使用者指南

AWS CloudShell

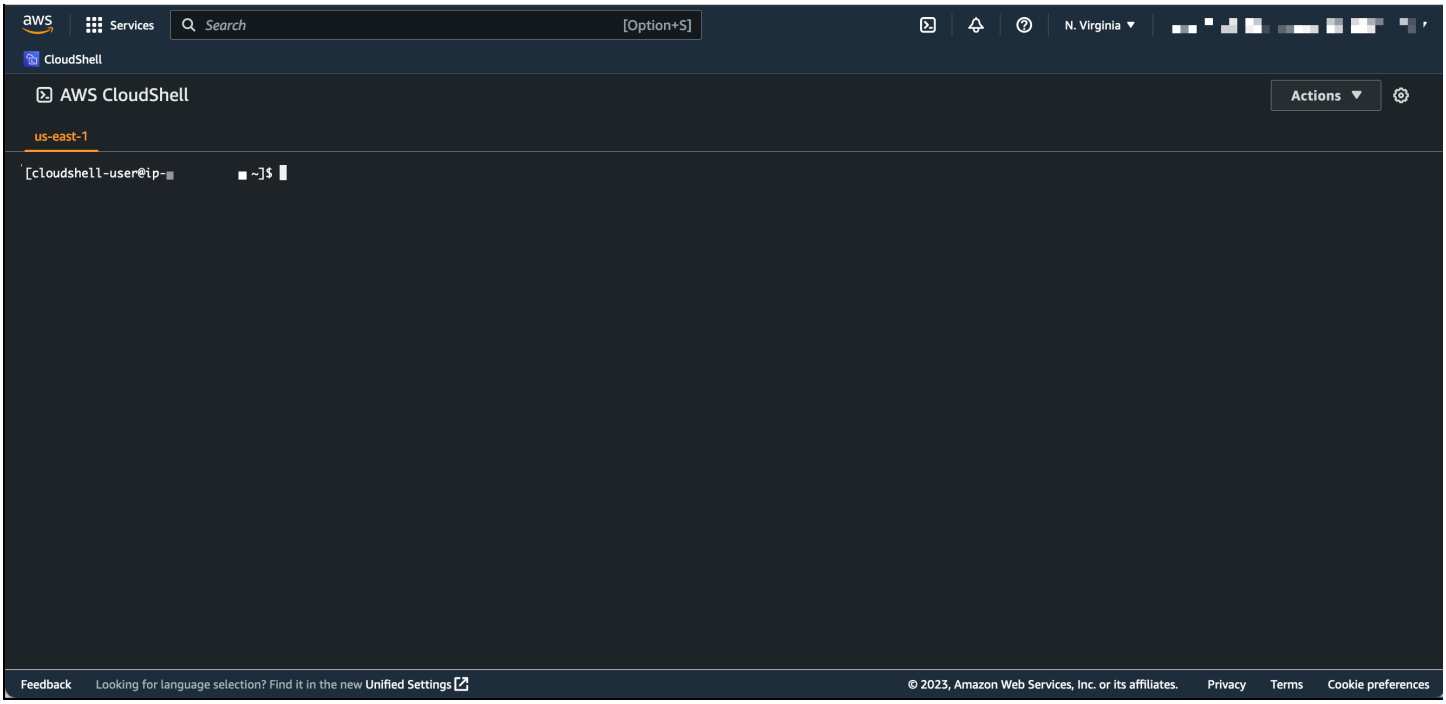
AWS CloudShell: 使用者指南

Table of Contents

什麼是 AWS CloudShell？

AWS CloudShell 是一個以瀏覽器為基礎的預先驗證殼層，您可以直接從. AWS Management Console 您可以 CloudShell 從幾種不同 AWS Management Console 的方式導覽至。如需的詳細資訊，請參閱如何開始使 AWS CloudShell？。

您可以使用偏好的 AWS CLI shell 執行命令Bash，例如 PowerShell、或Z shell。您無需下載或安裝命令行工具即可執行此操作。



當您啟動時 AWS CloudShell，就會建立以 Amazon Linux 2023 為基礎的運算環境。在此環境中，您可以存取廣泛的預先安裝開發工具、上傳和下載檔案的選項，以及在工作階段之間持續存在的檔案儲存空間。

（立即嘗試：AWS CloudShell 入門）

AWS CloudShell features

本主題說明如何 CloudShell 從主控台啟動、在偏好的命令列 shell 之間無縫切換， CloudShell 以及根據您的確切偏好進行自訂。此外，每個儲存空間最多可以使用 1 GB 的持續性儲存空間 AWS 區域，以及特定安全性功能如何保護 CloudShell 環境。

AWS Command Line Interface

您可以 AWS CloudShell 從啟動 AWS Management Console. 您用來登入主控台的 AWS 認證會自動在新的 Shell 工作階段中使用。由於 AWS 服務 使用 AWS CloudShell 者已預先驗證，因此您在使用第 2 AWS CLI 版進行互動時不需要設定認證。 AWS CLI 已預先安裝在殼層的運算環境中。

若要取得有關 AWS 服務 使用指令行介面進行互動的更多資訊，請參閱〈〉使用中的AWS服務AWS CloudShell。

外殼和開發工具

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。更具體地說，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gitmakepip、sudo、、tar、tmux、vim、wget、和zip。

殼層環境已預先設定，支援多種主要的主要軟體語言，例如Node.js和Python。這意味著，例如，您可以在不首先執行運Node.js行時安裝的情況下運行和Python項目。 PowerShell 使用者可以使用執.NET Core行階段。

在將這些檔案推送 AWS CloudShell 至由管理的遠端儲存庫之前，您可以認可在中建立或上傳至本機儲存庫的檔案 AWS CodeCommit。

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

有了 AWS CloudShell，您可以在每個儲存空間中使用高達 1 GB 的持續性儲存空 AWS 區域 間，而無需額外費用。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環境資源不同，主目錄中的資料會在工作階段之間持續存在。

如需持續性儲存裝置中保留資料的詳細資訊，請參閱持久性儲存。

安全

AWS CloudShell 環境及其使用者受到特定安全性功能的保護。這包括 IAM 許可管理、殼層工作階段限制以及用於文字輸入的安全貼上功能。

使用 IAM 進行許可管理

身為管理員，您可以使用 IAM 政策向使用 AWS CloudShell 者授與和拒絕許可。您也可以建立原則，以指定使用者可以在 shell 環境中執行的特定動作。如需詳細資訊，請參閱 使用 IAM 政策管理 AWS CloudShell 存取和使用。

殼層工作階段管

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

用於文本輸入的安全粘貼

預設為啟用「安全貼上」。這項安全性功能會要求您確認要貼到 shell 中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

您可以根據自己的喜好自定義您的 AWS CloudShell 體驗。例如，您可以變更螢幕版面配置 (多個索引標籤)、顯示的文字大小，以及在淺色與深色介面主題之間切換。如需詳細資訊，請參閱 自訂您的AWS CloudShell經驗。

您也可以安裝自己的軟體並修改啟動殼層指令碼來擴充 shell 環境。

工作階段還

工作階段還原功能會還原您在 CloudShell 終端機中跨單一或多個瀏覽器索引標籤執行的工作階段。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，直到因為非使用中工作階段而停止殼層 若要繼續使用 CloudShell 工作階段，請在終端機視窗中按任意鍵。如需命令介面工作階段的詳細資訊，請參閱命令介面

工作階段還原還原還原每個終端機索引標籤中的最新終端輸出和執行程序。

> **Note**
>
> 行動應用程式中無法使用工作階段還原。

定價 AWS CloudShell

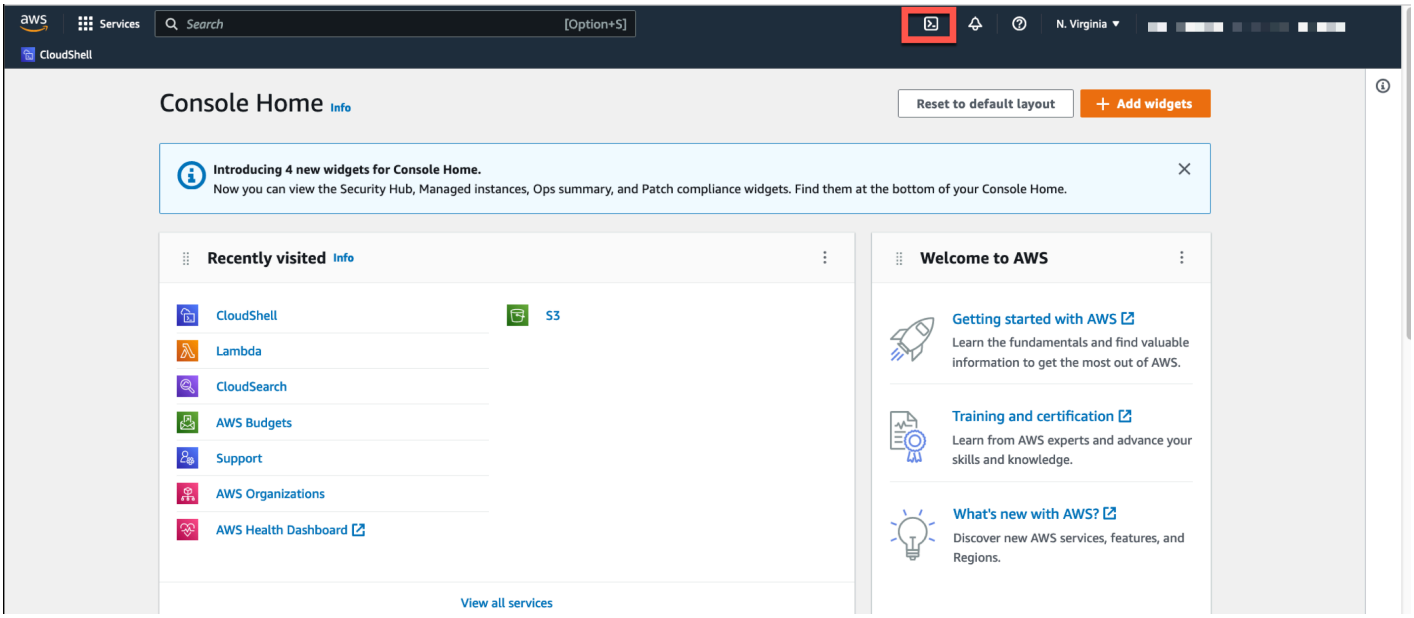
AWS CloudShell 是一個 AWS 服務 可用的，不收取額外費用。不過，您需要為執行的其他 AWS 資源付費 AWS CloudShell。此外，標準數據傳輸速率也適用。如需詳細資訊，請參閱 AWS CloudShell 定價。

如需詳細資訊，請參閱 的服務配額和限制AWS CloudShell。

如何開始使用 AWS CloudShell？

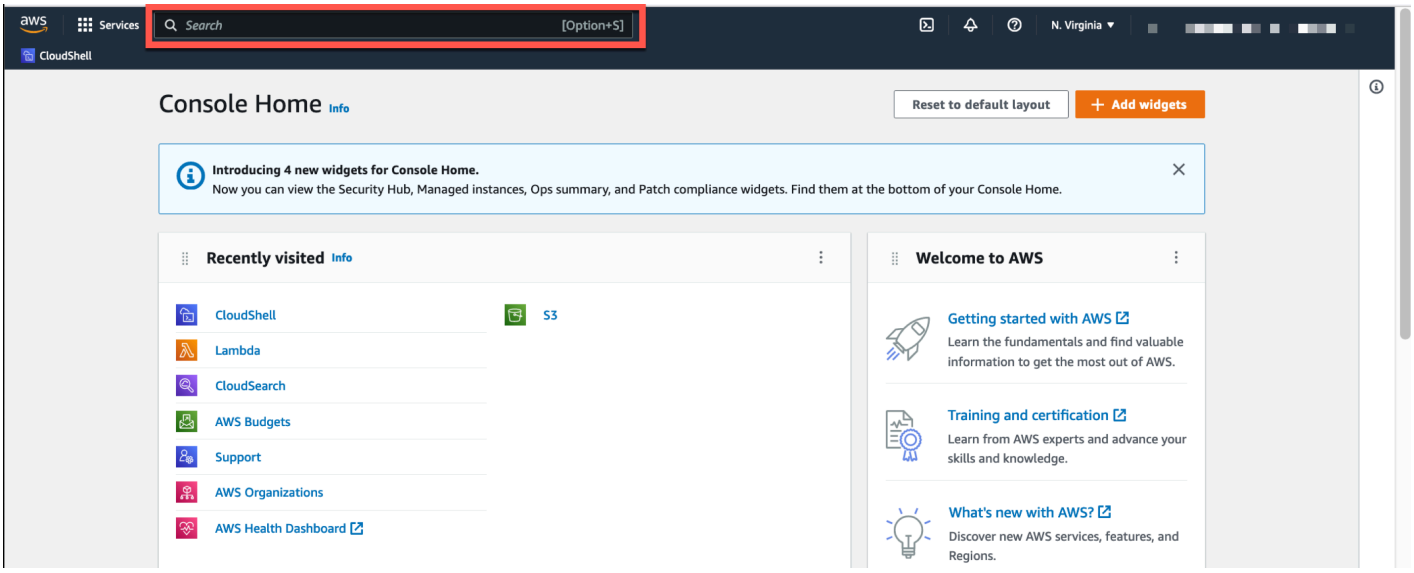
若要開始使用殼層，請登入 AWS Management Console 並選擇下列其中一個選項：

- 在導覽列上，選擇CloudShell圖示。



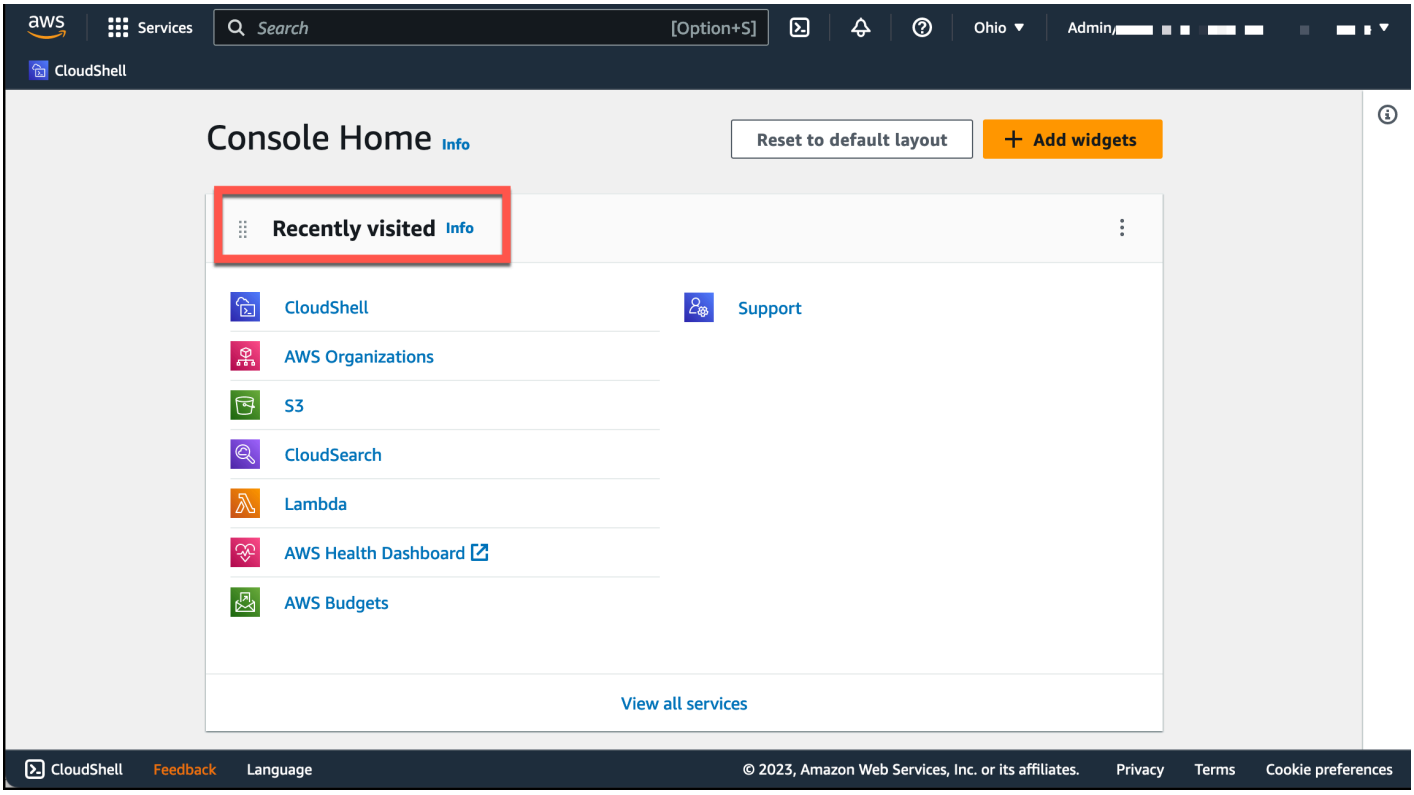
- 在 [搜尋] 方塊中，輸入「CloudShell」，然後選擇CloudShell。

  此步驟會以全螢幕開啟您的 CloudShell 工作階段。

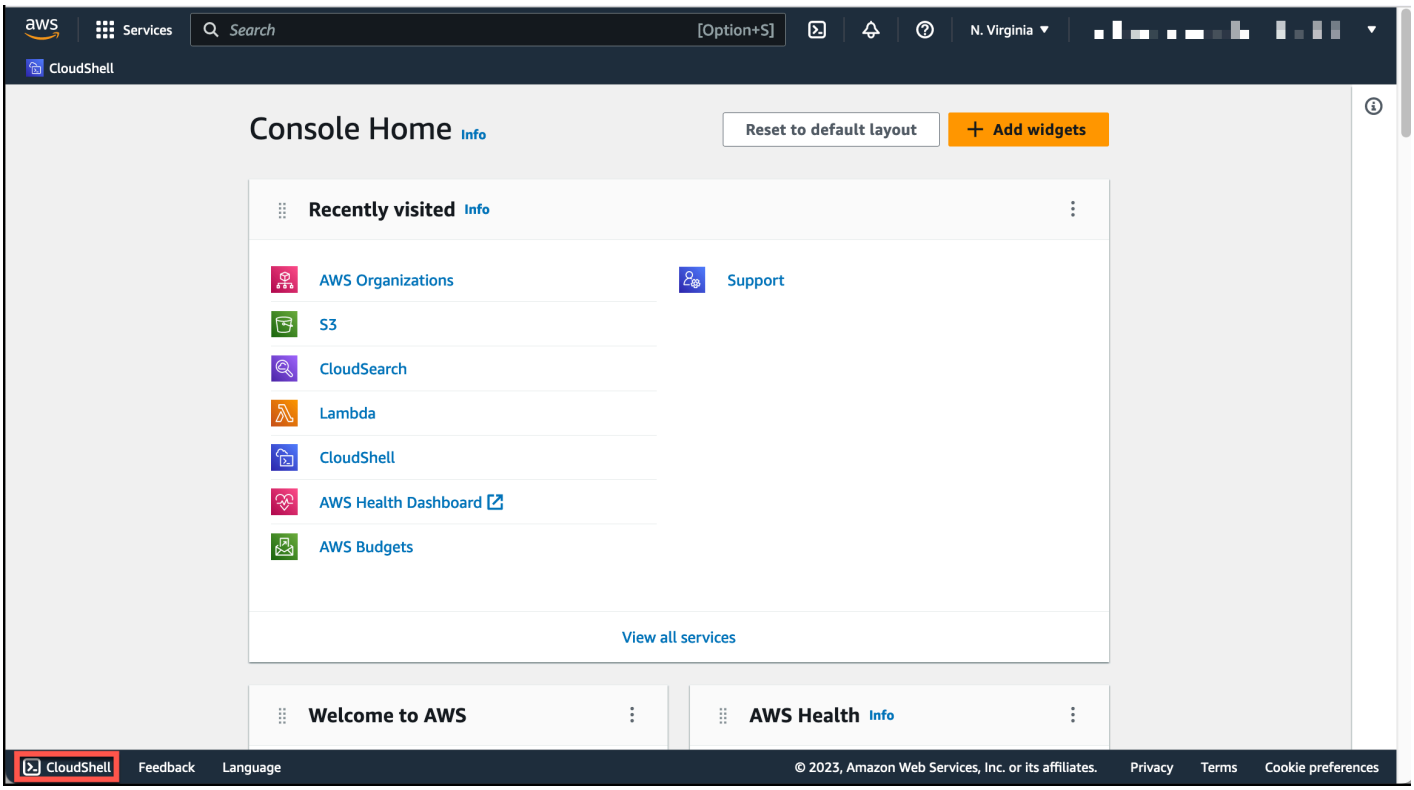


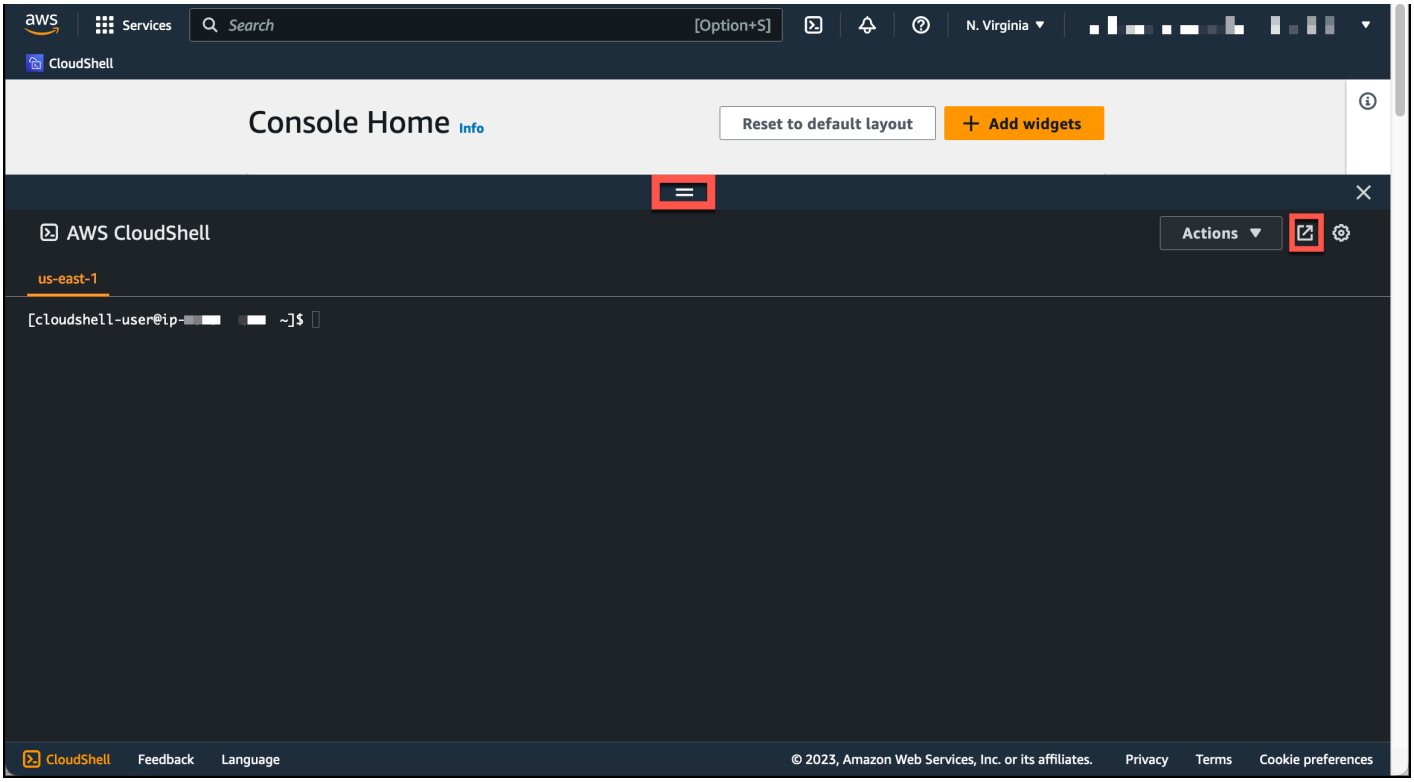
- 在最近訪問的小部件中，選擇CloudShell。

  此步驟會以全螢幕開啟您的 CloudShell 工作階段。



- 選擇CloudShell主機左下方的 (位於)。Console Toolbar您可以透過拖曳來調整 CloudShell 工作階段的高度=。





您也可以按一下 [在新瀏覽器標籤中開啟]，將 CloudShell 工作階段切換為全螢幕。

如需有關如何使用登入 AWS Management Console 和執行重要工作的指示 AWS CloudShell，請參閱入門 AWS CloudShell。

關鍵 AWS CloudShell 主題

- AWS CloudShell 入門
- 使用 AWS CloudShell
- 使用中的AWS服務AWS CloudShell
- 自訂您的AWS CloudShell經驗
- AWS CloudShell運算環境：規格和軟體

AWS CloudShell 常見問

以下是一些常見問題的答案 AWS CloudShell。

如需更多關注安全性的常見問題集，請參閱 AWS CloudShell 安全問題

如何開始使用 AWS CloudShell？

您可以透過幾個步驟來開始啟動 AWS CloudShell AWS Management Console。若要這麼做，請使用您的 AWS 帳戶 或 IAM 登入資料登入主控台，網址為 https://console.aws.amazon.com/console/home。

如需詳細資訊，請參閱 AWS CloudShell入門。

我需要存取哪些內容 AWS CloudShell？

因為您可以 AWS CloudShell 從存取 AWS Management Console，因此您必須是可以提供有效帳戶別名或 ID、使用者名稱和密碼的 IAM 使用者。

若要在主控台 AWS CloudShell 上啟動，您需要附加政策提供的 IAM 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。

AWS CloudShell 上面有什麼Console Toolbar？

位於 CloudShell 左下方的圖示 AWS Management Console。

如何 AWS CloudShell 在上啟動Console Toolbar？

您可以選擇主機左下角的CloudShell圖示Console Toolbar來在上啟動 AWS CloudShell 。

哪些 AWS 區域 是 AWS CloudShell 可用的？

如需受支援 AWS 區域 和相關聯服務端點的清單，請參閱中的AWS CloudShell 頁面Amazon Web Services 一般參考。

當您在? CloudShell 上啟動時，如 AWS CloudShell 果選取的區域無法使用，則會指派哪 AWS 區域 一個Console Toolbar？

預設「區域」會指派給最接近所選區域的「區域」。如需詳細資訊，請參閱選取區域 AWS CloudShell、啟動並選擇殼層。

您可以執行提供權限的命令，以管理與預設「地區」不同的「區域」中的資源。如需詳細資訊，請參閱在中使用 AWS 區域。

我可以在哪些類型的殼中使用 AWS CloudShell？

在中 AWS CloudShell，您可以使用Bash shell PowerShell、或執行指令Z shell。若要切換 shell，請在命令提示字元中使用下列格式輸入您要使用的 shell 名稱：

- `bash`：使用 Bash shell
- `pwsh`：使用 PowerShell
- `zsh`：使用 Z shell

我可以使用哪些網頁瀏覽器 AWS CloudShell？

AWS CloudShell 支持谷歌瀏覽器的最新版本, 火狐瀏覽器, Microsoft 邊緣, 和蘋果 Safari 瀏覽器.

如何建立和管理我的 AWS CloudShell 環境？

您的 AWS CloudShell 環境是根據每個區域的 IAM 使用者 ID 建立和管理。您可以`UserId`通過運行來檢查`aws sts get-caller-identity`。環境由該特定區域中的 IAM 使用者 ID 擁有。如果您變更 IAM `UserId` 或區域，您將能夠存取不同的 AWS CloudShell 環境。

當我啟動 AWS CloudShell 時，我可以使用哪些網頁瀏覽器Console Toolbar？

您可以 CloudShell 啟動Console Toolbar使用谷歌瀏覽器的最新版本, Microsoft 邊緣, 火狐瀏覽器, 和蘋果 Safari 瀏覽器.

我可以 AWS CloudShell 在啟動時下載檔案Console Toolbar嗎？

是的，您可以 CloudShell 在啟動時下載檔案Console Toolbar。您可以使用最新版本的谷歌瀏覽器和 Microsoft 邊緣瀏覽器下載文件。

目前，您無法使用火狐瀏覽器和蘋果 Safari 瀏覽器下載文件。

我的殼層環境中預先安裝了哪些軟體？

使用針對 AWS CloudShell 工作階段建立的 shell，您可以在偏好的命令列 shell (Bash PowerShell、和Z shell) 之間無縫切換。您也可以存取預先安裝的工具和公用程式 Make pipsudo，例如 tar、tmux、Vim、、Wget和Zip。

shell 環境已預先配置，支持大多數主要軟件語言。例如，您可以使用它來執行Node.js和Python專案，而不必先執行執行階段安裝。 PowerShell 使用者可以使用執.NET Core行階段。

您可以將使用命令介面建立的檔案或使用 shell 介面上傳的檔案，新增至使用的預先安裝版本管理的版本控制存放庫。git

如需詳細資訊，請參閱 預裝軟體。

我可以安裝在 shell 環境中不可用的軟件嗎？

是的， AWS CloudShell 使用者擁有sudo權限，可以從命令列安裝軟體。如需詳細資訊，請參閱 在 shell 環境中安裝第三方軟件。

我可以限制使用者可以在其中執行的動作 AWS CloudShell嗎？

是的，您可以控制使用者可以在中執行的動作 AWS CloudShell。例如，您可以允許使用者存取，AWS CloudShell 但禁止他們在 shell 環境中上傳或下載檔案。或者，您也可以完全防止使用者存取 AWS CloudShell。如需詳細資訊，請參閱 使用 IAM 政策管理 AWS CloudShell 存取和使用。

如果我想更改我正 AWS 區域 在使用的位置，如何從我的主目錄移動數據 AWS CloudShell？

若要將資 AWS CloudShell 料從一個區域移 AWS 區域 至另一個區域，請先將一個區域中的主目錄內容下載到您的本機電腦，然後從該目錄上傳到另一個區域的主目錄。如需詳細資訊，請參閱 教學課程：在本機電腦和AWS CloudShell。

我可以增加由於用戶不活動而確定何 AWS CloudShell 時超時的限制嗎？

如果您不 AWS CloudShell 使用鍵盤或指標進行互動，則 shell 工作階段會在大約 20-30 分鐘後自動結束。正在運行的進程不算作交互。因為 CloudShell 是專為重點、以工作為基礎的活動所設計，因此目前沒有增加此逾時限制的計劃。

如果您想要使用 AWS 服務 具有更靈活的逾時來執行終端型任務，建議您使用雲端 IDE AWS Cloud9，或啟動並連接至 Amazon EC2 執行個體。

我可以 AWS Console Mobile Application 從主屏幕訪問 AWS CloudShell嗎？

是，您可以 AWS CloudShell 登入 AWS Console Mobile Application Console Mobile Application 來存取。如需詳細資訊，請參閱 AWS Console Mobile Application 使用者指南。

我該如何 AWS CloudShell 啟動 AWS Console Mobile Application？

您可以 AWS CloudShell 使用下列其中一種方法啟動：

1. 選取導覽列底部的AWS CloudShell圖示。
2. 選取 [服務] 功能表AWS CloudShell上的。

AWS CloudShell 在中使用時，我可以在 iOS 和 Android 鍵盤上使用輔助按鍵 AWS Console Mobile Application嗎？

可以，您可以在 iOS 和 Android 鍵盤上使用輔助按鍵。如需詳細資訊，請參閱 AWS 主控台行動應用程式使用指南。

我可以將分 AWS CloudShell 頁顯示分頁分割成多個分頁 AWS Console Mobile Application嗎？

不可以，您目前無法在行動應用程式上執行多個 AWS CloudShell 分頁。

我可以 AWS CloudShell 在移動設備Console Toolbar上訪問嗎？

否，您目前無法 AWS CloudShell 在行動裝置Console Toolbar上存取。

AWS CloudShell 入門

此入門教學課程說明如何使用 shell 命令列介面啟動AWS CloudShell和執行重要工作。

首先，您登入AWS Management Console並選取AWS 區域。然後，您可以 CloudShell 在新的瀏覽器視窗和要使用的 shell 類型中啟動。

接下來，您可以在主目錄中建立新資料夾，並從本機電腦上傳檔案至該資料夾。您可以使用預先安裝的編輯器處理該檔案，然後再從命令列以程式的形式執行該檔案。最後，您可以呼叫AWS CLI命令來建立 Amazon S3 儲存貯體，並將檔案做為物件新增至儲存貯體。

先決條件

IAM 許可

您可以透AWS CloudShell過將下列AWS受管政策附加到 IAM 身分 (例如使用者、角色或群組) 來取得許可：

- AWSCloudShellFullAccess：為用戶提供對AWS CloudShell及其功能的完全訪問權限。

在本教學課程中，您還可以與之互動AWS 服務。更具體地說，您可以透過建立 S3 儲存貯體並將物件新增至該儲存貯體來與 Amazon 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 簡單儲存服務使用者指南中的 Amazon S3 動作。

練習檔案

本練習還涉及上傳和編輯隨後從命令行界面作為程序運行的文件。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 IAM 使用者資訊以存取AWS Management Console. 如果您已經在主機中，請跳至步驟 2。

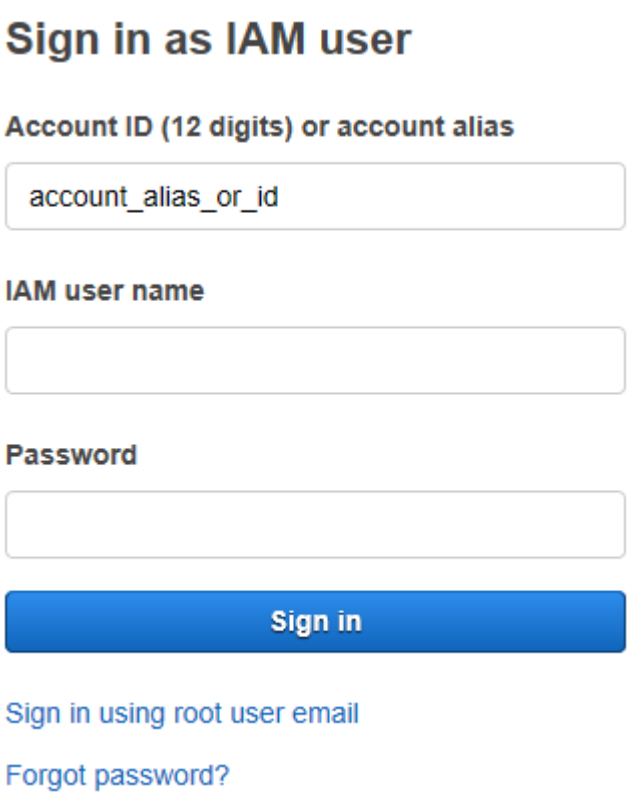
- 您可以使AWS Management Console用 IAM 使用者登入 URL 或前往主登入頁面來存取。

  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。

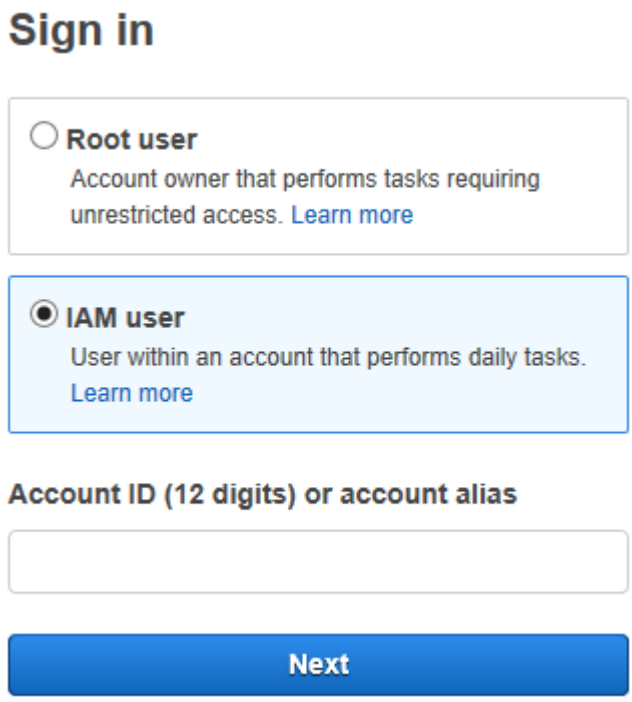
    ```
    https://account_alias_or_id.signin.aws.amazon.com/console/
    ```

  - 輸入您的 IAM 登入憑證，然後選擇 [登入]。

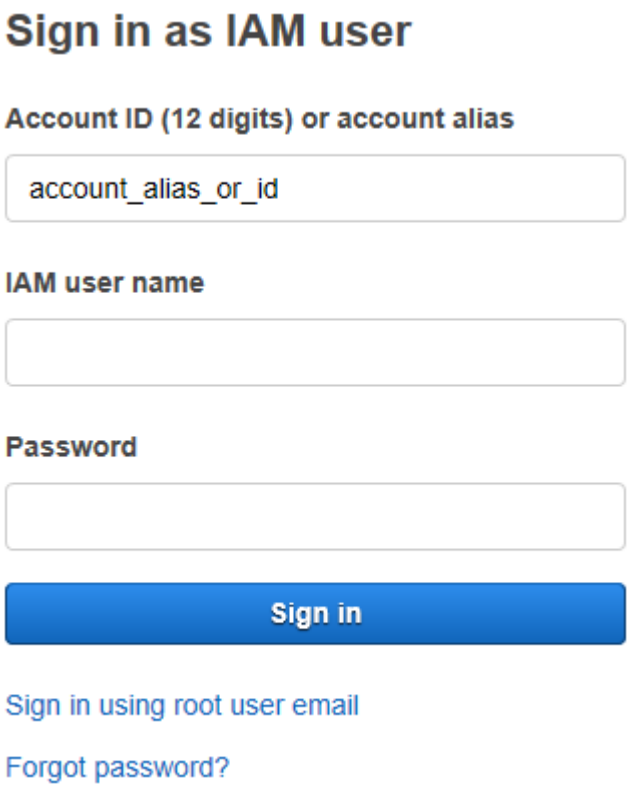


Main sign-in page

- 打開以下位置。https://aws.amazon.com/console/
- 如果您先前未使用此瀏覽器登入，則會顯示主要登入頁面。選擇 IAM 使用者，輸入帳戶別名或帳戶 ID，然後選擇下一步。



- 如果您之前已經以 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。如果是這樣，請輸入您的 IAM 登入憑證，然後選擇「登入」。



> **Note**
>
> 您也可以以 root 使用者身分登入。此身分對帳戶中的所有AWS 服務資源具有完整存取權。我們強烈建議您不要將 root 使用者用於日常工作，甚至是管理工作。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：選擇一個區域AWS CloudShell，啟動並選擇一個外圍程序

在此步驟中，您可以AWS CloudShell從主控台介面啟動，選擇可用的AWS 區域，然後切換至您偏好的ShellBash，例如 PowerShell、或Z shell。

1. 若要選擇AWS 區域要在其中工作，請移至「選取地區」功能表，然後選取支援的「AWS地區」以進行工作。(可用區域會反白顯示。)

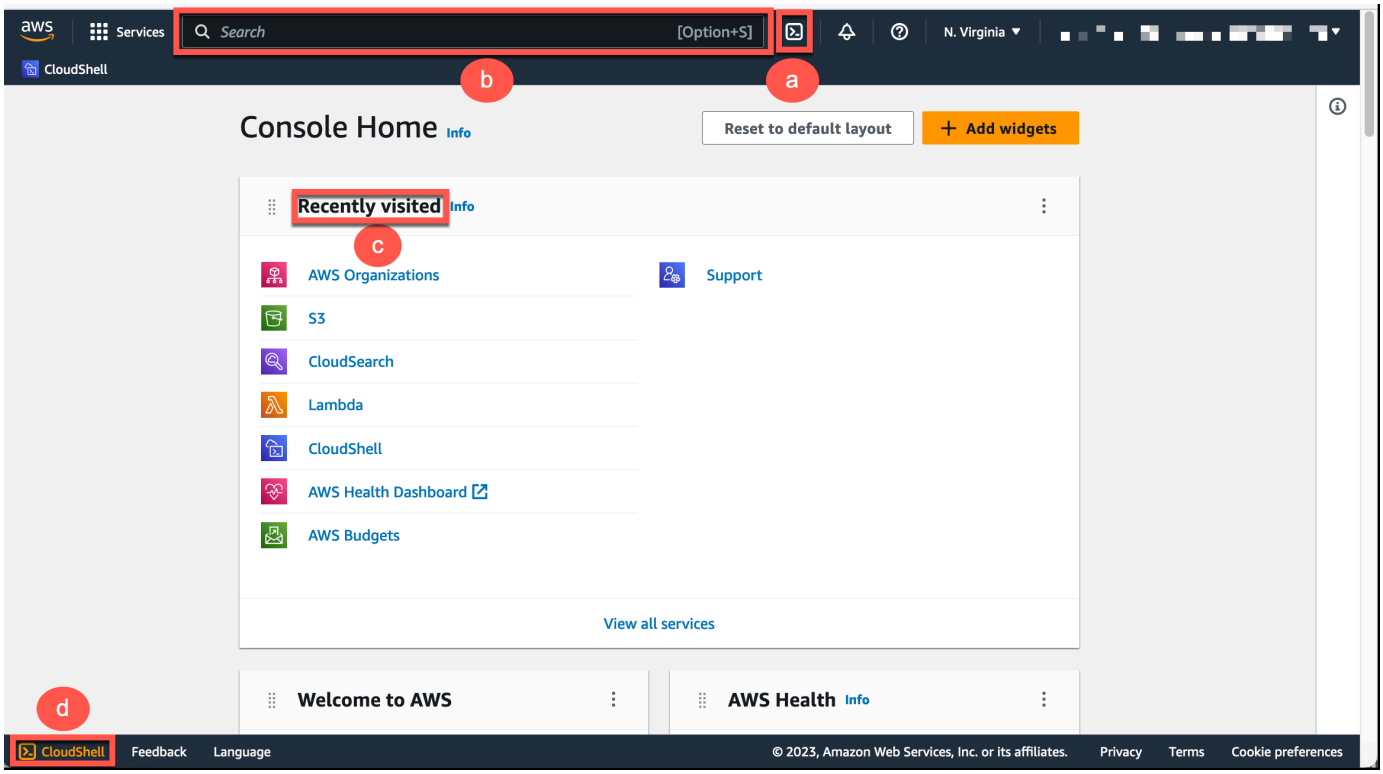
> **Important**
>
> 如果切換「區域」，則介面會重新整理，並且所選AWS 區域取的名稱會顯示在指令行文字上方。您添加到永久存儲中的任何文件僅在此相同中可用AWS 區域。如果您變更區域，則可存取不同的儲存空間和檔案。

> **Important**
>
> 如 CloudShell 果您在主機的左下角啟動 CloudShell 時Console Toolbar，選取的區域無法使用，則預設 [區域] 會設定為最接近所選區域的 [區域]。您可以執行提供權限的命令，以管理與預設「地區」不同的「區域」中的資源。如需詳細資訊，請參閱在中使用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) CloudShell 不提供eu-south-2，則預設「地區」會設定為最接近歐洲 (西班牙) eu-west-1 的歐洲 (愛爾蘭) eu-south-2。

您將使用預設「地區」、「歐洲」(愛爾蘭) 的服務配額eu-west-1，同一個 CloudShell 工作階段也會在所有區域中還原。預設區域可能會變更，您將在 CloudShell 瀏覽器視窗中收到通知。

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

   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. 在最近訪問的小部件中，選擇CloudShell。
   4. 選擇CloudShell主機左下方的 (位於)。Console Toolbar
      - 若要調整工 CloudShell 作階段的高度，請拖曳=。
      - 若要將 CloudShell 工作階段切換為全螢幕，請按一下「在新瀏覽器標籤中開啟」 圖示。



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之互動的問題AWS CloudShell，請檢查資訊以識別並解決中的這些問題AWS CloudShell 疑難排解。

3. 若要選擇要使用的預先安裝的 shell，請在指令行提示下輸入其程式名稱。

   Bash

   ```
   bash
   ```

   如果切換至Bash，則指令提示下的符號會更新為$。

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

   PowerShell

   ```
   pwsh
   ```

   如果切換至 PowerShell，則指令提示下的符號會更新為PS>。

   Z shell

   ```
   zsh
   ```

   如果切換至Z shell，則指令提示下的符號會更新為%。

   如需殼層環境中預先安裝的版本的相關資訊，請參閱 AWS CloudShell 運算環境一節中的 shell 表格。

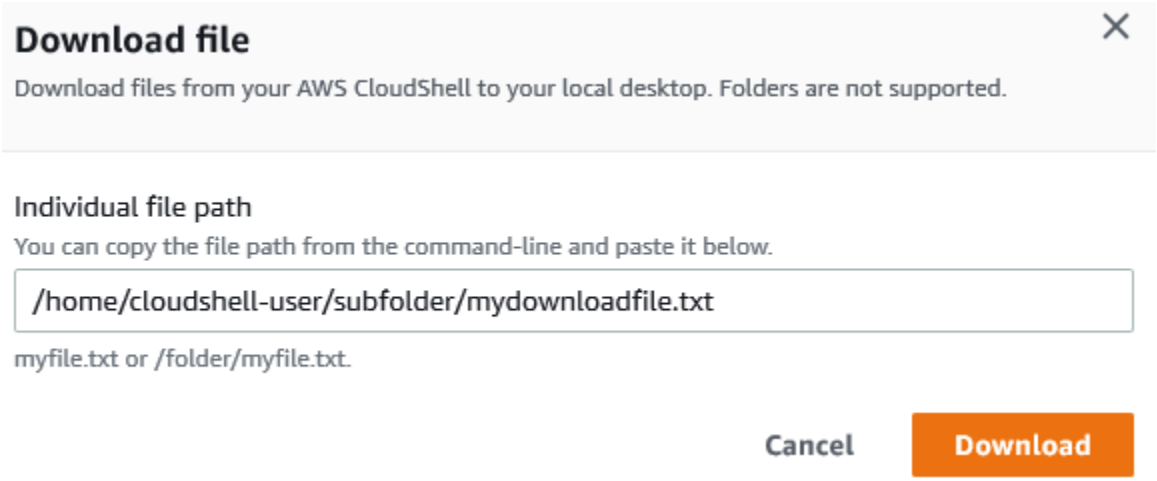
步驟 3：從下載文件 AWS CloudShell

此步驟會引導您完成下載檔案的過程。

1. 若要下載檔案，請移至「動作」，然後從選單中選擇「下載檔案」。

將顯示 [下載檔案] 對話方塊。

2. 在「下載檔案」對話方塊中，輸入要下載檔案的路徑。



> **Note**
>
> 指定要下載的檔案時，您可以使用絕對或相對路徑。對於相對路徑名稱，/home/cloudshell-user/默認情況下會自動添加到開頭。因此，要下載名為的文件mydownload-file，以下兩個都是有效路徑：
>
> - 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，將顯示一個對話方塊。您可以使用此對話方塊，以預設應用程式開啟檔案。或者，您可以將檔案儲存到本機電腦上的資料夾中。

> **Note**
>
> 當您 CloudShell 在上啟動時，無法使用 [下載] 選項Console Toolbar。您可以從 CloudShell 控制台或使用 Chrome 網絡瀏覽器下載文件。如需如何下載檔案的詳細資訊，請參閱步驟 3：從中下載檔案AWS CloudShell。

步驟 4：將檔案上傳至 AWS CloudShell

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，請在提示下輸入下列指令：

   pwd

   當您按下 Enter 鍵時，殼層會傳回您目前的工作目錄 (例如，/home/cloudshell-user)。
2. 要將文件上傳到此目錄，請轉到操作，然後從菜單中選擇上傳文件。

   將顯示 [上傳檔案] 對話方塊。
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，會顯示確認add_prog.py已新增至主目錄根目錄的訊息。
6. 若要建立檔案的目錄，請輸入「建立目錄」指令：mkdir mysub_dir。
7. 要將上傳的文件從主目錄的根目錄移動到新目錄，請使用以下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。

   命令提示字元會更新，指出您已變更工作目錄。
9. 若要檢視目前目錄的內容mysub_dir，請輸入ls指令。

   會列出工作目錄的內容。這包括您剛剛上傳的文件。

步驟 5：從中刪除文件 AWS CloudShell

此步驟說明如何從中移除檔案AWS CloudShell。

1. 若要從中移除檔案AWS CloudShell，請使用標準的 shell 指令，例如 rm (移除)。

   rm my-file-for-removal
2. 若要移除多個符合指定準則的檔案，請執行指find令。

下列範例會移除名稱中包含字尾「.pdf」的所有檔案。

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

> **Note**
>
> 假設你停止AWS CloudShell在一個特定的AWS 區域. 然後，該區域永久性儲存體中的資料會在指定時間後自動移除。如需詳細資訊，請參閱持續性儲存區。

步驟 6：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 創建備份文件

      若要使用 tar 建立備份檔案，請輸入下列指令：

      ```
      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 創建備份文件

      若要使用 zip 建立備份檔案，請輸入下列指令：

      ```
      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. 將備份文件傳輸到外部 CloudShell

   您可以使用下列其中一個選項將備份檔案傳輸到外部 CloudShell：

   a. 在本地計算機上下載備份文件

      您可以下載在上一步中創建的文件。如需如何從中下載檔案的詳細資訊 CloudShell，請參閱從下載檔案AWS CloudShell。

      在「下載檔案」對話方塊中，輸入要下載檔案的路徑 (例如/tmp/tmp.iA99tD9L98/home.tar.gz)。

   b. 將備份檔案傳輸到 S3

      若要產生值區，請輸入下列指令：

      ```
      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**
      >
      > 可能需要支付數據傳輸費用。

3. 直接 Backup 到 S3 儲存貯體

   若要直接備份到 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：重新啟動殼層工作階段

> **Note**
>
> 基於安全性考量，如果您長時間未使用鍵盤或指標與 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，請參閱殼層工作階。

1. 若要重新啟動 shell 工作階段，請選擇動作，重新啟動AWS CloudShell。

   系統會通知您重新啟動會AWS CloudShell停止目前的所有作用中工作階段AWS 區域。

2. 若要確認，請選擇重新啟動。

   介面會顯示一則訊息，指出 CloudShell 計算環境正在停止。環境停止並重新啟動之後，您可以在新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。

第 8 步：刪除一個 shell 會話主目錄

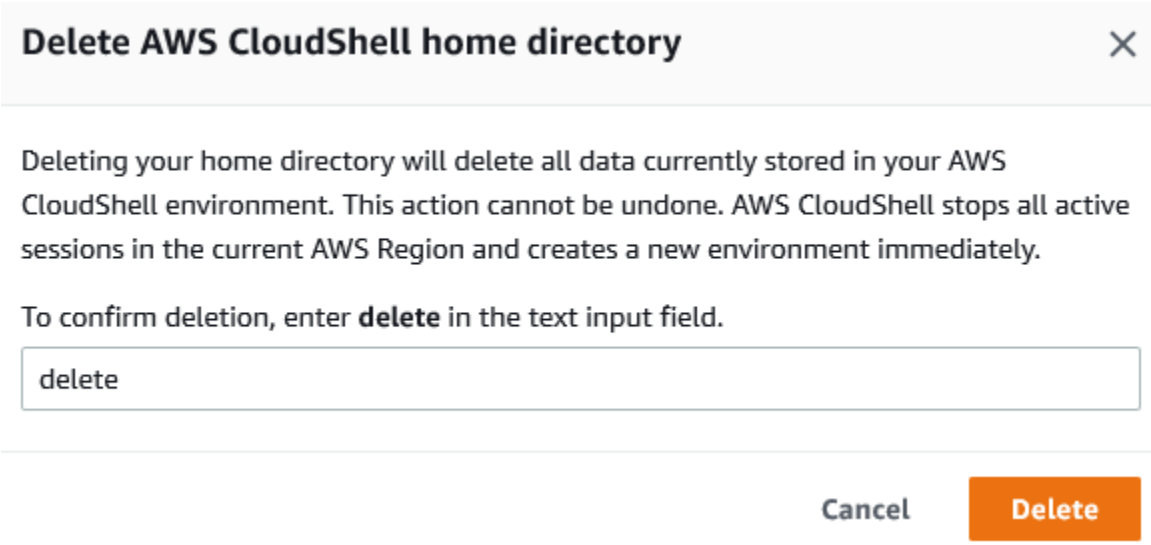
> **Warning**
>
> 刪除主目錄是不可復原的動作，其中儲存在主目錄中的所有資料都會永久刪除。但是，在以下情況下，您可能需要考慮此選項：
>
> - 您錯誤地修改了檔案，而且無法存取AWS CloudShell運算環境。刪除主目錄會返AWS CloudShell回其預設設定。
> - 您想AWS CloudShell立即從中刪除所有數據。如果您停止AWS CloudShell在某個AWS區域中使用，除非您在該地區AWS CloudShell再次啟動，否則會在保留期結束時自動刪除永久儲存裝置。
>
> 如果您的檔案需要長期儲存，請考慮 Amazon S3 或 CodeCommit.

1. 若要刪除殼層工作階段，請選擇動作，刪除 AWS CloudShell 主目錄。

   系統會通知您刪除AWS CloudShell主目錄會刪除目前儲存在您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 [刪除]。



   AWS 會 CloudShell 停止目前的所有作用中工作階段，AWS 區域並立即建立新環境。

   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`指令登出。然後，您可以按任意鍵重新連接並繼續使用AWS CloudShell。

步驟 9：編輯文件的代碼並使用命令行運行

此步驟示範如何使用預先安裝的Vim編輯器來處理檔案。然後，您可以從命令列以程式的形式執行該檔案。

1. 若要編輯您在上一個步驟中上傳的檔案，請輸入下列指令：

   ```
   vim add_prog.py
   ```

shell 介面會重新整理以顯示Vim編輯器。

2. 若要編輯中的檔案Vim，請按下I鍵。現在編輯內容，以便程序加起來三個數字，而不是兩個。

```
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**
>
> 如果您將文字貼到編輯器中並啟用「安全貼上」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「安全粘貼」功能，您可以在粘貼之前對其進行驗證。如果您對文字安全感到滿意，請選擇「貼上」。

3. 編輯程式後，按下Esc以進入指Vim令模式。然後，輸入`:wq`指令以儲存檔案並結束編輯器。

> **Note**
>
> 如果您是Vim命令模式的新手，最初可能會發現在命令模式和插入模式之間切換具有挑戰性。保存文件並退出應用程序時使用命令模式。插入新文本時使用插入模式。若要進入插入模式，請按I和，若要進入指令模式，請按Esc。如需中提供的其他工具Vim和其他可用工具的詳細資訊AWS CloudShell，請參閱開發工具和殼層公用程式。

4. 在主命令行界面上，運行以下程序並指定三個數字進行輸入。語法如下。

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

   指令行會顯示程式輸出：`The sum is 15`。

步驟 10：用AWS CLI來將檔案新增為 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，然後使用該PutObject方法將程式碼檔案新增為該儲存貯體中的物件。

> **Note**
>
> 在大多數情況下，您可以使用服務 (例如，將軟體檔案送出 CodeCommit至版本控制的儲存庫)。本教學課程說明如何使用AWS CLI中AWS CloudShell與其他 AWS 服務互動。使用此方法，您無需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. 若要在指定的值區中建立值區AWS 區域，請輸入下列指令：

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

> **Note**
>
> 如果您要在「區域」以外建立值us-east-1區，請create-bucket-configuration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，命令列會顯示類似下列輸出的服務回應。

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

> **Note**
>
> 如果您不遵守命名值區的規則，則會顯示下列錯誤：呼叫 CreateBucket作業時發生錯誤 (InvalidBucketName)：指定的值區無效。

2. 若要上傳檔案並將檔案新增為物件至您剛建立的值區，請呼叫方PutObject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是所儲存物件的雜湊值。您可以使用此雜湊來檢查上傳到 Amazon S3 之物件的完整性。

相關主題

- 使用中的AWS服務AWS CloudShell
- 教學課程：在本機電腦和AWS CloudShell
- 教學課程：使用 CodeCommit 於AWS CloudShell
- 使用 AWS CloudShell
- 自訂您的AWS CloudShell經驗

AWS CloudShell 教學課程

以下教程將使您能夠在使用時嘗試和測試不同的功能和集成AWS CloudShell。

主題

教學課程：在本機電腦和AWS CloudShell

使用此 CloudShell 介面，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。若要同時在本機電腦 CloudShell與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. 之間複製檔案時，請使用 S3 儲存貯體做為中介 CloudShell。
- Zip 文件：將多個文件壓縮在一個壓縮文件夾中，可以使用 CloudShell 界面上傳或下載。

> **Note**
>
> 由於 CloudShell 不允許傳入的網際網路流量，因此目前無法使用`scp`或`rsync`在本機電腦和 CloudShell 計算環境之間複製多個檔案之類的命令。

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，您需要授與許可的 IAM 政策，以執行下列 Amazon S3 API 動作：

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

如需 Amazon S3 動作的完整清單，請參閱 Amazon Storage Service API 參考資料中的動作。

AWS CloudShell使用 Amazon S3 將多個檔案上傳到

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值區。請選擇下列其中一個選項來上傳檔案：

   - AWS Management Console：用 drag-and-drop 於將檔案和資料夾上傳至值區。
   - AWS CLI：在本機電腦上安裝工具的版本後，使用命令列將檔案和資料夾上傳到值區。

   Using the console

   - 在以下網址開啟 Amazon S3 主控台：https://s3.console.aws.amazon.com/s3/。

     （如果您正在使用AWS CloudShell，則應該已經登錄到控制台。）
   - 在左側導覽窗格中，選擇儲存貯體，然後，選擇要上傳資料夾或檔案的目標儲存貯體名稱。您也可以選擇「建立值區」來建立自己選擇的值區。
   - 若要選擇您要上傳的檔案與資料夾，請選擇「上傳」。然後，將您的選取範圍拖放到列出目的地儲存貯體中物件的主控台視窗中，將您的選取範圍拖放到列出目的地儲存貯體中物件的主控台視窗中。

     您選擇的檔案會列在 Upload (上傳) 頁面上。
   - 選取勾選方塊以指示要加入的檔案。
   - 若要將選取的檔案新增至值區，請選擇 [上傳]。

> **Note**
>
> 如需使用主控台時的完整組態選項，請參閱我該如何上傳檔案與資料夾至 S3 儲存貯體至 S3 儲存貯體至 S3 儲存貯體？ 在 Amazon Storage Service 使用者指南中。

Using AWS CLI

> **Note**
>
> 對於此選項，您需要在本機電腦上安裝此AWS CLI工具，並設定您的認證以進行AWS服務呼叫。如需詳細資訊，請參閱 AWS Command Line Interface 使用者指南。

- 啟動AWS CLI工具並執行下列aws s3命令，以將指定值區與本機電腦上目前目錄的內容同步化：

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

如果同步成功，則會針對每個新增至值區的物件顯示上傳訊息。

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

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

> **Note**
>
> 您也可以在sync指令中加入--exclude "<value>"和--include "<value>"參數，以執行模式比對，以排除或包含特定檔案或物件。
> 若要取得更多資訊，請參閱〈在AWS CLI指令參考中使用排除篩選和包括篩選〉。

如果同步成功，則會針對從值區下載至目錄的每個檔案顯示下載訊息。

> **Note**
>
> 使用 sync 命令，只有新的和更新的檔案會從來源目錄遞迴複製到目的地。

AWS CloudShell使用 Amazon S3 下載多個文件

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

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

> **Note**
>
> 您也可以在sync指令中加入--exclude "<value>"和--include "<value>"參數，以執行模式比對，以排除或包含特定檔案或物件。
> 若要取得更多資訊，請參閱〈在AWS CLI指令參考中使用排除篩選和包括篩選〉。

   如果同步成功，則會針對每個新增至值區的物件顯示上傳訊息。

2. 將儲存貯體的內容下載到您的本機電腦。由於 Amazon S3 主控台不支援下載多個物件，因此您需要使用本機電腦上安裝的AWS CLI工具。

   從工AWS CLI具的命令列，執行以下命令：

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

   如果同步成功，命令列會針對目標目錄中更新或新增的每個檔案顯示下載訊息。

> **Note**
>
> 對於此選項，您需要在本機電腦上安裝此AWS CLI工具，並設定您的認證以進行AWS服務呼叫。如需詳細資訊，請參閱 AWS Command Line Interface 使用者指南。

使用壓縮文件夾上傳和下載多個文件

使用 zip/unzip 實用程序，您可以壓縮存檔中的多個文件，這些文件可以被視為單個文件。這些公用程式已預先安裝在 CloudShell 運算環境中。

如需預先安裝工具的詳細資訊，請參閱開發工具和殼層公用程式。

將多個文件上傳到AWS CloudShell使用壓縮文件夾

1. 在本機電腦上，將要上傳的檔案新增至壓縮資料夾。
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「上載檔案」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 歸檔的內容解壓縮至指定的目錄：

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

AWS CloudShell使用壓縮文件夾下載多個文件

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 (`/home/cloudshell-user/zip-folder/zipped-archive.zip`例如)，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到本機電腦的選項。
4. 在本地計算機上，您現在可以解壓縮下載的壓縮文件夾的內容。

教學課程：使用 CodeCommit 於AWS CloudShell

CodeCommit 是安全、可高度擴充和託管原始檔控制服務，可託管私有 Git 儲存庫。使用時AWS CloudShell，您可以使用公用程式 CodeCommit 在指令行上使git-remote-codecommit用。此公用程式已預先安裝在AWS CloudShell運算環境中，並提供從 CodeCommit 儲存庫推送和提取程式碼的簡單方法。此實用程序通過擴展 Git 來完成此操作。如需詳細資訊，請參閱 AWS CodeCommit 使用者指南。

本教學課程說明如何建立 CodeCommit 儲存庫，並將其複製到您的AWS CloudShell運算環境。您也會學習如何將檔案暫存並提交至複製的儲存庫，然後再將檔案推送至AWS雲端管理的遠端存放庫。

先決條件

如需 IAM 使用者需要使用的許可的詳細資訊AWS CloudShell，請參閱入門教學課程中的先決條件一節。您還需要 IAM 許可才能使用 CodeCommit。

此外，在開始之前，請確保具有以下內容：

- Git 命令和版本控制概念的基本理解
- 命令介面主目錄中的檔案，可提交至本機和遠端儲存庫。在本自學課程中，它被稱為`my-git-file`。

步驟 1：建立並複製 CodeCommit 儲存庫

1. 在 CloudShell 命令行界面中，輸入以下`codecommit`命令以創建名為的 CodeCommit 存儲庫`MyDemoRepo`。

```
aws codecommit create-repository --repository-name MyDemoRepo --repository-
description "My demonstration repository"
```

如果成功建立存放庫，命令列會顯示服務的回應。

```
{
    "repositoryMetadata": {
        "accountId": "111122223333",
        "repositoryId": "0dcd29a8-941a-1111-1111-11111111111a",
        "repositoryName": "MyDemoRepo",
        "repositoryDescription": "My demonstration repository",
        "lastModifiedDate": "2020-11-23T20:38:23.068000+00:00",
        "creationDate": "2020-11-23T20:38:23.068000+00:00",
        "cloneUrlHttp": "https://git-codecommit.eu-west-1.amazonaws.com/v1/repos/
MyDemoRepo",
        "cloneUrlSsh": "ssh://git-codecommit.eu-west-1.amazonaws.com/v1/repos/
MyDemoRepo",
        "Arn": "arn:aws:codecommit:eu-west-1:111111111111:MyDemoRepo"
     }
 )
```

2. 使用命令列為您的本機儲存庫建立新目錄，並將其設為您的工作目錄。

```
mkdir my-shell-repo
cd my-shell-repo
```

3. 若要複製遠端儲存庫，請使用`git clone`命令。（當您使用時 git-remote-codecommit，請使用 HTTPS（GRC）網址樣式）。

```
git clone codecommit::eu-west-1://MyDemoRepo
```

如果成功複製存放庫，命令列會顯示服務的回應。

```
Cloning into 'MyDemoRepo'...
warning: You appear to have cloned an empty repository.
```

4. 若要導覽至複製的儲存庫，請使用`cd`指令。

```
cd MyDemoRepo
```

第 2 步：在將文件推送到 CodeCommit 存儲庫之前對其進行分級並提交

1. 使用 Vim 編輯器或檔案上傳功能，將名為的檔案新增`my-git-file`至`MyDemoRepo`資料夾AWS CloudShell。若要了解如何使用這兩者，請參閱入門教學課程。

2. 若要儲存庫，請執行 gitadd 命令。

```
git add my-git-file
```

3. 要檢查文件是否已被暫存並準備好提交，請運行 gitstatus 命令。

```
git status
```

`my-git-file`會列示為新檔案，並以綠色文字顯示，表示已準備好可供認可。

4. 將此版本的暫存檔案提交至儲存庫。

```
git commit -m "first commit to repo"
```

> **Note**
>
> 如果系統要求您提供配置信息以完成提交，請使用以下格式。
>
> ```
> $ git config --global user.name "Jane Doe"
> $ git config --global user.email janedoe@example.com
> ```

5. 要將遠程存儲庫與本地存儲庫中所做的更改同步，請將更改推送到上游分支。

```
git push
```

教學課：使用 Amazon S3 物件建立 Amazon S3 物件預先簽章 URLAWS CloudShell

本教學說明如何建立立立立立立立立立立立立立立立立立立立立立立立立立立立立 Amazon S3 物件。由於物件擁有者在共用時會指定自己的安全性認證，因此任何接收預先簽署 URL 的人都可以在有限的時間內存取物件。

先決條件

- 具有AWSCloudShellFullAccess政策提供存取權限的 IAM 使用者。
- 有關建立預先簽署 URL 所需的 IAM 許可，請參閱 Amazon 簡單儲存服務使用者指南中的「與其他人共用物件」。

步驟 1：立立立立立立立立立立立立立立立立立立立立立立立立立

1. 若要取得可共用的 IAM 詳細資料，請從中呼叫`get-caller-identity`命令AWS CloudShell。

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

如果呼叫成功，指令行會顯示類似如下的回應。

```
{
    "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。此立立立立立立立立立立 IAM 角色。此角色會授與共用資源的共同作業人員最低權限。

```
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. 使用範本部署堆疊並呼叫`deploy`命令來建立 IAM 角色。

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

產生立預先簽章 URL

1. 在中使用您的編輯器AWS CloudShell，新增下列程式碼。此程式碼會建立 URL，提供同盟使用者直接存取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. 將代碼保存在名為的文件中share.py。
3. 從命令列執行以下 IAM 角色的 Amazon Resource Name (ARN)AWS CloudFormation。然後，在Python指令碼中使用它，取得暫時安全登入資料。

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

指令碼會傳回一個 URL，讓協同合作者按一下即可進AWS CloudShell入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。憑證會在一小時後到期。在這段時間之後，協作者便無法再存取值區。

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

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。

必要條件

- 您必須具備建立和推送至 Amazon ECR 儲存庫的必要許可。如需有關 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。

教程程序

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

1. 在您的主目錄中創建一個新文件夾。

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

2. 導覽至您建立的資料夾。

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

3. 創建一個空的碼頭文件。

```
touch Dockerfile
```

4. 例如nano Dockerfile，使用文字編輯器開啟檔案並將下列內容貼到檔案中。

```
# 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. 碼頭文件現在已準備就緒，可以構建。通過運行構建容器docker build。使用 easy-to-type 名稱標記容器，以便在以 future 的命令中使用。

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

請確定包含後置週期 (.)。

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

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

7. 現在，您有一個運作正常的 Docker 容器，您需要將其推送到 Amazon ECR 存儲庫。如果您有現有的 Amazon ECR 儲存庫，則可以略過此步驟。

執行下列命令以針對本教學建立 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 容器推送至該儲存庫。

   執行下列命令以取得泊塢視窗的 Amazon ECR 登入登入資料。

   ```
   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}
   ```

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

您現在已經成功地將您的碼頭容器部署到 Amazon ECR 儲存庫。若要從AWS CloudShell環境中移除您在本自學課程中建立的檔案，請執行以下指令。

- ```
  cd ~
  rm -rf ~/docker-cli-tutorial
  ```
- 刪除 Amazon ECR 儲存庫。

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

教學課程：使用 AWS CDK

本教學課程說明如何使用AWS Cloud Development Kit (AWS CDK).
  ```

必要條件

- 啟動您的帳戶以搭配使用AWS CDK. 如需使用啟動載入的相關資訊AWS CDK，請參閱 v2 開發人員指南中的AWS CDK啟動載入。如果您尚未啟動該帳戶，則可以在`cdk bootstrap`. CloudShell

- 確保您具有將資源部署到您的帳戶的適當權限。建議使用管理員權限。

教程程序

下列教學課程概述如何使用部署以 Docker 容器為基礎的 Lambda 函數。AWS CDK

1. 在您的主目錄中創建一個新文件夾。

```
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. 例如`nano cdk.json`，使用文字編輯器開啟檔案並將下列內容貼到檔案中。

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

6. 打開文`lib/docker-tutorial.js`件並將以下內容粘貼到其中。

```
// 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. 打開lib/Dockerfile並將以下內容粘貼到其中。

```
# 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. 打開文lib/hello.js件並將以下內容粘貼到其中。

```
// 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 來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 函數以確認並進行驗證。

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

您現在已經 Lambda 用. AWS CDK 如需詳細資訊AWS CDK，請參閱 AWS CDKv2 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

您現在已經 Lambda 用. AWS CDK 在AWS CDK專案內，執行下列命令以刪除關聯的資源。系統將提示您確認刪除。

-
```
npx cdk destroy DockerTutorialStack
```

- 若要從AWS CloudShell環境中移除您在本自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
```

使用 AWS CloudShell

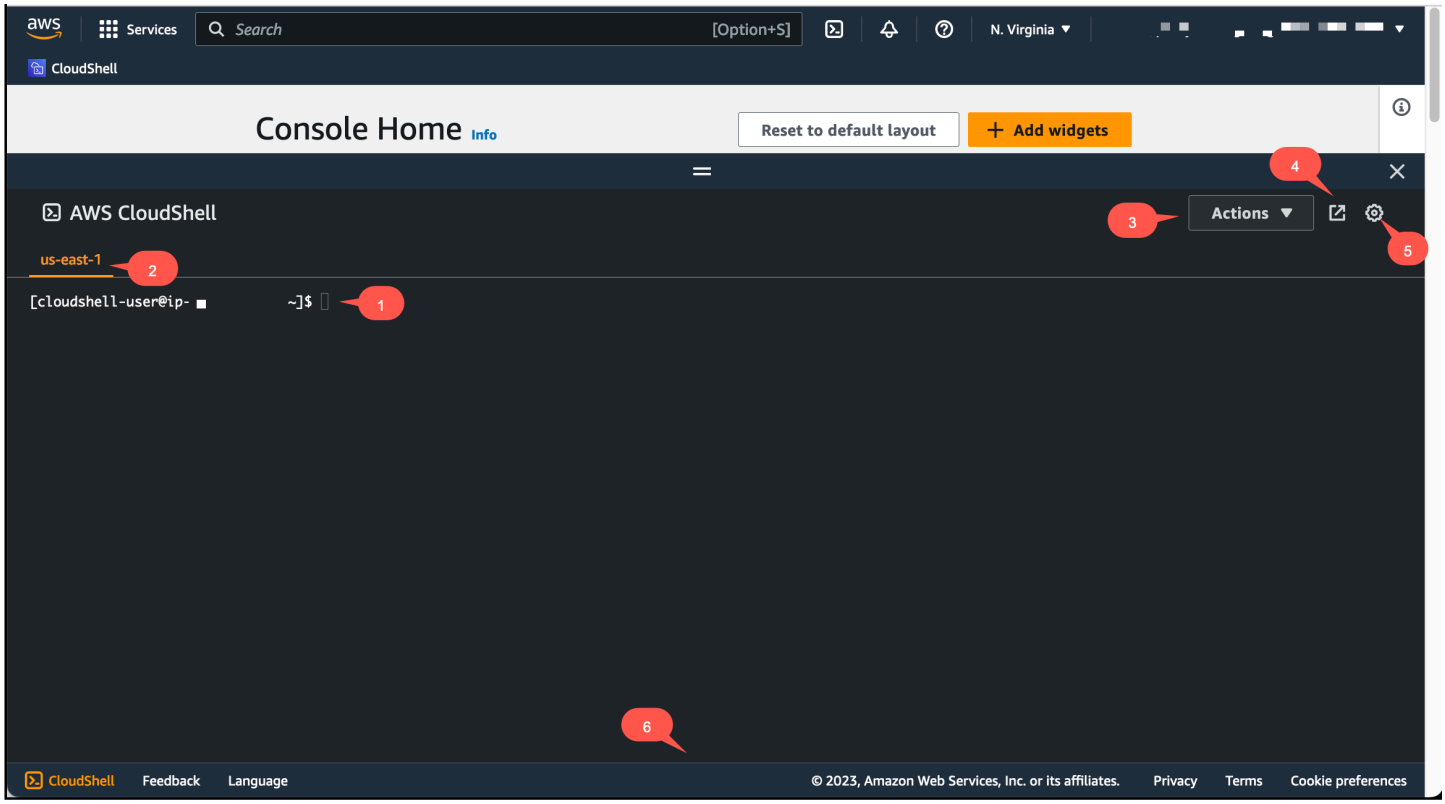
本節說明如何與支援的應用程式互動AWS CloudShell並執行特定動作。

主題

瀏覽介AWS CloudShell面

您可以從和導覽 CloudShell 介面功AWS Management Console能Console Toolbar。

下列螢幕擷取畫面指出幾個主要AWS CloudShell介面功能。



1. AWS CloudShell您可以使用偏好的 shell 來執行命令的命令列介面。目前的殼層類型由命令提示字元指示。

2. 終端機索引標籤，它使用目前AWS CloudShell正AWS 區域在執行的位置。
3. 「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以及刪除AWS CloudShell主目錄的選項。

> **Note**
>
> 當您 CloudShell 在上啟動時，無法使用 [下載] 選項Console Toolbar。

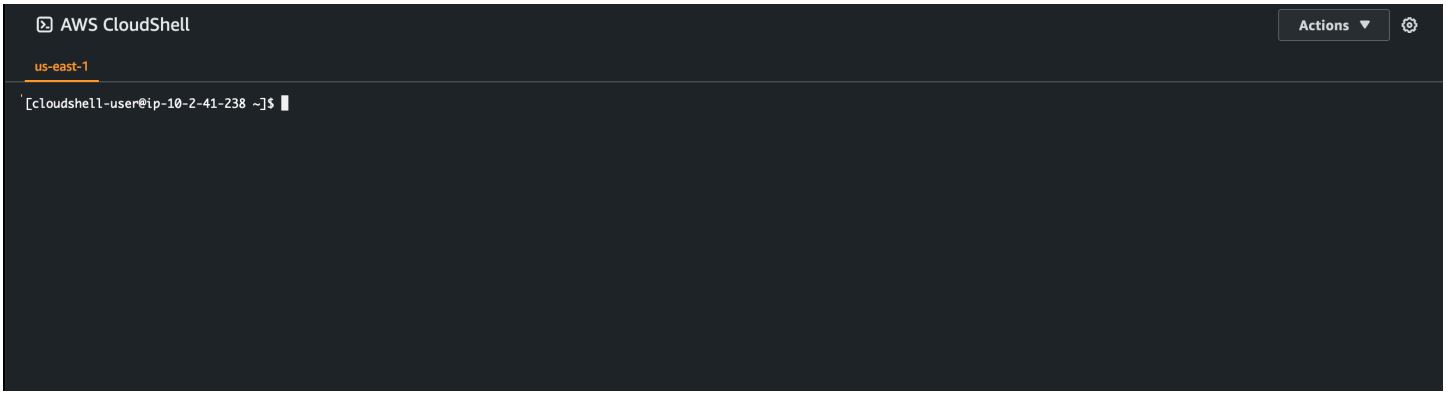
4. 「在新瀏覽器中開啟」索引標籤，提供以全螢幕存取 CloudShell 工作階段的選項。
5. 偏好設定選項，您可以使用此選項來自訂您的 shell 體驗。
6. 底部列提供下列選項：
   - CloudShell 從CloudShell圖標啟動。
   - 從意見反應圖示提供意見反應。選擇您要提交的意見反應類型、新增您的註解，然後選擇 [提交]。
     - 若要提交意見反應 CloudShell，請選擇下列其中一個選項：
       - 從主控台啟動 CloudShell，然後選擇 [意見反應]。新增您的註解，然後選擇 [提交]。
       - 選擇CloudShell主機左下角的Console Toolbar，然後選擇 [在新瀏覽器分頁中開啟] 圖示 [意見反應]。新增您的註解，然後選擇 [提交]。

     > **Note**
     >
     > 當您 CloudShell 在上啟動時，無法使用 [意見反應] 選項Console Toolbar。

   - 瞭解我們的隱私權政策和使用條款，以及自訂 Cookie 偏好設定。

工作在 AWS 區域

您AWS 區域正在運行的當前顯示在命令行界面上方。



您可以使用「區域」(Region) 選取器選取特定的「區域」AWS 區域 來選擇要在其中工作。變更區域之後，介面會在殼層工作階段連線至所選區域中執行的不同計算環境時重新整理。

> **Important**
>
> 每個儲存空間最多可以使用 1 GB 的永久儲存空間AWS 區域。永久性儲存裝置會儲存在您的主目錄 ($HOME) 中。這表示任何儲存在主目錄中的個人檔案、目錄、程式或指令碼都位於一個目錄中AWS 區域。此外，它們與位於主目錄中並存儲不同區域的位置不同。
> 永久性儲存空間中的檔案長期保留也會以每個區域為基礎進行管理。如需詳細資訊，請參閱持久性儲存。

指定您的預設AWS 區域值 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認證AWS CLI。指定殼層階段作業預設值AWS 區域的環境變數會在您AWS CloudShell從中的特定區域啟動時，AWS Management Console或在「區域」選取器中選擇選項時設定。

環境變數的優先順序高於由更新的AWS CLI認證檔案`aws configure`。因此，您無法執行`aws configure`命令來變更環境變數所指定的 Region。相反地，若要變更指AWS CLI令的預設「區域」，請將值指定給`AWS_REGION`環境變數。在接下來的範例中，請`us-east-1`以您所在的地區取代。

Bash or Zsh

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

設定環境變數會變更直到殼層工作階段結束或將變數設定為不同值之前使用的值。您可以在 shell 的啟動腳本中設置變量，以使變量在 future 的會話中持續存在。

PowerShell

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

如果您在 PowerShell 提示下設定環境變數，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，您可以通過將變量添加到您的 PowerShell 配置文件中來為所有 future 的 PowerShell 會話設置該變量。如需有關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，請執行指`aws configure list`令以顯示目前的AWS CLI組態資料。

> **Note**
>
> 對於特定AWS CLI指令，您可以使用指令行選項取代預設「區域」`--region`。若要取得更多資訊，請參閱《AWS Command Line Interface使用指南》中的指令行選項。

使用檔案和儲存

使用AWS CloudShell的界面，您可以將文件上傳到 shell 環境中並從下載文件。如需有關下載和上傳檔案的詳細資訊，請參閱開始使用AWS CloudShell。

為了確保您新增的任何檔案在工作階段結束後都可以使用，您應該瞭解永久儲存和暫存儲之間的差異。

- 持續性儲存空間：您每個儲存空間都有 1 GB 的永久儲存空間AWS 區域。永久性儲存位於您的主目錄中。
- 臨時存儲：臨時存儲在會話結束時回收。暫存儲位於主目錄之外的目錄中。

> **Important**
>
> 請務必在主目錄中保留您想要保留並用於 future shell 工作階段的檔案。例如，假設您透過執行`mv`命令將檔案移出主目錄。然後，當當前 shell 會話結束時，該文件被回收。

使用 Docker

AWS CloudShell完全支持 Docker，無需安裝或配置。您可以在其中定義，構建和運行 Docker 容器AWS CloudShell。您可以透過AWS CDK工具組部署以碼頭為基礎的資源 (例如以 Docker 容器為基礎

的 Lambda 函數)，也可以建置 Docker 容器，然後透過 Docker CLI 將它們推送至 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 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

搭配使用 Docker 時，有一定的限制和限制：AWS CloudShell

- 碼頭工人在環境中的空間有限。如果您擁有大型的個別映像檔，或是預先存在的 Docker 映像檔過多，可能會導致您無法提取、建立或執行其他映像檔的問題。如需 Docker 的詳細資訊，請參閱 Docker 文件指南。
- 只有特定區域才支援泊塢視窗。如需 Docker 支援哪些區域的相關資訊，請參閱碼頭區域。
- 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，瞭解如何解決這些問題的詳細資訊。

使用的協助工具功能AWS CloudShell

本主題說明如何使用協助工具功能CloudShell。您可以使用鍵盤瀏覽頁面上可聚焦的元素。您也可以自訂的外觀CloudShell，包括字體大小和界面主題。

鍵盤導航CloudShell

若要瀏覽頁面上可聚焦的元素，請按Tab。

CloudShell終端輔助功能

您可以使用Tab鍵入下列模式：

- 終端模式 (預設)— 在此模式下，終端捕獲您的Tab密鑰條目。對焦在終端上後，按下Tab僅訪問終端的功能。
- 導航模式— 在此模式下，終端不會捕獲您的Tab密鑰條目。新闻Tab瀏覽頁面上可聚焦的元素。

要在終端模式和導航模式之間切換，請按Ctrl+M。切換回去之後，標籤：導航出現在標題中，您可以使用Tab鍵以瀏覽整個頁面。

若要返回終端模式，請按Ctrl+M。或者，選擇X旁邊標籤：導航。

> **Note**
>
> 目前，CloudShell終端協助工具功能不適用於行動裝置。

在中選擇字體大小和界面主題CloudShell

您可以自訂的外觀CloudShell以適應您的視覺偏好。

- 字型大小— 從中選擇最小,小,中等,大，以及最大終端中的字體大小。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”。
- 主題— 選擇之間光和黑暗介面主題。如需變更介面主題的詳細資訊，請參閱the section called “變更介面主題”。

使用中的AWS服務AWS CloudShell

其中一個主要優點AWS CloudShell是您可以使用它從命令列介面管理您的AWS服務。這表示您不需要事先下載並安裝工具或設定憑據。啟動時AWS CloudShell，會建立已安裝下列AWS命令列工具的計算環境：

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

而且由於您已經登入AWS，因此在使用服務之前不需要在本機設定認證。您用來登入的認證AWS Management Console會轉寄到AWS CloudShell。

如果您要變更用於的預設「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「指定您的預設AWS 區域值 AWS CLI」。)

本主題的其餘部分AWS CloudShell將示範如何從命令列開始使用與選取的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 2AWS CLI 版提供的指令來使用的眾多AWS服務中的一些。如需完整清單，請參閱 AWS CLI 命令參考。

- DynamoDB
- AWS Cloud9
- 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AWS帳戶中命名的 NSQL 樣式表格。

```
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，請參閱使用AWS Command Line Interface者指南AWS CLI中的〈搭配使用 DynamoDB〉。

AWS Cloud9

AWS Cloud9是一種雲端整合開發環境 (IDE)，可用來在瀏覽器視窗中編寫、執行和偵錯。該環境具有代碼編輯器，調試器和終端。

以下`create-environment-ec2`命令創建具有指定設置的AWS Cloud9 EC2 開發環境。它會啟動 Amazon EC2 執行個體，然後從執行個體連線到環境。

```
aws cloud9 create-environment-ec2 --name my-demo-env --description "My demonstration
 development environment." --instance-type t2.micro --subnet-id subnet-1fab8aEX --
automatic-stop-time-minutes 60 --owner-arn arn:aws:iam::123456789012:user/MyDemoUser
```

如需詳細資訊，請參閱AWS Cloud9命令列參考。

Amazon EC2

Amazon EElastic Compute Cloud (Amazon EC2) 是一項 Web 服務，能夠提供安全且可調整大小的雲端。其旨在降低低達到 Web 規模運算的難度。

下列指`run-instances`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，請參閱使用AWS Command Line Interface者指南AWS CLI中的搭配使用 Amazon EC2。

S3 Glacier

S3 Glacier 和 S3 Glacier Deep Archive 是一種安全、耐用且低成本，適用於資料封存和長期備份。

下列create-vault指令會建立儲存庫 — 用來儲存歸檔的容器：

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

如需詳細資訊，請參閱使用AWS Command Line Interface者指南AWS CLI中的〈使用 Amazon S3 Glacier〉。

AWSElastic Beanstalk CLI CLI CLI CLI CLI

AWS Elastic BeanstalkCLI 提供了一個命令列界面，可簡化從本機儲存庫中進行的操作。在此內容中，環境為執行某個應用程式版本的AWS資源的集合。

以下create命令會在自訂的 Amazon Virtual Private Cloud (VPC) 中建立新環境。

```
$ 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 (CLI) 可提供多項高階命令。這些功能專門針對從本機開發環境中簡化叢集和任務所設計。Amazon ECS 叢集是任務或服務的邏輯分組。)。

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。此叢集配置使用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 SAMCLI 是在AWS Serverless Application Model模板和應用程序代碼上運行的命令行工具。您可以使用它執行多個任務。其中包括在本機叫用 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到AWS雲端。

下面的`init`命令初始化與作為參數傳遞所需的參數一個新的 SAM 項目：

```
sam init --runtime python3.7 --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 SAMCLI 命令參考。

自訂您的AWS CloudShell經驗

您可以自定義以下幾個方面AWS CloudShell經驗：

- 標籤佈局：將命令行界面拆分為多列和多行。
- 字型大小：調整命令行文本的大小。
- 顏色主題：在淺色和深色主題之間切換。
- 安全粘貼：開啟或關閉要求您在貼上多行文字之前驗證多行文字的功能。
- 恢復會話的 tmux：使用 tmux 會還原您的工作階段，直到工作階段變為非作用中。

您也可以通過以下方式擴展您的 shell 環境安裝您自己的軟體和修改啟動殼層指令碼。

將指令行顯示分割為多個頁籤

通過將命令行界面拆分為多個窗格來運行多個命令。

> **Note**
>
> 打開多個選項卡後，您可以通過單擊所選窗格中的任意位置來選擇要使用的選項卡。您可以通過選擇以關閉標簽x符號，這是旁邊的區域名稱。

- 選擇動作以及下列其中一個選項標籤佈局:
  - 新標籤：新增目前使用中分頁旁邊的新分頁。
  - 拆分為行：在當前活動標籤下方的行中添加一個新選項卡。
  - 拆分為列：在當前活動標籤旁邊的列中添加一個新選項卡。

  如果沒有足夠的空間完全顯示每個分頁，請捲動以查看整個分頁。您也可以選取分隔窗格的分割列，然後使用指標來增加或縮小窗格大小來拖曳它們。

變更字型大小

增加或減少指令行介面中顯示的文字大小。

1. 若要變更AWS CloudShell端子設定, 移至設定,偏好。
2. 選擇文字大小。您的選擇是最小,小,中等,大，以及最大。

變更介面主題

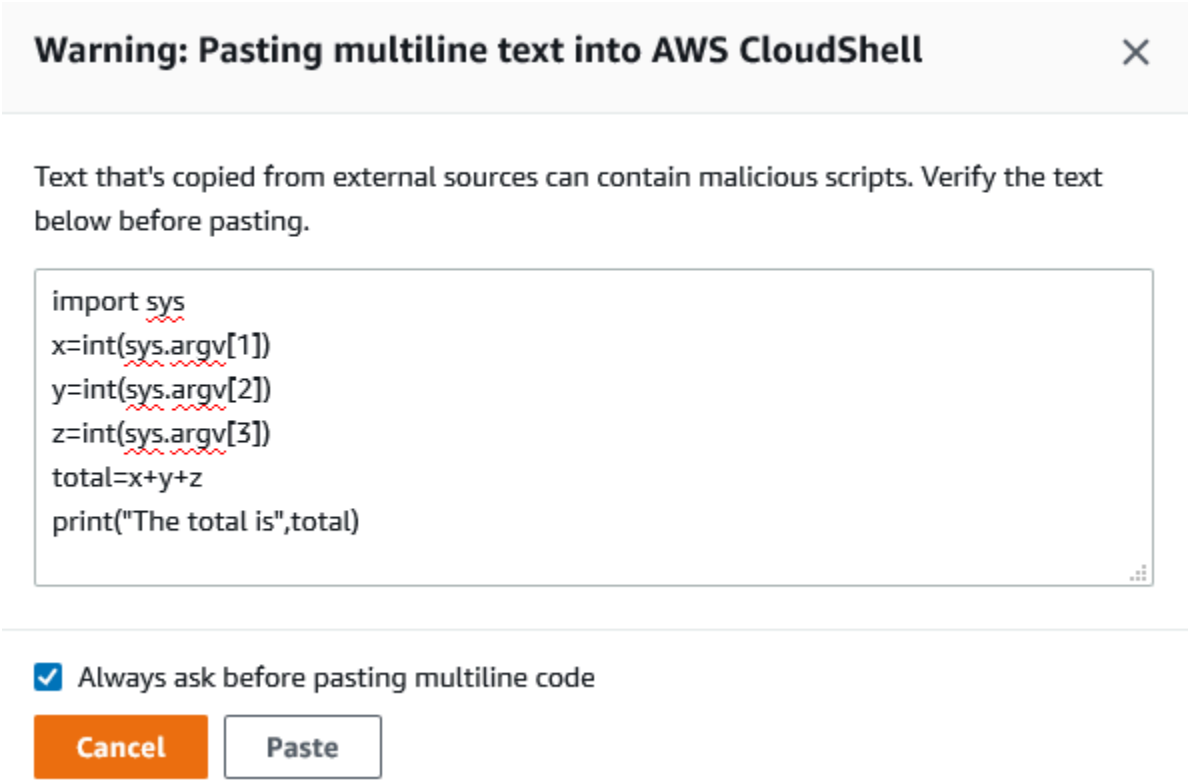
在指令行介面的淺色與深色主題之間切換。

1. 若要變更AWS CloudShell主題，移至設定,偏好。
2. 選擇光或者黑暗。

對多行文字使用安全貼上

安全貼上是一項安全性功能，會提示您確認您要貼到殼層中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的程式碼，這些程式碼會觸發 Shell 環境中的未預期行為

「安全貼上」對話方塊會顯示您複製到剪貼簿的完整文字。如果您對沒有安全風險感到滿意，請選擇貼上。



我們建議您啟用「安全貼上」，以便在指令碼中捕捉潛在的安全風險。您可以選擇開啟或關閉此功能偏好,啟用安全貼上和禁用安全粘貼。

使用tmux至工作階段還原

AWS CloudShell使用 tmux 在單個或多個瀏覽器選項卡中恢復會話。如果您重新整理瀏覽器標籤，它會繼續您的工作階段，直到工作階段變為非使用中狀 如需詳細資訊，請參閱工作階段還。

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，您可以從資料中心和網路架構中獲益，該架構專為滿足對安全性最敏感的組織的需求而打造。安全是 AWS 與您之間共同承擔的責任。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。

雲的安全性 — AWS 負責保護運行 AWS 雲中提供的所有服務的基礎設施，並為您提供可以安全使用的服務。我們的安全責任是我們的首要任務 AWS，並且我們的安全性有效性是由第三方審計師定期測試和驗證，作為AWS 合規計劃的一部分。

雲端安全性 — 您的責任取決於您使用的 AWS 服務，以及其他因素，包括資料的敏感性、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，遵循共同的責任模式。如需 AWS 服務安全性資訊，請參閱AWS 服務安全性說明文件頁面和符合性計劃 AWS 遵循工作範圍的AWS 服務。

下列主題說明如何設定 AWS CloudShell 以符合安全性與合規性目標。

主題

- 資料保護 AWS CloudShell
- AWS 的 Identity and Access Management CloudShell
- 登錄和監控 AWS CloudShell
- 符合性驗證 AWS CloudShell
- 韌性 AWS CloudShell
- 基礎結構安全 AWS CloudShell
- 中的配置和漏洞分析 AWS CloudShell
- 安全性最佳做法 AWS CloudShell
- AWS CloudShell 安全問題

資料保護 AWS CloudShell

AWS 共用責任模型適用於中的資料保護 AWS CloudShell。如此模型中所述， AWS 負責保護執行所有 AWS 雲端. 您負責維護在此基礎設施上託管內容的控制權。您還必須負責您所使用 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 共同責任模型和 GDPR 部落格文章。

基於資料保護目的，我們建議您使用 AWS IAM Identity Center 或 AWS Identity and Access Management (IAM) 保護 AWS 帳戶 登入資料並設定個別使用者。如此一來，每個使用者都只會獲得授與完成其任務所必須的許可。我們也建議您採用下列方式保護資料：

- 每個帳戶均要使用多重要素驗證 (MFA)。
- 使用 SSL/TLS 與 AWS 資源進行通訊。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。
- 使用 AWS 加密解決方案，以及其中的所有默認安全控制 AWS 服務。
- 使用進階的受管安全服務 (例如 Amazon Macie)，協助探索和保護儲存在 Amazon S3 的敏感資料。
- 如果您在透過命令列介面或 API 存取時需要經 AWS 過 FIPS 140-2 驗證的加密模組，請使用 FIPS 端點。如需 FIPS 和 FIPS 端點的相關資訊，請參閱聯邦資訊處理標準 (FIPS) 140-2 概觀。

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。這包括當您使用主控台、API AWS CloudShell 或 AWS SDK 時 AWS 服務 使用或其他使用時。 AWS CLI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

資料加密是指在靜態 (儲存時 AWS CloudShell) 和傳輸中 (在與服務端點之間 AWS CloudShell 傳輸時) 保護資料。

使用靜態加密 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，您可以免費獲得每個 AWS 區域 1 GB 的持續性存儲空間。持續性儲存空間位於主目錄 (`$HOME`) 中，而且對您來說是私有的。與在每個 shell 階段作業結束後回收的暫時環境資源不同，主目錄中的資料會持續存在。

儲存在中的資料加密 AWS CloudShell 是使用 AWS Key Management Service (AWS KMS) 所提供的加密金鑰來實作。這是用來建立和控制客戶主金鑰 (CMK) 的受管 AWS 服務，也就是用來加密儲存在 AWS CloudShell 環境中的客戶資料的加密金鑰。 AWS CloudShell 生成和管理加密密鑰，以代表客戶加密數據。

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。

預設情況下，用戶端網頁瀏覽器電腦與雲端架構之間的所有資料通訊 AWS CloudShell 都會透過 HTTPS/TLS 連線傳送所有資料進行加密。

您無需執行任何操作即可使用 HTTPS/TLS 進行通信。

AWS 的 Identity and Access Management CloudShell

AWS Identity and Access Management (IAM) 可協助系統管理員安全地控制 AWS 資源存取權。 AWS 服務 IAM 管理員控制哪些人可以通過身份驗證 (登入) 和授權 (具有權限) 來使用 CloudShell 資源。IAM 是您可以使用的 AWS 服務 ，無需額外付費。

主題

物件

您使用 AWS Identity and Access Management (IAM) 的方式會有所不同，具體取決於您在進行的工作 CloudShell。

服務使用者 — 如果您使用 CloudShell 服務執行工作，則管理員會為您提供所需的認證和權限。當您使用更多 CloudShell 功能來完成工作時，您可能需要其他權限。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。若您無法存取 CloudShell 中的某項功能，請參閱 疑難排解 AWS CloudShell 身分和存取。

服務管理員 — 如果您負責公司的 CloudShell 資源，您可能擁有完整的存取權 CloudShell。決定您的服務使用者應該存取哪些 CloudShell 功能和資源是您的工作。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。若要進一步瞭解貴公司如何搭配使用 IAM CloudShell，請參閱AWS 如何與 IAM CloudShell 搭配使用。

IAM 管理員：如果您是 IAM 管理員，建議您掌握如何撰寫政策以管理 CloudShell 存取權的詳細資訊。若要檢視可在 IAM 中使用的 CloudShell 基於身分的政策範例，請參閱。AWS 的身分型政策範例 CloudShell

使用身分驗證

驗證是您 AWS 使用身分認證登入的方式。您必須以 IAM 使用者身分或假設 IAM 角色進行驗證 (登入 AWS)。 AWS 帳戶根使用者

您可以使用透過 AWS 身分識別來源提供的認證，以聯合身分識別身分登入。 AWS IAM Identity Center (IAM 身分中心) 使用者、貴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識別的範例。當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 AWS 用同盟存取時，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。如需有關登入的詳細資訊 AWS，請參閱《AWS 登入 使用指南》 AWS 帳戶中的如何登入您的。

如果您 AWS 以程式設計方式存取，請 AWS 提供軟體開發套件 (SDK) 和命令列介面 (CLI)，以使用您的認證以加密方式簽署您的要求。如果您不使用 AWS 工具，則必須自行簽署要求。如需使用建議的方法自行簽署請求的詳細資訊，請參閱 IAM 使用者指南中的簽署 AWS API 請求。

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 建議您使用多重要素驗證 (MFA) 來增加帳戶的安全性。若要進一步了解，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》中的在 AWS中使用多重要素驗證 (MFA)。

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，該身分可完整存取該帳戶中的所有資源 AWS 服務和資源。此身分稱為 AWS 帳戶 root 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作業。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

最佳作法是要求人類使用者 (包括需要系統管理員存取權的使用者) 使用與身分識別提供者的同盟，才能使用臨時認證 AWS 服務 來存取。

聯合身分識別是來自企業使用者目錄的使用者、Web 身分識別提供者、Identi ty Center 目錄，或使用透過身分識別來源提供的認證進行存取 AWS 服務 的任何使用者。 AWS Directory Service同盟身分存取時 AWS 帳戶，他們會假設角色，而角色則提供臨時認證。

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，也可以連線並同步到自己身分識別來源中的一組使用者和群組，以便在所有應用程式 AWS 帳戶 和應用程式中使用。如需 IAM Identity Center 的相關資訊，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 使用者是您內部的身分，具 AWS 帳戶 有單一人員或應用程式的特定許可。建議您盡可能依賴暫時性憑證，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。如需詳細資訊，請參閱《IAM 使用者指南》https://docs.aws.amazon.com/IAM/latest/UserGuide/best-practices.html#rotate-credentials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。您無法以群組身分登入。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過程變得更為容易。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，但角色僅提供暫時憑證。若要進一步了解，請參閱《IAM 使用者指南》中的建立 IAM 使用者 (而非角色) 的時機。

IAM 角色

IAM 角色是您 AWS 帳戶 內部具有特定許可的身分。它類似 IAM 使用者，但不與特定的人員相關聯。您可以切換角色，在中暫時擔任 IAM 角色。 AWS Management Console 您可以透過呼叫 AWS CLI 或 AWS API 作業或使用自訂 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》中的使用 IAM 角色。

使用暫時憑證的 IAM 角色在下列情況中非常有用：

- 聯合身分使用者存取 – 若要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》中的為第三方身分供應商建立角色。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：您可以使用 IAM 角色，允許不同帳戶中的某人 (信任的委託人) 存取您帳戶的資源。角色是授予跨帳户存取權的主要方式。但是，對於某些策略 AWS 服務，您可以將策略直接附加到資源

（而不是使用角色作為代理）。若要了解跨帳戶存取角色和資源型政策間的差異，請參閱《IAM 使用者指南》中的 IAM 角色與資源類型政策的差異。

- 跨服務訪問 — 有些 AWS 服務 使用其他 AWS 服務功能。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。
  - 轉寄存取工作階段 (FAS) — 當您使用 IAM 使用者或角色在中執行動作時 AWS，您會被視為主體。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 會使用主體呼叫的權限 AWS 服務，並結合要求 AWS 服務 向下游服務發出要求。只有當服務收到需要與其 AWS 服務 他資源互動才能完成的請求時，才會發出 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給 AWS 服務。
  - 服務連結角色 — 服務連結角色是連結至. AWS 服務服務可以擔任代表您執行動作的角色。服務連結角色會顯示在您的中， AWS 帳戶 且屬於服務所有。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 執行個體上執行的應用程式以及發出 AWS CLI 或 AWS API 請求的臨時登入資料。這是在 EC2 執行個體內存放存取金鑰的較好方式。若要將 AWS 角色指派給 EC2 執行個體並提供給其所有應用程式，請建立連接至執行個體的執行個體設定檔。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性憑證。如需詳細資訊，請參閱《IAM 使用者指南》中的利用 IAM 角色來授予許可給 Amazon EC2 執行個體上執行的應用程式。

若要了解是否要使用 IAM 角色或 IAM 使用者，請參閱《IAM 使用者指南》中的建立 IAM 角色 (而非使用者) 的時機。

使用政策管理存取權

您可以透 AWS 過建立原則並將其附加至 AWS 身分識別或資源來控制中的存取。原則是一個物件 AWS ，當與身分識別或資源相關聯時，會定義其權限。 AWS 當主參與者 (使用者、root 使用者或角色工作階段) 提出要求時，評估這些原則。政策中的許可決定是否允許或拒絕請求。大多數原則會 AWS 以 JSON 文件的形式儲存在中。如需 JSON 政策文件結構和內容的相關資訊，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哪些內容。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。具有該原則的使用者可以從 AWS Management Console AWS CLI、或 AWS API 取得角色資訊。

身分型政策

身分型政策是可以連接到身分 (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。若要了解如何建立身分類型政策，請參閱《IAM 使用者指南》中的建立 IAM 政策。

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。受管理的策略是獨立策略，您可以將其附加到您的 AWS 帳戶. 受管政策包括 AWS 受管政策和客戶管理的策略。如需瞭解如何在受管政策及內嵌政策間選擇，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。主參與者可以包括帳戶、使用者、角色、同盟使用者或。 AWS 服務

資源型政策是位於該服務中的內嵌政策。您無法在以資源為基礎的政策中使用 IAM 的 AWS 受管政策。

存取控制清單 (ACL)

存取控制清單 (ACL) 可控制哪些委託人 (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 和 Amazon VPC 是支援 ACL 的服務範例。 AWS WAF若要進一步了解 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原則類型。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：許可界限是一種進階功能，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的限制。所有這類政策中的明確拒絕都會覆寫該允許。如需許可邊界的相關資訊，請參閱《IAM 使用者指南》中的 IAM 實體許可邊界。
- 服務控制策略（SCP）— SCP 是 JSON 策略，用於指定中組織或組織單位（OU）的最大權限。AWS Organizations AWS Organizations 是一種用於分組和集中管理您企業擁 AWS 帳戶 有的多個服務。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 限制成員帳戶中實體的權限，包括每個 AWS 帳戶根使用者帳戶。如需 Organizations 和 SCP 的相關資訊，請參閱《AWS Organizations 使用者指南》中的 SCP 運作方式。
- 工作階段政策：工作階段政策是一種進階政策，您可以在透過編寫程式的方式建立角色或聯合身分使用者的暫時工作階段時，作為參數傳遞。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。要了解如何在涉及多個政策類型時 AWS 確定是否允許請求，請參閱《IAM 使用者指南》中的政策評估邏輯。

AWS 如何與 IAM CloudShell 搭配使用

在您使用 IAM 管理存取權限之前 CloudShell，請先了解哪些 IAM 功能可搭配使用 CloudShell。

您可以搭配 AWS 使用的 IAM 功能 CloudShell

IAM 功能	CloudShell 支持
身分型政策	是
資源型政策	否
政策動作	是
政策資源	是
政策條件金鑰 (服務特定)	是

IAM 功能	CloudShell 支持
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

若要深入瞭解如何以 CloudShell 及其他 AWS 服務如何使用大多數 IAM 功能，請參閱 IAM 使用者指南中的搭配 IAM 使用的AWS 服務。

以身分識別為基礎的原則 CloudShell

支援身分型政策	是

身分型政策是可以連接到身分 (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。若要了解如何建立身分型政策，請參閱《IAM 使用者指南》中的建立 IAM 政策。

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。若要了解您在 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。

以身分識別為基礎的原則範例 CloudShell

若要檢視以 CloudShell 身分為基礎的原則範例，請參閱。AWS 的身分型政策範例 CloudShell

以資源為基礎的政策 CloudShell

支援以資源基礎的政策	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。主參與者可以包括帳戶、使用者、角色、同盟使用者或。 AWS 服務

若要啟用跨帳戶存取，您可以指定在其他帳戶內的所有帳戶或 IAM 實體，作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。當主體和資源位於不同時 AWS 帳戶，受信任帳戶中的 IAM 管理員也必須授與主體實體 (使用者或角色) 權限，才能存取資源。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 角色與資源型政策有何差異。

的政策動作 CloudShell

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哪些內容。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。原則動作通常與關聯的 AWS API 作業具有相同的名稱。有一些例外狀況，例如沒有相符的 API 操作的僅限許可動作。也有一些操作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與執行相關聯操作的許可。

若要查看 CloudShell 動作清單，請參閱服務授權參考 CloudShell中 AWS 定義的動作。某些動作可能有多個 API。

中的策略動作在動作之前 CloudShell 使用下列前置詞：

```
cloudshell
```

如需在單一陳述式中指定多個動作，請用逗號分隔。

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

若要檢視以 CloudShell 身分為基礎的原則範例，請參閱。AWS 的身分型政策範例 CloudShell

的政策資源 CloudShell

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哪些內容。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 (例如列出作業)，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 ARN 的清單，請參閱服務授權參考 CloudShell中 AWS 定義的資源。若要了解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。

若要檢視以 CloudShell 身分為基礎的原則範例，請參閱。AWS 的身分型政策範例 CloudShell

的政策條件索引鍵 CloudShell

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哪些內容。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，請使用邏輯OR運算來 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。如需詳細資訊，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條件金鑰。若要查看所有 AWS 全域條件金鑰，請參閱《IAM 使用者指南》中的AWS 全域條件內容金鑰。

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

若要檢視以 CloudShell 身分為基礎的原則範例，請參閱。AWS 的身分型政策範例 CloudShell

ACL 在 CloudShell

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。

阿巴克與 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。您可以將標籤附加到 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。

若要根據標籤控制存取，請使用 `aws:ResourceTag/key-name`、`aws:RequestTag/key-name` 或 `aws:TagKeys` 條件金鑰，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「什麼是 ABAC?」。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)。

使用臨時登入資料 CloudShell

支援臨時憑證	是

當您使用臨時憑據登錄時，有些 AWS 服務 不起作用。如需其他資訊，包括哪些 AWS 服務 與臨時登入資料搭配AWS 服務 使用，請參閱 IAM 使用者指南中的 IAM。

如果您使用除了使用者名稱和密碼以外的任何方法登入，則您正在 AWS Management Console 使用臨時認證。例如，當您 AWS 使用公司的單一登入 (SSO) 連結存取時，該程序會自動建立暫時認證。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。如需切換角色的相關資訊，請參閱《IAM 使用者指南》中的切換至角色 (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。

轉寄存取工作階段 CloudShell

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，您會被視為主體。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 會使用主體呼叫的權限 AWS 服務，並結合要求 AWS 服務 向下游服務發出要求。只有當服務收到需要與其 AWS 服務 他資源互動才能完成的請求時，才會發出 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給 AWS 服務。

> **Warning**
>
> 變更服務角色的權限可能會中斷 CloudShell 功能。只有在 CloudShell 提供指引時才編輯服務角色。

服務連結角色 CloudShell

支援服務連結角色。	否

服務連結角色是連結至. AWS 服務服務可以擔任代表您執行動作的角色。服務連結角色會顯示在您的中， AWS 帳戶 且屬於服務所有。IAM 管理員可以檢視，但不能編輯服務連結角色的許可。

AWS 的身分型政策範例 CloudShell

根據預設，使用者和角色不具備建立或修改 CloudShell 資源的權限。他們也無法使用 AWS Management Console、 AWS Command Line Interface (AWS CLI) 或 AWS API 來執行工作。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。

若要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。

有關由定義的動作和資源類型的詳細資訊 CloudShell，包括每種資源類型的 ARN 格式，請參閱服務授權參考 CloudShell中的 AWS 動作、資源和條件金鑰。

主題

- 政策最佳實務
- 使用 CloudShell 主控台
- 允許使用者檢視他們自己的許可

政策最佳實務

以身分識別為基礎的政策會決定某人是否可以建立、存取或刪除您帳戶中的 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 受管原則並邁向最低權限權限 — 若要開始授與使用者和工作負載的權限，請使用可授與許多常見使用案例權限的AWS 受管理原則。它們可用在您的 AWS 帳戶. 建議您透過定義特定於您使用案例的 AWS 客戶管理政策，進一步降低使用權限。如需詳細資訊，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。如需如何使用 IAM 套用許可的詳細資訊，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。您也可以使用條件來授與對服務動作的存取權 (如透過特定) 使用這些動作 AWS 服務，例如 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，可協助您編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 IAM Access Analyzer 政策驗證。
- 需要多因素身份驗證 (MFA) — 如果您的案例需要 IAM 使用者或根使用者 AWS 帳戶，請開啟 MFA 以獲得額外的安全性。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》中的設定 MFA 保護的 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，您必須擁有最少一組許可。這些權限必須允許您列出和檢視有關 AWS 帳戶. CloudShell 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。

您不需要為僅對 AWS CLI 或 AWS API 進行呼叫的使用者允許最低主控台權限。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。

若要確保使用者和角色仍可使用 CloudShell 主控台，請同時將 CloudShell *ConsoleAccess*或受*ReadOnly* AWS 管理的原則附加至實體。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。此原則包含在主控台上或以程式設計方式使用 AWS CLI 或 AWS API 完成此動作的權限。

```
{
    "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": "*"
        }
    ]
}
```

疑難排解 AWS CloudShell 身分和存取

使用下列資訊可協助您診斷和修正使用和 IAM 時可能會遇到的 CloudShell 常見問題。

主題

- 我沒有執行操作的授權 CloudShell
- 我沒有授權執行 iam：PassRole

- 我想允許我以外的人訪 AWS 帳戶 問我的 CloudShell 資源

我沒有執行操作的授權 CloudShell

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 系統管理員。您的管理員提供您的登入憑證。

我沒有授權執行 iam：PassRole

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

有些 AWS 服務 允許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。

名為 marymajor 的 IAM 使用者嘗試使用主控台在 CloudShell 中執行動作時，發生下列範例錯誤。但是，該動作要求服務具備服務角色授與的許可。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 系統管理員。您的管理員提供您的登入憑證。

我想允許我以外的人訪 AWS 帳戶 問我的 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資源的許可。

如需進一步了解，請參閱以下內容：

- 若要瞭解是否 CloudShell 支援這些功能，請參閱AWS 如何與 IAM CloudShell 搭配使用。
- 若要了解如何提供對您所擁有資源 AWS 帳戶 的存取權，請參閱《IAM 使用者指南》中您擁有的另一 AWS 帳戶 個 IAM 使用者提供存取權限。
- 若要了解如何向第三方提供對資源的存取權 AWS 帳戶，請參閱 IAM 使用者指南中的提供第三方 AWS 帳戶 擁有的存取權。
- 若要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)。
- 若要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 角色與資源型政策的差異。

使用 IAM 政策管理 AWS CloudShell 存取和使用

透過 AWS Identity and Access Management (IAM) 可提供的存取管理資源，管理員可以將許可授與 IAM 使用者。這樣，這些用戶就可以訪問 AWS CloudShell 和使用環境的功能。系統管理員也可以建立原則，以精細層級指定這些使用者可以在 shell 環境中執行的動作。

系統管理員授與使用者存取權的最快方法是透過 AWS 受管理的原則。AWS 受管政策是由建立和管理的獨立政策 AWS。的下列 AWS 受管政策 AWS CloudShell 可附加至 IAM 身分：

- AWS CloudShellFullAccess：授予使用 AWS CloudShell 的許可以及對所有功能的完整存取權。

該AWS CloudShellFullAccess政策使用萬用字元 (*) 字元來授予 IAM 身分 (使用者、角色或群組) 的完整存取權限 CloudShell和功能。如需有關此原則的詳細資訊，請參閱AWS CloudShellFullAccess受AWS 管理策略使用指南中的。

> **Note**
>
> 也可以啟動具有下列 AWS 受管政策的 IAM 身分 CloudShell。不過，這些原則會提供廣泛的權限。因此，我們建議您僅在這些政策對 IAM 使用者的工作角色至關重要時，才授予這些政策。
>
> - 管理員：為 IAM 使用者提供完整存取權限，並允許他們將權限委派給中的每個服務和資源 AWS。
> - 開發人員進階使用者：讓 IAM 使用者能夠執行應用程式開發工作，並建立和設定支援 AWS 感知應用程式開發的資源和服務。
>
> 如需附加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)。

管理 AWS CloudShell 使用自訂策略中允許的動作

若要管理 IAM 使用者可以執行的動作 CloudShell，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 使用者需要執行下列動作的許可：

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

如果附加政策未明確允許其中一個動作，則當您嘗試啟動時，會傳回 IAM 許可錯誤 CloudShell。

AWS CloudShell 權限

名稱	所授予之許可的描述	是否需要啟動 CloudShell？
`cloudshell:CreateEnvironment`	創建一個 CloudShell 環境，在 CloudShell 會話開始時檢索佈局，並從後端的 Web 應用程序中保存當前佈局。此權限僅預期*為中所`Resource`述的值the section called “下列項目的	是

名稱	所授予之許可的描述	是否需要啟動 CloudShell？
	IAM 政策範例 CloudShell”。	
cloudshell:CreateSession	從連接至 CloudShell 環境 AWS Management Console。	是
cloudshell:GetEnvironmentStatus	讀取 CloudShell 環境的狀態。	是
cloudshell:DeleteEnvironment	刪除 CloudShell 環境。	否
cloudshell:GetFileDownloadUrls	產生預先簽署的 Amazon S3 URL，用於透過 CloudShell CloudShell Web 介面下載檔案。	否
cloudshell:GetFileUploadUrls	產生預先簽署的 Amazon S3 URL，用於透過 CloudShell CloudShell 網頁界面上傳檔案。	否
cloudshell:PutCredentials	將用於登入的認證轉寄 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

下列項目的 IAM 政策範例 CloudShell

下列範例顯示如何建立原則以限制可存取的使用者 CloudShell。這些範例也會顯示可在 shell 環境中執行的動作。

此下列原則會強制執行完全拒絕存取 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 但禁止他們產生用於檔案上傳和下載的預先簽署 URL。用戶仍然可以使用客戶端例如在環境之間傳輸文件。`wget`

```
{
    "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，此原則會防止您用 AWS Management Console 來登入的認證轉送至 CloudShell 環境。具有此政策的 IAM 使用者需要在中手動設定其登入資料 CloudShell。

```
{
    "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": "*"
     }]
```

```
}
```

存取權限 AWS 服務

CloudShell 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`權限。

的這個預先驗證功能使使用 AWS CLI起來 CloudShell 很方便。不過，IAM 使用者仍需要從命令列呼叫的明確許可。 AWS 服務

例如，假設 IAM 使用者必須建立 Amazon S3 儲存貯體，並將檔案做為物件上傳到他們。您可以建立明確允許這些動作的原則。IAM 主控台提供互動式視覺化編輯器，可指導建立 JSON 格式政策文件的程序。建立政策後，您可以將其附加到相關的 IAM 身分 (使用者、群組或角色)。

如需附加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)。

登錄和監控 AWS CloudShell

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

監視活動 CloudTrail

AWS CloudShell 與提供使用者 AWS CloudTrail、角色或 AWS 服務 中所採取之動作記錄的服務整合 AWS CloudShell。 CloudTrail 擷取 AWS CloudShell 作為事件的所有 API 呼叫。擷取的呼叫包括來自 AWS CloudShell 主控台的呼叫和 AWS CloudShell API 的程式碼呼叫。

如果您建立追蹤，您可以啟用連續交付 CloudTrail 事件到 Amazon Simple Storage Service (Amazon S3) 儲存貯體。這包括 AWS CloudShell.

如果您不設定追蹤記錄，仍然可以透過 CloudTrail 主控台中的 Event history (事件歷史記錄) 檢視最新的事件。使用收集的資訊 CloudTrail，您可以發現有關請求的各種資訊。例如，您可以決定向 AWS 發出的請求 CloudShell、了解提出請求的來源 IP 地址、提出請求的人員以及提出請求的時間。

AWS CloudShell 在 CloudTrail

下表列出儲存在 CloudTrail 記錄檔中的 AWS CloudShell 事件。

> **Note**
>
> AWS CloudShell 事件，其中包括：
>
> - *表示這是一個非變異（只讀）API 調用。
> - 這個詞Environment與託管 shell 體驗的計算環境的生命週期有關。
> - 這個詞Layout還原 CloudShell終端中的所有瀏覽器選項卡。

CloudShell 活動在 CloudTrail

事件名稱	描述
createEnvironment	創建 CloudShell 環境時發生。
createSession	當 CloudShell 環境從連接時發生 AWS Management Console。
deleteEnvironment	當一個 CloudShell 環境被刪除時發生。
deleteSession	當刪除當前瀏覽器選 CloudShell 項卡中運行的選項卡中的會話時發生。
getEnvironmentStatus*	當檢索 CloudShell 環境的狀態時發生。
getFileDownloadUrls*	當產生用於透 CloudShell 過 CloudShell Web 介面下載檔案的預先簽署 Amazon S3 URL 時，會發生這種情況。
getFileUploadUrls*	在產生用於透過 CloudShell CloudShell Web 介面上傳檔案的預先簽署 Amazon S3 URL 時發生。
getLayout*	當檢索會話開始時的 CloudShell 佈局發生。
putCredentials	當轉發用於登錄到的憑據 AWS Management Console 時 CloudShell 發生。

事件名稱	描述
redeemCode*	在 CloudShell環境中擷取重新整理權杖的工作流程開始時發生。您可以稍後在putCredentials 指令中使用此權杖來存取 CloudShell 環境。
sendHeartBeat	發生以確認 CloudShell 會話處於活動狀態。
startEnvironment	啟動 CloudShell 環境時發生。
stopEnvironment	當正在運行的 CloudShell 環境停止時發生。
updateLayout	當保存從後端的 Web 應用程序的當前佈局時發生。

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。

EventBridge AWS CloudShell 動作的規則

使用 EventBridge 規則時，您可以指定在 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，以根據記錄為記錄檔中事件的 AWS CloudShell 動作，指定要採取的 CloudTrail 目標動作。

例如，您可以 AWS CLI使用put-rule指令建立 EventBridge 規則。呼put-rule叫必須至少包含 EventPattern 或 ScheduleExpression。觀察到相符事件時，會觸發規則。 EventPatterns 對 EventPattern 於 AWS CloudShell 事件：

```
{ "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 EventBridge中的事件和事件模式。

符合性驗證 AWS CloudShell

協力廠商稽核人員會評估 AWS 服務的安全性與合規性，作為多項 AWS 合規計畫的一部分。

AWS CloudShell 適用於以下合規計劃：

SOC

AWS 系統與組織控制 (SOC) 報告是獨立的第三方檢驗報告，展示如何 AWS 達成關鍵合規性控制與目標。

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

PCI

支付卡行業數據安全標準（PCI DSS）是由 PCI 安全標準委員會管理的專有信息安全標準，該委員會由美國運通，發現金融服務，JCB 國際，全 MasterCard 球和 Visa Inc 成立。

服務	SDK	PCI
AWS CloudShell	CloudShell	✓

ISO 和 CSA 之星認證和服務

AWS 擁有符合 ISO/IEC 27001:2013、27017:2015、27017:2019, 27701:2019, 2019 年 1 月 31 日, 9001:2015 和中國國家安全局之星中央管理委員會 V4.0 認證。

服務	SDK	ISO 和 CSA 之星認證和服務
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 () 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：

- 第三方評估機構 (3PAO) 評估：此服務目前正由我們的第三方評估機構進行評估。
- 聯合授權委員會（JAB）審查：此服務目前正在進行 JAB 審查。
- 國防信息系統局（DISA）審查：此服務目前正在進行 DISA 審查。

服務	SDK	DoD CC SRG IL2 (East/West)	DoD 抄送 SRG IL2 () GovCloud	DoD 抄送 SRG IL4 () GovCloud	DoD 抄送 SRG IL5 () GovCloud	DoD CC SRG IL6（秘密區域） AWS
AWS CloudShell	CloudShell	3 PAO 評估	N/A	N/A	N/A	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 使受 HIPAA 規範的涵蓋實體及其業務夥伴能夠安全地處理、儲存和傳輸受保護的健康資訊 (PHI)。此外，截至 2013 年 7 月，為此類客戶 AWS 提供標準化的商業夥伴增補合約（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環境中的 AWS 服務。例如，當您建立 IAM 政策時，請使用 Amazon 資源名稱 (ARN) 和讀取 AWS CloudTrail 日誌。

MTCS

多層雲端安全性 (MTCS) 是新加坡的一項運作安全管理標準 (SPRING SS 584)，以 ISO 27001/02 資訊安全管理系統 (ISMS) 標準為基礎。

服務	SDK	美國東部 (俄亥俄州)	美國東部 (維吉尼亞北部)	美國西部 (奧勒岡州)	美國西部 (加利福尼亞州)	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（國家塞古里達）認證計劃是由財政和公共行政部和 CCN（國家密碼中心）開發的。這包括足夠保護資料所需的基本原則和最低要求。

服務	SDK	高
AWS CloudShell	CloudShell	✓

FINMA

瑞士金融市場監管局（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一直致力於滿足瑞士金融服務監管機構和客戶對雲端服務提供商的高度期望。

服務	SDK	FINMA
AWS CloudShell	CloudShell	✓

PiTuKri

AWS PiTuKri 符合要求，證明我們持續致力於滿足芬蘭交通與通訊局 Traficom 對雲端服務提供者的高度期望。

服務	SDK	PiTuKri
AWS CloudShell	CloudShell	✓

如需特定合規計劃範圍內的 AWS 服務清單，請參閱合規計劃 AWS 服務範圍內的合規計。如需一般資訊，請參閱AWS 規範計劃AWS。

您可以使用下載協力廠商稽核報告 AWS Artifact。如需詳細資訊，請參閱下載 AWS Artifact 中的報告。

您在使用時的合規責任取決 AWS CloudShell 於資料的敏感性、公司的合規目標以及適用的法律和法規。 AWS 提供下列資源以協助遵循法規：

- 安全性與合規性快速入門指南 — 這些部署指南討論架構考量，並提供在上部署以安全為重點和遵循法規遵循的基準環境的步驟。 AWS
- 建構 HIPAA 安全性與合規性白皮書 — 本白皮書說明公司如何使用建立符合 HIPAA 標準的應用 AWS 程式。
- AWS 合規資源AWS — 此工作簿和指南集合可能適用於您的產業和所在地。

- 使用AWS Config 開發人員指南中的規則評估資源 — 此 AWS Config 服務會評估您的資源組態符合內部實務、產業準則和法規的程度。
- AWS Security Hub— 此 AWS 服務提供安全狀態的全面檢視，協助您檢查您 AWS 是否符合安全性產業標準和最佳做法。

韌性 AWS CloudShell

AWS 全球基礎架構是圍繞區 AWS 域和可用區域建立的。 AWS 區域提供多個實體分離和隔離的可用區域，這些區域透過低延遲、高輸送量和高度備援的網路連線。透過可用區域，您可以設計與操作的應用程式和資料庫，在可用區域之間自動容錯移轉而不會發生中斷。可用區域的可用性、容錯能力和擴充能力，均較單一或多個資料中心的傳統基礎設施還高。

如需區域和可用區域的相關 AWS 資訊，請參閱AWS 全域基礎結構。

除了 AWS 全球基礎架構外，還 AWS CloudShell 支援特定功能，以支援您的資料復原和備份需求。

- 提交您建立並新增至的檔案 AWS CodeCommit。這是由 Amazon Web Services 託管的版本控制服務，您可以使用它在雲端中私有存放和管理資產。這些資產可以由文檔，源代碼和二進制文件組成。如需詳細資訊，請參閱 教學課程：使用 CodeCommit 於AWS CloudShell。
- 使用 AWS CLI 呼叫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，請參閱入門教學課程。

基礎結構安全 AWS CloudShell

作為託管服務， AWS CloudShell 受到 AWS 全球網絡安全的保護。有關 AWS 安全服務以及如何 AWS 保護基礎結構的詳細資訊，請參閱AWS 雲端安全 若要使用基礎架構安全性的最佳做法來設計您的 AWS 環境，請參閱安全性支柱架構良 AWS 好的架構中的基礎結構保護。

您可以使用 AWS 已發佈的 API 呼叫透 AWS CloudShell 過網路存取。用戶端必須支援下列項目：

- 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，請求必須使用存取索引鍵 ID 和與 IAM 主體相關聯的私密存取索引鍵來簽署。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 會自動為您計算環境的系統套件安裝安全性修補程式。

中的配置和漏洞分析 AWS CloudShell

AWS CloudShell 使用者有責任確保他們在運算環境中安裝的任何軟體都已修補並保持最新狀態。

安全性最佳做法 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。因為這些最佳實務可能不適合或無法滿足您的環境，所以請將其視為實用建議，不要當成指示。

一些安全性最佳做法 AWS CloudShell

- 使用 IAM 許可和政策來控制存取權限， AWS CloudShell 並確保使用者只能執行其角色所需的那些動作 (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 存取和使用。
- 請勿在 IAM 實體 (例如使用者、角色或工作階段名稱) 中包含敏感資料。
- 啟用「保持安全貼上」功能，可在您從外部來源複製的文字中 catch 測潛在的安全風險。預設為啟用「安全貼上」。如需詳細資訊，請參閱 對多行文字使用安全貼上。
- 如果您將協力廠商應用程式安裝到的計算環境，請熟悉「共用安全性責任模型」 AWS CloudShell。
- 在編輯影響使用者 shell 體驗的 shell 指令碼之前，先準備復原機制。如需詳細資訊，請參閱 使用指令碼修改您的殼層。
- 將您的程式碼安全地存放在版本控制系統中，例如 AWS CodeCommit。

AWS CloudShell 安全問題

有關此安全性的常見問題解答 AWS 服務。

- 啟動和啟動 shell 會話時使用的 AWS 過程 CloudShell 和技術是什麼？
- 是否有可能限制網絡訪問 CloudShell？
- 我可以自訂我的 CloudShell 環境嗎？
- 我的$HOME目錄實際存儲在哪裡 AWS 雲端？

- 是否有可能加密我的$HOME目錄？
- 我可以在$HOME目錄上執行病毒掃描嗎？

啟動和啟動 shell 會話時使用的 AWS 過程 CloudShell 和技術是什麼？

登入時 AWS Management Console，您需要輸入 IAM 使用者登入資料。而且，當您 CloudShell 從主控台介面啟動時，這些認證會用於呼叫 CloudShell API，以建立服務的運算環境。然後 AWS Systems Manager 會為計算環境建立工作階段，並 CloudShell 將命令傳送至該工作階段。

返回保安常見問題列表

是否有可能限制網絡訪問 CloudShell？

您可以透過與您的網路供應商連線 CloudShell 來限制網路存取。或者，您可以使用 IAM 許可、明確拒絕存取 CloudShell 或不提供任何 IAM 許可，以及使用隱含拒絕 IAM 功能。如需詳細資訊，請參閱使用 IAM 政策管理 AWS CloudShell 存取和用量。

返回保安常見問題列表

我可以自訂我的 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協力廠商軟體。只有安裝在您$HOME目錄中的軟體才會在工作階段之間保留。

如AWS 同共用職責模型所定義，您必須負責所安裝之應用程式的必要組態與管理。

返回保安常見問題列表

我的$HOME目錄實際存儲在哪裡 AWS 雲端？

Amazon S3 提供用於將資料存放$HOME在您的中的基礎設施。

返回保安常見問題列表

是否有可能加密我的$HOME目錄？

您$HOME目錄中的資料已使用 Amazon S3 加密加密。

返回保安常見問題列表

我可以在$HOME目錄上執行病毒掃描嗎？

目前，無法對$HOME目錄運行病毒掃描。此功能的 Support 正在審核中。

返回保安常見問題列表

AWS CloudShell運算環境：規格和軟體

當您啟動時AWS CloudShell，系統會建立以 Amazon Linux 2023 為基礎的運算環境來託管殼層體驗。該環境配置了計算資源（vCPU 和內存），並提供各種預先安裝的軟件，可以從命令行界面訪問。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運算環境都會指派下列 CPU 和記憶體資源：

- 1 個 vCPU (虛擬中央處理器)
- 2 GiB 記憶體

此外，環境會以下列儲存區組態佈建：

- 1 GB 永久性儲存裝置 (工作階段結束後仍會保留儲存)

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 協議，該協議允許用戶的 Web 瀏覽器和AWS雲中的 CloudShell 服務之間進行雙向交互式通信。如果您在私人網絡中使用瀏覽器，則代理服務器和防火牆可能會促進對 Internet 的安全訪問。 WebSocket 通信通常可以遍歷代理服務器而沒有問題。但在某些情況下，代理服務器會阻 WebSockets 止正常工作。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`。

如果重複發生此錯誤，請參閱 Proxy 伺服器的說明文件，以確保其設定為允許 WebSockets。或者，您也可以聯絡網路的系統管理員。

> **Note**
>
> 如果您想要透過允許列出特定 URL 來定義精細的權限，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指令會傳送至該 Systems Manager 工作階段。)

Systems Manager 使 StreamUrl 用的格式為`wss://ssmmessages.region.amazonaws.com/v1/data-channel/session-id?stream=(input|output)`。
此區域代表支援之區AWS域的地區識別碼AWS Systems Manager，`us-east-2`例如美國東部 (俄亥俄) 區域。
因為工作階段識別碼是在特定 Systems Manager 工作階段成功啟動之後建立的，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 ManagerAPI 參考中的StartSession作業。

預裝軟體

> **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的版本號碼。相反，我們描述了如何檢查安裝了哪個版本。若要檢查安裝的版本，請輸入程式名稱，然後輸入`--version`選項 (例如`git --version`)。

貝殼

預裝外殼

名稱	描述	版本資訊
Bash	Bash 外殼是的預設殼層應用程式AWS CloudShell。	`bash --version`
PowerShell (PWSH)	提供命令行界面和腳本語言支持， PowerShell 是建立在微軟的 .NET 命令語言運行時之上。 PowerShell 使用稱為`cmdlets`接受並返回 .NET 對象的輕量級命令。	`pwsh --version`
Z 型外殼 (Z 系列)	Z 殼牌，也被稱為`zsh`，是伯恩殼牌的擴展版本，它提供了主題和插件增強的自定義支持。	`zsh --version`

AWS命 CLI 行界面

CLI

名稱	描述	版本資訊
AWS CDKCLI 工具包	CLI 命令工具AWS CDK包是與您的AWS CDK應用程序交互的主要工具。cdk它會執行您的應用程式、詢問您定義的應用程式模型，以及產生和部AWS CloudFormation署由. AWS CDK 如需詳細資訊，請參閱AWS CDK工具組。	`cdk --version`
AWS CLI	這AWS CLI是一個命令行界面，您可以使用它從命令行管理多個AWS服務並使用腳本自動化它們。如需詳細資訊，請參閱使用中的AWS服務AWS CloudShell。 如需如何確保使用最多 up-to-date AWS CLI版本 2 的相關資訊，請參閱安裝AWS CLI到您的主目錄。	`aws --version`
EB CLI	AWS Elastic BeanstalkCLI 提供命令列介面，可簡化從本機存放庫建立、更新和監控環境的作業。 如需詳細資訊，請參閱開發人員指南中的使用 Elastic Beanstalk 命令列介面 (EB CLI)。AWS Elastic Beanstalk	`eb --version`

名稱	描述	版本資訊
Amazon ECS CLI	Amazon Elastic Container Service (Amazon ECS) 命令列界面 (CLI) 提供高階命令，以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 彈性容器服務開發人員指南中的使用 Amazon ECS 命令列界面。	`ecs-cli --version`
AWS SAM CLI	AWS SAMCLI 是在AWS Serverless Application Model 模板和應用程序代碼上運行的命令行工具。您可以執行多項工作。其中包括在本機叫用 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到雲端。AWS 如需詳細資訊，請參閱AWS Serverless Application Model 開發人員指南中的 AWS SAMCLI 命令參考。	`sam --version`

名稱	描述	版本資訊
AWS Tools for PowerShell	這AWS Tools for PowerShell些 PowerShell 模組是建立在公開的功能上的AWS SDK for .NET. 使用AWS Tools for PowerShell，您可以從命令列對資AWS源執行指 PowerShell 令碼作業。 AWS CloudShell預先安裝的模組化版本 (AWS.Tools)。AWS Tools for PowerShell 如需詳細資訊，請參閱使用AWS Tools for PowerShell者指南 PowerShell中的使用 AWS 工具。	`pwsh --Command ' Get-Module -ListAvailable -Name AWS.Tools.Common'`

執行階段和 AWS 開發套件：Node.js 和 Python 3

執行階段和 AWS 開發套件

名稱	描述	版本資訊
Node.js（與故宮）	Node.js 是一個 JavaScript 運行時，旨在使其更容易應用異步編程技術。如需詳細資訊，請參閱 Node.js 官方網站上的文件。 npm 是一個軟件包管理器，可以訪問 JavaScript模塊的在線註冊表。如需詳細資訊，請參閱官方 npm 網站上的文件。	• Node.js: `node --version` • 故宮：`npm --version`
Node.js JavaScript 中適用的軟體套件	軟體開發套件 (SDK) 可為 AWS 服務 (包括 Amazon S3、Amazon EC2、Dynam	`npm -g ls --depth 0 2>/dev/null | grep aws-sdk`

名稱	描述	版本資訊
	oDB 和 Amazon SWF) 提供 JavaScript物件，協助簡化程式碼撰寫作業。如需詳細資訊，請參閱《AWS SDK for JavaScript 開發人員指南》https://docs.aws.amazon.com/sdk-for-javascript/latest/developer-guide/ 。	
Python	Python 3 是準備在外殼環境中使用。Python 3 現在被認為是編程語言的默認版本（對 Python 2 的支持將於 2020 年 1 月結束）。有關更多信息，請參閱 Python 官方網站上的文檔。 此外，預先安裝的是 pip，即 Python 的軟件包安裝程序。您可以使用此命令行程序從在線索引（如 Python 包索引）安裝 Python Package。如需詳細資訊，請參閱 Python 封裝授權單位提供的文件。	• Python 3: `python3 --version` • 點子：`pip3 --version`
適用於 Python (Boto3) 的 SDK	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。SDK 提供 easy-to-use物件導向 API，以及對. AWS 服務 如需詳細資訊，請參閱 Boto3 文件。	`pip3 list | grep boto3`

開發工具和殼層公用程式

開發工具和殼層公用程式

名稱	描述	版本資訊
bash-completion	bash-complete 是 shell 函數的集合，允許通過按 Tab 鍵自動完成部分類型的命令或參數。您可以在中找到 bash 完成支持的軟件包。`/usr/share/bash-completion/completions` 若要為套件的指令設定自動完成功能，必須取得程式檔案的來源。例如，要為 Git 命令設置自動完成功能，請添加以下行，以`.bashrc`便每當您的 AWS CloudShell會話啟動時都可以使用該功能： `source /usr/share/bash-completion/completions/git` 如果您想要使用自訂完成指令碼，請將它們新增至永久性主目錄 (`$HOME`)，並直接在中取得`.bashrc`。 如需詳細資訊，請參閱上的專案的「讀我檔案」頁面 GitHub。	`dnf info bash-completion`
CodeCommit Git 的實用程序	git-remote-codecommit 是一個實用程序，它提供了一種通過擴展 Git 從 CodeCommit 存儲庫中推送和提取代碼的簡單方	`pip3 list | grep git-remote-codecommit`

名稱	描述	版本資訊
	法。這是支援使用聯合存取、身分識別提供者和臨時登入資料建立之連線的建議方法。 如需詳細資訊，請參閱AWS CodeCommit使用指南 git-remote-codecommit中的 AWS CodeCommit HTTPS 連線的設定步驟。	
Git	Git 是一個分散式版本控制系統，透過分支工作流程和內容暫存來支援現代軟體開發實務。有關更多信息，請參閱 Git 官方網站上的文檔頁面。	`git --version`
iputils	iputils 套件包含適用於 Linux 網路的公用程式。如需有關所提供公用程式的詳細資訊，請參閱上的 iputils 儲存庫。GitHub	一個 iputils 工具的例子：`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器修改的輸出。如需詳細資訊，請參閱上 GitHub的 jq 手冊。	`jq --version`
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`

名稱	描述	版本資訊
make	make 公用程式會用makefiles 來自動執行一組工作並組織程式碼編譯。如需詳細資訊，請參閱 GNU 製作文件。	`make --version`
man	man 指令提供指令行公用程式和工具的手冊頁面。例如，man ls傳回列出目錄內容之ls命令的手冊頁。有關更多信息，請參閱維基百科手冊頁上的條目。	`man --version`
nano	nano 是用於基於文本的界面的小型和用戶友好的編輯器。如需詳細資訊，請參閱 GNU nano 文件。	`nano --version`
procps	procps 是一個系統管理實用程序，可用於監視和停止當前正在運行的進程。如需詳細資訊，請參閱 README 檔案，其中列出可以透過 procps 執行的程式。	`ps --version`
SSH 用戶端	SSH 用戶端會使用安全殼層通訊協定來與遠端電腦進行加密通訊。OpenSSH 是已預先安裝的 SSH 用戶端。如需詳細資訊，請參閱 OpenBSD 所維護的 OpenSSH 網站。	`ssh -V`

名稱	描述	版本資訊
sudo	使用 sudo 公用程式，使用者可以使用其他使用者 (通常是超級使用者) 的安全性權限來執行程式。當您需要以系統管理員身份安裝應用程序時，Sudo 非常有用。如需詳細資訊，請參閱 Sudo 手冊。	`sudo --version`
tar	tar 是一個命令行實用程序，您可以使用它將多個文件分組到單個存檔文件中（通常稱為 tarball）。如需詳細資訊，請參閱 GNU tar 文件。	`tar --version`
tmux	tmux 是終端多路復用器，您可以使用它在多個窗口中同時運行不同的程序。有關更多信息，請參閱提供 tmux 簡潔介紹的博客。	`tmux -V`
unzip	如需詳細資訊，請參閱壓縮/解壓縮。	
VIM	vim 是一個可自定義的編輯器，您可以通過基於文本的界面進行交互。如需詳細資訊，請參閱 vim.org 上提供的文件資源。	`vim --version`
wget	wget 是一種計算機程序，用於從命令行中指定的端點指定的 Web 服務器中檢索內容。如需詳細資訊，請參閱 GNU Wget 文件。	`wget --version`

名稱	描述	版本資訊

拉鍊/解壓	zip /解壓縮公用程序使用的存檔文件格式，提供無損數據壓縮而不會丟失數據。呼叫 zip 指令，將檔案分組並壓縮到單一歸檔中。使用 unzip 將檔案從歸檔解壓縮到指定的目錄中。	`unzip --version`
		
		`zip --version`

Docker	Docker 是用於開發，運輸和運行應用程序的開放平台。Docker 可讓您將應用程式與基礎架構分開，以便快速交付軟體。它使您可以在其中構建碼頭文件AWS CloudShell，並使用 CDK 構建碼頭資產。如需 Docker 支援哪些區域的資訊，請參閱碼頭區域。您應該注意，Docker 在環境中的空間有限。如果您擁有較大的單個圖像或過多預先存在的 Docker 映像，則可能會導致問題。如需 Docker 的詳細資訊，請參閱 D ocker 文件指南。	`docker --version`

安裝AWS CLI到您的主目錄

就像 CloudShell 環境中預先安裝的其他軟體一樣，此AWS CLI工具會自動更新排程的升級和安全性修補程式。如果您想確保您擁有的 up-to-date 版本最多AWS CLI，可以選擇在 shell 的主目錄中手動安裝此工具。

> **Important**
>
> 您需要在主目錄AWS CLI中手動安裝您的副本，以便在下次啟動 CloudShell 工作階段時可用。需要此安裝，因為在您完成 shell 階段作業之後，$HOME會刪除新增至以外目錄的檔案。此

外，安裝此副本之後AWS CLI，它不會自動更新。換句話說，您有責任管理更新和安全性修補程式。
如需AWS共用責任模型的詳細資訊，請參閱資料保護 AWS CloudShell。

安裝 AWS CLI

1. 在命 CloudShell 令行中，使用curl命令將AWS CLI已安裝的壓縮副本傳輸到 shell：

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

2. 解壓縮壓縮文件夾：

```
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，我們建議您同時更新PATH環境變數，以便在執行指aws令時不需要指定安裝工具的路徑：

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

> **Note**
>
> 如果將此變更還原為PATH，則依預設，不具有指定路徑的指aws令會使用預先安裝AWS CLI的版本。

在 shell 環境中安裝第三方軟件

> **Note**
>
> 建議您先檢閱「共用安全性責任模型」，然後再將任何協力廠商應用程式安裝至計算環境。AWS CloudShell

依預設，所有AWS CloudShell使用者都具有 sudo 權限。因此，您可以使用sudo命令來安裝 shell 運算環境中尚未提供的軟體。例如，您可以sudo與 DNF 套件管理公用程式搭配使用來安裝cowsay，該公用程式會產生含有訊息的母牛的 ASCII 藝術圖片：

```
sudo dnf install cowsay
```

然後，您可以通過鍵入啟動新安裝的程序echo "Welcome to AWS CloudShell" | cowsay。

> **Important**
>
> Package 管理公用程式，例如 DNF 安裝程式在目錄中 (例如)/usr/bin，這些公用程式會在 shell 工作階段結束時回收這些程式。這表示每個工作階段會安裝和使用額外的軟體。

使用指令碼修改您的殼層

如果您想要修改預設的 shell 環境，您可以編輯每次 shell 環境啟動時執行的 shell 指令碼。每當默認的 bash shell 啟動時，.bashrc腳本都會運行。

> **Warning**
>
> 如果您不正確地修改.bashrc檔案，之後可能無法存取您的 shell 環境。最好在編輯之前製作文件副本。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，您仍然可以登入另一個 shell，而且可以復原任何變更。
>
> 如果您在錯誤修改.bashrc或任何其他檔案之後失去存取權限，您可以刪除主目錄AWS CloudShell來回復其預設設定。

在此程序中，您將修改指.bashrc令碼，讓殼層環境自動切換為執行 Z 殼層。

1. .bashrc使用文本編輯器打開（例如 Vim）：

```
vim .bashrc
```

2. 在編輯器介面中，按 I 鍵開始編輯，然後新增以下內容：

```
zsh
```

3. 要退出並保存編輯過的.bashrc文件，請按Esc進入 Vim 命令模式並輸入以下內容：

```
:wq
```

4. 使用命source令重新載入.bashrc檔案：

```
source .bashrc
```

當指令行介面再次可用時，提示符號已變更%為，表示您現在正在使用 Z 殼層。

AWS CloudShell從 Amazon 2 遷移到 Amazon Linux 2023

AWS CloudShell這是基於 Amazon Linux 2（AL2），已遷移到 Amazon Linux 2023（AL2023）。有關 AL2023 的更多信息，請參閱 Amazon 2023 用戶指南中的什麼是 Amazon Linux 2023（AL2023）。

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

AL2023 對開發工具提供了一些改進，包括更新版本的軟件包，例如 Node .js 18 和 Python 3.9。

> **Note**
>
> 在 AL2023 中，Python2 不再隨您的 CloudShell環境一起出貨。

有關 AL2 和 AL2023 之間主要差異的更多信息，請參閱 Amazon 2023 用戶指南中的比較 Amazon Linux 2 和 Amazon Linux 2023。

如果您有任何問題，請聯繫AWS Support。您也可以在中搜尋答案並張貼問題AWS re:Post。輸入時 AWS re:Post，您可能需要登入AWS。

AWS CloudShell移轉問題

以下是有關從 AL2 遷移到 AL2023 的一些常見問題的答案。AWS CloudShell

此遷移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？

除了您的AWS CloudShell環境以外，其他任何服務或資源都不會受此移轉影響。這包括您可能從內部建立或存取的資源AWS CloudShell。例如，如果您已建立在 AL2 上執行的 Amazon EC2 執行個體，則不會移轉至 AL2023。

移轉至 AL2023 後，哪些套件已變更？

AWS CloudShell環境目前包括預安裝的軟體。若要瞭解預先安裝軟體的完整清單，請參閱預先安裝的軟體。 AWS CloudShell將繼續提供這些軟件包，但 Python 2 除外。有關 AL2 和 AL2023 提供的軟件包之間的完整差異，請參閱比較 AL2 和 AL2023。如果客戶的特定套件和版本需求在移轉至 AL2023 後將不再符合，我們建議您聯絡 Sup AWS port 部門以提交要求。

我可以選擇退出遷移嗎？

答案是否定的。 AWS CloudShell環境由管理AWS，因此，所有環境都已升級到 AL2023。

我可以建立AWS CloudShell環境的備份嗎？

AWS CloudShell將繼續保留用戶主目錄。如需詳細資訊，請參閱 AWS CloudShell. 如果您的主資料夾中儲存了任何檔案或組態，並且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。

AWS CloudShell 疑難排解

使用時AWS CloudShell，您可能會遇到問題，例如當您使用 shell 命令列介面啟動 CloudShell 或執行關鍵工作時。本章所涵蓋的資訊涵蓋如何疑難排解您可能會遇到的一些常見問題。

如需各種相關問題的解答 CloudShell，請參閱AWS CloudShell常見問題集。您也可以在AWS CloudShell討論區中搜尋答案並張貼問題。當您進入這個論壇時，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，可以使用下列解決方案來解決這些錯誤。

主題

無法啟動環境。若要重試，請重新整理瀏覽器，或選取「動作」、「重新啟動 AWS CloudShell

問題：當您嘗試AWS CloudShell從啟動時AWS Management Console，即使您具有 IAM 管理員所需的許可，並且已重新整理瀏覽器或重新啟動，您仍然會遭到拒絕存取 CloudShell。

解決方案：請聯絡 S AWSup port。

(回到頁首)

無法啟動環境。您沒有必要的權限。要求 IAM 管理員授予存取權 AWS CloudShell

問題：當您嘗試AWS CloudShell從啟動時AWS Management Console，系統會拒絕您存取，並通知您沒有必要的權限。

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 IAM 管理員為您提供必要的許可。他們可以透過新增附加的AWS受管理原則 (AWSCloudShellFullAccess) 或內嵌的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 存取和使用。

(回到頁首)

無法訪問AWS CloudShell命令行

問題：修改計算環境使用的檔案後，您無法存取中的命令列AWS CloudShell。

解決方案：如果您在錯誤修改`.bashrc`或任何其他檔案之後失去存取權，您可以刪除主目錄以回復AWS CloudShell其預設設定。

(回到頁首)

無法偵測外部 IP 位址

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

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

原因:ping 實用程序使用 Internet 控制消息協議（ICMP）將迴聲請求數據包發送到目標主機。它等待迴聲從目標回復。由於中未啟用 ICMP 通訊協定AWS CloudShell，因此 ping 公用程式不會在命令介面的運算環境中運作。

解決方案：由於中不支援 ICMPAWS CloudShell，您可以執行下列命令來安裝 Netcat。Netcat 是一種計算機網絡實用程序，用於使用 TCP 或 UDP 讀取和寫入網絡連接。

```
sudo yum install nc
nc -zv www.amazon.com 443
```

(回到頁首)

在準備您的終端機時出現一些問題

問題：嘗試AWS CloudShell使用 Microsoft Edge 瀏覽器存取時，您無法啟動殼層工作階段，而且瀏覽器會顯示錯誤訊息。

原AWS CloudShell因:與早期版本的 Microsoft 邊緣不兼容。您可以AWS CloudShell使用支持的瀏覽器的最新四個主要版本進行訪問。

解決方案：從 Microsoft 網站安裝邊緣瀏覽器的更新版本。

(回到頁首)

箭頭鍵無法正常工作 PowerShell

問題：在正常操作中，您可以使用箭頭鍵瀏覽命令行界面並向後和向前掃描命令歷史記錄。但是，當您在某些版本的 PowerShell on 中按方向鍵時AWS CloudShell，可能會錯誤地輸出字母。

原因:箭頭鍵輸出字母不正確的情況是 Linux 上運行的 PowerShell 7.2.x 版本的已知問題。

解法：若要去除修改方向鍵行為的逸出序列，請編輯設定 PowerShell 檔並將`$PSStyle`變數設定為`PlainText`。

1. 在命AWS CloudShell令行中，輸入以下命令以打開配置文件。

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

> **Note**
>
> 如果您已經在中 PowerShell，也可以使用以下命令在編輯器中打開配置文件。
>
> ```
> vim $PROFILE
> ```

2. 在編輯器中，轉到文件的現有文本的末尾，按 i 進入插入模式，然後添加以下語句。

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

3. 進行編輯後，按Esc進入指令模式。接下來，輸入以下命令以保存文件並退出編輯器。

```
:wq
```

> **Note**
>
> 您的變更會在您下次開始時生效 PowerShell。

(回到頁首)

不支援的 Web 通訊端會導致無法啟動 CloudShell 工作階段

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

原因： CloudShell 取決於WebSocket 協議，該協議允許您的 Web 瀏覽器和AWS CloudShell. 如果您在私人網絡中使用瀏覽器，則代理服務器和防火牆可能會促進對 Internet 的安全訪問。 WebSocket 通信通常可以遍歷代理服務器而沒有問題。但是，在某些情況下，代理服務器會阻 WebSockets 止正常工作。如果發生這個問題，就 CloudShell 無法啟動 shell 工作階段，而且連線的嘗試最終會逾時。

解決方案：連線逾時可能是因為不支援以外的問題所造成 WebSockets。如果是這種情況，請先重新整理 CloudShell 指令行介面所在的瀏覽器視窗。

如果重新整理後仍然收到逾時錯誤，請參閱 Proxy 伺服器的說明文件。並且，請確保您的代理服務器配置為允許 Web 套接字。或者，請聯絡您網路的系統管理員。

> **Note**
>
> 假設您想要透過允許列出特定 URL 來定義精細的權限。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指令會傳送至該 Systems Manager 工作階段。
> Systems Manager 使用的格式為 `wss://ssmmessages.region.amazonaws.com/v1/data-channel/session-id?stream=(input|output)`。 StreamUrl
> 該地區代表由支持AWS 區域的區域標識符AWS Systems Manager。例如，`us-east-2`是美國東部 (俄亥俄) 區域的區域識別碼。
> 因為工作階段識別碼是在特定 Systems Manager 工作階段順利啟動之後建立的，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 ManagerAPI 參考中的StartSession作業。

(回到頁首)

無法匯入AWSPowerShell.NetCore模組

問題：當您匯入 AWSPowerShell. NetCore模組中 PowerShell `Import-Module -Name AWSPowerShell.NetCore`，您會收到下列錯誤訊息：

導入模塊：指定的模塊 '. 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:該`AWSPowerShell.NetCore`模組已由. 中的每個服務 AWS .Tools 模組取代。AWS CloudShell

解決方案：可能不再需要任何明確的 import 陳述式，也可能需要變更為相關的每項服務 AWS .Tools 模組。

Example

Example

- 在大多數情況下，只要不使用 .NET 類型，就不需要任何明確的 import 語句。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AWS Tools for PowerShell.

(回到頁首)

使用時碼頭沒有運行 AWS CloudShell

問題：使AWS CloudShell用時 Docker 未正常運行。您收到下列錯誤訊息：`docker: Cannot connect to the Docker daemon at unix:///var/run/docker.sock. Is the docker daemon running?`。

解決方案：嘗試重新啟動環境。當您在不支援 Docker 的區域中AWS CloudShell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，如需哪些區域支援使用 Docker 容器的相關資訊AWS CloudShell，請參閱 Docker 區域。

碼頭工人已經用完了磁盤空間

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:碼頭文件超出了中的可用磁盤空間。AWS CloudShell這可能是由於單個圖像大或預先存在的 Docker 圖像過多而引起的。

解決方案：運行`df -h`以查找磁盤使用情況。運行`sudo du -sh /folder/folder1`以計算您認為可能很大的某些文件夾的大小，並考慮刪除其他文件以釋放空間。一種選擇是考慮通過運`docker rmi`行來刪除未使用的 Docker 映像。您應該注意 Docker 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`超時並繼續重試

問題：當您運行時，`docker push`它會超時並繼續重試，但沒有成功。

原因:這可能是由於缺少權限，推送到錯誤的存儲庫或缺少身份驗證引起的。

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存放庫。執行`docker login`以正確驗證。確保您擁有推送至 Amazon ECR 儲存庫的所有必要許可。

AWS CloudShell 支援的瀏覽器

下表會列出 AWS CloudShell 支援的瀏覽器。

Web 瀏覽器支援

瀏覽器	版本
Google Chrome	最近三個主要版本
Mozilla Firefox	最近三個主要版本
Microsoft Edge	最近三個主要版本
適用於 macOS 的 Apple Safari	最近兩個主要版本

支援的AWS地區 AWS CloudShell

本節涵蓋的支援地AWS區和選擇加入地區的清單。AWS CloudShell如需的AWS服務端點和配額清單CloudShell，請參閱中的AWS CloudShell頁面Amazon Web Services 一般參考。

以下是支援的AWS區域AWS CloudShell：

- 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)
- 亞太區域 (孟買)
- 亞太區域 (大阪)
- 亞太區域 (首爾)
- 亞太區域 (雪梨)
- 亞太區域 (新加坡)
- 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 (巴黎)
- 歐洲 (斯德哥爾摩)
- 南美洲 (聖保羅)

GovCloud 地區

以下是支援的 GovCloud 區域 CloudShell：

- AWS GovCloud (美國東部)
- AWS GovCloud (美國西部)

選擇加入區域

依預設未啟用選擇加入區域。您必須手動啟用這些區域才能使用它們。如需詳細資訊，請參閱管理 AWS 區域。以下是支援的選擇加入區域： CloudShell

- 非洲 (開普敦)
- 亞太區域 (香港)
- 亞太區域 (雅加達)
- 歐洲 (米蘭)
- Middle East (Bahrain)
- 中東 (阿拉伯聯合大公國)

泊塢視窗支援的區域

AWS CloudShell運算環境僅支援下列區域中的 Docker 容器：

- 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 (孟買)
- 亞太區域 (雪梨)
- 亞太區域 (新加坡)
- 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)
- Europe (Paris)
- 南美洲 (聖保羅)

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：

- 永久性儲存
- 每月使用量
- 指令大小
- 並發砲彈
- 殼層工作階
- 網路存取與資料傳輸
- 系統檔案和頁面重新載入

持久性儲存

使用AWS CloudShell，您可以免費獲得 1 GB 的持續性存儲空間。AWS 區域永久性儲存體位於您的主目錄 ($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環境資源不同，主目錄中的資料會在工作階段之間持續存在。

如果您停止AWS CloudShell在中使用AWS 區域，資料會在上次工作階段結束後保留在該區域的永久性儲存空間中 120 天。120 天後，除非您採取行動，否則系統會自動從該區域的永久性儲存區中刪除您的資料。您可以通過AWS CloudShell再次啟動來防止刪除AWS 區域。如需詳細資訊，請參閱步驟 2：選取區域、啟動 AWSCloudShell，然後選擇殼層。

> **Note**
>
> 使用情境
> Márcia 曾AWS CloudShell將檔案儲存在其主目錄中，分為兩種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) 使用AWS CloudShell，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 決定透過再次啟動AWS CloudShell並選取美國東部 (維吉尼亞北部) 區域來避免回收她的主目錄。由於她持續使用歐洲 (愛爾蘭) 進行 shell 工作階段，因此她在該地區的永久儲存空間不受影響。

每月使用量

您的.AWS 區域 中的每個人都有AWS CloudShell每月使用配額AWS 帳戶。如果您在達到該區域的每月配額AWS CloudShell後嘗試存取，則會顯示一則訊息，說明無法啟動 shell 環境的原因。

> **Note**
>
> 如果您需要增加每月用量配額，請聯絡 AWS Support。

指令大小

指令大小不能超過 65412 個字元。

> **Note**
>
> 如果您打算執行超過 65412 個字元的指令，請使用您選擇的語言建立指令碼，然後從指令行介面執行它。如需有關可從命令列介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。
>
> 若要查看如何建立指令碼，然後從命令列介面執行指令碼的範例，請參閱〈教學課程：入門〉AWS CloudShell。

並行砲彈

- 並發 shell：您的帳戶最多可以同時運行 10 個AWS 區域 shell。

殼層工作階

- 非作用中工作階段：AWS CloudShell是互動 shell 環境，如果您在 20—30 分鐘內沒有使用鍵盤或指標與其互動，則 shell 工作階段將結束。正在執行的進程不算作互動。
- 長時間執行的工作階段：即使使用者在該期間定期與其互動，連續執行約 12 小時的 shell 工作階段也會自動結束。

網路存取與資料傳輸

以下限制適用於您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。
- 入站：您無法存取入埠。沒有公有 IP 地址可用。

> **Warning**
>
> 存取公用網際網路後，某些使用者可能會從AWS CloudShell環境匯出資料的風險。建議 IAM 管理員透過 IAM 工具管理受信任AWS CloudShell使用者的允許清單。如需有關如何明確拒絕特定使用者存取的資訊，請參閱管理 AWS CloudShell 使用自訂策略中允許的動作。

資料傳輸：對於大型檔案而言，上傳和下載檔案AWS CloudShell可能會很慢。或者，您可以使用命令介面的命令列界面，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：如果您錯誤地修改了計算環境所需的檔案，則在存取或使用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獲得存取權。
- 重新載入頁面：若要重新載入AWS CloudShell介面，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順序。

AWS CloudShell 使用者指南的文件歷程記錄

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
新的自學課程已加入至AWS CloudShell使用者指南	已新增兩個新教學課程，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
特定AWS CloudShell區域支援的 Docker 容器	部分區域AWS CloudShell已新增對 Docker 容器的 Support 援。	2023 年 12 月 27 日
AWS CloudShell已遷移到現在使用 Amazon AL2023	AWS CloudShell現在使用 AL2023，並已從 Amazon Linux 2 遷移。	2023 年 12 月 4 日
適用於的新 AWS 區域 AWS CloudShell	AWS CloudShell現已在下列AWS地區正式推出： • 美國西部 (加利佛尼亞北部) • 非洲 (開普敦) • 亞太區域 (香港) • 亞太區域 (大阪) • 亞太區域 (首爾) • 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)	2023 年 6 月 16 日

	• Middle East (Bahrain) • 中東 (阿拉伯聯合大公國)	
AWS CloudShell在上啟動 Console Toolbar	選擇以Console Toolbar在主機左下方的啟動 CloudShell CloudShell。	2023 年 3 月 28 日
新AWS區域 AWS CloudShell	AWS CloudShell現在可在下列地AWS區使用： • 加拿大 (中部) • 歐洲 (倫敦) • 南美洲 (聖保羅)	2022 年 10 月 6 日
AWS CloudShell在美國 AWS 中支援 GovCloud	AWS CloudShellAWS GovCloud (美國) 區域現已支援。	2022 年 6 月 29 日
安全問題	其他常見問題集中在安全問題上。	2022 年 4 月 14 日
網絡插座	已新增說明使用 WebSocket通訊協定 CloudShell之網路需求的區段。	2022 年 3 月 21 日
疑難排解方向鍵 PowerShell	請按照以下步驟修復按下時不正確輸出字母的箭頭鍵。	2022 年 2 月 7 日
Tab 鍵自動完成	說明如何使用 bash-complete 的新文件，允許按 Tab 鍵自動完成部分類型的命令或引數。	2021 年 9 月 24 日
指定AWS區域	有關指定AWS CLI命令預設值 AWS 區域的文件。	2021 年 5 月 11 日
在 PDF 和點燃版本中進行格式化	修正表格儲存格中的影像大小和文字。	2021 年 3 月 10 日

選定AWS區域的正式推出 AWS CloudShell (GA) 版本	AWS CloudShell現已在下列AWS地區正式推出： • 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 (雪梨) • 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。