aws

使用者指南

AWS Elemental MediaStore

AWS Elemental MediaStore: 使用者指南

Table of Contents

什麼是 AWS Elemental MediaStore？

AWS Elemental MediaStore 是一種影片創作和儲存服務，提供即時創作所需的高效能和立即一致性。使用 MediaStore，您可以將視訊資產當做容器中的物件來管理，以建置可靠的雲端式媒體工作流程。

若要使用此服務，請將物件從來源 (例如編碼器或資料摘要) 上傳至您在 MediaStore 中建立的容器。

MediaStore 當您需要強大的一致性、低延遲的讀取和寫入，以及處理大量並行請求的能力時，是儲存分散視訊檔案的絕佳選擇。如果您不交付即時串流影片，請考慮改用 Amazon Simple Storage Service (Amazon S3)。

主題

AWS Elemental MediaStore 概念和術語

ARN

Amazon Resource Name。

Body

要上傳至物件中的資料。

(位元組) 範圍

要加以定址之物件資料的子集。如需詳細資訊，請參閱 HTTP 規格中的範圍。

容器

保存物件的命名空間。容器會有一個您可用於附加寫入和擷取物件以及附加存取政策的端點。

端點

MediaStore 服務的入口點，以 HTTPS 根 URL 的形式提供。

ETag

實體標籤，這是物件資料的雜湊值。

資料夾

容器的劃分區。資料夾可以保存物件及其他資料夾。

項目

用來指稱物件及資料夾的詞彙。

物件

一種資產，類似於 Amazon S3 對象。物件是存放在 MediaStore 中的基本實體。此服務接受所有檔案類型。

發送服務

MediaStore 被視為創建服務，因為它是媒體內容傳遞的發佈點。

路徑

物件或資料夾的唯一識別符，這可表示其在容器中的位置。

部分

物件的資料子集 (區塊)。

政策

IAM 政策。

資源

在 AWS 中您可以使用的實體。會為每個 AWS 資源指派 Amazon Resource Name (ARN)，此名稱可做為唯一識別符。在中 MediaStore，這是資源及其 ARN 格式：

- 容器：`aws:mediastore:region:account-id:container/:containerName`

相關服務

- Amazon CloudFront 是全球內容交付網路 (CDN) 服務，可將資料和影片安全地傳遞給觀眾。使用 CloudFront 以最佳的效能交付內容。如需詳細資訊，請參閱 Amazon CloudFront 開發人員指南。
- AWS CloudFormation 是一項服務，可幫助您來塑造模型及設定 AWS 資源。您可以建立一個範本來描述所AWS CloudFormation需的所有AWS資源 (例如 MediaStore 容器)，並為您處理佈建和設定這

些資源。您不需要個別建立及設定 AWS 資源，並費心思考依存性，AWS CloudFormation 能處理一切。如需詳細資訊，請參閱 AWS CloudFormation 使用者指南。

- AWS CloudTrail這是一項服務，可讓您監控對帳戶 CloudTrail API 發出的呼叫，包括 AWS 管理主控台的呼叫AWS CLI，以及其他服務。如需詳細資訊，請參閱 AWS CloudTrail 使用者指南。
- Amazon CloudWatch 是AWS雲端資源和您執行的應用程式的監控服務AWS。使用 CloudWatch 事件追蹤中容器和物件狀況的變更 MediaStore。如需詳細資訊，請參閱 Amazon CloudWatch 文件。
- AWS Identity and Access Management (IAM) 這個 Web 服務可讓您安全地控制使用者對 AWS 資源的存取。使用 IAM 控制誰可以使用 AWS 資源 (身分驗證)，以及使用者可以何種方式使用哪些資源 (授權)。如需詳細資訊，請參閱設定 AWS Elemental MediaStore。
- Amazon Simple Storage Service (Amazon S3) 是專為從任何位置存放和擷取任意數量資料而建立的物件儲存。如需詳細資訊，請參閱 Amazon S3 說明文件。

存取 AWS Elemental MediaStore

您可以使 MediaStore 用下列任何一種方法存取：

- AWS 管理主控台-本指南中的程序說明如何使用 AWS 管理主控台執行任務 MediaStore。若要使 MediaStore 用主控台存取：

```
https://<region>.console.aws.amazon.com/mediastore/home
```

- AWS Command Line Interface— 如需詳細資訊，請參閱《AWS Command Line Interface使用者指南》。若要使 MediaStore 用 CLI 端點存取：

```
aws mediastore
```

- MediaStore API — 如果您使用的是 SDK 無法使用的程式設計語言，請參閱 AWS Elemental MediaStoreAPI 參考，以取得有關 API 動作以及如何發出 API 請求的資訊。若要使 MediaStore 用 REST API 端點進行存取：

```
https://mediastore.<region>.amazonaws.com
```

- AWS 開發套件 如果您使用 AWS 提供之適用開發套件的程式設計語言，即可使用開發套件來存取 MediaStore。SDK 可簡化身分驗證、與您的開發環境輕鬆整合，並可輕鬆存取 MediaStore 命令。如需詳細資訊，請參閱 Amazon Web Services 適用工具。
- 適用於 Windows 的 AWS 工具 PowerShell — 如需詳細資訊，請參閱使AWS Tools for Windows PowerShell用者指南。

AWS Elemental 的定價 MediaStore

與其他AWS產品一樣，使用沒有合約或最低承諾 MediaStore。我們向您收取的費用僅限內容進入服務時的每 GB 提取費，以及您在服務中存放內容的每 GB 月費。如需詳細資訊，請參閱 AWS Elemental MediaStore 定價。

適用於 AWS Elemental 的區域和端點 MediaStore

為了減少應用程式中的資料延遲，請 MediaStore 提供區域端點來提出要求：

```
https://mediastore.<region>.amazonaws.com
```

若要檢視可用 AWS 區域的完整清單 MediaStore ，請參閱 AWS Elemental MediaStore 端點和 AWS 一般參考中的配額。

設定 AWS Elemental MediaStore

本節將引導您完成設定使用者以存取 AWS Elemental 所需的步驟 MediaStore。如需有關的身分識別與存取管理的背景和其他資訊 MediaStore，請參閱AWS Elemental Identity and Access Management MediaStore。

若要開始使用 AWS Elemental MediaStore，請完成以下步驟。

主題

註冊 AWS 帳戶

如果您還沒有 AWS 帳戶，請完成以下步驟建立新帳戶。

註冊 AWS 帳戶

1. 開啟 https://portal.aws.amazon.com/billing/signup。
2. 請遵循線上指示進行。

   部分註冊程序需接收來電，並在電話鍵盤輸入驗證碼。

   註冊 AWS 帳戶時，會建立 AWS 帳戶根使用者。根使用者有權存取該帳戶中的所有 AWS 服務和資源。作為最佳安全實務，將管理存取權指派給管理使用者，並且僅使用根使用者來執行需要根使用者存取權的任務。

註冊程序完成後，AWS 會傳送一封確認電子郵件給您。您可以隨時登錄 https://aws.amazon.com/ 並選擇 我的帳戶，以檢視您目前的帳戶活動並管理帳戶。

建立管理使用者

註冊後，請保護AWS 帳戶AWS 帳戶根使用者、啟用和建立系統管理使用者AWS IAM Identity Center，這樣您就不會將 root 使用者用於日常工作。

保護您的 AWS 帳戶根使用者

1. 選擇 根使用者 並輸入您的 AWS 帳戶電子郵件地址，以帳戶擁有者身分登入 AWS Management Console。在下一頁中，輸入您的密碼。

   如需使用根使用者登入的說明，請參閱 AWS 登入使用者指南中的以根使用者身分登入。
2. 若要在您的根使用者帳戶上啟用多重要素驗證 (MFA)。

   如需指示，請參閱《IAM 使用者指南》中的為 AWS 帳戶根使用者啟用虛擬 MFA 裝置 (主控台)。

建立管理使用者

1. 啟用 IAM 身分識別中心。

   如需指示，請參閱《AWS IAM Identity Center使用指南》AWS IAM Identity Center中的「啟用」。
2. 在 IAM 身分中心中，將管理存取權授與管理使用者。

   若要取得有關使用IAM Identity Center 目錄做為身分識別來源的自學課程，請參閱《使用指南》IAM Identity Center 目錄中的「以預設值設定使用AWS IAM Identity Center者存取」。

以管理員的身分登入

- 若要使用您的 IAM 身分中心使用者登入，請使用建立 IAM 身分中心使用者時傳送至您電子郵件地址的登入 URL。

  如需有關如何使用 IAM Identity Center 使用者登入的說明，請參閱《AWS 登入 使用者指南》中的登入 AWS存取入口網站。

AWS Elemental 入門 MediaStore

本入門教學課程說明如何使用 AWS Elemental MediaStore 建立容器和上傳物件。

主題

步驟 1：存取 AWS Elemental MediaStore

設定 AWS 帳戶並建立使用者和角色之後，您就可以登入 AWS Elemental 的主控台 MediaStore。

若要存取 AWS Elemental MediaStore

- 請登入AWS Management Console並開啟 MediaStore 主控台，網址為 https://console.aws.amazon.com/mediastore/。

> **Note**
>
> 您可以使用您已為此帳戶建立的任何 IAM 登入資料來登入。如需有關建立 IAM 登入資料的詳細資訊，請參閱設定 AWS Elemental MediaStore。

步驟 2：建立容器

您可以使用 AWS Elemental 中的容器 MediaStore 來存放資料夾和物件。您可以使用容器將相關物件分組，其方式與使用目錄將檔案系統中的檔案分組相同。建立容器時，您不需付費；只有在您上傳物件到容器時才付費。

若要建立容器

1. 在 Containers (容器) 頁面上，選擇 Create container (建立容器)。
2. 在 Container name (容器名稱) 中，輸入容器的名稱。如需詳細資訊，請參閱容器名稱規則。

3. 選擇 [建立容器]。AWS Elemental 會 MediaStore 將新容器新增至容器清單中。容器一開始的狀態會是 Creating (建立中)，然後變更為 Active (啟用)。

步驟 3：上傳物件

您可以將物件 (各物件最多 25 MB) 上傳至容器或容器中的資料夾。若要將物件上傳至資料夾，請指定資料夾的路徑。如果資料夾已存在，AWS Elemental 會將物件 MediaStore 儲存在資料夾中。如果資料夾不存在，則服務會建立此資料夾，再將物件存放在其中。

> **Note**
>
> 物件檔案名稱只能包含字母、數字、句點 (.)、底線 (_)、波狀符號 (~) 和連字號 (-)。

若要上傳物件

1. 在 Containers (容器) 頁面上，選擇您剛才建立的容器的名稱。容器的詳細資訊頁面隨即出現。
2. 選擇 Upload object (上傳物件)。
3. 在 Target path (目標路徑) 中，輸入資料夾的路徑。例如：`premium/canada`。如果路徑中的任何資料夾尚未存在，AWS Elemental 會自動 MediaStore 建立這些資料夾。
4. 針對 Object (物件) 選擇 Browse (瀏覽)。
5. 導覽至適當的資料夾，然後選擇一個物件來上傳。
6. 選擇 Open (開啟)，然後選擇 Upload (上傳)。

步驟 4：存取物件

您可以將物件下載到指定的端點。

1. 在 Containers (容器) 頁面上，選擇有您所要下載之物件的容器的名稱。
2. 如果您想要下載的物件是在某個子資料夾中，請繼續選擇資料夾名稱，直到您看到該物件為止。
3. 選擇物件的名稱。
4. 在物件的詳細資訊頁面上，選擇 Download (下載)。

AWS Elemental 中的容器MediaStore

您可以使用 MediaStore 中的容器來存放您的資料夾和物件。相關物件可以在容器中進行分組，其方式與使用目錄將檔案系統中的檔案分組相同。建立容器時，您不需付費；只有在您上傳物件到容器時才付費。如需費用的詳細資訊，請參AWS ElementalMediaStore定價。

主題

容器名稱規則

為容器選擇名稱時，請記得下列項目：

- 此名稱在目前帳戶的目前 AWS 區域中必須是唯一的。
- 此名稱可以包含大寫字母、小寫字母、數字和底線 (_)。
- 名稱長度必須介於 1 至 255 個字元。
- 名稱區分大小寫。例如，您可以有名為 `myContainer` 的容器以及名為 `mycontainer` 的資料夾，因為這些名稱是唯一的。
- 不可在建立容器後加以重新命名。

建立容器

您可以為每個 AWS 帳戶建立最多 100 個容器。您可以視需要建立任何數量的資料夾，只要容器中的巢狀層級不超過 10 層。此外，您可以將任意數量的物件上傳到每個容器。

> **Tip**
>
> 您也可以透過使用 AWS CloudFormation 範本來自動建立容器。AWS CloudFormation 範本可管理五個 API 動作的資料：建立容器、設定存取記錄、更新預設容器政策、新增跨來

源資源共享 (CORS) 政策，以及新增物件生命週期政策。如需詳細資訊，請參閱《AWS CloudFormation 使用者指南》。

若要建立容器 (主控台)

1. 開啟MediaStore主控台https://console.aws.amazon.com/mediastore/。
2. 在 Containers (容器) 頁面上，選擇 Create container (建立容器)。
3. 在 Container (容器) 名稱中，輸入容器的名稱。如需詳細資訊，請參閱 容器名稱規則。
4. 選擇建立容器。AWS ElementalMediaStore會將新的容器新增至容器清單。容器一開始的狀態會是 Creating (建立中)，然後變更為 Active (啟用)。

若要建立容器 (AWS CLI)

- 在 AWS CLI 中，使用 `create-container` 命令：

```
aws mediastore create-container --container-name ExampleContainer --region us-
west-2
```

以下範例顯示傳回值：

```
{
    "Container": {
        "AccessLoggingEnabled": false,
        "CreationTime": 1563557265.0,
        "Name": "ExampleContainer",
        "Status": "CREATING",
        "ARN": "arn:aws:mediastore:us-west-2:111122223333:container/
ExampleContainer"
    }
}
```

檢視容器的詳細資訊

容器的詳細資訊包括容器政策、端點、ARN 和建立時間。

若要檢視容器詳細資訊 (主控台)

1. 開啟MediaStore主控台https://console.aws.amazon.com/mediastore/。
2. 在 Containers (容器) 頁面上，選擇容器的名稱。

   容器詳細資訊頁面隨即出現。此頁面分為兩個區段：

   - Objects (物件) 區段，其中列出容器中的物件和資料夾。
   - Container policy (容器政策) 區段，其中顯示與此容器相關聯之依資源而定的政策。如需資源政策的相關資訊，請參閱容器政策。

若要檢視容器詳細資訊 (AWS CLI)

- 在 AWS CLI 中，使用 `describe-container` 命令：

```
aws mediastore describe-container --container-name ExampleContainer --region us-
west-2
```

以下範例顯示傳回值：

```
{
    "Container": {
        "CreationTime": 1563558086.0,
        "AccessLoggingEnabled": false,
        "ARN": "arn:aws:mediastore:us-west-2:111122223333:container/
ExampleContainer",
        "Status": "ACTIVE",
        "Name": "ExampleContainer",
        "Endpoint": "https://aaabbbcccdddee.data.mediastore.us-
west-2.amazonaws.com"
    }
}
```

檢視容器清單

您可以檢視所有與您帳戶相關之容器的清單。

若要檢視容器的清單 (主控台)

- 開啟MediaStore主控台https://console.aws.amazon.com/mediastore/。

  Containers (容器) 頁面會出現，列出所有與您帳戶相關聯的容器。

若要檢視容器的清單 (AWS CLI)

- 在 AWS CLI 中，使用 `list-containers` 命令。

```
aws mediastore list-containers --region us-west-2
```

以下範例顯示傳回值：

```
{
    "Containers": [
        {
            "CreationTime": 1505317931.0,
            "Endpoint": "https://aaabbbcccdddee.data.mediastore.us-
west-2.amazonaws.com",
            "Status": "ACTIVE",
            "ARN": "arn:aws:mediastore:us-west-2:111122223333:container/
ExampleLiveDemo",
            "AccessLoggingEnabled": false,
            "Name": "ExampleLiveDemo"
        },
        {
            "CreationTime": 1506528818.0,
            "Endpoint": "https://fffggghhhiiijj.data.mediastore.us-
west-2.amazonaws.com",
            "Status": "ACTIVE",
            "ARN": "arn:aws:mediastore:us-west-2:111122223333:container/
ExampleContainer",
            "AccessLoggingEnabled": false,
            "Name": "ExampleContainer"
        }
    ]
}
```

刪除容器

只有當容器沒有任何物件時，您才可以將該容器刪除。

若要刪除容器 (主控台)

1. 開啟MediaStore主控台https://console.aws.amazon.com/mediastore/。
2. 在 Containers (容器) 頁面上，選擇容器名稱左側的選項。
3. 選擇 Delete (刪除)。

若要刪除容器 (AWS CLI)

- 在 AWS CLI 中，使用 `delete-container` 命令：

  ```
  aws mediastore delete-container --container-name=ExampleLiveDemo --region us-west-2
  ```

  此命令沒有傳回值。

AWS Elemental 中的政策MediaStore

您可以將下列一或多個政策套用至 AWS ElementalMediaStore容器：

- 容器政策-設定容器內所有資料夾及物件的存取權利。MediaStore設置默認策略，允許用户執行所有MediaStore操作。此政策指定所有操作必須透過 HTTPS 執行。建立容器之後，您可以編輯容器政策。
- 跨來源資源分享 (CORS) 政策-允許一個網域中的用户端 Web 應用程式與不同網域中的資源互動。MediaStore未設置默認 CORS 策略。
- 指標政策-允許MediaStore將指標傳送至亞馬遜CloudWatch。MediaStore不會設定預設指標政策。
- 物件生命週期政策-控制物件保留在MediaStore容器。MediaStore不會設定預設物件生命週期政策。

AWS 元素中的容器策略MediaStore

每個容器都有依資源而定的政策，可管理該容器中所有資料夾及物件的存取權利。預設政策 (這已自動附加至所有新的容器) 允許存取所有 AWS ElementalMediaStore操作。政策會指定此存取有其必要條件，也就是必須使用 HTTPS 進行操作。建立容器後，您就可以編輯容器中連接的政策。

您也可以指定物件生命週期政策，此政策會在容器中管理物件的過期日期。物件達到您指定的年齡上限後，此服務就會將這些物件從容器中刪除。

主題

檢視容器政策

您可以使用主控台或 AWS CLI 來檢視容器中依資源而定的政策。

若要檢視容器政策 (主控台)

1. 開啟MediaStore主控台https://console.aws.amazon.com/mediastore/。
2. 在 Containers (容器) 頁面上，選擇容器名稱。

   容器詳細資訊頁面隨即出現。政策會顯示在 Container policy (容器政策) 區段中。

若要檢視容器政策 (AWS CLI)

- 在 AWS CLI 中，使用 `get-container-policy` 命令：

```
aws mediastore get-container-policy --container-name ExampleLiveDemo --region us-
west-2
```

以下範例顯示傳回值：

```
{
  "Policy": {
    "Version": "2012-10-17",
    "Statement": [
      {
        "Sid": "PublicReadOverHttps",
        "Effect": "Allow",
        "Principal": {
          "AWS": "arn:aws:iam::111122223333:root",
        },
        "Action": [
          "mediastore:GetObject",
          "mediastore:DescribeObject",
        ],
        "Resource": "arn:aws:mediastore:us-
west-2:111122223333:container/ExampleLiveDemo/*",
        "Condition": {
          "Bool": {
            "aws:SecureTransport": "true"
          }
        }
      }
    ]
  }
}
```

編輯容器政策

您可以編輯預設容器政策中的許可，也可以建立取代預設政策的新政策。新的政策需要最多 5 分鐘就會生效。

若要編輯容器政策 (主控台)

1. 開啟MediaStore主控台https://console.aws.amazon.com/mediastore/。
2. 在 Containers (容器) 頁面上，選擇容器名稱。
3. 選擇 Edit Policy (編輯政策)。如需說明如何設定不同許可的範例，請參閱the section called “範例容器政策”。
4. 進行適當變更，然後選擇 Save (儲存)。

若要編輯容器政策 (AWS CLI)

1. 建立檔案，此檔案會定義容器政策：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PublicReadOverHttps",
      "Effect": "Allow",
      "Action": ["mediastore:GetObject", "mediastore:DescribeObject"],
      "Principal": "*",
      "Resource": "arn:aws:mediastore:us-
west-2:111122223333:container/ExampleLiveDemo/*",
      "Condition": {
        "Bool": {
            "aws:SecureTransport": "true"
        }
      }
    }
  ]
}
```

2. 在 AWS CLI 中，使用 `put-container-policy` 命令：

```
aws mediastore put-container-policy --container-name ExampleLiveDemo --
policy file://ExampleContainerPolicy.json --region us-west-2
```

此命令沒有傳回值。

範例容器政策

下列範例示範針對不同使用者群組所建構的容器政策。

主題

範例容器政策：預設

創建容器時，AWS 元素MediaStore會自動附加下列依資源而定的政策：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "MediaStoreFullAccess",
      "Action": [ "mediastore:*" ],
      "Principal":{
          "AWS" : "arn:aws:iam::<aws_account_number>:root"},
      "Effect": "Allow",
      "Resource": "arn:aws:mediastore:<region>:<owner acct number>:container/<container
 name>/*",
      "Condition": {
        "Bool": { "aws:SecureTransport": "true" }
      }
    }
  ]
}
```

政策已內建於服務，因此您不需要建立。不過，您可以編輯政策，如果默認策略中的權限與您想要用於容器的權限不一致，則在容器上。

指派給所有新容器的預設政策允許對容器進行所有 MediaStore 操作存取。政策會指定此存取有其必要條件，也就是必須使用 HTTPS 進行操作。

範例容器政策：透過 HTTPS 的公有讀取存取

此範例政策允許使用者透過 HTTPS 請求擷取物件。此政策允許任何人透過安全的 SSL/TLS 連線進行讀取存取：驗證過身分的使用者以及匿名使用者 (未登入的使用者)。此陳述式的名稱為 PublicReadOverHttps。這允許對任何物件 (如資源路徑結尾的 * 所指定) 進行 GetObject 和 DescribeObject 操作存取。政策會指定此存取有其必要條件，也就是必須使用 HTTPS 進行操作：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PublicReadOverHttps",
      "Effect": "Allow",
      "Action": ["mediastore:GetObject", "mediastore:DescribeObject"],
      "Principal": "*",
      "Resource": "arn:aws:mediastore:<region>:<owner acct number>:container/<container
 name>/*",
      "Condition": {
        "Bool": {
            "aws:SecureTransport": "true"
        }
      }
    }
  ]
}
```

範例容器政策：透過 HTTP 或 HTTPS 的公有讀取存取

此範例政策允許對任何物件 (如資源路徑結尾的 * 所指定) 進行 GetObject 和 DescribeObject 操作存取。此政策允許任何人進行讀取存取，包括所有驗證過身分的使用者以及匿名使用者 (未登入的使用者)：

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

```
      "Action": ["mediastore:GetObject", "mediastore:DescribeObject"],
      "Principal": "*",
      "Resource": "arn:aws:mediastore:<region>:<owner acct number>:container/<container
 name>/*",
      "Condition": {
        "Bool": { "aws:SecureTransport": ["true", "false"] }
      }
    }
  ]
}
```

範例容器政策：跨帳戶讀取存取 - 啟用 HTTP

此範例政策允許使用者透過 HTTP 請求擷取物件。這允許具備跨帳戶存取權且驗證過身分的使用者進行此存取。物件不一定要在使用 SSL/TLS 憑證的伺服器上託管：

```
{
  "Version" : "2012-10-17",
  "Statement" : [ {
    "Sid" : "CrossAccountReadOverHttpOrHttps",
    "Effect" : "Allow",
    "Principal" : {
      "AWS" : "arn:aws:iam::<other acct number>:root"
    },
    "Action" : [ "mediastore:GetObject", "mediastore:DescribeObject" ],
    "Resource" : "arn:aws:mediastore:<region>:<owner acct number>:container/<container
 name>/*",
    "Condition" : {
      "Bool" : {
        "aws:SecureTransport" : [ "true", "false" ]
      }
    }
  } ]
}
```

範例容器政策：透過 HTTPS 的跨帳戶讀取存取

此範例政策允許存取`GetObject`和`DescribeObject`操作 (如資源路徑結尾的 * 所指定) 進行操作 (如資源路徑結尾的 * 所指定) 進行操作 <other acct number>。政策會指定此存取有其必要條件，也就是必須使用 HTTPS 進行操作：

```
{
```

```
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "CrossAccountReadOverHttps",
      "Effect": "Allow",
      "Action": ["mediastore:GetObject", "mediastore:DescribeObject"],
      "Principal":{
        "AWS": "arn:aws:iam::<other acct number>:root"},
      "Resource": "arn:aws:mediastore:<region>:<owner acct number>:container/<container
 name>/*",
      "Condition": {
        "Bool": {
            "aws:SecureTransport": "true"
        }
      }
    }
  ]
}
```

範例容器政策：角色的跨帳戶讀取存取

此範例政策允許對任何由 <擁有者帳號> 所擁有的物件 (如資源路徑結尾的 * 所指定) 進行 GetObject 和 DescribeObject 操作存取。這允許 <其他帳號> 的任何使用者進行此存取 (如果該帳戶有擔任 <角色名稱> 所指定的角色)。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "CrossAccountRoleRead",
      "Effect": "Allow",
      "Action": ["mediastore:GetObject", "mediastore:DescribeObject"],
      "Principal":{
        "AWS": "arn:aws:iam::<other acct number>:role/<role name>"},
      "Resource": "arn:aws:mediastore:<region>:<owner acct number>:container/<container
 name>/*",
    }
  ]
}
```

範例容器政策：角色的跨帳戶完整存取

此範例政策允許跨帳戶存取來更新帳戶中的任何物件，只要使用者是透過 HTTP 登入即可。這也允許對已擔任指定角色的帳戶進行跨帳戶存取，透過 HTTP 或 HTTPS 來刪除、下載和描述物件：

- 第一個陳述式是 CrossAccountRolePostOverHttps。這允許對任何物件進行 PutObject 操作存取，並允許指定之帳戶的任何使用者進行此存取 (如果該帳戶有擔任 <角色名稱> 所指定的角色)。政策會指定此存取有其必要條件，也就是必須使用 HTTPS 進行操作 (提供 PutObject 存取權時，務必要包含此條件)。

  換言之，任何擁有跨帳戶存取權的委託人都可以存取 PutObject，但只能透過 HTTPS 進行。

- 第二個陳述式是 CrossAccountFullAccessExceptPost。這允許對任何物件進行除了 PutObject 以外的所有操作存取。這允許指定之帳戶的任何使用者進行此存取 (如果該帳戶有擔任 <角色名稱> 所指定的角色)。此存取沒有限定條件要求使用 HTTPS 進行操作。

  換言之，任何具有跨帳戶存取權的帳戶都可以進行 DeleteObject、GetObject 等存取 (但 PutObject 除外)，而且可以透過 HTTP 或 HTTPS 進行此存取。

  如果沒有將 PutObject 從第二個陳述式排除，則陳述式不會有效 (因為要是包含 PutObject，您就必須依條件明確設定 HTTPS)。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "CrossAccountRolePostOverHttps",
      "Effect": "Allow",
      "Action": "mediastore:PutObject",
      "Principal":{
        "AWS": "arn:aws:iam::<other acct number>:role/<role name>"},
      "Resource": "arn:aws:mediastore:<region>:<owner acct number>:container/<container
 name>/*",
      "Condition": {
        "Bool": {
            "aws:SecureTransport": "true"
        }
      }
    },
    {
      "Sid": "CrossAccountFullAccessExceptPost",
```

```
      "Effect": "Allow",
      "NotAction": "mediastore:PutObject",
      "Principal":{
        "AWS": "arn:aws:iam::<other acct number>:role/<role name>"},
      "Resource": "arn:aws:mediastore:<region>:<owner acct number>:container/<container
 name>/*"
    }
  ]
}
```

範例容器政策：限制存取特定 IP 地址

此範例政策允許存取所有 AWS ElementalMediaStore操作對指定容器中的物件進行操作。不過，要求必須源自於條件中所指定的 IP 地址範圍。

此陳述式中的條件會識別允許之 Internet Protocol Version 4 (IPv4) IP 地址的 198.51.100.* 範圍，但有一個例外：一九八

Condition 區塊使用 IpAddress 與 NotIpAddress 條件以及 aws:SourceIp 條件鍵，其為整個 AWS 的條件鍵。aws:sourceIp IPv4 值會使用標準 CIDR 表示法。如需詳細資訊，請參閱「」IP 地址條件運算子在 IAM 使用者指南中的。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AccessBySpecificIPAddress",
            "Effect": "Allow",
            "Action": [
                "mediastore:GetObject",
                "mediastore:DescribeObject"
            ],
            "Principal": "*",
            "Resource": "arn:aws:mediastore:<region>:<owner acct number>:container/
<container name>/*",
            "Condition": {
                "IpAddress": {
                    "aws:SourceIp": [
                        "198.51.100.0/24"
                    ]
                },
                "NotIpAddress": {
                    "aws:SourceIp": "198.51.100.188/32"
```

```
                }
            }
        }
    ]
}
```

AWS Elemental 中的跨來源資源分享 (CORS) 政策MediaStore

跨來源資源分享 (CORS) 會定義一種方式，讓載入單一個網域的用戶端 Web 應用程式，能與不同網域中的資源互動。藉助 AWS 元素中的 CORS 支持MediaStore，您可以使用MediaStore，並有選擇地允許跨源訪問MediaStore的費用。

> **Note**
>
> 如果您正在使用亞馬遜CloudFront要從具有 CORS 策略的容器中分發內容，請務必配置 AWS 元素的分配MediaStore（包括編輯緩存行為以設置 CORS 的步驟）。

本節提供 CORS 的概觀。副主題會說明如何使用 AWS Elemental 來啟用 CORSMediaStore控制台，或以編程方式使用MediaStoreREST API 和 AWS 軟件開發工具包。

主題

- CORS 使用案例情境
- 將 CORS 政策新增至容器
- 檢視 CORS 政策
- 編輯 CORS 政策
- 刪除 CORS 政策
- 對 CORS 問題進行故障診斷
- 範例 CORS 政策

CORS 使用案例情境

以下為使用 CORS 的情境範例：

- 方案 1：假設您正在分發 AWS Elemental 中的即時串流視訊MediaStore容器名稱為LiveVideo。您的使用者從特定來源 (如 `http://livevideo.mediastore.ap-southeast-2.amazonaws.com`) 載入資訊清單端點 `www.example.com`。您想要使用JavaScript視訊播放器，透過未經驗證的方式

來存取源自此容器的視訊。GET和PUT請求。瀏覽器通常會阻止JavaScript，但您可以在容器上設定 CORS 政策，明確地啟用這些來自www.example.com。

- 方案 2：假設您想要託管案例 1 中那個同樣來自MediaStore容器，但希望允許來自任何源的請求。您可設定 CORS 政策以允許使用萬用字元 (*) 來源，使任何來源的要求都能存取該視訊。

將 CORS 政策新增至容器

本節說明如何將跨來源資源分享 (CORS) 組態新增至 AWS ElementalMediaStore容器。CORS 允許載入同一個網域的用戶端 Web 應用程式，可以與不同網域中的資源互動。

若要設定容器以允許跨來源要求，請將 CORS 政策新增至容器。CORS 政策會定義一些規則，這些規則可識別您所允許可存取容器的來源、每個來源支援的操作 (HTTP 方法)，以及其他操作特定資訊。

當您將 CORS 政策新增至容器時，容器政策 (管理容器存取權利的政策) 照常適用。

若要新增 CORS 政策 (主控台)

1. 開啟MediaStore主控台https://console.aws.amazon.com/mediastore/。
2. 在 Containers (容器) 頁面上，選擇您要為其建立 CORS 政策之容器的名稱。

   容器詳細資訊頁面隨即出現。
3. 在 Container CORS policy (容器 CORS 政策) 區段中，選擇 Create CORS policy (建立 CORS 政策)。
4. 依 JSON 格式插入政策，然後選擇 Save (儲存)。

若要新增 CORS 政策 (AWS CLI)

1. 建立檔案，此檔案會定義 CORS 政策：

```
[
  {
    "AllowedHeaders": [
      "*"
    ],
    "AllowedMethods": [
      "GET",
      "HEAD"
    ],
    "AllowedOrigins": [
```

```
      "*"
    ],
    "MaxAgeSeconds": 3000
  }
]
```

2. 在 AWS CLI 中，使用 `put-cors-policy` 命令。

```
aws mediastore put-cors-policy --container-name ExampleContainer --cors-policy
 file://corsPolicy.json --region us-west-2
```

此命令沒有傳回值。

檢視 CORS 政策

跨來源資源分享 (CORS) 會定義一種方式，讓載入單一個網域的用戶端 Web 應用程式，能與不同網域中的資源互動。

若要檢視 CORS 政策 (主控台)

1. 開啟MediaStore主控台https://console.aws.amazon.com/mediastore/。
2. 在 Containers (容器) 頁面上，選擇您要檢視其 CORS 政策之容器的名稱。

   容器詳細資訊頁面隨即出現，並將 CORS 政策顯示在 Container CORS policy (容器 CORS 政策) 區段中。

若要檢視 CORS 政策 (AWS CLI)

- 在 AWS CLI 中，使用 `get-cors-policy` 命令：

```
aws mediastore get-cors-policy --container-name ExampleContainer --region us-west-2
```

以下範例顯示傳回值：

```
{
    "CorsPolicy": [
        {
            "AllowedMethods": [
                "GET",
```

```
                "HEAD"
            ],
            "MaxAgeSeconds": 3000,
            "AllowedOrigins": [
                "*"
            ],
            "AllowedHeaders": [
                "*"
            ]
        }
    ]
}
```

編輯 CORS 政策

跨來源資源分享 (CORS) 會定義一種方式，讓載入單一個網域的用戶端 Web 應用程式，能與不同網域中的資源互動。

若要編輯 CORS 政策 (主控台)

1. 開啟MediaStore主控台https://console.aws.amazon.com/mediastore/。
2. 在 Containers (容器) 頁面上，選擇您要編輯其 CORS 政策之容器的名稱。

   容器詳細資訊頁面隨即出現。
3. 在 Container CORS policy (容器 CORS 政策) 區段中，選擇 Edit CORS policy (編輯 CORS 政策)。
4. 對政策進行變更，然後選擇 Save (儲存)。

若要編輯 CORS 政策 (AWS CLI)

1. 建立檔案，此檔案會定義更新的 CORS 政策：

```
[
  {
    "AllowedHeaders": [
      "*"
    ],
    "AllowedMethods": [
      "GET",
      "HEAD"
```

```
    ],
    "AllowedOrigins": [
      "https://www.example.com"
    ],
    "MaxAgeSeconds": 3000
  }
]
```

2. 在 AWS CLI 中，使用 `put-cors-policy` 命令。

```
aws mediastore put-cors-policy --container-name ExampleContainer --cors-policy
 file://corsPolicy2.json --region us-west-2
```

此命令沒有傳回值。

刪除 CORS 政策

跨來源資源分享 (CORS) 會定義一種方式，讓載入單一個網域的用戶端 Web 應用程式，能與不同網域中的資源互動。從容器刪除 CORS 政策會移除跨來源要求許可。

若要刪除 CORS 政策 (主控台)

1. 開啟MediaStore主控台https://console.aws.amazon.com/mediastore/。
2. 在 Containers (容器) 頁面上，選擇您要刪除其 CORS 政策之容器的名稱。

   容器詳細資訊頁面隨即出現。
3. 在 Container CORS policy (容器 CORS 政策) 區段中，選擇 Delete CORS policy (刪除 CORS 政策)。
4. 選擇 Continue (繼續) 來確認，然後選擇 Save (儲存)。

若要刪除 CORS 政策 (AWS CLI)

- 在 AWS CLI 中，使用 `delete-cors-policy` 命令：

```
aws mediastore delete-cors-policy --container-name ExampleContainer --region us-
west-2
```

此命令沒有傳回值。

對 CORS 問題進行故障診斷

如果您在存取具有 CORS 政策的容器時遇到非預期行為，請依照下列步驟，對問題進行疑難排解。

1. 確認已將 CORS 政策附加至容器。

   如需指示，請參閱 the section called “檢視 CORS 政策”。

2. 使用您選擇的工具 (例如，瀏覽器的開發人員主控台)，擷取完整的要求和回應。確認附加至容器的 CORS 政策至少包含一個比對您的要求中資料的 CORS 規則，如下所示：

   a. 確認要求含有 `Origin` 標頭。

      如果缺少標頭，AWS ElementalMediaStore不會將請求視為跨來源請求，也不會在回應中傳送 CORS 回應標頭。

   b. 確認要求中的 `Origin` 標頭至少與特定 `AllowedOrigins` 的其中一個 `CORSRule` 元素比對相符。

      `Origin` 要求標頭中的配置、主機及連接埠值，都必須符合 `AllowedOrigins` 中的 `CORSRule`。例如，若將 `CORSRule` 設定為允許來源 `http://www.example.com`，則要求中的 `https://www.example.com` 及 `http://www.example.com:80` 來源，都不符合您組態中允許的來源。

   c. 確認要求中的 Method (若為預檢要求，則是 `Access-Control-Request-Method` 中指定的方法) 是相同 `AllowedMethods` 的其中一個 `CORSRule` 元素。

   d. 若是預檢要求，如果要求包含 `Access-Control-Request-Headers` 標頭，則請確認 `CORSRule` 對於 `AllowedHeaders` 標頭中的每個值，都包含了 `Access-Control-Request-Headers` 項目。

範例 CORS 政策

下列範例示範跨來源資源分享 (CORS) 政策。

主題

- 範例 CORS 政策：所有網域皆可的讀取存取
- 範例 CORS 政策：特定網域方可的讀取存取

範例 CORS 政策：所有網域皆可的讀取存取

下列政策允許任何網域的網頁從 AWS ElementalMediaStore容器。要求包含所有來自原始網域的 HTTP 標頭，而且服務只會回應來自原始網域的 HTTP GET 和 HTTP HEAD 要求。系統先將結果暫存於快取達 3,000 秒之後，再傳送一組新的結果。

```
[
  {
    "AllowedHeaders": [
      "*"
    ],
    "AllowedMethods": [
      "GET",
      "HEAD"
    ],
    "AllowedOrigins": [
      "*"
    ],
    "MaxAgeSeconds": 3000
  }
]
```

範例 CORS 政策：特定網域方可的讀取存取

下列政策允許的網頁從https://www.example.com從您的 AWS 元素中檢索內容MediaStore容器。要求包含所有來自 https://www.example.com 的 HTTP 標頭，而且服務只會回應來自 https://www.example.com 的 HTTP GET 和 HTTP HEAD 要求。系統先將結果暫存於快取達 3,000 秒之後，再傳送一組新的結果。

```
[
  {
    "AllowedHeaders": [
      "*"
    ],
    "AllowedMethods": [
      "GET",
      "HEAD"
    ],
    "AllowedOrigins": [
      "https://www.example.com"
    ],
    "MaxAgeSeconds": 3000
```

```
  }
]
```

AWS 元素中的物件生命週期政策MediaStore

對於每個容器，您可以建立物件生命週期政策，此政策可管理物件在容器中應存放的時間。當物件達到您指定的年齡上限後，AWS 元素MediaStore就會刪除物件。您可以刪除不再需要的物件以節省儲存成本。

您也可以指定 MediaStore 在物件到達一定存留期後，應將物件移動到不常存取儲存 (IA) 儲存體方案。存放在 IA 儲存體方案中的物件，其儲存和擷取的費率會和存放在標準儲存體方案中的物件不同。如需詳細資訊，請參閱 MediaStore 定價。

物件生命週期政策會包含規則，這些規則會依子資料夾規定物件的生命週期。(您無法將物件生命週期政策指派至個別物件)。您可以只將一個物件生命週期政策連接到容器，但您可以在每個物件生命週期政策中新增最多 10 個規則。如需更多詳細資訊，請參閱 物件生命週期政策的元件。

主題

物件生命週期政策的元件

物件生命週期政策會管理物件在 AWS 元素中保留的時間MediaStore容器。物件生命週期政策由一或多個規則組成，這些規則會規定物件的生命週期。規則可套用到一個資料夾、多個資料夾或者整個容器。

您可以只將一個物件生命週期政策連接到容器，且每個物件生命週期政策可包含最多 10 個規則。您無法將物件生命週期政策指派至個別物件。

物件生命週期政策中的規則

您可以建立三種規則類型：

- 刪除物件
- 生命週期轉換

暫時性資料

暫時性資料規則會設定物件在幾秒內過期。這種規則類型僅適用於在政策生效後新增到容器的物件。MediaStore 最多需要 20 分鐘的時間，才能將新的政策套用到容器。

暫時性資料的規則範例如下所示：

```
        {
            "definition": {
                "path": [ {"wildcard": "Football/index*.m3u8"} ],
                "seconds_since_create": [
                    {"numeric": [">", 120]}
                ]
            },
            "action": "EXPIRE"
        },
```

暫時性資料規則有三個部分：

- `path`：一律設為 `wildcard`。您可以使用此部分來定義要刪除的物件。您可以使用一或多個萬用字元，以星號 (*) 表示。每個萬用字元代表零個或多個字元的任何組合。例如，`"path": [ {"wildcard": "Football/index*.m3u8"} ]`，會套用到 `Football` 資料夾中符合 `index*.m3u8` 模式 (例如 index.m3u8、index1.m3us8 和 index123456.m3u8) 的所有檔案。單一規則最多可以包含 10 個 路徑。
- `seconds_since_create`：一律設為 `numeric`。您可指定介於 1-300 秒之間的值。您也可以將運算子設定為大於 (>) 或大於或等於 (>=)。
- `action`：一律設為 `EXPIRE`。

針對暫時性資料規則 (物件會在幾秒內過期)，在物件過期與刪除物件之間沒有延遲。

> **Note**
>
> `list-items` 回應中不會包含受暫時性資料規則限制的物件。此外，由於臨時數據規則而過期的對象不會發出CloudWatch事件，當它們到期時。

刪除物件

刪除物件規則會將物件設定為在幾天內過期。這種規則類型適用於容器中的所有物件，即使它們在建立原則之前已新增至容器。MediaStore 最多需要 20 分鐘的時間才能套用新的政策，但物件最多可能需要 24 小時的時間才能從容器中清除。

刪除物件的兩個規則範例如下所示：

```
        {
            "definition": {
                "path": [ { "prefix": "FolderName/" }  ],
                "days_since_create": [
                    {"numeric": [">" , 5]}
                ]
            },
            "action": "EXPIRE"
        },
        {
            "definition": {
                "path": [ { "wildcard": "Football/*.ts" }  ],
                "days_since_create": [
                    {"numeric": [">" , 5]}
                ]
            },
            "action": "EXPIRE"
        }
```

刪除物件規則有三個部分：

- `path`：設為 `prefix` 或 `wildcard`。您不可在相同規則中混合使用 `prefix` 和 `wildcard`。如果要同時使用兩者，則必須為 `prefix` 建立一個規則，並為 `wildcard` 建立另一個規則，如上述範例所示。
  - `prefix` - 如果您想要刪除特定資料夾內的所有物件，可以將路徑設為 `prefix`。如果參數是空的 (`"path": [ { "prefix": "" } ],`)，目標就是存放在目前容器內任何位置的所有物件。單一規則最多可以包含 10 個 `prefix` 路徑。
  - `wildcard` - 如果您要根據檔案名稱及/或檔案類型刪除特定物件，則可將路徑設為 `wildcard`。您可以使用一或多個萬用字元，以星號 (*) 表示。每個萬用字元代表零個或多個字元的任何組合。例如，`"path": [ {"wildcard": "Football/*.ts"} ],` 適用於 `Football` 資料夾中符合 `*.ts` 模式的所有檔案 (例如 filename.ts、filename1.ts，以及 filename123456.ts)。單一規則最多可以包含 10 個 `wildcard` 路徑。

- days_since_create：一律設為 numeric。您可指定介於 1-36,500 天之間的值。您也可以將運算子設定為大於 (>) 或大於或等於 (>=)。
- action：一律設為 EXPIRE。

若為刪除物件規則 (物件會在幾天內過期)，在物件過期與刪除物件之間可能會有些許延遲。不過，只要物件過期，帳單也會立即變更。例如，如果生命週期規則指定 10 days_since_create，在物件達到第 10 天後，此帳戶就不會針對此物件收費，即使尚未刪除此物件。

生命週期轉換

生命週期轉換規則會將物件設為在到達一定存留期後移動到不常存取 (IA) 儲存體方案 (以天數計)。存放在 IA 儲存體方案中的物件，其儲存和擷取的費率會和存放在標準儲存體方案中的物件不同。如需詳細資訊，請參閱 MediaStore 定價。

一旦物件移動到 IA 儲存體方案，您就無法將其移回標準儲存體方案。

生命週期轉換規則適用於容器中的所有物件，即使它們在建立原則之前已新增至容器。MediaStore 最多需要 20 分鐘的時間才能套用新的政策，但物件最多可能需要 24 小時的時間才能從容器中清除。

生命週期轉移規則的範例如下所示：

```
{
    "definition": {
        "path": [
            {"prefix": "AwardsShow/"}
        ],
        "days_since_create": [
            {"numeric": [">=" , 30]}
        ]
    },
    "action": "ARCHIVE"
}
```

生命週期轉換規則有三個部分：

- path：設為 prefix 或 wildcard。您不可在相同規則中混合使用 prefix 和 wildcard。如果要同時使用兩者，您必須為 prefix 建立一個規則，並為 wildcard 建立單獨建立另一個規則。
  - prefix – 如果您希望將特定資料夾內的所有物件移動到 IA 儲存體方案，您可以將路徑設為 prefix。如果參數為空白 ("path": [{ "prefix": "" }],)，則目標就會是儲存在目前容器中任何位置的所有物件。單一規則最多可以包含 10 個 prefix 路徑。

- wildcard – 如果您希望根據檔案名稱和 (或) 檔案類型，將特定物件移動到 IA 儲存體方案，您可以將路徑設為 wildcard。您可以使用一或多個萬用字元，以星號 (*) 表示。每個萬用字元代表零個或多個字元的任何組合。例如，"path": [{"wildcard": "Football/*.ts"}], 適用於 Football 資料夾中符合 *.ts 模式的所有檔案 (例如 filename.ts、filename1.ts，以及 filename123456.ts)。單一規則最多可以包含 10 個 wildcard 路徑。
- days_since_create：一律設為 "numeric": [">=" , 30]。
- action：一律設為 ARCHIVE。

範例

例如，名為 LiveEvents 的容器有四個子資料夾：Football、Baseball、Basketball 和 AwardsShow。指派給 LiveEvents 資料夾的物件生命週期政策可能如下所示：

```
{
    "rules": [
         {
            "definition": {
                "path": [
                    {"prefix": "Football/"},
                    {"prefix": "Baseball/"}
                ],
                "days_since_create": [
                    {"numeric": [">" , 28]}
                ]
            },
            "action": "EXPIRE"
        },
        {
            "definition": {
                "path": [ { "prefix": "AwardsShow/" }  ],
                "days_since_create": [
                    {"numeric": [">=" , 15]}
                ]
            },
            "action": "EXPIRE"
        },
        {
            "definition": {
                "path": [ { "prefix": "" }  ],
                "days_since_create": [
                    {"numeric": [">" , 40]}
```

```
                ]
            },
            "action": "EXPIRE"
        },
        {
            "definition": {
                "path": [ { "wildcard": "Football/*.ts" }  ],
                "days_since_create": [
                    {"numeric": [">" , 20]}
                ]
            },
            "action": "EXPIRE"
        },
        {
            "definition": {
                "path": [
                    {"wildcard": "Football/index*.m3u8"}
                ],
                "seconds_since_create": [
                    {"numeric": [">" , 15]}
                ]
            },
            "action": "EXPIRE"
        },
        {
                "definition": {
                    "path": [
                        {"prefix": "Program/"}
                    ],
                    "days_since_create": [
                        {"numeric": [">=" , 30]}
                    ]
                },
                "action": "ARCHIVE"
            }
    ]
}
```

上述政策會指定以下項目：

- 第一條規則指示 AWS 元素MediaStore來刪除存放在`LiveEvents/Football`資料夾和`LiveEvents/Baseball`文件夾之後，它們超過 28 天。

- 第二個規則會指示此服務在 LiveEvents/AwardsShow 資料夾中存放的物件年齡為 15 天以上後，將其刪除。
- 第三個規則會指示此服務在 LiveEvents 容器中任何位置存放的物件年齡超過 40 天後，將其刪除。這個規則適用於直接在 LiveEvents 容器中存放的物件，以及在容器的四個子資料夾中任一項存放的物件。
- 第四個規則會服務在 Football 資料夾中符合模式 *.ts 的物件超過 20 天後，將其刪除。
- 第五條規則指示服務刪除Football與模式相符的資料夾index*.m3u8之後，他們超過 15 秒。MediaStore將這些文件放在容器中 16 秒後刪除這些文件。
- 第六個規則會指示服務在物件存留期超過 30 天後，將 Program 資料夾中的物件移動到 IA 儲存體方案。

如需物件生命週期政策的更多範例，請參閱 範例物件生命週期政策。

將物件生命週期政策新增至容器

物件生命週期政策可讓您指定要將物件存放在容器多久。您會設定過期日期，且在過期日期之後，AWS 元素MediaStore就會刪除物件。此服務最多需要 20 分鐘的時間，才能將新政策套用至容器。

如需有關如何建構生命週期政策的資訊，請參閱 物件生命週期政策的元件。

> **Note**
>
> 若為刪除物件規則 (物件會在幾天內過期)，在物件過期與刪除物件之間可能會有些許延遲。不過，只要物件過期，帳單也會立即變更。例如，如果生命週期規則指定 10 days_since_create，在物件達到第 10 天後，此帳戶就不會針對此物件收費，即使尚未刪除此物件。

新增物件生命週期政策 (主控台)

1. 開啟MediaStore主控台https://console.aws.amazon.com/mediastore/。
2. 在 Containers (容器) 頁面上，選擇您要為其建立物件生命週期政策之容器的名稱。

   容器詳細資訊頁面隨即出現。
3. 在 Object lifecycle policy (物件生命週期政策) 區段中，選擇 Create object lifecycle policy (建立物件生命週期政策)。

4. 依 JSON 格式插入政策，然後選擇 Save (儲存)。

新增物件生命週期政策 (AWS CLI)

1. 建立檔案，此檔案會定義物件生命週期政策：

```
{
    "rules": [
         {
            "definition": {
                "path": [
                    {"prefix": "Football/"},
                    {"prefix": "Baseball/"}
                ],
                "days_since_create": [
                    {"numeric": [">" , 28]}
                ]
            },
            "action": "EXPIRE"
        },
        {
            "definition": {
                "path": [
                    {"wildcard": "AwardsShow/index*.m3u8"}
                ],
                "seconds_since_create": [
                    {"numeric": [">" , 8]}
                ]
            },
            "action": "EXPIRE"
        }
    ]
}
```

2. 在 AWS CLI 中，使用 `put-lifecycle-policy` 命令：

```
aws mediastore put-lifecycle-policy --container-name LiveEvents --lifecycle-
policy file://LiveEventsLifecyclePolicy.json --region us-west-2
```

此命令沒有傳回值。此服務會將指定政策連接至容器。

檢視物件生命週期政策

物件生命週期政策會指定物件在容器中應保留的時間。

檢視物件生命週期政策 (主控台)

1. 開啟MediaStore主控台https://console.aws.amazon.com/mediastore/。
2. 在 Containers (容器) 頁面上，選擇您要檢視其物件生命週期政策之容器的名稱。

   容器詳細資訊頁面隨即出現，物件生命週期政策顯示在 Object lifecycle policy (物件生命週期政策) 區段中。

查看物件生命週期政策 (AWS CLI)

- 在 AWS CLI 中，使用 `get-lifecycle-policy` 命令：

```
aws mediastore get-lifecycle-policy --container-name LiveEvents --region us-west-2
```

以下範例顯示傳回值：

```
{
    "LifecyclePolicy": "{
        "rules": [
            {
                "definition": {
                    "path": [
                        {"prefix": "Football/"},
                        {"prefix": "Baseball/"}
                    ],
                    "days_since_create": [
                        {"numeric": [">" , 28]}
                    ]
                },
             "action": "EXPIRE"
            }
        ]
    }"
}
```

編輯物件生命週期政策

您無法編輯現有的物件生命週期政策。不過，您可以透過上傳取代政策來變更現有的政策。此服務最多需要 20 分鐘的時間，才能將更新的政策套用至容器。

編輯物件生命週期政策 (主控台)

1. 開啟MediaStore主控台https://console.aws.amazon.com/mediastore/。
2. 在 Containers (容器) 頁面上，選擇您要編輯其物件生命週期政策之容器的名稱。

   容器詳細資訊頁面隨即出現。
3. 在 Object lifecycle policy (物件生命週期政策) 區段中，選擇 Edit object lifecycle policy (編輯物件生命週期政策)。
4. 對政策進行變更，然後選擇 Save (儲存)。

編輯物件生命週期政策 (AWS CLI)

1. 建立檔案，此檔案會定義更新的物件生命週期政策：

```
{
    "rules": [
         {
            "definition": {
                "path": [
                    {"prefix": "Football/"},
                    {"prefix": "Baseball/"}
                    {"prefix": "Basketball/"}
                ],
                "days_since_create": [
                    {"numeric": [">" , 28]}
                ]
            },
            "action": "EXPIRE"
        }
    ]
}
```

2. 在 AWS CLI 中，使用 `put-lifecycle-policy` 命令：

```
aws mediastore put-lifecycle-policy --container-name LiveEvents --lifecycle-
policy file://LiveEvents2LifecyclePolicy --region us-west-2
```

此命令沒有傳回值。此服務會將指定政策連接至容器，取代先前的政策。

刪除物件生命週期政策

刪除物件生命週期政策時，此服務最多需要 20 分鐘的時間，才能將變更套用至容器。

刪除物件生命週期政策 (主控台)

1. 開啟MediaStore主控台https://console.aws.amazon.com/mediastore/。
2. 在 Containers (容器) 頁面上，選擇您要刪除其物件生命週期政策之容器的名稱。

   容器詳細資訊頁面隨即出現。
3. 在 Object lifecycle policy (物件生命週期政策) 區段中，選擇 Delete lifecycle policy (刪除生命週期政策)。
4. 選擇 Continue (繼續) 來確認，然後選擇 Save (儲存)。

刪除物件生命週期政策 (AWS CLI)

- 在 AWS CLI 中，使用 `delete-lifecycle-policy` 命令：

```
aws mediastore delete-lifecycle-policy --container-name LiveEvents --region us-
west-2
```

此命令沒有傳回值。

範例物件生命週期政策

下列範例顯示物件生命週期政策。

主題

- 範例物件生命週期政策：在幾秒鐘內過期
- 範例物件生命週期政策：在幾天內過期
- 範例物件生命週期政策：轉換至不常存取的存取體方案

- 範例物件生命週期政策：多重規則
- 範例物件生命週期政策：清空容器

範例物件生命週期政策：在幾秒鐘內過期

以下政策會指定 MediaStore 刪除符合下列所有條件的物件：

- 物件是在政策生效後新增到容器的。
- 物件存放在 Football 資料夾中。
- 物件具有副檔名 m3u8。
- 物件已在容器中超過 20 秒。

```
{
    "rules": [
        {
            "definition": {
                "path": [
                    {"wildcard": "Football/*.m3u8"}
                ],
                "seconds_since_create": [
                    {"numeric": [ ">", 20 ]}
                ]
            },
            "action": "EXPIRE"
        }
    ]
}
```

範例物件生命週期政策：在幾天內過期

以下政策會指定 MediaStore 刪除符合下列所有條件的物件：

- 物件存放在 Program 資料夾中
- 物件具有副檔名 ts
- 物件已在容器中超過 5 天

```
{
```

```
    "rules": [
        {
            "definition": {
                "path": [
                    {"wildcard": "Program/*.ts"}
                ],
                "days_since_create": [
                    {"numeric": [ ">", 5 ]}
                ]
            },
            "action": "EXPIRE"
        }
    ]
}
```

範例物件生命週期政策：轉換至不常存取的存取體方案

以下政策會指定 MediaStore 在物件存留期超過 30 天後，將物件移動到不常存取 (IA) 儲存體方案。存放在 IA 儲存體方案中的物件，其儲存和擷取的費率會和存放在標準儲存體方案中的物件不同。

`days_since_create` 欄位必須設為 `"numeric": [">=" ,30]`。

```
{
    "rules": [
        {
            "definition": {
                "path": [
                    {"prefix": "Football/"},
                    {"prefix": "Baseball/"}
                ],
                "days_since_create": [
                    {"numeric": [">=" , 30]}
                ]
            },
            "action": "ARCHIVE"
        }
    ]
}
```

範例物件生命週期政策：多重規則

以下政策會指定 MediaStore 執行下列作業：

- 在物件存留期達 30 天後，將存放在 AwardsShow 資料夾中的物件移動到不常存取 (IA) 儲存體方案。
- 刪除副檔名為 m3u8 並且存放在 Football 資料夾中超過 20 秒的物件。
- 刪除存放在 April 資料夾中超過 10 天的物件。
- 刪除副檔名為 ts，且存放在 Program 資料夾中超過 5 天的物件。

```
{
    "rules": [
        {
            "definition": {
                "path": [
                    {"prefix": "AwardsShow/"}
                ],
                "days_since_create": [
                    {"numeric": [ ">=" , 30 ]}
                ]
            },
            "action": "ARCHIVE"
        },
        {
            "definition": {
                "path": [
                    {"wildcard": "Football/*.m3u8"}
                ],
                "seconds_since_create": [
                    {"numeric": [ ">", 20 ]}
                ]
            },
            "action": "EXPIRE"
        },
        {
            "definition": {
                "path": [
                    {"prefix": "April"}
                ],
                "days_since_create": [
                    {"numeric": [ ">", 10 ]}
                ]
            },
            "action": "EXPIRE"
        },
```

```
        {
            "definition": {
                "path": [
                    {"wildcard": "Program/*.ts"}
                ],
                "days_since_create": [
                    {"numeric": [ ">", 5 ]}
                ]
            },
            "action": "EXPIRE"
        }
    ]
}
```

範例物件生命週期政策：清空容器

以下物件生命週期政策會指定 MediaStore 在物件新增到容器的 1 天後刪除容器內的所有物件 (包括資料夾和子資料夾)。如果容器在套用這項政策之前保留了任何物件，MediaStore會在政策生效 1 天後刪除物件。此服務最多需要 20 分鐘的時間，才能將新政策套用至容器。

```
{
    "rules": [
        {
            "definition": {
                "path": [
                    {"wildcard": "*"}
                ],
                "days_since_create": [
                    {"numeric": [ ">=", 1 ]}
                ]
            },
            "action": "EXPIRE"
        }
    ]
}
```

AWS Elemental 中的指標政策 MediaStore

對於每個容器，您可以新增指標政策，以允許 AWS Elemental 將指標傳送 MediaStore 到亞馬遜 CloudWatch。新的政策需要最多 20 分鐘就會生效。如需每個 MediaStore 量度的說明，請參閱MediaStore 度量。

指標政策包含下列項目：

- 在容器層級啟用或停用指標的設定。
- 從零到五之間任意數量的規則，用在物件層級啟用指標。如果政策包含規則，則每個規則都必須包含下列兩項：
  - 一個物件群組，定義要包括在群組中的物件。定義可以是路徑或檔案名稱，但不能超過 900 個字元。有效字元包括：a-z、A-Z、0-9、_ (底線)、= (等號)、: (冒號)、. (句號)、- (連字號)、~ (波浪號)、/ (正斜線) 和 * (星號)。可接受萬用字元 (*)。
  - 允許您參照物件群組的物件群組名稱。名稱不能超過 30 個字元。有效字元為 a-z、A-Z、0-9 和 _ (底線)。

如果物件符合多個規則， CloudWatch 會顯示每個相符規則的資料點。例如，如果一個物件符合兩個名為`rule1` AND 的規則`rule2`，則 CloudWatch 會顯示這些規則的兩個資料點。第一個具有 `ObjectGroupName=rule1` 的維度，第二個具有 `ObjectGroupName=rule2` 的維度。

主題

新增指標政策

指標政策包含指定 AWS Elemental MediaStore 傳送給亞馬遜的指標的規則 CloudWatch。如需指標政策的範例，請參閱指標政策範例。

新增指標政策 (主控台)

1. 開啟主 MediaStore 控台，網址為 https://console.aws.amazon.com/mediastore/。
2. 在 Containers (容器) 頁面上，選擇您要為其新增指標政策之容器的名稱。

   容器詳細資訊頁面隨即出現。
3. 在 Metric policy (指標政策) 區段中，選擇 Create metric policy (建立指標政策)。
4. 依 JSON 格式插入政策，然後選擇 Save (儲存)。

檢視指標政策

您可以使用主控台或 AWS CLI 來檢視容器的指標政策。

檢視指標政策 (主控台)

1. 開啟主 MediaStore 控台，網址為 https://console.aws.amazon.com/mediastore/。
2. 在 Containers (容器) 頁面上，選擇容器名稱。

   容器詳細資訊頁面隨即出現。政策會顯示在 Metric policy (指標政策) 區段中。

編輯指標政策

指標政策包含指定 AWS Elemental MediaStore 傳送給亞馬遜的指標的規則 CloudWatch。當您編輯現有的指標政策時，新政策需要長達 20 分鐘才會生效。如需指標政策的範例，請參閱指標政策範例。

編輯指標政策 (主控台)

1. 開啟主 MediaStore 控台，網址為 https://console.aws.amazon.com/mediastore/。
2. 在 Containers (容器) 頁面上，選擇容器名稱。
3. 在 Metric policy (指標政策) 區段中，選擇 Edit metric policy (編輯指標政策)。
4. 進行適當變更，然後選擇 Save (儲存)。

指標政策範例

下列範例示範針對不同使用案例所建構的指標政策。

主題

- 指標政策範例：容器層級指標
- 指標政策範例：路徑層級指標
- 指標政策範例：容器層級和路徑層級指標
- 指標政策範例：使用萬用字元的路徑層級指標
- 指標政策範例：具有重疊規則的路徑層級指標

指標政策範例：容器層級指標

此範例政策指出 AWS Elemental MediaStore 應該在容器層級將指標傳送 CloudWatch 至亞馬遜。例如，這包括 RequestCount 指標，此指標會計算對容器提出的 Put 請求數目。或者，您可以將其設定為 DISABLED。

由於此原則中沒有規則，因此 MediaStore 不會在路徑層級傳送量度。例如，您看不到對此容器內的特定資料夾提出的 Put 請求數。

```
{
   "ContainerLevelMetrics": "ENABLED"
}
```

指標政策範例：路徑層級指標

此範例政策指出 AWS Elemental 不 MediaStore 應在容器層級將指標傳送 CloudWatch 至亞馬遜。此外， MediaStore 應該為兩個特定資料夾中的物件傳送指標： baseball/saturday 和 football/saturday。 MediaStore 請求的指標如下：

- 對baseball/saturday資料夾的要求的 CloudWatch 維度為ObjectGroupName=baseballGroup。
- football/saturday 資料夾的請求具有維度 ObjectGroupName=footballGroup。

```
{
   "ContainerLevelMetrics": "DISABLED",
   "MetricPolicyRules": [
     {
       "ObjectGroup": "baseball/saturday",
       "ObjectGroupName": "baseballGroup"
     },
     {
       "ObjectGroup": "football/saturday",
       "ObjectGroupName": "footballGroup"
     }
   ]
}
```

指標政策範例：容器層級和路徑層級指標

此範例政策指出 AWS Elemental MediaStore 應該在容器層級將指標傳送 CloudWatch 至亞馬遜。此外， MediaStore 應該傳送兩個特定資料夾中物件的量度：`baseball/saturday`和`football/saturday`。 MediaStore 請求的指標如下：

- 對`baseball/saturday`資料夾的要求的 CloudWatch 維度為`ObjectGroupName=baseballGroup`。
- 對`football/saturday`資料夾的要求具有 CloudWatch 維度`ObjectGroupName=footballGroup`。

```
{
   "ContainerLevelMetrics": "ENABLED",
   "MetricPolicyRules": [
     {
       "ObjectGroup": "baseball/saturday",
       "ObjectGroupName": "baseballGroup"
     },
     {
       "ObjectGroup": "football/saturday",
       "ObjectGroupName": "footballGroup"
     }
   ]
}
```

指標政策範例：使用萬用字元的路徑層級指標

此範例政策指出 AWS Elemental MediaStore 應該在容器層級將指標傳送 CloudWatch 至亞馬遜。此外，還 MediaStore 應該根據物件的檔案名稱傳送物件的度量。萬用字元表示物件可以儲存在容器中的任何位置，而且它們可以有任何檔案名稱，只要它以 `.m3u8` 副檔名結尾即可。

```
{
   "ContainerLevelMetrics": "ENABLED",
   "MetricPolicyRules": [
     {
       "ObjectGroup": "*.m3u8",
       "ObjectGroupName": "index"
     }
   ]
 }
```

指標政策範例：具有重疊規則的路徑層級指標

此範例政策指出 AWS Elemental MediaStore 應該在容器層級將指標傳送 CloudWatch 至亞馬遜。此外， MediaStore 應該傳送兩個資料夾的量度：`sports/football/saturday`和`sports/football`。

`sports/football/saturday`資料夾 MediaStore 要求的量度具有的 CloudWatch 維度`ObjectGroupName=footballGroup1`。由於儲存在 `sports/football` 資料夾中的物件符合這兩個規則，因此 CloudWatch 會顯示這些物件的兩個資料點：一個具有 `ObjectGroupName=footballGroup2` 的維度，第二個具有 `ObjectGroupName=footballGroup1` 的維度。

```
{
   "ContainerLevelMetrics": "ENABLED",
   "MetricPolicyRules": [
     {
       "ObjectGroup": "sports/football/saturday",
       "ObjectGroupName": "footballGroup1"
     },
     {
       "ObjectGroup": "sports/football",
       "ObjectGroupName": "footballGroup2"
     }
   ]
 }
```

AWS Elemental 中的資料夾MediaStore

資料夾是容器內的劃分區。您使用資料夾細分容器，其方式與您建立子資料夾來劃分檔案系統中的資料夾相同。您可以建立最多 10 個層級的資料夾 (不包括容器本身)。

資料夾可選用；您可以選擇直接上傳物件給容器，而不是資料夾。不過，資料夾不失為組織物件的簡便方法。

若要將物件上傳至資料夾，請指定資料夾的路徑。如果資料夾已經存在，AWS ElementalMediaStore會將物件存放在資料夾中。如果資料夾不存在，則服務會建立此資料夾，再將物件存放在其中。

例如，假設您有一個名為`movies`，然後上傳一個名為`mlaw.ts`使用路徑`premium/canada`。AWS ElementalMediaStore將物件存放在 premium 資料夾底下的 canada 子資料夾中。如果這兩個資料夾都不存在，服務就會同時建立 `premium` 資料夾和 `canada` 子資料夾，然後將您的物件存放在 `canada` 子資料夾中。如果您只有指定 `movies` 容器 (沒有路徑)，服務則直接將物件存放在容器中。

AWS ElementalMediaStore當您刪除資料夾中的最後一個物件時， 會自動刪除該資料夾。服務也會刪除任何在該資料夾上層的空資料夾。例如，假設您有一個不含任何檔案、名為 premium 的資料夾，但其中包含一個名為 `canada` 的子資料夾。`canada` 子資料夾包含一個名為 `mlaw.ts` 的檔案。如果您刪除檔案 `mlaw.ts`，則服務會同時刪除 `premium` 和 `canada` 資料夾。此自動刪除動作僅適用於資料夾。服務並不會刪除空的容器。

主題

資料夾名稱規則

為資料夾選擇名稱時，請記得下列項目：

- 名稱只能包含以下字元：大寫字母 (A-Z)、小寫字母 (a-z)、數字 (0-9)、句點 (.)、連字號 (.)、連字號 (-)、波浪號 (~)、底線 (_)、等於符號 (=) 和冒號 (:)。
- 此名稱至少必須有一個字元。空文件夾名稱（ 如`folder1//folder3/` ）不允許。
- 名稱區分大小寫。例如，您在同一個容器或資料夾中可以有名為 `myFolder` 的資料夾以及名為 `myfolder` 的資料夾，因為這些名稱是唯一的。

- 此名稱在其父系容器或資料夾內必須是唯一的。例如，您可以在兩個不同的容器 (myfolder 和 movies/myfolder) 中各建立一個名為 sports/myfolder 的資料夾。
- 此名稱可以與其父容器相同。
- 不可在建立資料夾後加以重新命名。

建立資料夾

您可以在上傳物件時建立資料夾。若要將物件上傳至資料夾，請指定資料夾的路徑。如果資料夾已經存在，AWS ElementalMediaStore會將物件存放在資料夾中。如果資料夾不存在，則服務會建立此資料夾，再將物件存放在其中。

如需詳細資訊，請參閱 the section called “上傳物件”。

刪除資料夾

只有在資料夾是空的時，您才可以刪除資料夾；您無法刪除含有物件的資料夾。

AWS ElementalMediaStore當您刪除資料夾中的最後一個物件時， 會自動刪除該資料夾。服務也會刪除任何在該資料夾上層的空資料夾。例如，假設您有一個不含任何檔案、名為 premium 的資料夾，但其中包含一個名為 canada 的子資料夾。canada 子資料夾包含一個名為 mlaw.ts 的檔案。如果您刪除檔案 mlaw.ts，則服務會同時刪除 premium 和 canada 資料夾。此自動刪除動作僅適用於資料夾。服務並不會刪除空的容器。

如需詳細資訊，請參閱刪除物件。

AWS Elemental 中的物件MediaStore

AWS ElementalMediaStore資產即稱為對象。您可以將物件上傳至容器或容器中的資料夾。

在 MediaStore 中，您可以上傳、下載和刪除物件：

- 上傳 – 將物件新增至容器或資料夾。這與建立物件有所不同。您必須先在本機建立物件，然後才能將這些物件上傳至 MediaStore。
- 下載 – 將物件從 MediaStore 複製到其他位置。這不會從 MediaStore 移除物件。
- 刪除 – 從 MediaStore 完全移除物件。您可以個別刪除物件，或者可以新增物件生命週期政策，來自動在指定的持續時間後刪除容器中的物件。

MediaStore 接受所有檔案類型。

主題

上傳物件

您可以將物件上傳至容器，或容器中的資料夾。若要將物件上傳至資料夾，請指定資料夾的路徑。如果資料夾已經存在，AWS ElementalMediaStore將物件存放在資料夾中。如果資料夾不存在，則服務會建立此資料夾，再將物件存放在其中。如需資料夾的詳細資訊，請參閱AWS Elemental 中的資料夾MediaStore。

您可以使用 MediaStore 主控台或 AWS CLI 來上傳物件。

MediaStore 支援物件的區塊傳輸，這可透過將物件設為在上傳期間中仍可供下載來減少延遲。若要使用此功能，請將物件的上傳可用性設為 `streaming`。您可以在使用 API 上傳物件時，設定此標頭的值。如果您沒有在請求中指定此標頭，MediaStore 會指派預設值 `standard` 做為物件的上傳可用性。

標準上傳可用性的物件大小不可超過 25 MB，串流上傳可用性的物件大小限制為 10 MB。

> **Note**
>
> 物件資料名稱只能包含字母、數字、句點 (.)、底線 (_)、波狀符號 (~)、連字號 (-)、等號 (=) 和冒號 (:)。

若要上傳物件 (主控台)

1. 開啟MediaStore主控台https://console.aws.amazon.com/mediastore/。
2. 在 Containers (容器) 頁面上，選擇容器的名稱。容器的詳細資訊面板隨即出現。
3. 選擇 Upload object (上傳物件)。
4. 在 Target path (目標路徑) 中，輸入資料夾的路徑。例如： `premium/canada`。如果您所指定的路徑中有任何資料夾尚不存在，服務會自動建立該資料夾。
5. 在 Object (物件) 區段中，選擇 Browse (瀏覽)。
6. 導覽至適當的資料夾，然後選擇一個物件來上傳。
7. 選擇 Open (開啟)，然後選擇 Upload (上傳)。

> **Note**
>
> 如果名稱相同的檔案已經存在於選取的資料夾，服務會將原始檔案取代為上傳的檔案。

若要上傳物件 (AWS CLI)

- 在 AWS CLI 中，使用 `put-object` 命令。您也可以包含下列任何參數：`content-type`、`cache-control` (允許呼叫者控制物件的快取行為) 以及 `path` (將物件放在容器的某個資料夾中)。

> **Note**
>
> 上傳物件後，就無法編輯 `content-type`、`cache-control` 或 `path`。

```
aws mediastore-data put-object --endpoint https://
aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com --body README.md --path /
```

```
folder_name/README.md --cache-control "max-age=6, public" --content-type binary/
octet-stream --region us-west-2
```

以下範例顯示傳回值：

```
{
    "ContentSHA256":
 "74b5fdb517f423ed750ef214c44adfe2be36e37d861eafe9c842cbe1bf387a9d",
    "StorageClass": "TEMPORAL",
    "ETag": "af3e4731af032167a106015d1f2fe934e68b32ed1aa297a9e325f5c64979277b"
}
```

檢視物件的清單

您可以使用 AWS 元素MediaStore主控台來檢視存放在容器最上層或在資料夾中的項目 (物件和資料夾)。不會顯示存放在目前容器或資料夾之子資料夾中的項目。無論容器中有多少資料夾或子資料夾，您都可以使用 AWS CLI 來檢視容器中的物件及資料夾清單。

若要檢視特定容器中的物件清單 (主控台)

1. 開啟MediaStore主控台https://console.aws.amazon.com/mediastore/。
2. 在 Containers (容器) 頁面上，選擇有您所要檢視之資料夾的容器的名稱。
3. 從清單中選擇資料夾的名稱。

   詳細資訊頁面隨即出現，並顯示所有存放在該資料夾中的資料夾及物件。

若要檢視特定資料夾中的物件清單 (主控台)

1. 開啟MediaStore主控台https://console.aws.amazon.com/mediastore/。
2. 在 Containers (容器) 頁面上，選擇有您所要檢視之資料夾的容器的名稱。

   詳細資訊頁面隨即出現，並顯示所有存放在該容器中的資料夾及物件。

若要檢視特定容器中的物件及資料夾清單 (AWS CLI)

- 在 AWS CLI 中，使用 `list-items` 命令：

```
aws mediastore-data list-items --endpoint https://
aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com --region us-west-2
```

以下範例顯示傳回值：

```
{
    "Items": [
        {
            "ContentType": "image/jpeg",
            "LastModified": 1563571859.379,
            "Name": "filename.jpg",
            "Type": "OBJECT",
            "ETag":
 "543ab21abcd1a234ab123456a1a2b12345ab12abc12a1234abc1a2bc12345a12",
            "ContentLength": 3784
        },
        {
            "Type": "FOLDER",
            "Name": "ExampleLiveDemo"
        }
    ]
}
```

> **Note**
>
> 受 seconds_since_create 規則限制的物件不會包含在 list-items 回應中。

若要檢視特定資料夾中的物件及資料夾清單 (AWS CLI)

- 在 AWS CLI 中，使用 list-items 命令，並在請求的結尾包含指定的資料夾名稱：

```
aws mediastore-data list-items --endpoint https://
aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com --path /folder_name --
region us-west-2
```

以下範例顯示傳回值：

```
{
```

```
    "Items": [
        {
            "Type": "FOLDER",
            "Name": "folder_1"
        },
        {
            "LastModified": 1563571940.861,
            "ContentLength": 2307346,
            "Name": "file1234.jpg",
            "ETag":
 "111a1a22222a1a1a222abc333a444444b55ab1111ab2222222222ab333333a2b",
            "ContentType": "image/jpeg",
            "Type": "OBJECT"
        }
    ]
}
```

> **Note**
>
> 受 `seconds_since_create` 規則限制的物件不會包含在 `list-items` 回應中。

檢視物件的詳細資訊

上傳數據元後，AWS 元素MediaStore會存放一些詳細資訊，例如修改日期、內容長度、ETag (實體標籤) 和內容類型。若要了解如何使用物件的中繼資料，請參閱MediaStore與 HTTP 快取的互動。

若要檢視物件的詳細資訊 (主控台)

1. 開啟MediaStore主控台https://console.aws.amazon.com/mediastore/。
2. 在 Containers (容器) 頁面上，選擇有您所要檢視之物件的容器的名稱。
3. 如果您想要檢視的物件是在某個資料夾中，請繼續選擇資料夾名稱，直到您看到該物件為止。
4. 選擇物件的名稱。

   詳細資訊頁面隨即出現，並顯示物件的相關資訊。

若要檢視物件的詳細資訊 (AWS CLI)

- 在 AWS CLI 中，使用 `describe-object` 命令：

```
aws mediastore-data describe-object --endpoint https://
aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com --path /folder_name/
file1234.jpg --region us-west-2
```

以下範例顯示傳回值：

```
{
    "ContentType": "image/jpeg",
    "LastModified": "Fri, 19 Jul 2019 21:32:20 GMT",
    "ContentLength": "2307346",
    "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9eeeeee4dd89ff7f555555555555da6d3"
}
```

下載物件

您可以使用主控台來下載物件。您可以使用 AWS CLI 來下載一個物件，或是僅下載物件的一部分。

若要下載物件 (主控台)

1. 開啟MediaStore主控台https://console.aws.amazon.com/mediastore/。
2. 在 Containers (容器) 頁面上，選擇有您所要下載之物件的容器的名稱。
3. 如果您想要下載的物件是在某個資料夾中，請繼續選擇資料夾名稱，直到您看到該物件為止。
4. 選擇物件的名稱。
5. 在 Object details (物件詳細資訊) 頁面上，選擇 Download (下載)。

若要下載物件 (AWS CLI)

- 在 AWS CLI 中，使用 `get-object` 命令：

```
aws mediastore-data get-object --endpoint https://
aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com --path=/folder_name/
README.md README.md --region us-west-2
```

以下範例顯示傳回值：

```
{
    "ContentLength": "2307346",
```

```
    "ContentType": "image/jpeg",
    "LastModified": "Fri, 19 Jul 2019 21:32:20 GMT",
    "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9eeeeee4dd89ff7f555555555555da6d3",
    "StatusCode": 200
}
```

若要下載物件的一部分 (AWS CLI)

- 在 AWS CLI 中，使用 `get-object` 命令，並指定範圍。

```
aws mediastore-data get-object --endpoint https://
aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com --path /folder_name/
README.md --range="bytes=0-100" README2.md --region us-west-2
```

以下範例顯示傳回值：

```
{
    "StatusCode": 206,
    "ContentRange": "bytes 0-100/2307346",
    "ContentLength": "101",
    "LastModified": "Fri, 19 Jul 2019 21:32:20 GMT",
    "ContentType": "image/jpeg",
    "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9eeeeee4dd89ff7f555555555555da6d3"
}
```

刪除物件

AWS ElementalMediaStore提供了從容器中刪除物件的不同選項：

- 刪除個別物件。無需支付費用。
- 清空容器以一次刪除容器內的所有物件。因為此程序使用 API 呼叫，所以會收取一般 API 費用。
- 新增物件生命週期政策，以便在物件達到特定存留期時刪除物件。無需支付費用。

刪除物件

您可以使用主控台或 AWS CLI 來個別刪除物件。或者，您可以新增物件生命週期政策，在物件達到容器中的特定存留期後自動刪除物件，或清空容器以刪除該容器內的所有物件。

> **Note**
>
> 當您刪除資料夾中的唯一物件時，AWS ElementalMediaStore會自動刪除此資料夾以及任何在該資料夾上層的空資料夾。例如，假設您有一個不含任何檔案、名為 `premium` 的資料夾，但其中包含一個名為 `canada` 的子資料夾。`canada` 子資料夾包含一個名為 `mlaw.ts` 的檔案。如果您刪除檔案 `mlaw.ts`，則服務會同時刪除 `premium` 和 `canada` 資料夾。

若要刪除物件 (主控台)

1. 開啟MediaStore主控台https://console.aws.amazon.com/mediastore/。
2. 在 Containers (容器) 頁面上，選擇具備您要刪除之物件的容器名稱。
3. 如果您想要刪除的物件是在某個資料夾中，請繼續選擇資料夾名稱，直到您看到該物件為止。
4. 選擇物件名稱左側的選項。
5. 選擇 Delete (刪除)。

若要刪除物件 (AWS CLI)

- 在 AWS CLI 中，使用 `delete-object` 命令。

  範例：

  ```
  aws mediastore-data --region us-west-2 delete-object --endpoint=https://
  aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com --path=/folder_name/
  README.md
  ```

  此命令沒有傳回值。

清空容器

您可以清空容器，以刪除儲存在容器內的所有物件。或者，您可以新增物件生命週期政策，在物件於容器中存在超過特定存留期後自動刪除物件，或者您可以個別刪除物件。

清空容器 (主控台)

1. 開啟MediaStore主控台https://console.aws.amazon.com/mediastore/。
2. 在 Containers (容器) 頁面上，為您要清空的容器選擇選項。

3. 選擇 Empty container (清空容器)。會出現確認訊息。
4. 在文字欄位中輸入容器名稱以確認您要清空的容器，然後選擇空白。

AWS Elemental 中的安全性 MediaStore

雲端安全是 AWS 最重視的一環。身為 AWS 的客戶，您將能從資料中心和網路架構中獲益，這些都是專為最重視安全的組織而設計的。

安全性是 AWS 與您共同肩負的責任。共同的責任模式將其稱為雲端的安全性和雲端中的安全性：

- 雲端本身的安全 – AWS 負責保護執行 AWS 雲端 內 AWS 服務的基礎設施。AWS 提供的服務，也可讓您安全使用。第三方稽核人員會定期測試和驗證我們安全性的有效性，作為 AWS 合規計畫的一部分。要了解適用於 AWS Elemental 的合規計劃 MediaStore，請參閱AWS合規計劃的合規計劃AWS服務範的服務。
- 雲端內部的安全 – 您的責任取決於所使用的 AWS 服務。您也必須對其他因素負責，包括資料的機密性、您公司的請求和適用法律和法規。

本文件可協助您瞭解如何在使用時套用共同責任模型 MediaStore。下列主題說明如何設定 MediaStore 以符合安全性與合規性目標。您也會學到如何使用其他可協助您監控和保護 MediaStore 資源的AWS服務。

主題

- AWS Elemental 中的資料保護 MediaStore
- AWS Elemental Identity and Access Management MediaStore
- AWS Elemental MediaStore 中的日誌記錄和監控
- AWS Elemental 的合規驗證 MediaStore
- AWS Elemental 的韌性 MediaStore
- AWS Elemental 中的基礎設施安全 MediaStore
- 預防跨服務混淆代理人

AWS Elemental 中的資料保護 MediaStore

AWS共同責任模型適用於 AWS Elemental 中的資料保護 MediaStore。如此模型所述，AWS 負責保護執行所有 AWS 雲端 的全球基礎設施。您負責維護在此基礎設施上託管內容的控制權。您也必須負責您所使用 AWS 服務 的安全組態和管理任務。如需有關資料隱私權的更多相關資訊，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，建議您使用 AWS IAM Identity Center 或 AWS Identity and Access Management (IAM) 保護 AWS 帳戶 憑證，並設定個人使用者。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 AWS CloudTrail 設定 API 和使用者活動日誌記錄。
- 使用 AWS 加密解決方案，以及 AWS 服務 內的所有預設安全控制項。
- 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 AWS 時，需要 FIPS 140-2 驗證的加密模組，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) 140-2 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 Name (名稱) 欄位。這包括當您使用主控台、API MediaStore 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

MediaStore 使用業界標準 AES-256 演算法加密靜態容器和物件。我們建議您通過以下方式使用 MediaStore 來保護您的數據：

- 建立容器原則以控制該容器中所有資料夾和物件的存取權限。如需詳細資訊，請參閱the section called “容器政策”。
- 建立跨來源資源共用 (CORS) 原則，以允許選擇性地跨來源存取您的資源。 MediaStore 您可以使用 CORS，允許在一個網域載入的用戶端 Web 應用程式與不同網域中的資源互動。如需詳細資訊，請參閱the section called “CORS 政策”。

AWS Elemental Identity and Access Management MediaStore

AWS Identity and Access Management (IAM) 是一種 AWS 服務，讓管理員能夠安全地控制對 AWS 資源的存取權。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 MediaStore 資源。IAM 是一種您可以免費使用的 AWS 服務。

主題

對象

AWS Identity and Access Management (IAM) 的使用方式會不同，需視您在 MediaStore 中所執行的工作而定。

服務使用者 — 如果您使用 MediaStore 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 MediaStore 功能來完成工作時，您可能需要其他權限。瞭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 MediaStore 中的某項功能，請參閱 AWS Elemental MediaStore 身分和存取的疑難排解。

服務管理員 — 如果您負責公司的 MediaStore 資源，您可能擁有完整的存取權 MediaStore。決定您的服務使用者應該存取哪些 MediaStore 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 MediaStore，請參閱AWS Elemental 如何與 IAM MediaStore 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 MediaStore 存取權的詳細資訊。若要檢視可在 IAM 中使用的 MediaStore 基於身分的政策範例，請參閱。AWS Elemental 於身份的政策範例 MediaStore

使用身分驗證

身分驗證是使用身分憑證登入 AWS 的方式。您必須以 AWS 帳戶根使用者、IAM 使用者身分，或擔任 IAM 角色進行 驗證（登入至 AWS）。

您可以使用透過身分來源 AWS IAM Identity Center 提供的憑證，以聯合身分登入 AWS。(IAM Identity Center) 使用者、貴公司的單一登入身分驗證和您的 Google 或 Facebook 憑證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。您 AWS 藉由使用聯合進行存取時，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。如需有關登入至 AWS 的詳細資訊，請參閱《AWS 登入 使用者指南》中的如何登入您的 AWS 帳戶。

如果您是以程式設計的方式存取 AWS，AWS 提供軟體開發套件 (SDK) 和命令列介面 (CLI)，以便使用您的憑證透過密碼編譯方式簽署您的請求。如果您不使用 AWS 工具，您必須自行簽署請求。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) 以提高帳戶的安全。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》中的在 AWS 中使用多重要素驗證 (MFA)。

AWS 帳戶 根使用者

如果是建立 AWS 帳戶，您會先有一個登入身分，可以完整存取帳戶中所有 AWS 服務 與資源。此身分稱為 AWS 帳戶 根使用者，使用建立帳戶時所使用的電子郵件地址和密碼即可登入並存取。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

最佳實務是要求人類使用者（包括需要管理員存取權的使用者）搭配身分提供者使用聯合功能，使用暫時憑證來存取 AWS 服務。

聯合身分是來自您企業使用者目錄的使用者、Web 身分供應商、AWS Directory Service、Identity Center 目錄或透過身分來源提供的憑證來存取 AWS 服務 的任何使用者。聯合身分存取 AWS 帳戶時，會擔任角色，並由角色提供暫時憑證。

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來源中的一組使用者和群組，以便在您的所有 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 AWS 帳戶 中的一種身分，具備單一人員或應用程式的特定許可。建議您盡可能依賴暫時憑證，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。如需詳細資訊，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。如需進一步了解，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 中的一種身分，具備特定許可。它類似 IAM 使用者，但不與特定的人員相關聯。您可以在 AWS Management Console 中透過切換角色來暫時取得 IAM 角色。您可以透過呼叫 AWS CLI 或 AWS API 操作，或是使用自訂 URL 來取得角色。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 — 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，並取得由角色定義的許可。如需有關聯合角色的詳細資訊，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。
- 跨帳戶存取權 – 您可以使用 IAM 角色，允許不同帳戶中的某人（信任的委託人）存取您帳戶中的資源。角色是授予跨帳戶存取權的主要方式。但是，針對某些 AWS 服務，您可以將政策直接連接到資源（而非使用角色作為代理）。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 角色與資源類型政策的差異。
- 跨服務存取 – 有些 AWS 服務 會使用其他 AWS 服務 中的功能。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。
  - 轉發存取工作階段 (FAS)：當您使用 IAM 使用者或角色在 AWS 中執行動作時，系統會將您視為主體。當您使用某些服務時，您可能會執行一個動作，而該動作之後會在不同的服務中啟動另一個動作。FAS 使用主體的許可呼叫 AWS 服務，搭配請求 AWS 服務 以向下游服務發

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之後，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》https://docs.aws.amazon.com/IAM/latest/UserGuide/access_forward_access_sessions.html。

- 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。
- 服務連結角色 – 服務連結角色是一種連結到 AWS 服務 的服務角色類型。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 AWS 帳戶 中，並由該服務所擁有。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 – 針對在 EC2 執行個體上執行並提出 AWS CLI 和 AWS API 請求的應用程式，您可以使用 IAM 角色來管理暫時憑證。這是在 EC2 執行個體內儲存存取金鑰的較好方式。如需指派 AWS 角色給 EC2 執行個體並提供其所有應用程式使用，您可以建立連接到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暫時憑證。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。

如需了解是否要使用 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

您可以透過建立政策並將其附加到 AWS 身分或資源，在 AWS 中控制存取。政策是 AWS 中的一個物件，當其和身分或資源建立關聯時，便可定義其許可。AWS 會在主體（使用者、根使用者或角色工作階段）發出請求時評估這些政策。政策中的許可，決定是否允許或拒絕請求。大部分政策以 JSON 文件形式儲存在 AWS 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。若要授與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。具備該政策的使用者便可以從 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。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。受管政策則是獨立的政策，您可以將這些政策附加到 AWS 帳戶 中的多個使用者、群組和角色。受管政策包含 AWS 管理政策和客戶管理政策。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。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。主體可以包括帳戶、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 Simple Storage Service (Amazon S3)、AWS WAF 和 Amazon VPC 是支援 ACL 的服務範例。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 支援其他較少見的政策類型。這些政策類型可設定較常見政策類型授與您的最大許可。

- 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。如需許可範圍的更多相關資訊，請參閱《IAM 使用者指南》中的 IAM 實體許可範圍。
- 服務控制政策 (SCP) – SCP 是 JSON 政策，可指定 AWS Organizations 中組織或組織單位 (OU) 的最大許可。AWS Organizations 服務可用來分組和集中管理您企業所擁有的多個 AWS 帳戶。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 AWS 帳戶根使用者。如需組織和 SCP 的更多相關資訊，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。如需更多資訊，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。如需瞭解 AWS 在涉及多種政策類型時如何判斷是否允許一項請求，請參閱 IAM 使用者指南中的政策評估邏輯。

AWS Elemental 如何與 IAM MediaStore 搭配使用

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

您可以搭配 AWS Elemental 使用的 IAM 功能 MediaStore

IAM 功能	MediaStore 支持
身分型政策	是
資源型政策	是
政策動作	是
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)	部分
臨時憑證	是
主體許可	是

IAM 功能	MediaStore 支持
服務角色	是
服務連結角色	否

若要深入瞭解如何以 MediaStore 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 MediaStore

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。若要瞭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。

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，因為這會套用至附加的使用者或角色。如要瞭解您在 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 MediaStore

若要檢視以 MediaStore 身分為基礎的原則範例，請參閱。AWS Elemental 於身份的政策範例 MediaStore

以資源為基礎的政策 MediaStore

支援以資源基礎的政策	是

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。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。主體可以包括帳戶、使用者、角色、聯合身分使用者或 AWS 服務。

若要啟用跨帳戶存取權，您可以指定在其他帳戶內的所有帳戶或 IAM 實體，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。當主體和資源在不同的 AWS 帳戶 中時，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) 存取資源的許可。其透過將身分型政策附加到實體來授予許可。不過，如果資源型政策會為相同帳戶中的主體授與存取，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。

> **Note**
>
> MediaStore 也支援容器原則，這些原則可定義哪些主參與者實體 (帳戶、使用者、角色和同盟使用者) 可以對容器執行動作。如需詳細資訊，請參閱容器政策。

的政策動作 MediaStore

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。政策動作的名稱通常會和相關聯的 AWS API 操作相同。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 MediaStore 動作清單，請參閱服務授權參考 MediaStore中的 AWS Elemental 定義的動作。

中的策略動作在動作之前 MediaStore 使用下列前置詞：

```
mediastore
```

如需在單一陳述式中指定多個動作，請用逗號分隔。

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

若要檢視以 MediaStore 身分為基礎的原則範例，請參閱。AWS Elemental 於身份的政策範例 MediaStore

的政策資源 MediaStore

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 MediaStore 資源類型及其 ARN 的清單，請參閱服務授權參考資料 MediaStore中的 AWS Elemental 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 Elemental MediaStore 定義的動作。

MediaStore 容器資源具有以下 ARN：

```
arn:${Partition}:mediastore:${Region}:${Account}:container/${containerName}
```

如需 ARN 格式的詳細資訊，請參閱 Amazon Resource Name (ARN) 和 AWS 服務命名空間。

例如，若要在您的陳述式中指定 AwardsShow 容器，請使用以下 ARN：

```
"Resource": "arn:aws:mediastore:us-east-1:111122223333:container/AwardsShow"
```

的政策條件索引鍵 MediaStore

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。若您為單一條件索引鍵指定多個值，AWS 會使用邏輯 OR 操作評估條件。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與該 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 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 MediaStore 條件金鑰清單，請參閱服務授權參考資料 MediaStore中的 AWS Elemental 的條件金鑰。若要了解可以使用條件金鑰的動作和資源，請參閱 AWS Elemental 定義的動作 MediaStore。

若要檢視以 MediaStore 身分為基礎的原則範例，請參閱。AWS Elemental 於身份的政策範例 MediaStore

ACL 在 MediaStore

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 MediaStore

支援 ABAC (政策中的標籤)	部分

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 AWS 中，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)，以及許多 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 MediaStore

支援臨時憑證	是

您使用臨時憑證進行登入時，某些 AWS 服務 無法運作。如需詳細資訊，包括那些 AWS 服務 搭配臨時憑證運作，請參閱《IAM 使用者指南》中的可搭配 IAM 運作的 AWS 服務。

如果您使用使用者名稱和密碼之外的任何方法登入 AWS Management Console，則您正在使用臨時憑證。例如，當您使用公司的單一登入(SSO)連結存取 AWS 時，該程序會自動建立臨時憑證。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。如需切換角色的詳細資訊，請參閱 IAM 使用者指南中的切換至角色 (主控台)。

您可使用 AWS CLI 或 AWS API，手動建立臨時憑證。接著，您可以使用這些臨時憑證來存取 AWS。AWS 建議您動態產生臨時憑證，而非使用長期存取金鑰。如需詳細資訊，請參閱 IAM 中的暫時性安全憑證。

的跨服務主體權限 MediaStore

支援轉寄存取工作階段 (FAS)	是

當您使用 IAM 使用者或角色在 AWS 中執行動作時，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 使用主體的許可呼叫 AWS 服務，搭配請求 AWS 服務 以向下游服務發出請求。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

之後，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》。

MediaStore 的服務角色

支援服務角色	是

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 服務。

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

服務連結角色 MediaStore

支援服務連結角色。	否

服務連結角色是一種連結到 AWS 服務 的服務角色類型。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 AWS 帳戶 中，並由該服務所擁有。IAM 管理員可以檢視，但不能編輯服務連結角色的許可。

如需建立或管理服務連結角色的詳細資訊，請參閱可搭配 IAM 運作的 AWS 服務。在表格中尋找服務，其中包含服務連結角色欄中的 Yes。選擇 Yes (是) 連結，以檢視該服務的服務連結角色文件。

AWS Elemental 於身份的政策範例 MediaStore

根據預設，使用者和角色不具備建立或修改 MediaStore 資源的權限。他們也無法使用 AWS Management Console、AWS Command Line Interface (AWS CLI) 或 AWS API 執行任務。若要授與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 MediaStore，包括每種資源類型的 ARN 格式，請參閱服務授權參考 MediaStore中的 AWS Elemental 的動作、資源和條件金鑰。

主題

- 政策最佳實務
- 使用 MediaStore 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 MediaStore 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 受管政策並朝向最低權限許可的目標邁進：如需開始授予許可給使用者和工作負載，請使用 AWS 受管政策，這些政策會授予許可給許多常用案例。它們可在您的 AWS 帳戶 中使用。我們建議您定義特定於使用案例的 AWS 客戶管理政策，以便進一步減少許可。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。
- 套用最低許可許可 – 設定 IAM 政策的許可時，請僅授予執行任務所需的權限。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。您也可以使用條件來授予對服務動作的存取權，前提是透過特定 AWS 服務（例如 AWS CloudFormation）使用條件。如需更多資訊，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。如需更多資訊，請參閱 IAM 使用者指南中的 IAM Access Analyzer 政策驗證。
- 需要多重要素驗證 (MFA)：如果存在需要 AWS 帳戶中 IAM 使用者或根使用者的情況，請開啟 MFA 提供額外的安全性。如需在呼叫 API 操作時請求 MFA，請將 MFA 條件新增至您的政策。如需更多資訊，請參閱 IAM 使用者指南中的設定 MFA 保護的 API 存取。

有關 IAM 中最佳實務的更多相關資訊，請參閱 IAM 使用者指南中的 IAM 最佳安全實務。

使用 MediaStore 主控台

若要存取 AWS Elemental MediaStore 主控台，您必須擁有最少一組權限。這些權限必須允許您列出和檢視有關AWS 帳戶. MediaStore 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，您不需要允許其最基本主控台許可。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。

若要確保使用者和角色仍可使用 MediaStore 主控台，請同時將 MediaStore *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。此政策包含在主控台上，或是使用 AWS CLI 或 AWS API 透過編寫程式的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": "*"
        }
    ]
}
```

AWS Elemental MediaStore 身分和存取的疑難排解

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 MediaStore 常見問題。

主題

我沒有執行操作的授權 MediaStore

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 mediastore:*GetWidget* 許可時發生。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 mediastore: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 mediastore:*GetWidget* 動作存取 *my-example-widget* 資源。

如需任何協助，請聯絡您的 AWS 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 MediaStore。

有些 AWS 服務 允許您傳遞現有的角色至該服務，而無須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 MediaStore 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。

如需任何協助，請聯絡您的 AWS 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 MediaStore 資源

您可以建立一個角色，讓其他帳戶中的使用者或您的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對象取得該角色。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：

- 若要瞭解是否 MediaStore 支援這些功能，請參閱AWS Elemental 如何與 IAM MediaStore 搭配使用。
- 如需了解如何存取您擁有的所有 AWS 帳戶 所提供的資源，請參閱《IAM 使用者指南》中的將存取權提供給您所擁有的另一個 AWS 帳戶 中的 IAM 使用者。
- 如需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擁有的 AWS 帳戶。
- 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 角色與資源型政策的差異。

AWS Elemental MediaStore 中的日誌記錄和監控

本章節會概述 AWS Elemental MediaStore 中的記錄和監控選項，可讓您用來保障安全。如需 MediaStore 記錄和監控的詳細資訊，請參閱AWS Elemental 中的監控和標記 MediaStore。

監控是維護 AWS Elemental MediaStore 及您 AWS 解決方案可靠性、可用性和效能的重要部分。您應該從AWS解決方案的所有部分收集監視資料，以便在發生多點失敗時更輕鬆地偵錯。 AWS提供數種工具來監控您的 MediaStore 資源並回應潛在事件。

Amazon CloudWatch 警報

您可以使用 CloudWatch 警示來監視指定期間內的單一量度。如果指標超過指定臨界值，則會向 Amazon SNS 主題或 AWS Auto Scaling 政策傳送通知。 CloudWatch 警報不會叫用動作，因為它們處於特定狀態。必須是狀態已變更並維持了所指定的時間長度，才會呼叫動作。如需更多詳細資訊，請參閱 使用監控 CloudWatch。

AWS CloudTrail 日誌

CloudTrail 提供使用者、角色或AWS服務所採取之動作的記錄AWS Elemental MediaStore。使用收集的資訊 CloudTrail，您可以判斷提出的要求 MediaStore、提出要求的 IP 位址、提出要求的人員、提出要求的時間，以及其他詳細資訊。如需詳細資訊，請參閱使用 CloudTrail 記錄 API 呼叫。

AWS Trusted Advisor

Trusted Advisor 會利用為成千上萬名 AWS 客戶提供服務的過程中，學習到的最佳實務。Trusted Advisor 會檢查您的 AWS 環境，並在有可能節省成本、提升系統可用性與效能或填補安全漏洞時，向您提出建議。所有 AWS 客戶都能存取 5 個 Trusted Advisor 檢查。商業或企業支援方案客戶，可以檢視所有 Trusted Advisor 檢查。

如需詳細資訊，請參閱AWS Trusted Advisor。

AWS Elemental 的合規驗證 MediaStore

要瞭解 AWS 服務 是否在特定法規遵循方案範圍內，請參閱法規遵循方案範圍內的 AWS 服務，並選擇您感興趣的法規遵循方案。如需一般資訊，請參閱 AWS 法規遵循方案。

您可使用 AWS Artifact 下載第三方稽核報告。如需詳細資訊，請參閱 AWS Artifact 中的下載報告。

您使用 AWS 服務 時的法規遵循責任取決於資料的敏感度、您的公司的合規目標，以及適用的法律和法規。AWS 提供以下資源協助您處理法規遵循事宜：

- 安全與合規快速入門指南 – 這些部署指南討論在 AWS 上部署以安全及合規為重心的基準環境的架構考量和步驟。
- Amazon Web Services 的 HIPAA 安全與法規遵循架構：本白皮書說明公司可如何運用 AWS 來建立符合 HIPAA 規定的應用程式。

> **Note**
>
> 並非全部的 AWS 服務 都符合 HIPAA 資格。如需詳細資訊，請參閱 HIPAA 資格服務參照。

- AWS 合規資源：這組手冊和指南可能適用於您的產業和位置。
- AWS 客戶合規指南：透過合規的角度瞭解共同的責任模式。這份指南橫跨多個架構 (包含國家標準技術研究所 (NIST)、支付卡產業安全標準委員會 (PCI) 和國際標準組織 (ISO))，總結保護 AWS 服務的最佳實務並將指導方針對應至安全控制。
- AWS Config 開發人員指南中的使用規則評估資源：AWS Config 服務可評估您的資源組態對於內部實務、業界準則和法規的合規狀態。
- AWS Security Hub – 此 AWS 服務 可供您全面檢視 AWS 中的安全狀態。Security Hub 使用安全控制，可評估您的 AWS 資源並檢查您的法規遵循是否符合安全業界標準和最佳實務。如需支援的服務和控制清單，請參閱 Security Hub controls reference。
- AWS Audit Manager – 此 AWS 服務 可協助您持續稽核 AWS 使用情況，以簡化管理風險與法規與業界標準的法規遵循方式。

AWS Elemental 的韌性 MediaStore

AWS 全球基礎架構是以 AWS 區域 與可用區域為中心建置的。AWS 區域 提供多個分開且隔離的實際可用區域，並以具備低延遲、高輸送量和高度備援特性的聯網相互連結。透過可用區域，您可以設計與操作的應用程式和資料庫，在可用區域之間自動容錯移轉而不會發生中斷。可用區域的可用性、容錯能力和可擴展性能力，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 與可用區域的詳細資訊，請參閱AWS全球基礎架構。

除了AWS全球基礎架構之外，還 MediaStore 提供多種功能，協助支援您的資料恢復能力和備份需求。

AWS Elemental 中的基礎設施安全 MediaStore

身為受管服務，AWS Elemental MediaStore 受到AWS全球網路安全的保護。如需有關 AWS 安全服務以及 AWS 如何保護基礎設施的詳細資訊，請參閱 AWS 雲端安全。若要使用基礎設施安全性的最佳實務來設計您的 AWS 環境，請參閱安全支柱 AWS 架構良好的框架中的基礎設施保護。

您可以使用AWS已發佈的 API 呼叫透 MediaStore 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，您可以使用 AWS Security Token Service (AWS STS) 產生臨時安全憑證來簽署請求。

預防跨服務混淆代理人

混淆代理人問題屬於安全性問題，其中沒有執行動作許可的實體可以強制具有更多許可的實體執行該動作。在 AWS 中，跨服務模擬可能會導致混淆代理人問題。在某個服務（呼叫服務）呼叫另一個服務（被呼叫服務）時，可能會發生跨服務模擬。可以操縱呼叫服務來使用其許可，以其不應有存取許可的方式對其他客戶的資源採取動作。為了預防這種情況，AWS 提供的工具可協助您保護所有服務的資料，而這些服務主體已獲得您帳戶中資源的存取權。

我們建議在資源政策中使用aws:SourceArn和aws:SourceAccount全域條件上下文金鑰，以限制 AWS Elemental MediaStore 為資源提供其他服務的許可。如果您想要僅允許一個資源與跨服務存取相關聯，則請使用 aws:SourceArn。如果您想要允許該帳戶中的任何資源與跨服務使用相關聯，請使用 aws:SourceAccount。

防範混淆代理人問題的最有效方法是使用 aws:SourceArn 全域條件內容索引鍵，以及資源的完整 ARN。如果不知道資源的完整 ARN，或者如果您指定了多個資源，請使用 aws:SourceArn 全域內容條件索引鍵搭配萬用字元 (*) 來表示 ARN 的未知部分。例如 arn:aws:*servicename*:*:*123456789012*:*。

如果 aws:SourceArn 值不包含帳戶 ID (例如 Amazon S3 儲存貯體 ARN)，您必須使用這兩個全域條件內容索引鍵來限制許可。

的值aws:SourceArn必須是在您的區域和帳戶中 MediaStore 發佈 CloudWatch 記錄的組態。

下列範例顯示如何在中使用aws:SourceArn和aws:SourceAccount全域條件前後關聯鍵字 MediaStore 來避免混淆的副問題。

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Sid": "ConfusedDeputyPreventionExamplePolicy",
    "Effect": "Allow",
    "Principal": {
      "Service": "servicename.amazonaws.com"
```

```
    },
    "Action": "servicename:ActionName",
    "Resource": [
      "arn:aws:servicename:::ResourceName/*"
    ],
    "Condition": {
      "ArnLike": {
        "aws:SourceArn": "arn:aws:servicename:*:123456789012:*"
      },
      "StringEquals": {
        "aws:SourceAccount": "123456789012"
      }
    }
  }
}
```

AWS Elemental 中的監控和標記 MediaStore

監控是維護 AWS Elemental 及其他AWS解決方案之可靠性、可靠性、可靠性 MediaStore 和效能的重要環節。 AWS提供下列監控工具以監看 MediaStore、在發生錯誤時回報，並自動適時採取動作：

- AWS CloudTrail 擷取您 AWS 帳戶發出或代表發出的 API 呼叫和相關事件，並傳送記錄檔案至您指定的 Simple Storage Service (Amazon S3) 儲存貯體。您可以找出哪些使用者和帳戶呼叫 AWS、發出呼叫的來源 IP 地址，以及呼叫的發生時間。如需詳細資訊，請參閱 AWS CloudTrail 使用者指南。
- Amazon 會即時 CloudWatch監控您的AWS資源，以及您AWS在上執行的應用程式。您可以收集和追蹤指標、建立自訂儀表板，以及設定警示，在特定指標達到您指定的閾值時通知您或採取動作。例如，您可以使用 CloudWatch 追蹤 CPU 使用量或其他 Amazon EC2 執行個體指標，並在需要時自動啟動新的執行個體。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南。
- Amazon E CloudWatch vents 可傳送系統事件串流，以說明AWS資源發生的變動。AWS服務一般能在幾秒內將事件通知交付至 E CloudWatch vents，但有時候會耗費一分鐘或更長的時間。CloudWatch Events 啟用自動的事件驅動運算，因為您可以在這些事件發生時，編寫監看特定事件與在其他AWS服務內觸發自動化動作的規則。如需詳細資訊，請參閱 Amazon E CloudWatch vents 使用者指南。
- Amazon CloudWatch Logs 可讓您從 Amazon EC2 執行個體監控、存放及存取來自 Amazon EC2 執行個體 CloudTrail，以及其他來源的日誌檔案。 CloudWatch 日誌可監控日誌檔案中的資訊，並在達到特定閾值時通知您。您也可以將日誌資料存檔在高耐用性的儲存空間。如需詳細資訊，請參閱 Amazon CloudWatch Logs 使用者指南。

您也可以用標籤的形式 MediaStore 將中繼資料指定許可 每個標籤都包含您定義的金鑰和值。標籤可讓您更輕鬆地管理、搜尋和篩選資源。您可以使用標籤管理您在 AWS 管理主控台的 AWS 資源、建立所有 AWS 資源的用量和帳單報告，以及在基礎設施自動化活動中篩選資源。

主題

- 使用記錄 AWS Elemental MediaStore API 呼叫AWS CloudTrail
- MediaStore 使用亞馬遜監控 AWS Elemental CloudWatch
- 標記 AWS Elemental MediaStore 資源

使用記錄 AWS Elemental MediaStore API 呼叫AWS CloudTrail

AWS Elemental MediaStore 已與整合AWS CloudTrail，這項服務可提供由使用者、角色或中AWS服務所採取之動作的記錄 MediaStore。 CloudTrail 擷取 MediaStore 為事件的 API 呼叫子集，包括來自MediaStore主控台的呼叫，以及來自對 MediaStore API 發出的程式碼呼叫。如果您建定線索，就可以將 CloudTrail 事件持續交付至 Amazon S3 儲存貯體，包括的事件 MediaStore。即使您未設定線索，依然可以透過 CloudTrail 主控台中的 Event history (事件歷史記錄) 檢視最新事件。您可以使用收集的資訊來 CloudTrail判斷提交給和的請求 MediaStore、提出請求的 IP 地址、提出請求的對象、提出請求的時間，以及其他更多其他資訊。

若要進一步了解 CloudTrail，包括如何設定許可，請參閱《AWS CloudTrail使用者指南》。

主題

- AWS 中的元素 MediaStore資訊 CloudTrail
- 範例：AWS Elemental MediaStore 日誌檔項目

AWS 中的元素 MediaStore資訊 CloudTrail

CloudTrail 當您建定AWS帳戶時，系統會在您的帳戶中啟用。AWS Elemental 發生支援的事件活動時 MediaStore，系統便會將該活動記錄至事件，並將其他AWS服務 CloudTrail事件記錄到 E vent history (事件歷史記錄) 您可以檢視、搜尋和下載 AWS 帳戶的最新事件。如需詳細資訊，請參閱使用 CloudTrail 事件歷程記錄檢視事件。

如需您 AWS 帳戶中正在進行事件的記錄 (包含 MediaStore 的事件)，請建立線索。追蹤可讓您 CloudTrail 將日誌檔案交付至 Amazon S3 儲存貯體。根據預設，當您在主控台建立線索時，線索會套用到所有 AWS 區域。該追蹤會記錄來自 AWS 分割區中所有區域的事件，並將日誌檔案交付到您指定的 Amazon S3 儲存貯體。此外，您可以設定其他AWS服務，以進一步分析及處理 CloudTrail 日誌中所收集的事件資料。如需詳細資訊，請參閱下列主題：

- 建立追蹤的概觀
- CloudTrail 支援的服務與整合
- 設定的 Amazon SNS 通知 CloudTrail
- 從多個區域接收 CloudTrail 日誌檔案，以及從多個帳戶接收 CloudTrail 日誌檔案

AWS Elemental MediaStore 支援將下列操作記錄為 CloudTrail 日誌檔案事件：

- `CreateContainer`

- DeleteContainer
- DeleteContainerPolicy
- DeleteCorsPolicy
- DescribeContainer
- GetContainerPolicy
- GetCorsPolicy
- ListContainers
- PutContainerPolicy
- PutCorsPolicy

每一筆事件或日誌項目都會包含產生請求者的資訊。身分資訊可協助您判斷下列事項：

- 該請求是否使用根使用者或使用者憑證提出
- 提出該請求時，是否使用了特定角色或聯合身分使用者的臨時安全憑證
- 該請求是否由另一項 AWS 服務提出

如需詳細資訊，請參閱 CloudTrail 使用者身分元素。

範例：AWS Elemental MediaStore 日誌檔項目

追蹤是一種組態，能讓事件以日誌檔案的形式交付至您指定的 Amazon S3 儲存貯體。 CloudTrail 日誌檔案包含一個或多個日誌項目。一個事件為任何來源提出的單一請求，並包含請求動作、請求的日期和時間、請求參數等資訊。 CloudTrail 日誌檔案並非依公有 API 呼叫追蹤記錄的堆疊排序，因此不會以任何特定順序出現。

以下範例顯示的 CloudTrail 日誌項目會示範CreateContainer操作：

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "ABCDEFGHIJKL123456789",
        "arn": "arn:aws:iam::111122223333:user/testUser",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
```

```
        "userName": "testUser",
        "sessionContext": {
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2018-07-09T12:55:42Z"
            }
        },
        "invokedBy": "signin.amazonaws.com"
    },
    "eventTime": "2018-07-09T12:56:54Z",
    "eventSource": "mediastore.amazonaws.com",
    "eventName": "CreateContainer",
    "awsRegion": "ap-northeast-1",
    "sourceIPAddress": "54.239.119.16",
    "userAgent": "signin.amazonaws.com",
    "requestParameters": {
        "containerName": "TestContainer"
    },
    "responseElements": {
        "container": {
            "status": "CREATING",
            "creationTime": "Jul 9, 2018 12:56:54 PM",
            "name": " TestContainer ",
            "aRN": "arn:aws:mediastore:ap-northeast-1:111122223333:container/
TestContainer"
        }
    },
    "requestID":
 "MNCTGH4HRQJ27GRMBVDPIVHEP4LO2BN6MUVHBCPSHOAWNSOKSXCO24B2UEOBBND5DONRXTMFK3TOJ4G7AHWMESI",
    "eventID": "7085b140-fb2c-409b-a329-f567912d704c",
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```


MediaStore 使用亞馬遜監控 AWS Elemental CloudWatch

您可以 MediaStore 使用 AWS Elemental 監控 AWS Elemental CloudWatch，這會收集原始資料並將該資料處理成可讀指標。 CloudWatch 將統計資料保留 15 個月，以便您存取歷史資訊，並更清楚 Web 應用程式或服務的執行效能。您也可以設定留意特定閾值的警示，當滿足這些閾值時傳送通知或採取動作。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南。

AWS提供下列監控工具以監看 MediaStore、在發生錯誤時回報，並自動適時採取動作：

- Amazon CloudWatch Logs 可讓您從 AWS Elemental 等AWS服務中監控、存放及存取日誌檔案 MediaStore。您可以使用 CloudWatch Logs 來監控使用日誌資料的應用程式和系統。例如，CloudWatch 日誌可以追蹤應用程式日誌中發生的錯誤數量，並在錯誤率超過您指定的閾值時通知您。 CloudWatch Logs 會使用您的日誌資料進行監控，所以不需要變更程式碼。例如，您可以監視應用程式記錄檔中的特定常值字詞 (例如 "ValidationException「)，或計算特定時段內發出的`PutObject`要求數目。找到您要搜尋的詞彙時， CloudWatch Logs 便會將資料回報至您指定的指 CloudWatch 標。日誌資料會在移轉和靜態時加密。
- Amazon E CloudWatch vents 提供描述資AWS源變更的系統事件，例如 MediaStore 物件。AWS服務一般能在幾秒內將事件通知交付至 E CloudWatch vents，但有時候會耗費一分鐘或更長的時間。您可以設定許可，以比對事件 (例如`DeleteObject`請求)，並將這些規則路由至一或多個目標函數或串流。 CloudWatch 事件在操作變更時會查覺到。此外，E CloudWatch vents 會回應這些操作變更並視需要進行更正動作，透過傳送訊息來回應環境、啟用功能、執行變更和擷取狀態資訊。

CloudWatch 日誌

存取記錄會為對容器中的物件所做的請求提供詳細的記錄。存取記錄對於許多應用程式都相當實用，例如安全和存取稽核。您也可藉由該資訊了解自己的客戶群，並掌握 MediaStore 帳單。 CloudWatch 記錄的分類如下：

- 日誌串流是共享相同來源的一系列日誌事件。
- 日誌群組是共享相同保留、監控和存取控制設定的日誌串流群組。當您在容器上啟用存取記錄時，MediaStore 會建立名稱如下的記錄群組`/aws/mediastore/MyContainerName`。您可以定義日誌群組，並指定放入每個群組的串流。可以屬於一個日誌群組的日誌串流數量並沒有配額。

根據預設，日誌將無限期保留且永遠不會過期。您可以調整每個日誌群組的保留政策，維持無限期保留，或選擇 1 天至 10 年之間的保留期間。

設定許可 CloudWatch

使用AWS Identity and Access Management (IAM) 建立可讓 AWS Elemental MediaStore 存取亞馬遜的角色 CloudWatch。您必須執行這些步驟，才能為您的帳戶發佈 CloudWatch 記錄。 CloudWatch 自動發佈您帳戶的指標。

若要允許 MediaStore 存取 CloudWatch

1. 前往網址 https://console.aws.amazon.com/iam/ 開啟 IAM 主控台。
2. 在 IAM 主控台的導覽窗格中，選擇 Policies (政策)，接著選擇 Create Policy (建立政策)。

3. 選擇 JSON 標籤，並將下列政策貼上：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "logs:DescribeLogGroups",
                "logs:CreateLogGroup"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogStream",
                "logs:DescribeLogStreams",
                "logs:PutLogEvents"
            ],
            "Resource": "arn:aws:logs:*:*:log-group:/aws/mediastore/*"
        }
    ]
}
```

此原則可讓您 MediaStore 為AWS帳戶內任何區域中的任何容器建立記錄群組和記錄資料流。

4. 選擇 Review policy (檢閱政策)。
5. 在 Review policy (檢閱政策) 頁面上，針對 Name (名稱)，輸入 **MediaStoreAccessLogsPolicy**，然後選擇 Create policy (建立政策)。
6. 在 IAM 主控台的導覽窗格中，選擇 Roles (角色)，然後選擇 Create role (建立角色)。
7. 選擇 Another AWS account (另一個 AWS 帳戶) 角色類型。
8. 針對 Account ID (帳戶 ID)，輸入您的 AWS 帳戶 ID。
9. 選擇 Next: Permissions (下一步：許可)。
10. 在搜尋方塊中，輸入 **MediaStoreAccessLogsPolicy**。
11. 選取新政策旁的核取方塊，然後選擇 Next: Tags (下一步：標籤)。
12. 選擇 Next: Review (下一步：檢閱) 來預覽新使用者。
13. 針對 Role name (角色名稱)，輸入 **MediaStoreAccessLogs**，然後選擇 Create role (建立角色)。

14. 在確認訊息中，選擇您剛建立的角色名稱 (**MediaStoreAccessLogs**)。
15. 在角色的 Summary (摘要) 頁面上，選擇 Trust relationships (信任關係) 標籤。
16. 選擇 Edit trust relationship (編輯信任關係)。
17. 在政策文件中，將委託人變更為 MediaStore 服務。它應該如下所示：

```
"Principal": {
   "Service": "mediastore.amazonaws.com"
},
```

整個政策看起來應該如下所示：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "mediastore.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {}
    }
  ]
}
```

18. 選擇 Update Trust Policy (更新信任政策)。

啟用容器的存取記錄

根據預設，AWS Elemental MediaStore 不會收集存取日誌。當您在容器上啟用存取記錄日誌時，Amazon 會將存放在該容器中物件的存取日誌交 MediaStore 付至 Amazon CloudWatch。這些存取日誌會為對容器中存放的任何物件所做的請求提供詳細的記錄。這資訊可能包含要求類型、要求中指定的資源，以及要求的處理時間和日期。

> **Important**
>
> 在 MediaStore 容器上啟用存取記錄不會產生額外費用。不過，此服務提供給您的任何日誌檔都會產生一般儲存費用。(您可以隨時刪除這些日誌檔。) AWS 不會估算日誌檔傳遞的資料傳輸費，但會收取用於存取日誌檔之一般資料傳輸率的費用。

若要啟用存取記錄 (AWS CLI)

- 在 AWS CLI 中，使用 `start-access-logging` 命令：

```
aws mediastore start-access-logging --container-name LiveEvents --region us-west-2
```

此命令沒有傳回值。

停用容器的存取記錄

當您停用容器上的存取記錄時，AWS Elemental MediaStore 會停止將存取日誌傳送到亞馬遜 CloudWatch。系統不會儲存這些存取日誌而且這些日誌不供擷取。

停用存取記錄 (AWS CLI)

- 在 AWS CLI 中，使用 `stop-access-logging` 命令：

```
aws mediastore stop-access-logging --container-name LiveEvents --region us-west-2
```

此命令沒有傳回值。

AWS Elemental 中的存取記錄疑難排解 MediaStore

當 AWS Elemental MediaStore 存取日誌未出現在亞馬遜中時 CloudWatch，請參閱下表以瞭解潛在原因和解決方案。

> **Note**
>
> 請務必啟用 AWS CloudTrail 日誌，以協助故障診斷程序。

徵狀	問題可能是 ...	嘗試這麼做 ...
即使 CloudTrail 記錄已啟用，您也看不到任何 CloudTrail 事件。	IAM 角色不存在或其名稱、許可或信任政策不正確。	建立具有正確名稱、許可和信任政策的角色。請參閱 the section called “設定許可 CloudWatch”。
您已提交 `DescribeContainer` API 請求，但回應顯示 `AccessLoggingEnabled` 參數具有值 `False`。此外，您沒有看見適用於進行成功 `DescribeLogGroup` 、`CreateLogGroup` 、`DescribeLogStream` 或 `CreateLogStream` 呼叫之 `MediaStoreAccessLogs` 角色 的任何 CloudTrail 事件。	IAM 角色不存在或其名稱、許可或信任政策不正確。	建立具有正確名稱、許可和信任政策的角色。請參閱 the section called “設定許可 CloudWatch”。
	容器上沒有啟用存取記錄。	啟用容器的存取日誌。請參閱 the section called “啟用存取記錄”。

在 CloudTrail 主控台上，您會看到與`MediaStoreAccessLogs` 角色相關的存取遭拒錯誤的事件。此 CloudTrail 事件可能包含下列行：
 	
`"eventSource": "logs.amazonaws.com",`
 	
`"errorCode": "AccessDenied",`
 	
`"errorMessage": "User: arn:aws:sts::111122223333:assumed-role/MediaStoreAccessLogs/MediaStoreAccessLogsSession is not authorized to perform: logs:DescribeLogGroups on resource: arn:aws:logs:us-west-2:1111`
	IAM 角色沒有 AWS Elemental 的正確許可 MediaStore。	將 IAM 角色更新為具備正確的許可和信任政策。請參閱 the section called “設定許可 CloudWatch”。

徵狀	問題可能是 ...	嘗試這麼做 ...
`22223333:log-group::log-stream:",`		
您沒有看到適用於一或多個整個容器的任何日誌。	您的帳戶可能超過每個區域每個帳戶的日誌群組 CloudWatch 配額。請參閱 Amazon CloudWatch 日誌使用者指南中的日誌群組配額。	在 CloudWatch 主控台上，判斷您的帳戶是否已達到記錄群組的 CloudWatch 配額。如果必要，請求增加配額。
您會看到一些登入 CloudWatch，但並非您希望看到的所有記錄檔。	您的帳戶可能已超過每個區域每個帳戶每秒的交易 CloudWatch 配額。請參閱 Amazon CloudWatch 日誌使用者指南`PutLogEvents` 中的配額。	每個區域每個帳戶每秒 CloudWatch 交易，請求增加配額。

存取日誌格式

存取日誌檔是由一系列的 JSON 格式的日誌記錄所組成，其中每個日誌記錄代表一個請求。日誌中欄位的順序可能有所不同。以下是包含兩個日誌記錄的範例日誌：

```
{
  "Path": "/FootballMatch/West",
  "Requester": "arn:aws:iam::111122223333:user/maria-garcia",
  "AWSAccountId": "111122223333",
  "RequestID":
 "aaaAAA111bbbBBB222cccCCC333dddDDD444eeeEEE555fffFFF666gggGGG777hhhHHH888iiiIII999jjjJJJ",
  "ContainerName": "LiveEvents",
  "TotalTime": 147,
  "BytesReceived": 1572864,
  "BytesSent": 184,
  "ReceivedTime": "2018-12-13T12:22:06.245Z",
  "Operation": "PutObject",
```

```
  "ErrorCode": null,
  "Source": "192.0.2.3",
  "HTTPStatus": 200,
  "TurnAroundTime': 7,
  "ExpiresAt": "2018-12-13T12:22:36Z"
}
 {
  "Path": "/FootballMatch/West",
  "Requester": "arn:aws:iam::111122223333:user/maria-garcia",
  "AWSAccountId": "111122223333",
  "RequestID":
 "dddDDD444eeeEEE555fffFFF666gggGGG777hhhHHH888iiiIII999jjjJJJ000cccCCC333bbbBBB222aaaAAA",
  "ContainerName": "LiveEvents",
  "TotalTime": 3,
  "BytesReceived": 641354,
  "BytesSent": 163,
  "ReceivedTime": "2018-12-13T12:22:51.779Z",
  "Operation": "PutObject",
  "ErrorCode": "ValidationException",
  "Source": "198.51.100.15",
  "HTTPStatus": 400,
  "TurnAroundTime": 1,
  "ExpiresAt": null
}
```

下列清單說明日誌記錄欄位：

AWSAccountId

用來提出請求之帳戶的 AWS 帳戶 ID。

BytesReceived

MediaStore 伺服器所接收到請求內文中的位元組數。

BytesSent

MediaStore 伺服器所傳送請求內文中的位元組數。此值通常與在伺服器回應中包含的 `Content-Length` 標頭值相同。

ContainerName

收到請求的容器名稱。

ErrorCode

錯 MediaStore 誤代碼 (例如InternalServerError)。如果沒有錯誤發生，則會顯示 - 字元。即使狀態碼為 200 (表示關閉連線或在伺服器開始串流回應後發生錯誤)，仍可能會顯示錯誤碼。

ExpiresAt

物件的過期日期和時間。此值是以套用至容器的生命週期原則transient data rule中所設定的到期天數為基礎。此值是 ISO-8601 日期時間，且會根據服務請求之主機的系統時鐘。如果生命週期原則沒有套用至物件的暫時性資料規則，或是沒有套用至容器的生命週期原則，則此欄位的值為null。此欄位僅適用於下列操作：PutObjectGetObject、DescribeObject、和DeleteObject。

HTTPStatus

回應的數字 HTTP 狀態碼。

操作

已執行的操作，例如 PutObject 或 ListItems。

路徑

容器中的路徑，此容器為物件的存放位置。如果操作不會採用路徑參數，則會出現 - 字元。

ReceivedTime

收到請求的時間。此值是 ISO-8601 日期時間，且會根據服務請求之主機的系統時鐘。

要求者

用來提出請求的帳戶使用者 Amazon Resource Name (ARN)。若是未經授權的請求，這個值會是 anonymous。如果在身分驗證完成前要求失敗，則此欄位可能會從日誌中遺失。對於這類要求，ErrorCode 可能會識別授權問題。

RequestID

由 AWS Elemental 產生的字串，是由 AWS Elemental MediaStore 可唯一識別每項請求。

來源

請求者的明顯網際網路地址或進行呼叫的 AWS 服務服務委託人。如果中間代理伺服器與防火牆會將提出請求的機器地址模糊處理，此值會設為 Null。

TotalTime

從伺服器角度計算的請求所經過的毫秒數 (ms)。此值是從服務收到您請求的時間開始，計算到回應傳送出最後一組位元組的時間。從用戶端角度進行的測量會受網路延遲影響，因此會從伺服器角度來進行此值的測量。

TurnAroundTime

處理您要求所 MediaStore 花費的毫秒數。此值是從收到您要求的最後位元組的時間開始，計算到回應傳送出第一組位元組的時間。

日誌中欄位的順序可能有所不同。

記錄狀態變更會在一段時間後生效

記錄容器狀態的變更，要一段時間後才會實際影響到日誌檔交付。例如，若已啟用容器記錄，則在接下來的一小時內提出之請求，可能有些會記錄下來，有些則不會。如果您停用容器 B 的記錄，系統可能會繼續交付下一個小時的日誌，有些則不會。在所有情況下，新的設定最終都會生效，您無需採取任何進一步動作。

伺服器日誌交付最佳作法

存取日誌記錄會依最佳作法交付。已適合設定為在交付日誌記錄中記錄結果的容器的多數請求。大多數的日誌記錄會於記錄後的數小時內交付，但也可以常交付。

並不保證存取記錄的完成程度與時間先後順序。特定要求的日誌記錄，可能會在實際處理要求之後很久才交付，或者有可能完全不會交付。存取日誌的目的在於讓您能了解容器流量的真實狀態。雖然日誌記錄極少會缺失，但存取記錄並不代表所有請求的完整記錄。

其遵循存取記錄功能的最佳做法，在 AWS 入口網站提供的用量報告 (AWS Management Console上的帳單和成本管理報告) 中，可能包含一或多個未出現在已交付存取日誌中的存取請求。

存取記錄格式的程式設計考量

有時候，我們可能要在新增欄位，來擴展存取日誌的格式。必須寫入剖析存取日誌的程式碼，才能處理它不了解的其他欄位。

CloudWatch 活動

Amazon E CloudWatch vents 可讓您自動化AWS服務，並自動回應系統事件 (例如應用性的問題或資源的變動)。您可編寫簡單的規則，來指示您在意的事件，以及當事件符合規則時所要自動執行的動作。

> **Important**
>
> AWS服務一般能在幾秒內將事件通知交付至 E CloudWatch vents，但有時候會耗費一分鐘或更長的時間。

當檔案上傳至容器或從容器中移除時， CloudWatch 服務中會連續觸發兩個事件：

1. the section called “物件狀態變更事件”
2. the section called “容器狀態變更事件”

如需訂閱這些事件的相關資訊，請參閱 Amazon CloudWatch。

可以自動觸發的動作如下：

- 呼叫 AWS Lambda 函數
- 呼叫 Amazon EC2 執行命令
- 將事件轉傳至 Amazon Kinesis Data Streams
- 啟動 AWS Step Functions 狀態機器
- 通知 Amazon SNS 主題或AWS SMS佇列

下列是 AWS Elemental MediaStore 使用 CloudWatch 事件的部分範例：

- 每當容器建立時啟動 Lambda 函數
- 在物件被刪除時通知 Amazon SNS 主題

如需詳細資訊，請參閱 Amazon E CloudWatch vents 使用者指南。

主題

- AWS Elemental MediaStore 物件狀態變更事件
- AWS Elemental MediaStore 容器狀態變更事件

AWS Elemental MediaStore 物件狀態變更事件

物件的狀態已變更時 (當物件已上傳或刪除時)，將會發佈此事件。

> **Note**
>
> 由於暫時性資料規則而過期的物件不會在過期時發出 CloudWatch事件。

如需訂閱此事件的相關資訊，請參閱 Amazon CloudWatch。

物件已更新

```
{
  "version": "1",
  "id": "6a7e8feb-b491-4cf7-a9f1-bf3703467718",
  "detail-type": "MediaStore Object State Change",
  "source": "aws.mediastore",
  "account": "111122223333",
  "time": "2017-02-22T18:43:48Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:mediastore:us-east-1:111122223333:MondayMornings/Episode1/
Introduction.avi"
  ],
  "detail": {
    "ContainerName": "Movies",
    "Operation": "UPDATE",
    "Path":"TVShow/Episode1/Pilot.avi",
    "ObjectSize":123456,
    "URL": "https://a832p1qeaznlp9.files.mediastore-us-west-2.com/Movies/
MondayMornings/Episode1/Introduction.avi"
  }
}
```

物件已移除

```
{
  "version": "1",
  "id": "6a7e8feb-b491-4cf7-a9f1-bf3703467718",
  "detail-type": "MediaStore Object State Change",
  "source": "aws.mediastore",
  "account": "111122223333",
  "time": "2017-02-22T18:43:48Z",
  "region": "us-east-1",
  "resources": [
```

```
    "arn:aws:mediastore:us-east-1:111122223333:Movies/MondayMornings/Episode1/
Introduction.avi"
  ],
  "detail": {
    "ContainerName": "Movies",
    "Operation": "REMOVE",
    "Path":"Movies/MondayMornings/Episode1/Introduction.avi",
    "URL": "https://a832p1qeaznlp9.files.mediastore-us-west-2.com/Movies/
MondayMornings/Episode1/Introduction.avi"
  }
}
```

AWS Elemental MediaStore 容器狀態變更事件

容器的狀態已變更時 (當容器已新增或刪除時)，將會發佈此事件。如需訂閱此事件的相關資訊，請參閱 Amazon CloudWatch。

容器已建立

```
{
  "version": "1",
  "id": "6a7e8feb-b491-4cf7-a9f1-bf3703467718",
  "detail-type": "MediaStore Container State Change",
  "source": "aws.mediastore",
  "account": "111122223333",
  "time": "2017-02-22T18:43:48Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:mediastore:us-east-1:111122223333:container/Movies"
  ],
  "detail": {
    "ContainerName": "Movies",
    "Operation": "CREATE"
    "Endpoint": "https://a832p1qeaznlp9.mediastore-us-west-2.amazonaws.com"
  }
}
```

容器已移除

```
{
  "version": "1",
  "id": "6a7e8feb-b491-4cf7-a9f1-bf3703467718",
  "detail-type": "MediaStore Container State Change",
```

```
  "source": "aws.mediastore",
  "account": "111122223333",
  "time": "2017-02-22T18:43:48Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:mediastore:us-east-1:111122223333:container/Movies"
  ],
  "detail": {
    "ContainerName": "Movies",
    "Operation": "REMOVE"
  }
}
```

MediaStore 使用亞馬遜 CloudWatch 指標監控 AWS Elemental

您可以 MediaStore 使用 AWS Elemental 監控 AWS Elemental CloudWatch，這會收集原始資料並將該資料處理成可讀指標。 CloudWatch保留統計資料會保留 15 個月，以便您存取歷史資訊，並更清楚 Web 應用程式或服務的執行效能。您也可以設定留意特定閾值的警示，當滿足這些閾值時傳送通知或採取動作。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南。

對於 AWS Elemental MediaStore，您可能想要在該指標達到特定閾值時觀看`BytesDownloaded`並傳送電子郵件給自己。

使用 CloudWatch 主控台檢視指標

指標會先依服務命名空間分組，再依各命名空間內不同的維度組合分類。

1. 請登入AWS Management Console並開啟 CloudWatch 主控台，網址為 https://console.aws.amazon.com/cloudwatch/。
2. 在導覽窗格中，選擇 Metrics (指標)。
3. 在「所有測量結果」下，選擇 AWS/MediaStore 命名空間。
4. 選擇要檢視指標的指標維度。例如，選擇 `Request metrics by container` 以檢視已傳送至容器之不同請求類型的指標。

若要使用 AWS CLI 來檢視指標

- 在命令提示中，使用下列命令：

```
aws cloudwatch list-metrics --namespace "AWS/MediaStore"
```

AWS Elemental MediaStore 指標

下表列出 AWS Elemental MediaStore 傳送的目標指標 CloudWatch。

> **Note**
>
> 若要檢視指標，您必須將指標政策新增至容器， MediaStore 才能將指標傳送至 Amazon CloudWatch。

指標	描述
RequestCount	針對 MediaStore 容器提出的 HTTP 請求總數，以操作類型 (Put、Get、Delete、Describe, 、List) 分隔。 單位：計數 有效維度： • 容器名稱 • 物件群組名稱 • 要求類型 有效的統計資訊：總和
4xxErrorCount	對此發出的 HTTP 要求數目 MediaStore 會導致 4xx 錯誤。 單位：計數 有效維度： • 容器名稱 • 物件群組名稱 • 要求類型 有效的統計資訊：總和

指標	描述
5xxErrorCount	對此發出的 HTTP 要求數目 MediaStore 會導致發生 5xx 錯誤。 單位：計數 有效維度： • 容器名稱 • 物件群組名稱 • 要求類型 有效的統計資訊：總和
BytesUploaded	針對 MediaStore 容器提出之請求所上傳的位元組數目，且回應包含內文。 單位：位元組 有效維度： • 容器名稱 • 物件群組名稱 有效的統計資訊：平均 (每個請求的位元組數)、總和 (每個期間的位元組數)、範例計數、最小值 (相容於 P0.0)、最大值 (相同於 p100)，任何介於 p0.0 與 p99.9 的百分位數

指標	描述
BytesDownloaded	針對 MediaStore 容器提出之請求所下載的位元組數目，且回應包含內文。 單位：位元組 有效維度： • 容器名稱 • 物件群組名稱 有效的統計資訊：平均 (每個請求的位元組數)、總和 (每個期間的位元組數)、範例計數、最小值 (相容於 P0.0)、最大值 (相同於 p100)，任何介於 p0.0 與 p99.9 的百分位數
TotalTime	從伺服器角度計算的請求所經過的毫秒數。此值是從 MediaStore 收到您要求的時間開始，計算到回應傳送出最後一組位元組的時間。從用戶端角度進行的測量會受網路延遲影響，因此會從伺服器角度來進行此值的測量。 單位：毫秒 有效維度： • 容器名稱 • 物件群組名稱 • 要求類型 有效的統計資訊：平均、最小值 (相容於 P0.0)、最大值 (相同於 p100)、任何介於 p0.0 與 p100 的百分位數

指標	描述
TurnaroundTime	處理您要求所 MediaStore 花費的毫秒數。此值是從 MediaStore 收到您要求的最後位元組的時間開始，計算到回應傳送出第一組位元組的時間。 單位：毫秒 有效維度： • 容器名稱 • 物件群組名稱 • 要求類型 有效的統計資訊：平均、最小值 (相容於 P0.0)、最大值 (相同於 p100)、任何介於 p0.0 與 p100 的百分位數
ThrottleCount	對此發出的 HTTP 請求數量 MediaStore 已經過限制。 單位：計數 有效維度： • 容器名稱 • 物件群組名稱 • 要求類型 有效的統計資訊：總和

標記 AWS Elemental MediaStore 資源

標籤是一種自訂屬性標籤，可讓您指派或由 AWS 指派給 AWS 資源。每個標籤有兩個部分：

- 標籤鍵 (例如，CostCenter、Environment 或 Project)。標籤鍵會區分大小寫。
- 選用欄位，稱為標籤值 (例如 111122223333 或 Production)。忽略標籤值基本上等同於使用空字串。與標籤鍵相同，標籤值會區分大小寫。

標籤可協助您執行以下操作：

- 識別和組織您的 AWS 資源。許多 AWS 服務支援標籤，因此您可以對來自不同服務的資源指派相同的標籤，以指出資源是相關的。例如，您可以將相同的標籤指派給指派給AWS Elemental MediaLive輸入的 AWS Elemental MediaStore ##。
- 追蹤您的 AWS 成本。您會在 AWS Billing and Cost Management 儀表板上啟用這些標籤。AWS 會使用標籤來分類您的成本，並提供每月成本分配報告給您。如需詳細資訊，請參閱《AWS Billing 使用者指南》中的使用成本分配標籤。

以下各節提供有關 AWS Elemental 標籤的詳細資訊 MediaStore。

AWS Elemental 支援的資源 MediaStore

AWS Elemental MediaStore 支援標記中的下列資源：

- ##

如需新增和管理標籤的詳細資訊，請參閱管理標籤。

AWS Elemental MediaStore 不支援 AWS Identity and Access Management (IAM) 的以標籤為基礎的存取控制功能。

標籤命名和使用慣例

下列基本命名和使用慣例適用於搭配 AWS Elemental MediaStore 資源使用標籤：

- 每個資源的上限為 50 個標籤。
- 對於每一個資源，每個標籤金鑰必須是唯一的，且每個標籤金鑰只能有一個值。
- 最大標籤索引鍵長度為 128 個 UTF-8 形式的 Unicode 字元。
- 最大標籤值長度為 256 個 UTF-8 形式的 Unicode 字元。
- 允許的字元包括可用 UTF-8 表示的英文字母、數字、空格，還有以下特殊字元：. : + = @ _ / - (連字號)。Amazon EC2 資源允許任何字元。
- 標籤鍵與值皆區分大小寫。做為最佳實務，請決定大寫標籤的策略，並一致地在所有資源類型中實作該策略。例如，決定要使用 `Costcenter`、`costcenter` 還是 `CostCenter`，並針對所有標籤使用相同的慣例。避免針對相似的標籤使用不一致的大小寫處理。
- 標籤禁止使用 `aws:` 字首，它保留給 AWS 使用。您不可編輯或刪除具此字首的標籤金鑰或值。具此字首的標籤，不會算在每個資源配額的標籤計數內。

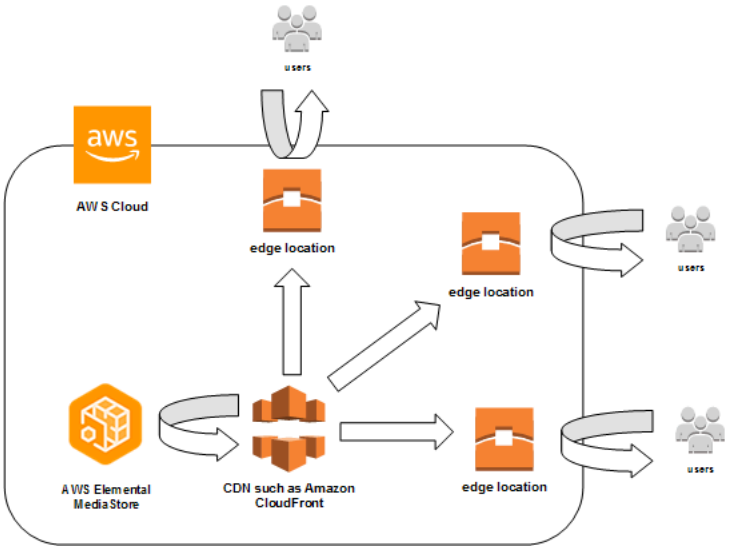
管理標籤

標籤是由資源上的 `Key` 和 `Value` 屬性組成。您可以使用AWS CLI或 MediaStore API 來新增、編輯或刪除這些屬性的值。如需使用標籤的相關資訊，請參閱 AWS Elemental MediaStore API 參考中的以下各節：

- CreateContainer
- ListTagsForResource
- 資源
- TagResource
- UntagResource

使用內容交付網路 (CDN)

您可以使用亞馬遜等內容交付網路 (CDN) CloudFront 來提供存放在 AWS Elemental 中的內容 MediaStore。CDN 是快取影片等內容的全球分佈伺服器。當使用者要求提供您的內容時，CDN 將此要求路由至產生最低延遲的節點。如果您的內容已於該節點中快取，CDN 就會立即提供該內容。如果您的內容目前不在該節點，CDN 會從您的來源 (例如您的容 MediaStore 器) 擷取該內容，並將其散佈給使用者。



主題

- 允許亞馬遜 CloudFront 訪問您的 AWS Elemental MediaStore 容器
- AWS Elemental 與 HTTP 快取 MediaStore的互動

允許亞馬遜 CloudFront 訪問您的 AWS Elemental MediaStore 容器

您可以使用亞馬遜 CloudFront 為您存放在 AWS Elemental 容器中的內容提供服務 MediaStore。您可以下列其中一種方法：

- 使用原始存取控制 (OAC)-(建議) 如果您AWS 區域支援的 OAC 功能，請使用此選項 CloudFront。
- 使用共用 Secrets-如果您AWS 區域不支援的 OAC 功能，請使用此選項 CloudFront。

使用原始存取控制 (OAC)

您可以使用 Amazon 的來源存取控制 (OAC) 功能，透過改善的安全性 CloudFront 來保護 AWS Elemental 來 MediaStore 源的安全性。您可以針對 MediaStore來源的 CloudFront 要求啟用AWS

簽章版本 4 (SIGv4)，並設定何時及是否 CloudFront 應簽署要求。您可以透 CloudFront 過主控台、API、SDK 或 CLI 存取的 OAC 功能，而且使用無需額外費用。

如需搭配使用 OAC 功能的詳細資訊 MediaStore，請參閱 Amazon CloudFront 開發人員指南中的限制對 MediaStore 來源的存取。

使用共用 Secrets

如果您AWS 區域不支援 Amazon 的 OAC 功能 CloudFront，您可以將政策附加到 AWS Elemental MediaStore 容器，以授予讀取權限或更高的權限 CloudFront。

> **Note**
>
> 如果您AWS 區域支援 OAC 功能，我們建議您使用此功能。下列程序需要您設定 MediaStore 並使 CloudFront 用共用密碼，以限制對 MediaStore 容器的存取。若要遵循最佳安全性做法，此手動組態需要定期輪替密碼。透過 MediaStore 原始 OAC，您可以指示使用 SIGv4 簽署 CloudFront 要求，並將其轉寄至 MediaStore 簽名比對，無需使用和輪換機密。這樣可以確保在提供媒體內容之前自動驗證請求，從 MediaStore 而使媒體內容的傳遞更加 CloudFront 簡單，更安全。

允許訪 CloudFront 問您的容器（控制台）

1. 請在以下位置開啟 MediaStore 主控台。 https://console.aws.amazon.com/mediastore/
2. 在 Containers (容器) 頁面上，選擇容器名稱。

   容器詳細資訊頁面隨即出現。
3. 在容器政策區段中，附加授予讀取存取權或更高權限給 Amazon 的政策 CloudFront。

   Example

   下列範例原則類似於透過 HTTPS 進行公開讀取存取的範例原則，符合這些需求，因為它允許`GetObject`任何透過 HTTPS 向您網域提交要求的使用者的`DescribeObject`命令。此外，下列範例原則更能保護您的工作流程，因為只有在要求透過 HTTPS 連線發生且包含正確的 Referer 標頭時，才允許 CloudFront 存取 MediaStore 物件。

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

```
      "Sid": "CloudFrontRead",
      "Effect": "Allow",
      "Principal": "*",
      "Action": [
        "mediastore:GetObject",
        "mediastore:DescribeObject"
      ],
      "Resource": "arn:aws:mediastore:<region>:<owner acct
number>:container/<container name>/*",
      "Condition": {
        "StringEquals": {
          "aws:Referer": "<secretValue>"
        },
        "Bool": {
          "aws:SecureTransport": "true"
        }
      }
    }
]}
```

4. 在 Container CORS policy (容器 CORS 政策) 區段中，指派允許適當存取等級的政策。

> **Note**
>
> 只有在您想要讓使用者存取瀏覽器型播放程式時，CORS 政策才為必要。

5. 請記下列詳細資訊：

   - 指定給容器的資料端點。您可以在 Containers (容器) 頁面的 Info (資訊) 區段中找到這項資訊。在中 CloudFront，資料端點稱為原始網域名稱。
   - 存放物件之容器中的資料夾結構。在中 CloudFront，這稱為原點路徑。請注意，這是選擇性設定。如需有關原始路徑的詳細資訊，請參閱 Amazon CloudFront 開發人員指南。

6. 在中 CloudFront，建立設定為提供 AWS Elemental 內容的發佈 MediaStore。您將需要您在前面步驟中收集的資訊。

將原則附加至 MediaStore 容器之後，您必須設定 CloudFront 為僅針對原始要求使用 HTTPS 連線，並新增具有正確密碼值的自訂標頭。

若 CloudFront 要設定透過 HTTPS 連線存取您的容器，其中包含 Revirer 標頭 (主控台) 的密碼值

1. 開啟主 CloudFront 控台。

2. 在「起源」頁面上，選擇您的 MediaStore來源。
3. 選擇 編輯 。
4. 僅針對通訊協定選擇 HTTPS。
5. 在「新增自訂標頭」區段中，選擇「新增標頭」。
6. 對於「名稱」，選擇「參考者」。對於值，請使用<secretValue>您在容器政策中使用的相同字串。
7. 選擇 [儲存] 並讓變更部署。

AWS Elemental 與 HTTP 快取 MediaStore的互動

AWS Elemental 可 MediaStore 存放物件，以便透過亞馬遜等內容交付網路 (CDN) 正確且有效率地快取這些物件 CloudFront。當一般使用者或 CDN 從中擷取物件時 MediaStore，服務會傳回影響物件快取行為的 HTTP 標頭。HTTP 1.1 快取行為的標準可在 RFC2616 第 13 節中取得)。這些標頭包括：

- **ETag** (不可自訂) -實體標籤標題是 MediaStore 傳送回應的唯一識別符。符合標準的 CDN 和網頁瀏覽器會使用此標籤做為快取物件的索引鍵。 MediaStore 上載時，會自動ETag為每個物件產生一個。您可以檢視物件的詳細資料，以判斷其 ETag 值。
- **Last-Modified**(不可自訂) — 此標頭的值表示修改物件的日期和時間。 MediaStore 上載物件時會自動產生此值。
- **Cache-Control** (可自訂) — 此標頭的值會控制物件在 CDN 查看它是否已遭修改前，應該快取的時間長度。當您使用 CLI 或 API 將物件上傳到 MediaStore 容器時，您可以將此標頭設定為任何值。完整一組的有效值會在 HTTP/1.1 文件中加以說明。如果您在上傳物件時未設定此值，則擷取物件時不 MediaStore 會傳回此標頭。

  快取控制標頭常見的使用案例是指定快取物件的持續時間。例如，假設您的視訊資訊清單檔案經常遭編碼器覆寫。您可以將 max-age 設為 10，表示該物件應該快取只有 10 秒。或者，假設您有永遠不會遭覆寫的儲存視訊區段。您可以將這個物件的 max-age 設定為 31536000 來快取約 1 年的長度。

條件式請求

有條件的請求 MediaStore

MediaStore 對條件式要求 (使用要求標頭 (如 RFC7232 中所述)If-Modified-Since 和If-None-Match無條件要求的回應完全相同。這意味著，當 MediaStore 收到有效的GetObject請求時，即使客戶端已經擁有該對象，服務始終返回該對象。

對 CDN 的條件請求

代表提供內容的 CDN MediaStore 可以透過傳回來處理條件式要求`304 Not Modified`，如 RFC7232 第 4.1 節所述。這表示系統已不需要傳輸完整的物件內容，因為請求者已經有符合條件請求的物件。

CDN (以及與 HTTP/1.1 相容的其他快取) 會將這些決策基於原始服務器轉發的 `ETag` 和 `Cache-Control` 標頭。若要控制 CDN 查詢 MediaStore 原始伺服器以取得重複擷取物件的更新頻率，請在將這些物件上傳至時設定這些物件的`Cache-Control`標頭 MediaStore。

AWS Elemental 中的配額 MediaStore

Service Quotas 主控台提供有關 AWS Elemental vice MediaStore 配額的相關資訊。除了檢視預設配額之外，您還可以使用 Service Quotas 主控台來請求增加配額以取得可調整配額。

下表說明 AWS Elemental 中的配額 (先前稱為限制) MediaStore。配額是您 AWS 帳戶的服務資源或操作數目最大值。

> **Note**
>
> 若要將配額指派給帳戶內的個別容器，請聯絡 AWS Support 或您的客戶經理。此選項可協助您在容器之間劃分帳戶層級限制，以防止一個容器使用您的整個配額。

資源或操作	預設配額	說明
容器	100	此帳戶中可建立的容器數量上限。
資料夾層級	10	容器中可建立的資料夾層級數量上限。您可以視需要建立任何數量的資料夾，只要容器中的巢狀層級不超過 10 層。
資料夾	無限制	您可以視需要建立任何數量的資料夾，只要容器中的巢狀層級不超過 10 層。
物件大小	25 MB	單一物件的檔案大小上限。
物件	無限制	您可以將任意數量的物件上傳到帳戶中的資料夾或容器。
DeleteObject API 請求的速率	100	您每秒可以提出的操作請求上限數量。其他請求會受到調節。 您可以要求增加配額。
DescribeObject API 請求的速率	1,000	您每秒可以提出的操作請求上限數量。其他請求會受到調節。 您可以要求增加配額。

資源或操作	預設配額	說明
標準上傳可用性的 GetObjectAPI 要求比率	1,000	您每秒可以提出的操作請求上限數量。其他請求會受到調節。 您可以要求增加配額。
串流上傳可用性的 GetObjectAPI 要求比率	25	您每秒可以提出的操作請求上限數量。其他請求會受到調節。 您可以要求增加配額。
ListItems API 請求的速率	5	您每秒可以提出的操作請求上限數量。其他請求會受到調節。 您可以要求增加配額。
區塊傳輸編碼的 PutObjectAPI 要求率 (也稱為串流上傳可用性)	10	您每秒可以提出的操作請求上限數量。其他請求會受到調節。 您可以要求增加配額。在請求中，指定請求的 TPS 和平均物件大小。
標準上傳可用性的 PutObjectAPI 要求比率	100	您每秒可以提出的操作請求上限數量。其他請求會受到調節。 您可以要求增加配額。在請求中，指定請求的 TPS 和平均物件大小。
指標政策中的規則	10	您可以包含在指標政策中的規則數量上限。
物件生命週期政策中的規則	10	您可以包含在物件生命週期政策中包含的規則數量上限。

AWS Elemental MediaStore 相關資訊

下表列出您在使用 AWS Elemental 時發現有用的相關資源 MediaStore。

- 課程和研討會 — 連結至以角色為基礎的專門課程以及自主進度實驗室，協助加強您的AWS技能，並取得實際體驗。
- AWS開發人員中心 — 研究教學課程、下載工具，以及了解AWS開發人員活動。
- AWS開發人員工具 – 連結至開發人員工具、軟體開發人員工具、開發人員工具、軟體開發人員工具、軟體開發人員AWS工具、
- 入門資源中心 — 了解如何設定AWS 帳戶、加入AWS社群，以及啟動您的第一個應用程式。
- 實用的教學課 step-by-step 程-按照啟動您的第一個應用程式AWS。
- AWS白皮書 — 連結至完整的技術AWS白皮書清單，其中涵蓋了架構、安全和成本等主題，並由AWS解決方案架構師或其他技術專家撰寫。
- AWS Support 中心 – 建立和管理您的 AWS Support 案例的中心。這也包含與其他實用資源的連結，例如論壇、技術常見問答集、服務運作狀態以及 AWS Trusted Advisor。
- AWS Support— 有關的資訊的主要網頁AWS Support，它是快速回應支援頻道 one-on-one，可協助您在雲端中建置並執行應用程式。
- 聯絡我們 – 查詢有關 AWS 帳單、帳戶、事件、濫用與其他問題的聯絡中心。
- AWS 網站條款 – 我們的著作權與商標；您的帳戶、授權與網站存取；以及其他主題的詳細資訊。

使用者指南的文件歷史記錄

下表說明此版本 AWS Elemental 的文件 MediaStore。如需有關此文件更新的通知，您可以訂閱 RSS 摘要。

變更	描述	日期
原始存取控制 (OAC) 改善	新增如何搭配 AWS Elemental 使用者指南的相關資訊 MediaStore。	2023 年 4 月 17 日
配額更新	已更正的配額值和說明`Rules in a Metric Policy`。	2022 年 10 月 25 日
ExpiresAt 字段	存取記錄現在包含一個`ExpiresAt` 欄位，根據容器生命週期原則中的暫時性資料規則，指出物件的到期日期和時間。	2020 年 7
生命週期變化規	您現在可以將生命週期轉換規則新增到物件的生命週期政策中，將物件設為在到達一定存留期後移動到不常存取 (IA) 儲存體方案。	2020 年 4 月 20 日
空容器	您現在可以一次刪除容器內的所有物件。	2020 年 4 月 7 日
Support 亞馬遜 CloudWatch 指標	您可以設定指標原則來指定 MediaStore 傳送的量度 CloudWatch。	2020 年 3 月 30 日
刪除物件規則中的萬用字元	在物件生命週期政策中，您現在可以在刪除物件規則中使用萬用字元。這可讓您根據要在特定天數後刪除服務的檔案名稱或副檔名來指定檔案。	2019 年 12 月 20 日

物件生命週期原	您現在可以將規則新增至物件生命週期政策，以秒為單位指定過期時間。	2019 年 9
AWS CloudFormation 支援	您現在可以使用 AWS CloudFormation 範本來自動建立容器。AWS CloudFormation 範本可管理五個 API 動作的資料：建立容器、設定存取記錄、更新預設容器政策、新增跨來源資源共享 (CORS) 政策，以及新增物件生命週期政策。	2019 年 5 月 17 日
串流上傳可用性的配額	對於具有串流上傳可用性的物件 (物件的區塊傳輸)，`PutObject` 作業不能超過 10 TPS，且 `GetObject` 作業不能超過 25 TPS。	2019 年 4 月 8 日
物件的區塊傳輸	新增對物件區塊傳輸的支援。此功能可讓您指定物件可在完全上傳前供下載使用。	2019 年 4 月 5 日
存取記錄	AWS Elemental MediaStore 現在支援存取記錄，應容取記錄。	2019 年 2 月 25 日
物件生命週期原	新增對物件生命週期政策的支援，此政策可管理目前容器中物件的過期日期。	2018 年 12 月 12 日
增加物件大小配額	物件大小的配額現在是 25 MB。	2018 年 10 月 10 日
增加物件大小配額	物件大小的配額現在是 20 MB。	2018 年 9 月 6 日

AWS CloudTrail 整合	CloudTrail 整合內容已更新，以配合 CloudTrail 服務的最新變更。	2018 年 7 月 12 日
CDN 協同合作	已新增有關如何將 AWS Elemental MediaStore 與內容交付網路 (CDN) (例如亞馬遜) 搭配使用的資訊 CloudFront。	2018 年 4
CORS 配置	AWS Elemental MediaStore 現在支援跨來源資源共享 (CORS)，讓載入單一個網域的用戶端 Web 應用程式，能與不同網域中的資源互動。	2018 年 2 月 7 日
新服務與指南	這是影片創作和儲存服務、AWS Elemental 和 AWS Ele mental MediaStore MediaStore 使用者指南的初始版本。	2017 年 11 月 27 日

Note

- AWS媒體服務不適用於應用程式或需要故障安全性能的情況下，例如生命安全操作、導航或通訊系統、空中交通管制或生命支援機器，而服務無法使用、中斷或故障可能導致死亡、人身傷害、財產損害或環境損害。

AWS 詞彙表

如需最新的 AWS 術語，請參閱《AWS 詞彙表 參考》中的 AWS 詞彙表。

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。