAM 使用者指南》中的建立 IAM 政策 (主控台)。

如需有關 CloudShell 定義的動作和資源類型的詳細資訊，包括每種資源類型的 ARNs 格式，請參閱《服務授權參考》中的 AWS CloudShell 的動作、資源和條件索引鍵。

主題

政策最佳實務

身分型政策會判斷您帳戶中的某個人員是否可以建立、存取或刪除 CloudShell 資源。這些動作可能會讓您的 AWS 帳戶產生費用。當您建立或編輯身分型政策時，請遵循下列準則及建議事項：

- 開始使用 AWS 受管政策並邁向最低權限許可 – 若要開始將許可授予您的使用者和工作負載，請使用將許可授予許多常見使用案例的 AWS 受管政策。它們可在您的 中使用 AWS 帳戶。我們建議您定義特定於使用案例 AWS 的客戶受管政策，進一步減少許可。如需更多資訊，請參閱 IAM 使用者指南中的 AWS 受管政策或任務職能的AWS 受管政策。
- 套用最低權限許可 – 設定 IAM 政策的許可時，請僅授予執行任務所需的許可。為實現此目的，您可以定義在特定條件下可以對特定資源採取的動作，這也稱為最低權限許可。如需使用 IAM 套用許可的更多相關資訊，請參閱 IAM 使用者指南中的 IAM 中的政策和許可。
- 使用 IAM 政策中的條件進一步限制存取權 – 您可以將條件新增至政策，以限制動作和資源的存取。例如，您可以撰寫政策條件，指定必須使用 SSL 傳送所有請求。如果透過特定 例如 使用服務動作 AWS 服務，您也可以使用條件來授予其存取權 AWS CloudFormation。如需詳細資訊，請參閱 IAM 使用者指南中的 IAM JSON 政策元素：條件。
- 使用 IAM Access Analyzer 驗證 IAM 政策，確保許可安全且可正常運作 – IAM Access Analyzer 驗證新政策和現有政策，確保這些政策遵從 IAM 政策語言 (JSON) 和 IAM 最佳實務。IAM Access

Analyzer 提供 100 多項政策檢查及切實可行的建議，可協助您撰寫安全且實用的政策。如需詳細資訊，請參閱《IAM 使用者指南》中的使用 IAM Access Analyzer 驗證政策。

- 需要多重要素驗證 (MFA) – 如果您的案例需要 IAM 使用者或 中的根使用者 AWS 帳戶，請開啟 MFA 以提高安全性。如需在呼叫 API 操作時請求 MFA，請將 MFA 條件新增至您的政策。如需詳細資訊，請參閱《IAM 使用者指南》https://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_mfa_configure-api-require.html中的透過 MFA 的安全 API 存取。

如需 IAM 中最佳實務的相關資訊，請參閱 IAM 使用者指南中的 IAM 安全最佳實務。

使用 CloudShell 主控台

若要存取 AWS CloudShell 主控台，您必須擁有一組最低許可。這些許可必須允許您列出和檢視 中 CloudShell 資源的詳細資訊 AWS 帳戶。如果您建立比最基本必要許可更嚴格的身分型政策，則對於具有該政策的實體 (使用者或角色) 而言，主控台就無法如預期運作。

對於僅呼叫 AWS CLI 或 AWS API 的使用者，您不需要允許最低主控台許可。反之，只需允許存取符合他們嘗試執行之 API 操作的動作就可以了。

為了確保使用者和角色仍然可以使用 CloudShell 主控台，也請將 CloudShell *ConsoleAccess*或 *ReadOnly* AWS 受管政策連接到實體。如需詳細資訊，請參閱《IAM 使用者指南》中的新增許可到使用者。

允許使用者檢視他們自己的許可

此範例會示範如何建立政策，允許 IAM 使用者檢視附加到他們使用者身分的內嵌及受管政策。此政策包含在主控台或使用 或 AWS CLI AWS API 以程式設計方式完成此動作的許可。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
```

```
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

對 AWS CloudShell 身分和存取進行故障診斷

使用以下資訊來協助您診斷和修正在使用 CloudShell 和 IAM 時可能遇到的常見問題。

主題

- 我無權在 CloudShell 中執行動作
- 我未獲得執行 iam:PassRole 的授權
- 我想要允許 以外的人員 AWS 帳戶 存取我的 CloudShell 資源

我無權在 CloudShell 中執行動作

如果您收到錯誤，告知您未獲授權執行動作，您的政策必須更新，允許您執行動作。

下列範例錯誤會在mateojackson IAM 使用者嘗試使用主控台檢視一個虛構 *my-example-widget* 資源的詳細資訊，但卻無虛構 awes:*GetWidget* 許可時發生。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 awes:GetWidget on resource: my-example-widget
```

在此情況下，必須更新 mateojackson 使用者的政策，允許使用 awes:*GetWidget* 動作存取 *my-example-widget* 資源。

如果您需要協助，請聯絡您的 AWS 管理員。您的管理員提供您的簽署憑證。

我未獲得執行 iam:PassRole 的授權

如果您收到錯誤，告知您無權執行 iam:PassRole動作，您的政策必須更新，以允許您將角色傳遞給 CloudShell。

有些 AWS 服務 可讓您將現有角色傳遞給該服務，而不是建立新的服務角色或服務連結角色。如需執行此作業，您必須擁有將角色傳遞至該服務的許可。

當名為 的 IAM marymajor 使用者嘗試使用主控台在 CloudShell 中執行動作時，會發生下列範例錯誤。但是，動作請求服務具備服務角色授予的許可。Mary 沒有將角色傳遞至該服務的許可。

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

在這種情況下，Mary 的政策必須更新，允許她執行 iam:PassRole 動作。

如果您需要協助，請聯絡您的 AWS 管理員。您的管理員提供您的簽署憑證。

我想要允許 以外的人員 AWS 帳戶 存取我的 CloudShell 資源

您可以建立一個角色，讓其他帳戶中的使用者或您組織外部的人員存取您的資源。您可以指定要允許哪些信任物件取得該角色。針對支援基於資源的政策或存取控制清單 (ACL) 的服務，您可以使用那些政策來授予人員存取您的資源的許可。

如需進一步了解，請參閱以下內容：

- 若要了解 CloudShell 是否支援這些功能，請參閱 AWS CloudShell 如何與 IAM 搭配使用。
- 若要了解如何 AWS 帳戶 在您擁有的 資源之間提供存取權，請參閱《IAM 使用者指南》中的在您擁有 AWS 帳戶 的另一個 中提供存取權給 IAM 使用者。
- 若要了解如何將資源的存取權提供給第三方 AWS 帳戶，請參閱《IAM 使用者指南》中的將存取權提供給第三方 AWS 帳戶 擁有。
- 如需了解如何透過聯合身分提供存取權，請參閱 IAM 使用者指南中的將存取權提供給在外部進行身分驗證的使用者 (聯合身分)。
- 如需了解使用角色和資源型政策進行跨帳戶存取之間的差異，請參閱《IAM 使用者指南》中的 IAM 中的跨帳戶資源存取。

使用 IAM 政策管理 AWS CloudShell 存取和用量

透過 提供的存取管理資源 AWS Identity and Access Management，管理員可以將許可授予 IAM 使用者。如此一來，這些使用者可以存取 AWS CloudShell 和使用環境的功能。管理員也可以建立政策，在精細層級指定這些使用者可以使用 shell 環境執行的動作。

管理員授予使用者存取權的最快速方法是透過 AWS 受管政策。AWS 受管政策是由 建立和管理的獨立政策 AWS。下列 受 AWS 管政策 AWS CloudShell 可以連接到 IAM 身分：

- AWS CloudShellFullAccess：授予許可，以 AWS CloudShell 完整存取所有功能。

AWS CloudShellFullAccess 政策使用萬用字元 (*) 字元為 IAM 身分 （使用者、角色或群組） 提供 CloudShell 和功能的完整存取權。如需此政策的詳細資訊，請參閱《 AWS 受管政策使用者指南》中的 AWS CloudShellFullAccess。

> **Note**
>
> 具有下列 AWS 受管政策的 IAM 身分也可以啟動 CloudShell。不過，這些政策提供廣泛的許可。因此，我們建議您只在這些政策對 IAM 使用者的任務角色至關重要時，才授予這些政策。
>
> - 管理員：為 IAM 使用者提供完整存取權，並允許他們將許可委派給其中的每個服務和資源 AWS。
> - 開發人員進階使用者：可讓 IAM 使用者執行應用程式開發任務，並建立和設定支援 AWS 感知應用程式開發的資源和服務。
>
> 如需連接受管政策的詳細資訊，請參閱《IAM 使用者指南》中的新增 IAM 身分許可 （主控台）。

AWS CloudShell 使用自訂政策在 中管理允許的動作

若要管理 IAM 使用者可以使用 CloudShell 執行的動作，請建立使用 CloudShellPolicy 受管政策做為範本的自訂政策。或者，編輯內嵌在相關 IAM 身分 （使用者、群組或角色） 中的內嵌政策。

例如，您可以允許 IAM 使用者存取 CloudShell，但防止他們轉送用於登入 的 CloudShell 環境登入資料 AWS Management Console。

Important

若要 AWS CloudShell 從 啟動 AWS Management Console，IAM 使用者需要下列動作的許可：

- `CreateEnvironment`
- `CreateSession`
- `GetEnvironmentStatus`
- `StartEnvironment`

如果連接的政策未明確允許其中一個動作，當您嘗試啟動 CloudShell 時，會傳回 IAM 許可錯誤。

AWS CloudShell 許可

名稱	所授予之許可的描述	啟動 CloudShell 時需要？
`cloudshell:CreateEnvironment`	建立 CloudShell 環境、在 CloudShell 工作階段開始時擷取配置，並從後端的 Web 應用程式儲存目前的配置。此許可僅預期 `*` 的值，`Resource`如中所述the section called “CloudShell 的 IAM 政策範例”。	是
`cloudshell:CreateSession`	從 連線至 CloudShell 環境 AWS Management Console。	是

名稱	所授予之許可的描述	啟動 CloudShell 時需要？
cloudshell:GetEnvironmentStatus	讀取 CloudShell 環境的狀態。	是
cloudshell:DeleteEnvironment	刪除 CloudShell 環境。	否
cloudshell:GetFileDownloadUrls	產生預先簽章的 Amazon S3 URLs，用於使用 CloudShell Web 介面透過 CloudShell 下載檔案。這不適用於 VPC 環境。	否
cloudshell:GetFileUploadUrls	產生預先簽章的 Amazon S3 URLs，用於使用 CloudShell Web 介面透過 CloudShell 上傳檔案。這不適用於 VPC 環境。	否
cloudshell:DescribeEnvironments	描述環境。	否
cloudshell:PutCredentials	將用來登入 的登入資料轉送 AWS Management Console 至 CloudShell。	否
cloudshell:StartEnvironment	啟動已停止的 CloudShell 環境。	是
cloudshell:StopEnvironment	停止正在執行的 CloudShell 環境。	否
cloudshell:ApproveCommand	核准從其他 AWS Service 主控台傳送至 CloudShell 的命令。	否

CloudShell 的 IAM 政策範例

下列範例示範如何建立政策來限制誰可以存取 CloudShell。這些範例也會顯示可在 shell 環境中執行的動作。

以下政策強制完全拒絕存取 CloudShell 及其功能。

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Sid": "DenyCloudShell",
        "Effect": "Deny",
        "Action": [
            "cloudshell:*"
        ],
        "Resource": "*"
    }]
}
```

以下政策允許 IAM 使用者存取 CloudShell，但封鎖他們產生預先簽章URLs 以進行檔案上傳和下載。使用者仍然可以使用 wget 等用戶端，在環境中來回傳輸檔案。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
        "Sid": "AllowUsingCloudshell",
        "Effect": "Allow",
        "Action": [
            "cloudshell:*"
        ],
        "Resource": "*"
    },
    {
        "Sid": "DenyUploadDownload",
        "Effect": "Deny",
        "Action": [
            "cloudshell:GetFileDownloadUrls",
            "cloudshell:GetFileUploadUrls"
        ],
        "Resource": "*"
    }]
```

```
}
```

下列政策允許 IAM 使用者存取 CloudShell。不過，政策 AWS Management Console 會防止您用來登入 的登入資料轉送至 CloudShell 環境。具有此政策的 IAM 使用者需要在 CloudShell 中手動設定其登入資料。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
        "Sid": "AllowUsingCloudshell",
        "Effect": "Allow",
        "Action": [
            "cloudshell:*"
        ],
        "Resource": "*"
    },
    {
        "Sid": "DenyCredentialForwarding",
        "Effect": "Deny",
        "Action": [
            "cloudshell:PutCredentials"
        ],
        "Resource": "*"
    }]
}
```

下列政策允許 IAM 使用者建立 AWS CloudShell 環境。

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Sid": "CloudShellUser",
        "Effect": "Allow",
        "Action": [
            "cloudshell:CreateEnvironment",
            "cloudshell:CreateSession",
            "cloudshell:GetEnvironmentStatus",
            "cloudshell:StartEnvironment"
        ],
        "Resource": "*"
     }]
```

```
}
```

建立和使用 CloudShell VPC 環境所需的 IAM 許可

若要建立和使用 CloudShell VPC 環境，IAM 管理員必須啟用對 VPC 特定 Amazon EC2 許可的存取。本節列出建立和使用 VPC 環境所需的 Amazon EC2 許可。

若要建立 VPC 環境，指派給您角色的 IAM 政策必須包含下列 Amazon EC2 許可：

- `ec2:DescribeVpcs`
- `ec2:DescribeSubnets`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeDhcpOptions`
- `ec2:DescribeNetworkInterfaces`

- `ec2:CreateTags`
- `ec2:CreateNetworkInterface`
- `ec2:CreateNetworkInterfacePermission`

我們也建議包括：

- ec2:DeleteNetworkInterface

> **Note**
>
> 此許可不是強制性的，但 CloudShell 需要此許可才能清除其建立的 ENI 資源（為 CloudShell VPC 環境建立ENIs 會加上ManagedByCloudShell金鑰標籤）。如果此許可未啟用，您必須在每次 CloudShell VPC 環境使用後手動清除 ENI 資源。

授予完整 CloudShell 存取權的 IAM 政策，包括對 VPC 的存取

下列範例顯示如何啟用 CloudShell 的完整許可，包括對 VPC 的存取：

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

```
    {
      "Sid": "AllowCloudShellOperations",
      "Effect": "Allow",
      "Action": [
        "cloudshell:*"
      ],
      "Resource": "*"
    },
    {
      "Sid": "AllowDescribeVPC",
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeDhcpOptions",
        "ec2:DescribeNetworkInterfaces",
        "ec2:DescribeSubnets",
        "ec2:DescribeSecurityGroups",
        "ec2:DescribeVpcs"
      ],
      "Resource": "*"
    },
    {
      "Sid": "AllowCreateTagWithCloudShellKey",
      "Effect": "Allow",
      "Action": [
        "ec2:CreateTags"
      ],
      "Resource": "arn:aws:ec2:*:*:network-interface/*",
      "Condition": {
        "StringEquals": {
          "ec2:CreateAction": "CreateNetworkInterface"
        },
        "ForAnyValue:StringEquals": {
          "aws:TagKeys": "ManagedByCloudShell"
        }
      }
    },
    {
      "Sid": "AllowCreateNetworkInterfaceWithSubnetsAndSG",
      "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterface"
      ],
      "Resource": [
        "arn:aws:ec2:*:*:subnet/*",
```

```
        "arn:aws:ec2:*:*:security-group/*"
      ]
    },
    {
      "Sid": "AllowCreateNetworkInterfaceWithCloudShellTag",
      "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterface"
      ],
      "Resource": "arn:aws:ec2:*:*:network-interface/*",
      "Condition": {
        "ForAnyValue:StringEquals": {
          "aws:TagKeys": "ManagedByCloudShell"
        }
      }
    },
    {
      "Sid": "AllowCreateNetworkInterfacePermissionWithCloudShellTag",
      "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterfacePermission"
      ],
      "Resource": "arn:aws:ec2:*:*:network-interface/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/ManagedByCloudShell": ""
        }
      }
    },
    {
      "Sid": "AllowDeleteNetworkInterfaceWithCloudShellTag",
      "Effect": "Allow",
      "Action": [
        "ec2:DeleteNetworkInterface"
      ],
      "Resource": "arn:aws:ec2:*:*:network-interface/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/ManagedByCloudShell": ""
        }
      }
    }
  ]
```

```
}
```

將 IAM 條件金鑰用於 VPC 環境

您可以針對 VPC 設定使用 CloudShell 特定條件金鑰，為您的 VPC 環境提供額外的許可控制。您也可以指定 VPC 環境可以和不能使用的子網路和安全群組。

CloudShell 在 IAM 政策中支援下列條件金鑰：

- `CloudShell:VpcIds` – 允許或拒絕一或多個 VPCs
- `CloudShell:SubnetIds` – 允許或拒絕一或多個子網路
- `CloudShell:SecurityGroupIds` – 允許或拒絕一或多個安全群組

> **Note**
>
> 如果修改可存取公有 CloudShell 環境的使用者許可，以對`cloudshell:createEnvironment`動作新增限制，他們仍然可以存取現有的公有環境。不過，如果您想要修改具有此限制的 IAM 政策，並停用其對現有公有環境的存取，您必須先更新具有限制的 IAM 政策，然後確保帳戶中的每個 CloudShell 使用者使用 CloudShell Web 使用者介面 (動作 → 刪除 CloudShell 環境) 手動刪除現有的公有環境。

具有 VPC 設定條件金鑰的範例政策

下列範例示範如何使用條件金鑰進行 VPC 設定。建立具有所需限制的政策陳述式之後，請附加目標 使用者或角色的政策陳述式。

確保使用者僅建立 VPC 環境，並拒絕建立公有環境

為了確保使用者只能建立 VPC 環境，請使用拒絕許可，如下列範例所示：

```
{
  "Statement": [
    {
      "Sid": "DenyCloudShellNonVpcEnvironments",
      "Action": [
        "cloudshell:CreateEnvironment"
      ],
      "Effect": "Deny",
      "Resource": "*",
```

```
      "Condition": {
        "Null": {
          "cloudshell:VpcIds": "true"
        }
      }
    }
  ]
}
```

拒絕使用者存取特定 VPC、子網路或安全群組

若要拒絕使用者存取特定 VPC，請使用 StringEquals 來檢查 cloudshell:VpcIds 條件的值。下列範例拒絕使用者存取 vpc-1和 vpc-2：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "EnforceOutOfVpc",
      "Action": [
        "cloudshell:CreateEnvironment"
      ],
      "Effect": "Deny",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "cloudshell:VpcIds": [
            "vpc-1",
            "vpc-2"
          ]
        }
      }
    }
  ]
}
```

若要拒絕使用者存取特定 VPC，請使用 StringEquals 來檢查 cloudshell:SubnetIds 條件的值。下列範例拒絕使用者存取 subnet-1和 subnet-2：

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

```
    {
      "Sid": "EnforceOutOfSubnet",
      "Action": [
        "cloudshell:CreateEnvironment"
      ],
      "Effect": "Deny",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "cloudshell:SubnetIds": [
            "subnet-1",
            "subnet-2"
          ]
        }
      }
    }
  ]
}
```

若要拒絕使用者存取特定 VPC，請使用 StringEquals 來檢查 cloudshell:SecurityGroupIds 條件的值。下列範例拒絕使用者存取 sg-1和 sg-2：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "EnforceOutOfSecurityGroups",
      "Action": [
        "cloudshell:CreateEnvironment"
      ],
      "Effect": "Deny",
      "Resource": "*",
      "Condition": {
        "ForAnyValue:StringEquals": {
          "cloudshell:SecurityGroupIds": [
            "sg-1",
            "sg-2"
          ]
        }
      }
    }
  ]
}
```

允許使用者使用特定 VPC 組態建立環境

若要允許使用者存取特定 VPCs，請使用 StringEquals檢查cloudshell:VpcIds條件的值。下列範例允許使用者存取 vpc-1和 vpc-2：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "EnforceStayInSpecificVpc",
      "Action": [
        "cloudshell:CreateEnvironment"
      ],
      "Effect": "Allow",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "cloudshell:VpcIds": [
            "vpc-1",
            "vpc-2"
          ]
        }
      }
    }
  ]
}
```

若要允許使用者存取特定 VPCs，請使用 StringEquals檢查cloudshell:SubnetIds條件的值。下列範例允許使用者存取 subnet-1和 subnet-2：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "EnforceStayInSpecificSubnets",
      "Action": [
        "cloudshell:CreateEnvironment"
      ],
      "Effect": "Allow",
      "Resource": "*",
      "Condition": {
        "ForAllValues:StringEquals": {
          "cloudshell:SubnetIds": [
```

```
            "subnet-1",
            "subnet-2"
          ]
        }
      }
    }
  ]
}
```

若要允許使用者存取特定 VPCs，請使用 StringEquals檢查cloudshell:SecurityGroupIds條件的值。下列範例允許使用者存取 sg-1和 sg-2：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "EnforceStayInSpecificSecurityGroup",
      "Action": [
        "cloudshell:CreateEnvironment"
      ],
      "Effect": "Allow",
      "Resource": "*",
      "Condition": {
        "ForAllValues:StringEquals": {
          "cloudshell:SecurityGroupIds": [
            "sg-1",
            "sg-2"
          ]
        }
      }
    }
  ]
}
```

存取 的許可 AWS 服務

CloudShell 會使用您用來登入 的 IAM 登入資料 AWS Management Console。

> **Note**
>
> 若要使用您用來登入 的 IAM 登入資料 AWS Management Console，您必須擁有 `cloudshell:PutCredentials`許可。

CloudShell 的預先驗證功能可讓您方便使用 AWS CLI。不過，IAM 使用者仍需要從命令列 AWS 服務呼叫 的明確許可。

例如，假設 IAM 使用者需要建立 Amazon S3 儲存貯體並將檔案上傳為物件。您可以建立明確允許這些動作的政策。IAM 主控台提供互動式視覺化編輯器，引導您完成建置 JSON 格式政策文件的程序。建立政策之後，您可以將政策連接至相關的 IAM 身分 （使用者、群組或角色）。

如需連接受管政策的詳細資訊，請參閱《IAM 使用者指南》中的新增 IAM 身分許可 （主控台）。

在 CloudShell 中存取 Amazon Q CLI 功能的許可

若要在 CloudShell 中使用 Amazon Q CLI 功能，例如內嵌建議、聊天和翻譯，請確定您擁有必要的 IAM 許可。如果您無法存取 CloudShell 中的 Amazon Q CLI 功能，請聯絡您的管理員，為您提供必要的 IAM 許可。如需詳細資訊，請參閱《Amazon Q 開發人員使用者指南》中的 Amazon Q 開發人員的身分型政策範例。

在 中記錄和監控 AWS CloudShell

本主題說明如何使用 CloudTrail 記錄和監控 AWS CloudShell 活動和效能。

使用 CloudTrail 監控活動

AWS CloudShell 已與 服務整合 AWS CloudTrail，此服務提供使用者、角色或 AWS 服務 所採取動作的記錄 AWS CloudShell。CloudTrail 會將 的所有 API 呼叫擷取 AWS CloudShell 為事件。擷取的呼叫包括來自 AWS CloudShell 主控台的呼叫，以及對 AWS CloudShell API 的程式碼呼叫。

如果您建立線索，則可以將 CloudTrail 事件持續交付至 Amazon Simple Storage Service (Amazon S3) 儲存貯體。這包括 的事件 AWS CloudShell。

即使您未設定追蹤，依然可以透過 CloudTrail 主控台中的事件歷史記錄檢視最新事件。使用 CloudTrail 收集的資訊，您可以探索有關請求的各種資訊。例如，您可以判斷向 AWS CloudShell 提出的請求，也可以了解提出請求的 IP 地址、提出請求的人員，以及提出請求的時間。

AWS CloudShell CloudTrail 中的

下表列出儲存在 CloudTrail 日誌檔案中 AWS CloudShell 的事件。

> **Note**
>
> AWS CloudShell 事件，包括：
>
> - * 表示它是非變動（唯讀）API 呼叫。
> - 該詞與託管 shell 體驗的運算環境生命週期`Environment`相關。
> - 此字詞會`Layout`還原 CloudShell 終端機中的所有瀏覽器索引標籤。

CloudTrail 中的 CloudShell 事件 CloudTrail

事件名稱	描述
`createEnvironment`	建立 CloudShell 環境時發生。
`createSession`	當 CloudShell 環境從 連線時發生 AWS Management Console。
`deleteEnvironment`	刪除 CloudShell 環境時發生。
`deleteSession`	刪除目前瀏覽器標籤中執行之 CloudShell 標籤中的工作階段時發生。
`getEnvironmentStatus*`	擷取 CloudShell 環境的狀態時發生。
`getFileDownloadUrls*`	當產生用於使用 CloudShell Web 介面透過 CloudShell 下載檔案的預先簽章 Amazon S3 URLs CloudShell 時發生。
`getFileUploadUrls*`	當產生用於使用 CloudShell Web 介面透過 CloudShell 上傳檔案的預先簽章 Amazon S3 URLs CloudShell 時發生。
`cloudshell:DescribeEnvironments`	描述環境。

事件名稱	描述
getLayout*	當擷取工作階段開始時的 CloudShell 配置時發生。
putCredentials	當用於登入 的登入資料轉送 AWS Management Console 至 CloudShell 時發生。
redeemCode*	當 CloudShell 環境中擷取重新整理權杖的工作流程開始時，便會發生。您稍後可以在 putCredentials 命令中使用此字符來存取 CloudShell 環境。
sendHeartBeat	發生以確認 CloudShell 工作階段處於作用中狀態。
startEnvironment	當 CloudShell 環境啟動時發生。
stopEnvironment	當執行中的 CloudShell 環境停止時發生。
updateLayout	當後端 Web 應用程式的目前配置儲存時，便會發生。

包含「Layout」一詞的事件會還原 CloudShell 終端機中的所有瀏覽器索引標籤。

AWS CloudShell 動作的 EventBridge 規則

使用 EventBridge 規則，您可以指定 EventBridge 收到符合規則的事件時要採取的目標動作。您可以定義規則，根據記錄為 CloudTrail 日誌檔案中事件 AWS CloudShell 的動作，指定要採取的目標動作。

例如，您可以使用 put-rule命令使用 建立 EventBridge 規則 AWS CLI。put-rule 呼叫必須至少包含 EventPattern 或 ScheduleExpression。具有 EventPattern 的規則會在觀察到符合事件時觸發。AWS CloudShell 事件的 EventPattern：

```
{ "source": [ "aws.cloudshell" ], "detail-type": [ "AWS API Call via CloudTrail" ],
 "detail": { "eventSource": [ "cloudshell.amazonaws.com" ] } }
```

如需詳細資訊，請參閱《Amazon EventBridge 使用者指南》中的 EventBridge 中的事件和事件模式。EventBridge

的合規驗證 AWS CloudShell

在多個 AWS 合規計畫中，第三方稽核人員會評估 AWS 服務的安全性和合規性。

AWS CloudShell 在以下合規計劃範圍內：

SOC

AWS 系統和組織控制 (SOC) 報告是獨立的第三方檢查報告，示範 如何 AWS 實現關鍵合規控制和目標。

服務	SDK	SOC 1、2、3
AWS CloudShell	CloudShell	✓

PCI

支付卡產業資料安全標準 (PCI DSS) 是由 PCI 安全標準委員會管理的專屬資訊安全標準，由 American Express、Discover Financial Services、JCB International、MasterCard Worldwide 及 Visa Inc. 共同創設。

服務	SDK	PCI
AWS CloudShell	CloudShell	✓

ISO 和 CSA STAR 認證和服務

AWS 具有符合 ISO/IEC 27001：2013、27017：2015、27018：2019、27701：2019、22301：2019、9001：2015 和 CSA STAR CCM v4.0 的認證。

服務	SDK	ISO 和 CSA STAR 認證和服務
AWS CloudShell	CloudShell	✓

FedRamp

聯邦風險與授權管理計劃 (FedRAMP) 是一項美國政府整體計劃，提供標準化的方法，為雲端產品和服務進行安全評估、授權和持續監控。

服務	SDK	FedRAMP Moderate (East/West)	FedRAMP High (GovCloud)
AWS CloudShell	CloudShell	✓	✓

DoD CC SRG

國防部 (DoD) 雲端運算安全要求指南 (SRG) 提供標準化的評定和授權程序，讓雲端服務提供者 (CSP) 取得 DoD 臨時授權，以便為 DoD 客戶提供服務。

透過 DoD CC SRG 評估和授權的服務將具有以下狀態：

- 第三方評估組織 (3PAO) 評估：此服務目前正在由我們的第三方評估者進行評估。
- 聯合授權委員會 (JAB) 審查：此服務目前正在進行 JAB 審查。
- Defense Information Systems Agency (DISA) Review：此服務目前正在進行 DISA 審查。

服務	SDK	DoD CC SRG IL2 (East/West)	DoD CC SRG IL2 (GovCloud)	DoD CC SRG IL4 (GovCloud)	DoD CC SRG IL5 (GovCloud)	DoD CC SRG IL6 (AWS 秘密區域)
AWS CloudShell	CloudShell	✓	✓	✓	✓	N/A

HIPAA BAA

1996 年健康保險流通與責任法案 (HIPAA) 是一項聯邦法律，要求制定國家標準以保護敏感的患者健康資訊在未經患者同意或不知情的情況下公開。

AWS 可讓受 HIPAA 約束的涵蓋實體及其商業夥伴安全地處理、儲存和傳輸受保護醫療資訊 (PHI)。此外，截至 2013 年 7 月， AWS 為此類客戶提供標準化商業夥伴增補合約 (BAA)。

服務	SDK	HIPAA BAA
AWS CloudShell	CloudShell	✓

IRAP

資訊安全註冊評估人員計劃 (IRAP) 可讓澳洲政府客戶驗證是否有適當的控制措施，並決定適當的責任模式，以便滿足由澳洲網路安全中心 (ACSC) 所制定的澳洲政府資訊安全手冊 (ISM) 要求。

服務	命名空間*	IRAP 受保護
AWS CloudShell	N/A	✓

* 命名空間可協助您識別整個 AWS 環境的服務。例如，當您建立 IAM 政策時，請使用 Amazon Resource Name (ARNs) 和 read AWS CloudTrail 日誌。

MTCS

多層級雲端安全 (MTCS) 是以 ISO 27001/02 資訊安全管理系統 (ISMS) 標準為基礎的營運新加坡安全管理標準 (SPRING SS 584)。

服務	SDK	美國東部（俄亥俄）	美國東部（維吉尼亞北部）	美國西部（奧勒岡）	美國西部（加利福尼亞北部）	新加坡	首爾
AWS CloudShell	CloudShell	✓	✓	✓	N/A	N/A	N/A

C5

雲端運算合規控制目錄 (C5) 是德國政府支持的認證計劃，由德國聯邦資訊安全辦公室 (BSI) 在德國推出，旨在協助組織在德國政府的「雲端供應商安全建議」範圍內使用雲端服務時展示針對常見網路攻擊的操作安全性。

服務	SDK	C5
AWS CloudShell	CloudShell	✓

ENS 高級

ENS (Esquema Nacional de Seguridad) 認證方案是由財務與公共行政部和 CCN （國家加密中心） 所開發。這包括為了充分保護資訊所需的基本原則和最低要求。

服務	SDK	ENS 高
AWS CloudShell	CloudShell	✓

FINMA

瑞士金融市場監督管理局 (FINMA) 是瑞士獨立的金融市場監管機構。 AWS與 FINMA 要求的一致性證明了我們持續致力於滿足瑞士金融服務監管機構和客戶對雲端服務供應商的更高期望。

服務	SDK	FINMA
AWS CloudShell	CloudShell	✓

PiTuKri

AWS 與 PiTuKri 要求保持一致，證明我們持續致力於滿足芬蘭運輸和通訊局 Traficom 為雲端服務供應商設定的更高期望。

服務	SDK	PiTuKri
AWS CloudShell	CloudShell	✓

如需在特定合規計劃範圍內 AWS 的服務清單，請參閱合規計劃範圍內的 AWS 服務。如需一般資訊，請參閱 AWS Compliance Programs。

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 AWS Artifact 中的報告。

您使用 時的合規責任 AWS CloudShell 取決於資料的機密性、您公司的合規目標，以及適用的法律和法規。 AWS 提供下列資源來協助合規：

- 安全與合規快速入門指南 – 這些部署指南討論架構考量，並提供在其中部署以安全為重心和以合規為重心的基準環境的步驟 AWS。
- HIPAA 安全與合規架構白皮書 – 此白皮書說明公司如何使用 AWS 來建立符合 HIPAA 規範的應用程式。
- AWS 合規資源 – 此工作手冊和指南集合可能適用於您的產業和位置。
- 《 AWS Config 開發人員指南》中的使用 規則評估資源 – AWS Config 服務會評估資源組態符合內部實務、產業準則和法規的程度。
- AWS Security Hub – AWS 此服務提供 內安全狀態的全方位檢視 AWS ，可協助您檢查是否符合安全產業標準和最佳實務。

中的彈性 AWS CloudShell

AWS 全球基礎設施是以 AWS 區域和可用區域為基礎建置。 AWS 區域提供多個實體分隔和隔離的可用區域，這些可用區域以低延遲、高輸送量和高備援聯網連接。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、容錯能力和擴展能力，均較單一或多個資料中心的傳統基礎設施還高。

如需 AWS 區域和可用區域的詳細資訊，請參閱 AWS 全球基礎設施。

除了 AWS 全球基礎設施之外， AWS CloudShell 還支援下列功能，以支援您的資料彈性和備份需求：

- 使用 AWS CLI 呼叫在 中指定主目錄中的檔案， AWS CloudShell 並將其新增為 Amazon S3 儲存貯體中的物件。如需範例，請參閱 入門 AWS CloudShell。

中的基礎設施安全 AWS CloudShell

作為受管服務， AWS CloudShell 受到 AWS 全球網路安全的保護。如需 AWS 安全服務以及 如何 AWS 保護基礎設施的相關資訊，請參閱AWS 雲端安全。若要使用基礎設施安全的最佳實務設計您的 AWS 環境，請參閱安全支柱 AWS Well-Architected Framework 中的基礎設施保護。

您可以使用 AWS 發佈的 API 呼叫， AWS CloudShell 透過網路存取 。使用者端必須支援下列專案：

- Transport Layer Security (TLS)。我們需要 TLS 1.2 並建議使用 TLS 1.3。

- 具備完美轉送私密(PFS)的密碼套件，例如 DHE (Ephemeral Diffie-Hellman)或 ECDHE (Elliptic Curve Ephemeral Diffie-Hellman)。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 主體相關聯的私密存取金鑰來簽署。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。

> **Note**
>
> 根據預設， AWS CloudShell 會自動為運算環境的系統套件安裝安全修補程式。

的安全最佳實務 AWS CloudShell

以下最佳實務為一般準則，並不代表完整的安全解決方案。由於這些最佳實務可能不適合或不適合您的環境，我們建議您將其視為有用的考量，而非處方藥。

的一些安全最佳實務 AWS CloudShell

- 使用 IAM 許可和政策來控制對 的存取， AWS CloudShell 並確保使用者只能執行其角色所需的動作（例如，下載和上傳檔案）。如需詳細資訊，請參閱使用 IAM 政策管理 AWS CloudShell 存取和用量。
- 請勿在 IAM 實體中包含敏感資料，例如使用者、角色或工作階段名稱。
- 保持啟用安全貼上功能，以捕捉您從外部來源複製的文字中的潛在安全風險。預設會啟用安全貼上。如需針對多行文字使用安全貼上的詳細資訊，請參閱針對多行文字使用安全貼上。
- 如果您將第三方應用程式安裝到 的運算環境，請熟悉共享安全責任模型 AWS CloudShell。
- 在編輯會影響使用者 shell 體驗的 shell 指令碼之前，請先準備復原機制。如需修改預設 shell 環境的詳細資訊，請參閱使用指令碼修改 shell。
- 將程式碼安全地存放在版本控制系統中。

AWS CloudShell 安全性FAQs

以下是 CloudShell 安全性常見問題的解答。

- 當您啟動 CloudShell 並啟動 shell 工作階段時，會使用哪些 AWS 程序和技術？
- 是否可以限制對 CloudShell 的網路存取？
- 我可以自訂 CloudShell 環境嗎？

- 我的$HOME目錄實際存放在哪裡 AWS 雲端？
- 是否可以加密我的$HOME目錄？
- 我可以在$HOME目錄上執行病毒掃描嗎？

當您啟動 CloudShell 並啟動 shell 工作階段時，會使用哪些 AWS 程序和技術？

登入時 AWS Management Console，請輸入您的 IAM 使用者登入資料。此外，當您從主控台介面啟動 CloudShell 時，這些登入資料會用於呼叫 CloudShell API，以建立服務的運算環境。接著會為運算環境建立 AWS Systems Manager 工作階段，CloudShell 會將命令傳送至該工作階段。

返回安全性FAQs清單

是否可以限制對 CloudShell 的網路存取？

對於公有環境，無法限制網路存取。如果您想要限制網路存取，您必須啟用僅建立 VPC 環境的許可，並拒絕建立公有環境。

如需詳細資訊，請參閱確保使用者僅建立 VPC 環境，並拒絕建立公有環境。

對於 CloudShell VPC 環境，網路設定會繼承自您的 VPC。在 VPC 中使用 CloudShell 可讓您控制 CloudShell VPC 環境的網路存取。

返回安全性FAQs清單

我可以自訂 CloudShell 環境嗎？

您可以為您的 CloudShell 環境下載並安裝公用程式和其他第三方軟體。工作階段之間只會保留安裝在$HOME目錄中的軟體。

根據AWS 共同責任模型的定義，您必須負責所安裝應用程式的必要組態和管理。

返回安全性FAQs清單

我的$HOME目錄實際存放在哪裡 AWS 雲端？

對於公有環境，用於在 中存放資料的基礎設施由 Amazon S3 $HOME提供。

對於 VPC 環境，您的$HOME目錄會在 VPC 環境逾時 （ 閒置 20-30 分鐘後 ） 或當您刪除或重新啟動環境時刪除。

返回安全性FAQs清單

是否可以加密我的$HOME目錄？

否，無法使用您自己的金鑰加密您的$HOME目錄。但是CloudShell 會在將$HOME目錄內容存放在 Amazon S3 時加密目錄內容。

返回安全性FAQs清單

我可以在$HOME目錄上執行病毒掃描嗎？

目前，無法執行$HOME目錄的病毒掃描。此功能的支援正在審核中。

返回安全性FAQs清單

我可以限制 CloudShell 的資料輸入或輸出嗎？

若要限制輸入或輸出，建議您使用 CloudShell VPC 環境。當您的 VPC 環境逾時（閒置 20-30 分鐘後）或當您刪除或重新啟動環境時，會刪除 VPC 環境的$HOME目錄。在動作功能表中，上傳和下載選項不適用於 VPC 環境。

返回安全性FAQs清單

AWS CloudShell 運算環境：規格和軟體

當您啟動時 AWS CloudShell，會建立以 Amazon Linux 2023 為基礎的運算環境來託管 shell 體驗。環境是以運算資源 (vCPU 和記憶體） 設定，並提供各式各樣的預先安裝軟體，可從命令列界面存取。請確定您在運算環境中安裝的任何軟體都已修補且為最新版本。您也可以透過安裝軟體和修改 shell 指令碼來設定預設環境。

運算環境資源

每個 AWS CloudShell 運算環境都會獲指派下列 CPU 和記憶體資源：

- 1 個 vCPU （虛擬中央處理單元）
- 2-GiB RAM

此外，會使用下列儲存組態佈建環境：

- 1-GB持久性儲存 （工作階段結束後儲存仍然存在）

如需詳細資訊，請參閱持久性儲存。

CloudShell 網路需求

WebSockets

CloudShell 取決於 WebSocket 通訊協定，該通訊協定允許使用者 Web 瀏覽器與 CloudShell 服務在 AWS 雲端中進行雙向互動式通訊。如果您在私有網路中使用瀏覽器，代理伺服器和防火牆可能會協助安全存取網際網路。WebSocket 通訊通常可以周遊代理伺服器，而不會發生問題。但在某些情況下，代理伺服器會阻止 WebSockets正常運作。如果發生此問題，您的 CloudShell 介面會報告下列錯誤：`Failed to open sessions : Timed out while opening the session`。

如果此錯誤重複發生，請參閱代理伺服器的文件，以確保其設定為允許 WebSockets 或者，您可以聯絡網路的系統管理員。

> **Note**
>
> 如果您想要允許列出特定 URLs 來定義精細許可，您可以新增 AWS Systems Manager 工作階段用來開啟 WebSocket 連線以傳送輸入和接收輸出的部分 URL。（您的 AWS CloudShell 命令會傳送至該 Systems Manager 工作階段。）
> Systems Manager 使用此 StreamUrl 的格式為 `wss://ssmmessages.region.amazonaws.com/v1/data-channel/session-id?stream=(input|output)`。
> 區域代表 支援的 AWS 區域的區域識別符 AWS Systems Manager，例如`us-east-2`美國東部（俄亥俄）區域。
> 由於 session-id 是在特定 Systems Manager 工作階段成功啟動後建立，因此您只能在更新 URL 允許清單`wss://ssmmessages.region.amazonaws.com`時指定 。如需詳細資訊，請參閱 AWS Systems Manager API 參考中的 StartSession 操作。

預先安裝的軟體

> **Note**
>
> 由於 AWS CloudShell 開發環境會定期更新以提供最新軟體的存取權，因此我們不會在本文件中提供特定版本編號。反之，我們會說明如何檢查已安裝的版本。若要檢查安裝的版本，請輸入程式名稱，後面接著 `--version`選項（例如，`git --version`)。

殼層

預先安裝的 shell

名稱	描述	版本資訊
Bash	Bash shell 是 的預設 shell 應用程式 AWS CloudShell。	`bash --version`
PowerShell (pwsh)	PowerShell 提供命令列界面和指令碼語言支援，建置在 Microsoft 的 .NET 命令語言執行期之上。PowerShell 使用名	`pwsh --version`

名稱	描述	版本資訊
	為 cmdlets的輕量型命令，接受並傳回 .NET 物件。	
Z Shell (zsh)	Z Shell 也稱為 zsh，是 Bourne Shell 的擴充版本，可為主題和外掛程式提供增強型自訂支援。	`zsh --version`

AWS 命令列界面 (CLI)

CLI

名稱	描述	版本資訊
AWS CDK 工具組 CLI	AWS CDK Toolkit 是 CLI 命令 cdk，是與您的 AWS CDK 應用程式互動的主要工具。它會執行您的應用程式、查詢您定義的應用程式模型，以及產生和部署 產生的 AWS CloudFormation 範本 AWS CDK。 如需詳細資訊，請參閱 AWS CDK Toolkit。	`cdk --version`
AWS CLI	AWS CLI 是命令列界面，可用來從命令列管理多個 AWS 服務，並使用指令碼將其自動化。如需詳細資訊，請參閱在 CloudShell 中從 CLI 管理 AWS 服務。 如需有關如何確保您使用的是 up-to-date AWS CLI 2 的資	`aws --version`

名稱	描述	版本資訊
	訊，請參閱 安裝 AWS CLI 到您的主目錄。	
EB CLI	CLI AWS Elastic Beanstalk 提供命令列界面，可簡化從本機儲存庫建立、更新和監控環境。 如需詳細資訊，請參閱《AWS Elastic Beanstalk 開發人員指南》中的使用 Elastic Beanstalk 命令列界面 (EB CLI)。	`eb --version`
Amazon ECS CLI	Amazon Elastic Container Service (Amazon ECS) 命令列界面 (CLI) 提供高階命令，可簡化叢集和任務的建立、更新和監控。 如需詳細資訊，請參閱《Amazon Elastic Container Service 開發人員指南》中的使用 Amazon ECS 命令列界面。	`ecs-cli --version`

名稱	描述	版本資訊
AWS SAM CLI	AWS SAM CLI 是一種命令列工具，可在 AWS Serverless Application Model 範本和應用程式程式碼上操作。您可以執行數個任務。這包括在本機叫用 Lambda 函數、為無伺服器應用程式建立部署套件，以及將無伺服器應用程式部署至 AWS 雲端。 如需詳細資訊，請參閱《 AWS Serverless Application Model 開發人員指南》中的 AWS SAM CLI 命令參考。	`sam --version`
AWS Tools for PowerShell	AWS Tools for PowerShell 是建置在 公開功能上的 PowerShell 模組 適用於 .NET 的 SDK。使用 AWS Tools for PowerShell，您可以從 PowerShell 命令列在您的 AWS 資源上編寫操作指令碼。 AWS CloudShell 預先安裝 的模組化版本 (AWS AWS Tools for PowerShell.Tools)。如需詳細資訊，請參閱AWS Tools for PowerShell 《 使用者指南》中的使用適用於 PowerShell 的 AWS 工具。	`pwsh --Command 'Get-AWSPowerShellVersion'`

執行期和 AWS SDKs：Node.js 和 Python 3

執行期和 AWS SDKs

名稱	描述	版本資訊
Node.js （含 npm)	Node.js 是一種 JavaScript 執行期，旨在更輕鬆地套用非同步程式設計技術。如需詳細資訊，請參閱官方 Node.js 網站上的文件。 npm 是套件管理員，可讓您存取 JavaScript 模組的線上登錄檔。如需詳細資訊，請參閱官方 npm 網站上的文件。	• Node.js： `node --version` • npm： `npm --version`
Node.js 中適用於 JavaScript 的 SDK	軟體開發套件 (SDK) 透過為 Amazon S3、Amazon EC2、DynamoDB 和 Amazon SWF 等 AWS 服務提供 JavaScript 物件，協助簡化編碼。如需詳細資訊，請參閱《適用於 JavaScript 的 AWS SDK 開發人員指南》。	`npm -g ls --depth 0 2>/dev/null | grep aws-sdk`
Python	Python 3 已準備好在 shell 環境中使用。Python 3 現在被視為程式設計語言的預設版本（對 Python 2 的支援已於 2020 年 1 月結束）。如需詳細資訊，請參閱官方 Python 網站上的文件。 此外，預先安裝 pip，Python 套件安裝程式。您可以使用此命令列程式從線上索引安裝 Python 套件，例如 Python 套件索引。如需詳細資訊，請參	• Python 3： `python3 --version` • pip： `pip3 --version`

名稱	描述	版本資訊
	閱 Python Packaging Authority 提供的文件。	
SDK for Python (Boto3)	Boto 是 Python 開發人員用來建立、設定和管理的軟體開發套件 (SDK) AWS 服務，例如 Amazon EC2 和 Amazon S3。開發套件提供easy-to-use物件導向 API，以及對 的低階存取 AWS 服務。 如需詳細資訊，請參閱 Boto3 文件。	`pip3 list | grep boto3`

開發工具和 shell 公用程式

開發工具和 shell 公用程式

名稱	描述	版本資訊
bash-completion	bash-completion 是 shell 函數的集合，可透過按 Tab 鍵自動完成部分輸入的命令或引數。您可以在 中找到 bash-completion 支援的套件`/usr/share/bash-completion/completions` 。 若要設定套件命令的自動完成，必須取得程式檔案。例如，若要設定 Git 命令的自動完成，請將以下行新增至 ，`.bashrc`以便在 AWS CloudShell 工作階段啟動時可以使用此功能：	`dnf info bash-completion`

名稱	描述	版本資訊
	`source /usr/share/bash-completion/completions/git` 如果您想要使用自訂完成指令碼，請將它們新增至您的持久性主目錄 (`$HOME`)，並直接在中取得它們`.bashrc`。 如需詳細資訊，請參閱 GitHub 上的專案的 README 頁面。	
cqlsh-expansion	`cqlsh-expansion` 是一種工具組，其中包含針對 Amazon Keyspaces 預先設定的 cqlsh 和協助程式，同時保持與 Apache Cassandra 的完整相容性。如需詳細資訊，請參閱使用 cqlsh 連線至 Amazon Keyspaces （適用於 Apache Cassandra) 開發人員指南中的 Amazon Keyspaces 。	`cqlsh-expansion --version`

名稱	描述	版本資訊
Docker	Docker 是開發、運送和執行應用程式的開放平台。Docker 可讓您將應用程式與基礎設施分開，以便快速交付軟體。它可讓您在內部建置 Dockerfile AWS CloudShell，並使用 CDK 建置 Docker 資產。如需 Docker 支援哪些 AWS 區域的詳細資訊，請參閱支援的 AWS 區域 AWS CloudShell。您應該知道 Docker 在環境中的空間有限。如果您有大型的個別映像，或預先存在太多的 Docker 映像，可能會導致問題。如需 Docker 的詳細資訊，請參閱 Docker 文件指南。	`docker --version`
Git	Git 是一種分散式版本控制系統，可透過分支工作流程和內容預備來支援現代軟體開發實務。如需詳細資訊，請參閱 Git 官方網站上的文件頁面。	`git --version`
iputils	iputils 套件包含 Linux 聯網的公用程式。如需所提供公用程式的詳細資訊，請參閱 GitHub 上的 iputils 儲存庫。	iputils 工具的範例： `arping -V`
jq	jq 公用程式會剖析 JSON 格式的資料，以產生由命令列篩選條件修改的輸出。如需詳細資訊，請參閱 GitHub 上託管的 jq 手冊。	`jq --version`

名稱	描述	版本資訊
kubectl	kubectl 是一種命令列工具，可使用 Kubernetes API 與 Kubernetes 叢集的控制平面進行通訊。	`kubectl --version`
make	make 公用程式使用 `makefiles` 來自動化任務集並組織程式碼編譯。如需詳細資訊，請參閱 GNU Make 文件。	`make --version`
man	man 命令提供命令列公用程式和工具的手動頁面。例如，會`man ls`傳回列出目錄內容之`ls`命令的手動頁面。如需詳細資訊，請參閱人物頁面上的 Wikipedia 項目。	`man --version`
nano	nano 是適用於文字型界面的小型且易於使用的編輯器。如需詳細資訊，請參閱 GNU nano 文件。	`nano --version`
OpenJDK 21	Amazon Corretto 21 是 OpenJDK 21 的長期支援 (LTS) 分佈。Amazon Corretto 是 Open Java Development Kit (OpenJDK) 的免費、多平台的生產就緒分佈。如需詳細資訊，請參閱 Corretto 21 使用者指南中的什麼是 Amazon Corretto 21？。	`java -version`

名稱	描述	版本資訊
procps	procps 是一種系統管理公用程式，可用來監控和停止目前執行中的程序。如需詳細資訊，請參閱列出可使用 procps 執行之程式的 README 檔案。	`ps --version`
psql	PostgreSQL 是一種功能強大的開放原始碼資料庫系統，使用標準 SQL 功能，同時提供強大的功能，以安全地管理和擴展複雜的資料操作。如需詳細資訊，請參閱什麼是 PostgreSQL。	`psql --version`
SSH 用戶端	SSH 用戶端使用安全 Shell 通訊協定與遠端電腦進行加密通訊。OpenSSH 是預先安裝的 SSH 用戶端。如需詳細資訊，請參閱 OpenBSD 維護的 OpenSSH 網站。 OpenBSD	`ssh -V`
sudo	使用 sudo 公用程式，使用者可以執行具有其他使用者安全許可的程式，通常是超級使用者。當您需要以系統管理員身分安裝應用程式時，Sudo 很有用。如需詳細資訊，請參閱 Sudo 手冊。	`sudo --version`
tar	tar 是一種命令列公用程式，可用來在單一封存檔案中將多個檔案分組 （通常稱為 tarball)。如需詳細資訊，請參閱 GNU tar 文件。	`tar --version`

名稱	描述	版本資訊
tmux	tmux 是一種終端機多工器，可用於在多個視窗中同時執行不同的程式。如需詳細資訊，請參閱提供 tmux 簡要簡介的部落格。	`tmux -V`
vim	vim 是一種可自訂的編輯器，您可以透過文字型界面與之互動。如需詳細資訊，請參閱 https：//vim.org 上提供的文件資源。	`vim --version`
wget	wget 是一種電腦程式，用於從命令列中端點指定的 Web 伺服器擷取內容。如需詳細資訊，請參閱 GNU Wget 文件。	`wget --version`

zip/unzip	zip/unzip 公用程式使用封存檔案格式，可提供無失真的資料壓縮，而不會遺失資料。呼叫 zip 命令，在單一封存中分組和壓縮檔案。使用 unzip 將檔案從封存擷取到指定的目錄。	`unzip --version`
		
		`zip --version`

安裝 AWS CLI 到您的主目錄

如同您 CloudShell 環境中預先安裝的其餘軟體， AWS CLI 工具會自動更新為排程升級和安全性修補程式。如果您想要確保您擁有up-to-date的 AWS CLI，您可以選擇在 shell 的主目錄中手動安裝工具。

> **Important**
>
> 您需要 AWS CLI 在主目錄中手動安裝 的複本，以便下次啟動 CloudShell 工作階段時可以使用。需要此安裝，因為新增至 外部目錄的檔案`$HOME`會在您完成 shell 工作階段後刪除。此外，在您安裝此副本之後 AWS CLI，不會自動更新。換句話說，管理更新和安全性修補程式是您的責任。

如需 AWS 共同責任模型的詳細資訊，請參閱 中的資料保護 AWS CloudShell。

安裝 AWS CLI

1. 在 CloudShell 命令列中，使用 curl命令將 AWS CLI 已安裝 的壓縮副本傳輸至 shell：

```
curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
```

2. 解壓縮壓縮的資料夾：

```
unzip awscliv2.zip
```

3. 若要將工具新增至指定的資料夾，請執行 AWS CLI 安裝程式：

```
sudo ./aws/install --install-dir /home/cloudshell-user/usr/local/aws-cli --bin-
dir /home/cloudshell-user/usr/local/bin
```

   如果安裝成功，命令列會顯示下列訊息：

```
You can now run: /home/cloudshell-user/usr/local/bin/aws --version
```

4. 為了方便起見，我們建議您也更新PATH環境變數，以便在執行aws命令時不需要指定工具安裝的路徑：

```
export PATH=/home/cloudshell-user/usr/local/bin:$PATH
```

> **Note**
>
> 如果您復原此變更至 PATH，則不具有指定路徑的aws命令 AWS CLI 預設會使用預先安裝的 版本。

在 shell 環境上安裝第三方軟體

> **Note**
>
> 建議您先檢閱共同安全責任模型，再將任何第三方應用程式安裝至 AWS CloudShell的運算環境。

根據預設， AWS CloudShell 所有使用者都有 sudo 許可。因此，您可以使用 sudo命令來安裝 shell 運算環境中尚未可用的軟體。例如，您可以使用 sudo搭配 DNF 套件管理公用程式來安裝 cowsay，這會產生具有訊息之牛的 ASCII 藝術圖片：

```
sudo dnf install cowsay
```

然後，您可以輸入 來啟動新安裝的程式echo "Welcome to AWS CloudShell" | cowsay。

> **Important**
>
> 套件管理公用程式，例如目錄 （ 例如，） 中的 dnf 安裝程式/usr/bin，這些公用程式會在 shell 工作階段結束時回收。這表示會根據工作階段安裝和使用其他軟體。

使用指令碼修改您的 shell

如果您想要修改預設 shell 環境，您可以編輯 shell 環境每次啟動時都會執行的 shell 指令碼。.bashrc 指令碼會在預設 bash shell 啟動時執行。

> **Warning**
>
> 如果您錯誤地修改.bashrc檔案，之後可能無法存取您的 shell 環境。最好先複製檔案再編輯。您也可以在編輯 時開啟兩個 Shell 來降低風險.bashrc。如果您在一個 shell 中失去存取權，您仍然會登入另一個 shell，並且可以轉返任何變更。
>
> 如果您在錯誤修改.bashrc或任何其他檔案後失去存取權，您可以透過刪除主目錄 AWS CloudShell 來返回其預設設定。

在此程序中，您將修改.bashrc指令碼，讓您的 shell 環境自動切換到執行 Z shell。

1. .bashrc 使用文字編輯器 （ 例如 Vim) 開啟 ：

```
vim .bashrc
```

2. 在編輯器界面中，按下 I 鍵開始編輯，然後新增下列項目：

```
zsh
```

3. 若要結束並儲存編輯.bashrc的檔案，請按 Esc 進入 Vim 命令模式，然後輸入下列內容：

```
:wq
```

4. 使用 source命令重新載入.bashrc檔案：

```
source .bashrc
```

當命令列界面再次可用時，提示符號 已變更為 % ，表示您正在使用 Z shell。

AWS CloudShell 從 AL2 遷移至 AL2023

AWS CloudShell以 Amazon Linux 2 (AL2) 為基礎的 已遷移至 Amazon Linux 2023 (AL2023)。如需 AL2023 的詳細資訊，請參閱《Amazon Linux 2023 使用者指南》中的什麼是 Amazon Linux 2023 (AL2023)。

透過 AL2023，您可以使用 CloudShell 提供的所有工具繼續存取現有的 CloudShell 環境。如需可用工具的詳細資訊，請參閱預先安裝的軟體。

AL2023 為開發工具提供了幾項改進，包括較新版本的套件，例如 Node.js 18 和 Python 3.9。

> **Note**
>
> 在 AL2023 Python 中，2 不再與您的 CloudShell 環境一起運送。

如需 AL2 和 AL2023 之間主要差異的詳細資訊，請參閱《Amazon Linux 2023 使用者指南》中的比較 Amazon Linux 2 和 Amazon Linux 2023。

如果您有任何問題，請聯絡 支援。您也可以在 中搜尋答案和張貼問題AWS re:Post。當您進入 時 AWS re:Post，您可能需要登入 AWS。

AWS CloudShell 遷移FAQs

以下是從 AL2 遷移到 AL2023 的一些常見問題的解答 AWS CloudShell。

遷移至 AL2023 是否會影響我的任何其他 AWS 資源，例如在 AL2 上執行的 Amazon EC2 執行個體？ AL2

您 AWS CloudShell 環境以外的服務或資源都不會受到此遷移的影響。這包括您可能已在 內建立或存取的資源 AWS CloudShell。例如，如果您已建立在 AL2 上執行的 Amazon EC2 執行個體，則不會遷移至 AL2023。

隨著遷移至 AL2023 而變更了哪些套件？

AWS CloudShell 環境目前包含預先安裝的軟體。若要了解預先安裝軟體的完整清單，請參閱預先安裝的軟體。 AWS CloudShell 將繼續交付這些套件，但 Python 2 除外。如需 AL2 和 AL2023 所提供套件的完整差異，請參閱比較 AL2 和 AL2023。對於具有遷移至 AL2023 後不再符合特定套件和版本需求的客戶，我們建議您聯絡 AWS Support 提交請求。

我可以選擇退出遷移嗎？

否，您無法選擇退出 migration。 AWS CloudShell 環境由 管理 AWS，因此，所有環境都已升級至 AL2023。

我可以建立 AWS CloudShell 環境的備份嗎？

AWS CloudShell 會繼續保留使用者主目錄。如需詳細資訊，請參閱 的服務配額和限制 AWS CloudShell。如果您的主資料夾中儲存了任何檔案或組態，而且您想要為相同的檔案或組態建立備份，請完成步驟 6：建立主目錄備份。

故障診斷 AWS CloudShell

使用 時 AWS CloudShell，您可能會遇到問題，例如當您啟動 CloudShell 或使用 shell 命令列界面執行關鍵任務時。本章涵蓋的資訊涵蓋了如何針對您可能遇到的一些常見問題進行故障診斷。

如需 CloudShell 各種問題的答案，請參閱AWS CloudShell FAQs。您也可以在AWS CloudShell 開發論壇中搜尋答案和張貼問題。當您進入此論壇時，您可能需要登入 AWS。您也可以直接聯絡我們。

故障診斷錯誤

當您遇到下列任何索引錯誤時，您可以使用下列解決方案來解決這些錯誤。

主題

拒絕的存取

問題：當您嘗試從 啟動 CloudShell 時 AWS Management Console，會收到「無法啟動環境」訊息。若要重試，請重新整理瀏覽器或選取動作以重新啟動 AWS CloudShell「。即使您擁有來自 IAM 管理員的必要許可，而且已重新整理瀏覽器或重新啟動 CloudShell，您也會被拒絕存取。

解決方案：請聯絡 AWS Support。

(回到頁首)

許可不足

問題：當您嘗試從 啟動 CloudShell 時 AWS Management Console，會收到「無法啟動環境」訊息。您沒有必要的許可。請您的 IAM 管理員授予 的存取權 AWS CloudShell。您被拒絕存取，並通知您沒有必要的許可。

原因：您用來存取的 IAM 身分 AWS CloudShell 缺少必要的 IAM 許可。

解決方案：請求您的 IAM 管理員為您提供必要的許可。他們可以透過新增連接的 AWS 受管政策 (AWSCloudShellFullAccess) 或內嵌內嵌內嵌政策來執行此操作。如需詳細資訊，請參閱使用 IAM 政策管理 AWS CloudShell 存取和用量。

(回到頁首)

無法存取 AWS CloudShell 命令列

問題：修改運算環境使用的檔案後，您無法存取命令列 AWS CloudShell。

解決方案：如果您在錯誤修改`.bashrc`或任何其他檔案後失去存取權，您可以透過刪除主目錄 AWS CloudShell 來返回其預設設定。

(回到頁首)

無法 ping 外部 IP 地址

問題：當您從命令列執行 ping 命令 （ 例如 `ping amazon.com`) 時，您會收到下列訊息。

```
ping: socket: Operation not permitted
```

原因：ping 公用程式使用網際網路控制訊息通訊協定 (ICMP) 將回應請求封包傳送至目標主機。它會等待回音從目標回覆。由於未在 中啟用 ICMP 通訊協定 AWS CloudShell，Ping 公用程式不會在 shell 的運算環境中操作。

解決方案：由於 不支援 ICMP AWS CloudShell，您可以執行下列命令來安裝 Netcat。Netcat 是一種電腦聯網公用程式，用於使用 TCP 或 UDP 讀取和寫入網路連線。

```
sudo yum install nc
```

```
nc -zv www.amazon.com 443
```

(回到頁首)

準備終端機時遇到一些問題

問題：嘗試 AWS CloudShell 使用 Microsoft Edge 瀏覽器存取 時，您無法啟動 shell 工作階段，瀏覽器會顯示錯誤訊息。

Cause：與舊版 Microsoft Edge AWS CloudShell 不相容。您可以使用支援瀏覽器 AWS CloudShell 的最新四個主要版本來存取 。

解決方案：從 Microsoft 網站安裝更新版本的 Edge 瀏覽器。

(回到頁首)

PowerShell 中的方向鍵無法正常運作

問題：在正常操作中，您可以使用方向鍵導覽命令列界面，並透過命令歷史記錄向後和向前掃描。但是，當您在 上按下特定 PowerShell 版本中的方向鍵時 AWS CloudShell，字母可能會輸出不正確。

原因：方向鍵輸出字母不正確的情況是 Linux 上執行 PowerShell 7.2.x 版本的已知問題。

解決方案：若要分割修改方向鍵行為的逸出序列，請編輯 PowerShell 設定檔並將`$PSStyle`變數設定為 `PlainText`。

1. 在 AWS CloudShell 命令列中，輸入下列命令以開啟設定檔檔案。

   ```
   vim ~/.config/powershell/Microsoft.PowerShell_profile.ps1
   ```

   > **Note**
   >
   > 如果您已在 PowerShell 中，您也可以使用下列命令在編輯器中開啟設定檔檔案。
   >
   > ```
   > vim $PROFILE
   > ```

2. 在編輯器中，前往檔案現有文字的結尾，按 i 進入插入模式，然後新增下列陳述式。

   ```
   $PSStyle.OutputRendering = 'PlainText'
   ```

3. 完成編輯後，按 Esc 進入命令模式。接著，輸入下列命令來儲存檔案並結束編輯器。

```
:wq
```

> **Note**
>
> 您的變更會在您下次啟動 PowerShell 時生效。

(回到頁首)

不支援的 Web Sockets 會導致無法啟動 CloudShell 工作階段

問題：當您嘗試啟動時 AWS CloudShell，會重複收到下列訊息：`Failed to open sessions : Timed out while opening the session`。

原因：CloudShell 取決於 WebSocket 通訊協定，允許 Web 瀏覽器與 之間的雙向互動式通訊 AWS CloudShell。如果您在私有網路中使用瀏覽器，代理伺服器和防火牆可能會協助安全存取網際網路。WebSocket 通訊通常可以周遊代理伺服器，而不會發生問題。但是，在某些情況下，代理伺服器會阻止 WebSockets正常運作。如果發生此問題，CloudShell 無法啟動 shell 工作階段，並且嘗試連線最終會逾時。

解決方案：連線逾時可能是由不支援的 WebSockets以外的問題所造成。如果是這種情況，請先重新整理 CloudShell 命令列界面所在的瀏覽器視窗。

如果您在重新整理後仍出現逾時錯誤，請參閱代理伺服器的文件。此外，請確定您的代理伺服器已設定為允許 Web Sockets。或者，請聯絡網路的系統管理員。

> **Note**
>
> 假設您想要透過允許列出特定 URLs 來定義精細許可。您可以新增 AWS Systems Manager 工作階段用來開啟 WebSocket 連線的部分 URL，以傳送輸入和接收輸出。您的 AWS CloudShell 命令會傳送至該 Systems Manager 工作階段。
> Systems Manager 所使用的此 StreamUrl 格式為 `wss://ssmmessages.region.amazonaws.com/v1/data-channel/session-id?stream=(input|output)`。
> 區域代表 AWS 區域 支援之 的區域識別符 AWS Systems Manager。例如， `us-east-2`是美國東部（俄亥俄）區域的區域識別符。

由於 session-id 是在特定 Systems Manager 工作階段成功啟動後建立，因此您只能在更新 URL 允許清單`wss://ssmmessages.region.amazonaws.com`時指定 。如需詳細資訊，請參閱 AWS Systems Manager API 參考中的 StartSession 操作。

(回到頁首)

無法匯入`AWSPowerShell.NetCore`模組

問題：當您透過 在 PowerShell 中匯入 AWSPowerShell.NetCore 模組時`Import-Module -Name AWSPowerShell.NetCore`，您會收到下列錯誤訊息：

Import-Module：指定的模組「AWSPowerShell.NetCore'未載入，因為在任何模組目錄中找不到有效的模組檔案。

原因：`AWSPowerShell.NetCore`模組由 中的 per-service. AWS Tools 模組取代 AWS CloudShell。

解決方案：可能不再需要任何明確的匯入陳述式，或需要變更為相關的每個服務 AWS。工具模組。

Example

Example

- 對於大多數情況下，只要不使用 .Net 類型，您就不需要任何明確的匯入陳述式。以下是匯入陳述式的範例。
  - `Get-S3Bucket`
  - `(Get-EC2Instance).Instances`

- 如果使用 .Net 類型，請匯入服務層級模組 (`AWS.Tools.<Service>`)。以下為範例語法。

```
Import-Module -Name AWS.Tools.EC2
$InstanceTag = [Amazon.EC2.Model.Tag]::new("Environment","Dev")
```

```
Import-Module -Name AWS.Tools.S3
$LifecycleRule = [Amazon.S3.Model.LifecycleRule]::new()
```

如需詳細資訊，請參閱 的第 4 版公告 AWS Tools for PowerShell。

(回到頁首)

使用 時，Docker 未執行 AWS CloudShell

問題：使用 時，Docker 未正確執行 AWS CloudShell。您會收到下列錯誤訊息：`docker: Cannot connect to the Docker daemon at unix:///var/run/docker.sock. Is the docker daemon running?`。

解決方案：嘗試重新啟動您的環境。當您 AWS CloudShell 在 GovCloud 區域中的 中執行 Docker 時，可能會發生此錯誤訊息。請確定您在支援的區域中執行 Docker AWS 。如需可使用 Docker 的區域清單，請參閱 支援的 AWS 區域 AWS CloudShell。

Docker 已用盡磁碟空間

問題：您收到下列錯誤訊息：`ERROR: failed to solve: failed to register layer: write [...]: no space left on device`。

原因：Dockerfile 超過可用的磁碟空間 AWS CloudShell。這可能是由於大型個別影像或太多預先存在的 Docker 影像所致。

解決方案：執行 `df -h` 以尋找磁碟用量。執行 `sudo du -sh /folder/folder1` 來評估您認為可能很大的特定資料夾大小，並考慮刪除其他檔案以釋放空間。其中一個選項是執行 來考慮移除未使用的 Docker 映像`docker rmi`。您應該知道 Docker 在環境中的空間有限，如需 Docker 的詳細資訊，請參閱 Docker 文件指南。

`docker push` 正在逾時並持續重試

問題：當您執行時`docker push`，它會逾時，並在沒有成功的情況下繼續重試。

原因：這可能是由於缺少許可、推送到錯誤的儲存庫或缺乏身分驗證所致。

解決方案：若要嘗試解決此問題，請確定您正在推送至正確的儲存庫。執行 `docker login`以正確驗證。請確定您擁有推送至 Amazon ECR 儲存庫的所有必要許可。

無法從我的 VPC 環境存取 AWS CloudShell VPC 內的資源

問題：使用我的 VPC 環境時，無法存取 AWS CloudShell VPC 內的資源。

原因：您的 AWS CloudShell VPC 環境會繼承 VPC 的網路設定。

解決方案：若要解決此問題，請確定您的 VPC 已正確設定以存取您的 資源。如需詳細資訊，請參閱 VPC 文件 將您的 VPC 連接到其他網路，以及 和 Network Access Analyzer 文件 Network Access

Analyzer。您可以在命令列提示字元或 AWS CloudShell VPC 主控台頁面中，於您的環境`ip -a`內執行命令，尋找 VPC 環境正在使用的 IPv4 地址。

用於我的 VPC 環境 AWS CloudShell 的 ENI 不會清除

問題：無法清除 AWS CloudShell 用於我的 VPC 環境的 ENI。

原因：您的角色未啟用 `ec2:DeleteNetworkInterface`許可。

解決方案：若要解決此問題，請確定您的角色已啟用`ec2:DeleteNetworkInterface`許可，如下列範例指令碼所示：

```
{
  "Effect": "Allow",
  "Action": [
    "ec2:DeleteNetworkInterface"
  ],
  "Condition": {
    "StringEquals": {
      "aws:ResourceTag/ManagedByCloudShell": ""
    }
  },
  "Resource": "arn:aws:ec2:*:*:network-interface/*"
}
```

具有僅 VPC 環境**`CreateEnvironment`**許可的使用者也可以存取公有 AWS CloudShell 環境

問題：只有 VPC 環境的`CreateEnvironment`許可限制使用者也可以存取公有 AWS CloudShell 環境。

原因：當您限制僅建立 VPC 環境的`CreateEnvironment`許可，且如果您已建立公有環境，您將保留對現有公有 CloudShell 環境的存取權，直到使用 Web 使用者介面刪除此環境為止。但是，如果您從未使用過 CloudShell，您將無法存取公有環境。

解決方案：若要限制對公 AWS CloudShell 有環境的存取，IAM 管理員必須先使用 限制更新 IAM 政策，然後使用者必須使用 AWS CloudShell Web 使用者介面手動刪除現有的公有環境。(動作 → 刪除 CloudShell 環境)。

支援的 AWS 區域 AWS CloudShell

本節涵蓋支援的 AWS 區域和選擇加入 區域的清單 AWS CloudShell。如需 CloudShell AWS 的服務端點和配額清單，請參閱 中的 AWS CloudShell 頁面Amazon Web Services 一般參考。

以下是 CloudShell、Docker 和 CloudShell VPC 環境支援的 AWS 區域：

- 美國東部 (俄亥俄)
- 美國東部 (維吉尼亞北部)
- 美國西部 (加利佛尼亞北部)
- 美國西部 (奧勒岡)
- 非洲 (開普敦)
- Asia Pacific (Hong Kong)
- 亞太區域 (雅加達)
- 亞太區域 (孟買)
- 亞太區域 (大阪)
- 亞太區域 (首爾)
- 亞太區域 (新加坡)
- 亞太區域 (雪梨)
- 亞太區域 (東京)
- 加拿大 (中部)
- 歐洲 (法蘭克福)
- 歐洲 (愛爾蘭)
- 歐洲 (倫敦)
- 歐洲 (米蘭)
- 歐洲 (巴黎)
- 歐洲 (斯德哥爾摩)
- Middle East (Bahrain)
- 中東 (阿拉伯聯合大公國)
- 南美洲 (聖保羅)

GovCloud 區域

以下是 CloudShell 支援的 GovCloud 區域：

- AWS GovCloud (US-East)
- AWS GovCloud (US-West)

> **Note**
>
> Docker 和 CloudShell VPC 環境可在 GovCloud 區域使用。

的服務配額和限制 AWS CloudShell

此頁面說明適用於下列區域的服務配額和限制：

持久性儲存

使用 AWS CloudShell時，您擁有 1 GB 的持久性儲存體，每個儲存體 AWS 區域 皆免費。持久性儲存體位於您的主目錄 ($HOME) 中，為您私有。與每個 shell 工作階段結束後回收的暫時性環境資源不同，主目錄中的資料會保留在工作階段之間。

> **Note**
>
> CloudShell VPC 環境沒有持久性儲存。當您的 VPC 環境逾時 （ 閒置 20-30 分鐘後 ） 或刪除環境時，$HOME 目錄會被刪除。

如果您在 AWS CloudShell 中停止使用 AWS 區域，資料會保留在該區域的持久性儲存體中，直到您的最後一個工作階段結束後 120 天。120 天後，除非您採取動作，否則您的資料會自動從該區域的持久性儲存中刪除。您可以再次在 AWS CloudShell 中啟動，以防止移除 AWS 區域。如需詳細資訊，請參閱步驟 2：選取區域 AWS CloudShell、啟動和選擇 shell。

> **Note**
>
> 使用案例
>
> Márcia 已使用 AWS CloudShell 將檔案存放在她的主目錄中的兩個位置 AWS 區域：美國東部 （ 維吉尼亞北部 ） 和歐洲 （ 愛爾蘭 ）。然後，她開始在歐洲 （ 愛爾蘭 ） AWS CloudShell 專門使用 ，並停止在美國東部 （ 維吉尼亞北部 ） 啟動 shell 工作階段。

在美國東部（維吉尼亞北部）刪除資料的截止日期之前，Márcia 決定再次啟動 AWS CloudShell 和選取美國東部（維吉尼亞北部）區域，以防止她的主目錄遭到回收。由於她持續使用歐洲（愛爾蘭）進行 Shell 工作階段，因此她在該區域的持久性儲存不會受到影響。

每月使用量

您 中的每個 AWS 區域 AWS 帳戶 都有每月用量配額 AWS CloudShell。此配額會合併該區域中所有 IAM 主體使用 CloudShell 所花費的總時間。如果您在達到該區域的每月配額後嘗試存取 CloudShell，則會顯示一則訊息，說明為何無法啟動 shell 環境。

使用 Service Quotas 主控台請求增加

您可以開啟 Service Quotas 主控台，請求提高每月用量配額。如需詳細資訊，請參閱「Service Quotas 使用者指南」中的請求提高配額。

並行 shell

AWS 區域 您最多可以為您的帳戶在每個 中同時執行 10 個 shell。

使用 Service Quotas 主控台請求提高

您可以開啟 Service Quotas 主控台，請求提高每個區域的配額。如需詳細資訊，請參閱「Service Quotas 使用者指南」中的請求提高配額。

命令大小

命令大小不能超過 65412 個字元。

Note

如果您想要執行超過 65412 個字元的命令，請使用您選擇的語言建立指令碼，然後從命令列界面執行。如需可從命令列界面存取之預先安裝軟體範圍的詳細資訊，請參閱預先安裝的軟體。若要查看 做為如何建立指令碼，然後從命令列界面執行指令碼的範例，請參閱教學課程：開始使用 AWS CloudShell。

Shell 工作階段

- 非作用中工作階段： AWS CloudShell 是一種互動式 shell 環境，如果您在 20-30 分鐘內未使用鍵盤或指標與其互動，則 shell 工作階段會結束。執行中的程序不會計入互動。

  如果您想要使用具有更靈活逾時的 AWS 服務執行終端型任務，建議您啟動並連線至 Amazon EC2 執行個體。
- 長時間執行的工作階段：即使使用者在此期間定期與其互動，持續執行約 12 小時的 shell 工作階段也會自動結束。

VPC 環境

每個 IAM 主體最多只能建立兩個 VPC 環境。

> **Note**
>
> 連線至您的私有 VPC 並存取其中的資源是免費的。私有 VPC 內的資料傳輸會包含在 VPC 帳單中，而透過 CloudShell 的 VPCs 之間的資料傳輸費用，會與您目前的 CloudShell 相同。

網路存取和資料傳輸

下列限制適用於您 AWS CloudShell 環境的傳入和傳出流量：

- 傳出：您可以存取公有網際網路。
- 傳入：您無法存取傳入連接埠。沒有可用的公有 IP 地址。

> **Warning**
>
> 透過存取公有網際網路，某些使用者可能會從 AWS CloudShell 環境匯出資料。我們建議 IAM 管理員透過 IAM 工具管理信任 AWS CloudShell 使用者的允許清單。如需如何明確拒絕特定使用者存取的資訊，請參閱 AWS CloudShell 使用自訂政策在 中管理允許的動作。

資料傳輸：大型檔案的上傳和下載檔案往返速度 AWS CloudShell 可能很慢。或者，您可以使用 shell 的命令列界面，將檔案從 Amazon S3 儲存貯體傳輸到您的環境。

系統檔案和頁面重新載入的限制

- 系統檔案：如果您不正確地修改運算環境所需的檔案，則存取或使用 AWS CloudShell 環境時可能會遇到問題。如果發生這種情況，您可能需要刪除主目錄才能重新取得存取權。
- 重新載入頁面：若要重新載入 AWS CloudShell 界面，請使用瀏覽器中的重新整理按鈕，而不是作業系統的預設快速鍵序列。

AWS CloudShell 使用者指南的文件歷史記錄

最近更新

下表說明 AWS CloudShell 使用者指南的重要變更。

變更	描述	日期
中的 Amazon Q CLI AWS CloudShell	已新增對在 中使用 Amazon Q CLI 功能的支援 AWS CloudShell。	2024 年 10 月 2 日
Amazon VPC AWS CloudShell 支援特定區域中的	新增支援在特定區域中建立和使用 AWS CloudShell VPC 環境。	2024 年 6 月 13 日
新的教學課程已新增至 AWS CloudShell 使用者指南	已新增兩個新的教學課程，詳細說明如何在內部建置 Docker 容器 AWS CloudShell 並將其推送至 Amazon ECR 儲存庫，以及如何透過 部署 Lambda 函數 AWS CDK。	2023 年 12 月 27 日
在特定區域中支援 AWS CloudShell 的 Docker 容器	在特定區域中 AWS CloudShell 已新增對 Docker 容器的支援。	2023 年 12 月 27 日
AWS CloudShell 已遷移至 現在使用 Amazon Linux 2023 (AL2023)	AWS CloudShell 現在使用 AL2023 且已從 Amazon Linux 2 遷移。	2023 年 12 月 4 日
的新 AWS 區域 AWS CloudShell	AWS CloudShell 現已在下列 AWS 區域正式推出： • 美國西部 (加利佛尼亞北部) • 非洲 (開普敦) • Asia Pacific (Hong Kong) • 亞太區域 (大阪) • 亞太區域 (首爾)	2023 年 6 月 16 日

	• 亞太區域 (雅加達) • 亞太區域 (新加坡) • Europe (Paris) • 歐洲 (斯德哥爾摩) • 歐洲 (米蘭) • Middle East (Bahrain) • 中東 (阿拉伯聯合大公國)	
在 AWS CloudShell 上啟動 Console Toolbar	選擇 CloudShellConsole Toolbar，在主控台左下角的 上啟動 CloudShell。	2023 年 3 月 28 日
的新 AWS 區域 AWS CloudShell	AWS CloudShell 現已在下列 AWS 區域提供： • 加拿大 (中部) • 歐洲 (倫敦) • 南美洲 (聖保羅)	2022 年 10 月 6 日
AWS CloudShell 美國 AWS GovCloud 支援	AWS CloudShell AWS GovCloud (US) 區域現在支援 。	2022 年 6 月 29 日
安全性FAQs	專注於安全問題的其他FAQs。	2022 年 4 月 14 日
Web 通訊端	已將 章節新增至說明 CloudShell 使用 WebSocket 通訊協定的網路需求。	2022 年 3 月 21 日
故障診斷 PowerShell 中的方向鍵	依照步驟修正按下時錯誤輸出字母的方向鍵。	2022 年 2 月 7 日
Tab 鍵自動完成	說明如何使用 bash-completion 的新文件，其允許透過按下 Tab 鍵自動完成部分輸入的命令或引數。	2021 年 9 月 24 日

指定 AWS 區域	指定 AWS 區域 AWS CLI 命令預設值的文件。	2021 年 5 月 11 日
PDF 和 Kindle 版本的格式	已修正資料表儲存格中的影像大小和文字。	2021 年 3 月 10 日
選定 AWS 區域中 的一般可用性 AWS CloudShell (GA) 版本	AWS CloudShell 現已在下列 AWS 區域正式推出： • 美國東部 (俄亥俄) • 美國東部 (維吉尼亞北部) • 美國西部 (奧勒岡) • 亞太區域 (東京) • 歐洲 (愛爾蘭) • 亞太區域 (孟買) • 亞太區域 (悉尼) • 歐洲 (法蘭克福)	2020 年 12 月 15 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。